A Coalgebraic View of ε-Transitions

Alexandra Silva* and Bram Westerbaan[1]

ICIS, Radboud University Nijmegen

Abstract. In automata theory, a machine transitions from one state to the next when it reads an input symbol. It is common to also allow an automaton to transition without input, via an ε-transition. These ε-transitions are convenient, e.g., when one defines the composition of automata. However, they are not necessary, and can be eliminated. Such ε-elimination procedures have been studied separately for different types of automata, including non-deterministic and weighted automata.
It has been noted by Hasuo that it is possible to give a coalgebraic account of ε-elimination for some automata using trace semantics (as defined by Hasuo, Jacobs and Sokolova).
In this paper, we give a detailed description of the ε-elimination procedure via trace semantics (missing in the literature). We apply this framework to several types of automata, and explore its boundary.
In particular, we show that is possible (by careful choice of a monad) to define an ε-removal procedure for all weighted automata *over the positive reals* (and certain other semirings). Our definition extends the recent proposals by Sakarovitch and Lombardy for these semirings.

1 Introduction

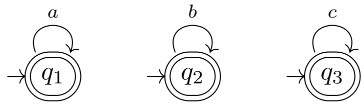
Automata are among the most basic structures in Computer Science. They have applications in a wide range of areas, including parsing, speech processing, and image recognition/generation software. Despite their simplicity, much research is still devoted to the semantics of automata and of related constructions.

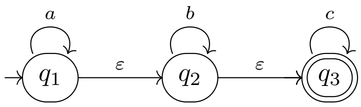
Coalgebra is a mathematical framework to study dynamical systems, of which automata are prime examples. Deterministic automata were the first automata to be studied as coalgebras in the seminal paper by Rutten [15]. Subsequently, various other types of automata and constructions were studied coalgebraically. This view has unified and generalized existing results and algorithms for different types of automata [17,1,2,18,3].

In this paper, we give a coalgebraic account of another concrete construction for automata: the elimination of ε-transitions. For this we use the abstract machinery of trace semantics. The advantage of this combination is two-fold. On the one hand, the concrete examples that the various types of automata provide clarify and ground the abstract notion of trace. On the other hand, trace semantics provides us with a uniform and intuitive definition for ε-elimination for many types of automata.

* Also affiliated with Centrum Wiskunde & Informatica (Amsterdam, The Netherlands) and HASLab / INESC TEC, Universidade do Minho (Braga, Portugal).

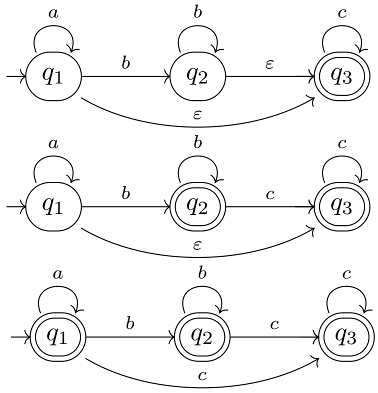
ε-Transitions are often useful at an intermediate stage. To illustrate this, let us show how to construct a non-deterministic automaton (without ε-transitions) that recognizes the language $a^*b^*c^*$. Note that it is easy to find automata recognizing the languages a^*, b^* and c^* (above, respectively).





If we compose these automata using ε-transitions, we obtain an automaton, on the left, that recognizes $a^*b^*c^*$. To obtain an automaton *without* ε-transitions that recognizes $a^*b^*c^*$ we incrementally eliminate the ε-transitions, as displayed below.

Hasuo and others [7,9] noted that result of the iterative process seen above can be captured using trace semantics in a Kleisli category, approach which we will discuss in more detail in Section 2.



In this paper, we take inspiration from [7,9] and we give an elaborate treatment of ε-elimination procedures using trace semantics. We extend their theory to include a class of weighted automata.

Though the process of ε-elimination for non-deterministic automata is classical and well-understood, for weighted automata things are less clear-cut, as witnessed by recent research [14,12]. The construction presented in this paper brings new results in comparison with the research presented in the aforementioned papers.

From a coalgebraic perspective, the challenge behind ε-elimination comes from the fact that many notions and definitions, such as bisimilarity for a functor, are given in a step-wise fashion. That is, the behavior of a certain system is fully determined by looking one step ahead at each time. This phenomenon, of having to deal with *multi-step* behavior, poses problems when having to model internal actions, such as ε-transitions, of a system. This is also present in concurrency theory, where internal actions (τ-steps) are discarded when defining weak bisimilarity. The theory presented in this paper might give a direction to improve the existing coalgebraic accounts of weak bisimilarity [19,4], which are not yet satisfactory.

The paper is organized as follows. In Section 2, we discuss the concrete construction for non-deterministic automata, we discuss how this paves the way to a coalgebraic account, and we introduce the idiosyncrasies behind the analogous construction for weighted automata. In Section 3, we present the general framework to formalize elimination of ε-transitions. In Section 4, we show how to model weighted automata in order to fit the framework. In Section 5, we discuss directions for future work, including the use of the framework for weak bisimilarity. In the appendix, we present extra material, including proofs omitted in the main text.

2 Motivation

In this section we describe the existing ε-elimination procedures for weighted and non-deterministic automata more thoroughly. We also recall some of the basic notions concerning automata. We present the material in a manner that is suited to the purposes of this paper. For instance, we represent these automata as coalgebras and make no mention of initial states.

2.1 Non-Deterministic Automata

We represent a **non-deterministic automaton** (NDA) with states X over the alphabet A as a map $\alpha\colon X \longrightarrow \wp(A \times X + 1)$, where $\wp$ is the powerset functor. Given $q, r \in X$ and $a \in A$, we write, omitting the coproduct injections,

$$\begin{array}{lcrr} q{\downarrow_\alpha} & \iff & * \in \alpha(q) & q \text{ is a final state} \\ q \xrightarrow{a}_\alpha r & \iff & (a,r) \in \alpha(q) & q \text{ has an } a\text{-transition to } r \end{array}$$

Let us recall the usual (language) semantics of α, i.e., which words a state $q \in X$ of α *accepts*. Let $w \equiv a_1 a_2 \cdots a_n$ be a word over A, and let $q \in X$. We say that q **accepts** w if there are $q_1, \ldots, q_n \in X$ such that

$$q \xrightarrow{a_1}_\alpha q_1 \xrightarrow{a_2}_\alpha \cdots \xrightarrow{a_n}_\alpha q_n \quad \text{and} \quad q_n{\downarrow_\alpha}. \tag{1}$$

So the semantics of α is captured by the map $[\![-]\!]_\alpha\colon X \longrightarrow \wp(A^*)$ given by

$$w \in [\![q]\!]_\alpha \qquad \iff \qquad q \text{ accepts } w,$$

where $q \in X$ and $w \in A^*$. So we will simply say that $[\![-]\!]_\alpha$ *is* the semantics of α.

ε-Transitions An NDA **with ε-transitions** (ε-NDA) with states X over an alphabet A is simply an NDA with states X over the alphabet $A + \{\varepsilon\}$,

$$\alpha\colon X \longrightarrow \wp((A + \{\varepsilon\}) \times X + 1),$$

but with a different semantics, which we define next.

Given a word $\tilde{w}$ over $A + \{\varepsilon\}$, let $\tilde{w}\backslash\varepsilon$ be the word on A one obtains by removing all the letters "ε" from $\tilde{w}$.

Let $w \in A^*$, and let $q \in X$. We say q **accepts** w (in the ε-NDA α) if there is $\tilde{w} \in (A + \{\varepsilon\})^*$ such that $w = \tilde{w}\backslash\varepsilon$ and q accepts $\tilde{w}$ in α seen as an NDA, as in (1).

Hence the semantics of α is the map $[\![-]\!]^\varepsilon_\alpha\colon X \longrightarrow \wp(A^*)$ given by, for $q \in X$,

$$[\![q]\!]^\varepsilon_\alpha \;=\; \big\{\, \tilde{w}\backslash\varepsilon\colon\;\; \tilde{w} \in [\![q]\!]_\alpha \,\big\}.$$

Or, more abstractly, $[\![-]\!]^\varepsilon_\alpha = \wp(-\backslash\varepsilon) \circ [\![-]\!]_\alpha$.

ε-Elimination for Non-Deterministic Automata Let α be an ε-NDA with states X and over an alphabet A. We construct an NDA $\alpha^{\#}\colon X \to \wp(A \times X + 1)$ which has the same semantics as α, in the sense that $[\![-]\!]_{\alpha^{\#}} = [\![-]\!]^{\varepsilon}_{\alpha}$. Since $\alpha^{\#}$ will have no ε-transitions we say "we have eliminated the ε-transitions".

The NDA $\alpha^{\#}$ is defined as follows. A state $q \in X$ has a transition in $\alpha^{\#}$ labelled by $a \in A$ to a state r if either this transition was already there in α or after a number of ε-transitions, starting from q, it is possible to make an a-transition to r. Formally:

$$q{\downarrow}_{\alpha^{\#}} \iff \left[\begin{array}{l} q \xrightarrow{\varepsilon}_{\alpha} q_1 \xrightarrow{\varepsilon}_{\alpha} \cdots \xrightarrow{\varepsilon}_{\alpha} q_n \quad \text{and} \quad q_n{\downarrow}_{\alpha} \\ \quad \text{for some } n \in \mathbb{N} \text{ and } q_1, \ldots, q_n \in X \end{array}\right.$$

$$q \xrightarrow{a}_{\alpha^{\#}} r \iff \left[\begin{array}{l} q \xrightarrow{\varepsilon}_{\alpha} q_1 \xrightarrow{\varepsilon}_{\alpha} \cdots \xrightarrow{\varepsilon}_{\alpha} q_n \quad \text{and} \quad q_n \xrightarrow{a}_{\alpha} r \\ \quad \text{for some } n \in \mathbb{N} \text{ and } q_1, \ldots, q_n \in X \end{array}\right.$$

Let $w \in A^*$ and $q \in X$. We leave it to the reader to verify that q accepts w in the ε-NDA α if and only if q accepts w in the NDA $\alpha^{\#}$, i.e., $[\![q]\!]_{\alpha^{\#}} = [\![q]\!]^{\varepsilon}_{\alpha}$. Hence, the following diagram commutes.

$$\begin{tikzcd} X \arrow[r, "{[\![-]\!]_{\alpha}}"] \arrow[dr, "{[\![-]\!]_{\alpha^{\#}}}"'] & \wp((A + \{\varepsilon\})^*) \arrow[d, "\wp(-\backslash\varepsilon)"] \\ & \wp(A^*) \end{tikzcd}$$

Note that $[\![-]\!]_{\alpha}$ is the semantics of α considered as an NDA.

Coalgebraic Formulation We want to find an abstract definition of $\alpha^{\#}$ so that it can be instantiated for other types of automata. To this end it turns out to be fruitful to consider the following variant of $\alpha^{\#}$. Let

$$\mathrm{tr}_{\alpha}\colon X \longrightarrow \wp(\,\mathbb{N} \times (A \times X + 1)\,)$$

be the map given by: for all $q, r \in X$, and $a \in A$, and $n \in \mathbb{N}$:

$$(n, (a, r)) \in \mathrm{tr}_{\alpha}(q) \iff \left[\begin{array}{l} q \xrightarrow{\varepsilon}_{\alpha} q_1 \xrightarrow{\varepsilon}_{\alpha} \cdots \xrightarrow{\varepsilon}_{\alpha} q_n \quad \text{and} \quad q_n \xrightarrow{a}_{\alpha} r \\ \quad \text{for some } q_1, \ldots, q_n \in X \end{array}\right.$$

$$(n, *) \in \mathrm{tr}_{\alpha}(q) \iff \left[\begin{array}{l} q \xrightarrow{\varepsilon}_{\alpha} q_1 \xrightarrow{\varepsilon}_{\alpha} \cdots \xrightarrow{\varepsilon}_{\alpha} q_n \quad \text{and} \quad q_n{\downarrow}_{\alpha} \\ \quad \text{for some } q_1, \ldots, q_n \in X \end{array}\right.$$

The map tr_{α} contains more information than $\alpha^{\#}$. For example, $\alpha^{\#}$ tells us if a state $q \in X$ is final by whether $* \in \alpha^{\#}(q)$. The map tr_{α} tells us more, namely whether a final state can be reached from the state q using exactly n ε-transitions by whether $(n, *) \in \mathrm{tr}_{\alpha}(q)$.

Note that we can recover $\alpha^{\#}$ from the map tr_{α}; we have

$$b \in \alpha^{\#}(q) \iff \exists n \in \mathbb{N} \quad (n, b) \in \mathrm{tr}_{\alpha}(q), \tag{2}$$

for all $q \in X$ and $b \in B$, where $B := A \times X + 1$. More categorically, we can formulate Statement (2) as:

$$\begin{tikzcd} X \arrow[r, "\mathrm{tr}_\alpha"] \arrow[rr, bend right, "\alpha^{\#}"'] & \wp(\,\mathbb{N}\cdot B\,) \arrow[r, "\wp(\nabla)"] & \wp(B) \end{tikzcd} \qquad \text{commutes.}$$

Here, $\mathbb{N}\cdot B$ is the countable coproduct and $\nabla\colon \mathbb{N}\cdot B \to B$ is the *codiagonal* given by $\nabla(n,b) = b$ for all $(n,b) \in \mathbb{N}\cdot B$.

We are interested in tr_α because it satisfies a recursive relation, namely

$$\begin{aligned} (0,b) \in \mathrm{tr}_\alpha(q) \quad &\iff \quad b \in \alpha(q) \\ (n+1,b) \in \mathrm{tr}_\alpha(q) \quad &\iff \quad \exists r \in X\big[\ (\varepsilon, r) \in \alpha(q) \ \wedge\ (n,b) \in \mathrm{tr}_\alpha(r)\ \big], \end{aligned} \tag{3}$$

where $q \in X$ and $n \in \mathbb{N}$ and $b \in B$.

The recursive relation (3) can be cast in an abstract form, and this allows us to define tr_α (and hence $\alpha^{\#}$) for different types of automata at once.

For this we will use the Kleisli category $\mathcal{K}\ell(\wp)$ of the monad $\wp$. Recall that a map $f\colon V \to \wp(W)$ is a morphism from V to W in $\mathcal{K}\ell(\wp)$, which we will write as $f\colon\ V \multimap W$.

Indeed, we will see that the map $\mathrm{tr}_\alpha\colon X \to \wp(\mathbb{N}\cdot B)$ is the unique morphism such that the following diagram commutes, in $\mathcal{K}\ell(\wp)$.

$$\begin{tikzcd} X \arrow[r, "\mathrm{tr}_\alpha", "\circ" marking] \arrow[d, "\alpha'"', "\circ" marking] & \mathbb{N}\cdot B \arrow[d, "\xi", "\circ" marking] \\ X + B \arrow[r, "\mathrm{tr}_\alpha + 1"', "\circ" marking] & \mathbb{N}\cdot B + B \end{tikzcd} \quad ,$$

where $\alpha'\colon X \longrightarrow \wp(X+B)$ is the composition of the following maps

$$X \xrightarrow{\ \alpha\ } \wp(\,(A + \{\varepsilon\}) \times X + 1\,) \xrightarrow{\ \cong\ } \wp(\,X + (A \times X + 1)\,)\ , \tag{4}$$

and $\xi\colon \mathbb{N}\cdot B \longrightarrow \wp(\mathbb{N}\cdot B + B)$ is given by $\xi(0,b) = \{b\}$, and $\xi(n+1,b) = \{(n,b)\}$, for all $b \in B$ and $n \in \mathbb{N}$.

We can formulate this more coalgebraically, as follows. Let F be the functor on $\mathcal{K}\ell(\wp)$ given by $F = - + B$. Then we can regard α' as an F-coalgebra,

$$X \overset{\alpha'}{\multimap} FX \ = X + B$$

The final F-coalgebra is ξ, and tr_α is the unique homomorphism from α' to ξ.

Such a unique homomorphism tr_α into the final coalgebra in a Kleisli category is called a *trace map* by Hasuo, Jacobs, and Sokolova [8].

We will use the observations above to study ε-elimination in a more general setting later on. But let us first consider the class of weighted automata.

2.2 Weighted Automata

Let S be a semiring (such as $\mathbb{R}$). A weighted automaton is similar to a non-deterministic automaton, but each transition and state carries a weight, $s \in S$. Depending on the semiring, one may think of the weight of the transition between two states q and r as the distance of the transition from q to r, or as the probability that α transitions from q to r. For more information on weighted automata, see [5].

We represent a **weighted automaton** over the semiring S with states X over an alphabet A by a map $\alpha\colon X \longrightarrow \mathscr{M}(A \times X + 1)$, where $\mathscr{M}$ is the *multiset monad* over S. Recall that

$$\mathscr{M}(X) \;=\; \big\{\varphi \mid \varphi\colon X \to S,\ \operatorname{supp}\varphi \text{ is finite}\,\big\}.$$

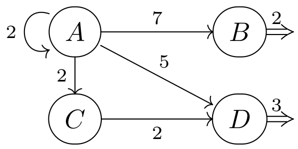
Given $q, r \in X$ and $a \in A$ and $s \in S$, we write

$$\begin{array}{lcll} q{\downarrow}^{s}_{\alpha} & \iff & s \,=\, \alpha(q)(*) & q \text{ outputs weight } s \\ q \xrightarrow{a|s}_{\alpha} r & \iff & s \,=\, \alpha(q)(a,r) & q \text{ has an } a\text{-transition to } r \text{ with weight } s \end{array}$$

The subscript α will be omitted whenever it is clear from the context.

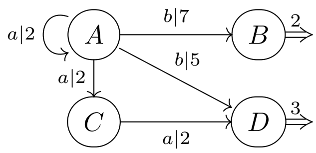
Let $a \in A$ be given. Note that formally there is an a-transition between any pair of states with *some* weight. We will typically only depict transitions with non-zero weight.

Semantics We explain the semantics of weighted automata by an example. Consider the following variation on a directed graph that represents a maze.



Suppose we stand at vertex A, and want to find the shortest path to exit the maze (via one of the exits, $\Rightarrow$). It is $A \to C \to D \Rightarrow$ with length 7.

Let us increase the complexity of the maze by adding some labels.



(5)

Again, we stand at A and want to find the shortest path to one of the exits, but this time we are only allowed to move along an ab-labelled path. That is, to exit the maze, we are only allowed to first move along an edge labelled by a, and then along an edge labelled by b, and then along $\Rightarrow$. Now the shortest path is $A \to A \to D \Rightarrow$ with length 10.

The maze in (5) can be represented by a weighted automaton α with states $X := \{A, B, C, D\}$ and alphabet $A := \{a, b\}$ over a semiring[1] on $\mathbb{R} \cup \{+\infty\}$ in a straightforward manner. When there is no c-labelled edge from one vertex to another we use a c-transition of weight $+\infty$, e.g., $\alpha(B)(a, D) = +\infty$. We interpret the symbol "$\overset{s}{\Rightarrow}$" at a vertex q to mean that q outputs s.

Note that we can express the length of the shortest ab-labelled path from A to an exit using α as follows.

$$\min_{q_1 \in X} \min_{q_2 \in X} \Big[\ \alpha(A)(\kappa_\ell(a, q_1)) \ +\ \alpha(q_1)(\kappa_\ell(b, q_2)) \ +\ \alpha(q_2)(\kappa_r(*))\ \Big]$$

Note that "+" and "min" form a semiring $\mathscr{T}_{\min}$ on $\mathbb{R} \cup \{+\infty\}$, called the *tropical semiring*. Confusingly, "+" is the *multiplication* of $\mathscr{T}_{\min}$ while "min" is the *addition*. Hence the *zero* of $\mathscr{T}_{\min}$ is $+\infty$ and the *one* is 0.

Observe that if we change the operations "+" and "min" (that is, if we change the semiring on $\mathbb{R} \cup \{+\infty\}$) we get different semantics $[\![-]\!]_\alpha$. For instance, if we take "+" and "max" instead, $[\![q]\!]_\alpha(w)$ will be the the length of the *longest* w-labelled path from q to an exit.

We now give the general definition of semantics for weighted automata. Let S be a semiring. Let $\alpha\colon X \longrightarrow \mathscr{M}(A \times X + 1)$ be a weighted automaton over S. Then the **semantics** of α is the map $[\![-]\!]_\alpha\colon X \longrightarrow S^{A^*}$, given by, for $q_1 \in X$, and a word $w = a_1 \cdots a_n \in A^*$,

$$[\![q_1]\!]_\alpha(w) \ :=\ \sum_{q_2 \in X} \cdots \sum_{q_{n+1} \in X} \Big(\prod_{i=1}^{n} \alpha(q_i)(a_i, q_{i+1})\Big) \cdot \alpha(q_{n+1})(*). \tag{6}$$

So a state in the weighted automaton α recognizes functions in S^{A^*}. These functions are usually referred to as *formal power series (over S)*.

Non-deterministic automata are a special case of weighted automata. Indeed, the reader can verify that if we take S to be the Boolean semiring then weighted automata over S correspond exactly to NDAs.

ε-Transitions A **weighted automaton with ε-transitions** α over a semiring S with states X and alphabet A is simply a weighted automaton over S with states X and alphabet $A + \{\varepsilon\}$.

To explain the semantics of α, we first consider the tropical case $S = \mathscr{T}_{\min}$. Following the earlier discussion of the semantics of ordinary weighted automata over $\mathscr{T}_{\min}$ and shortest paths, it seems natural to define the semantics of α to be the map $[\![-]\!]^\varepsilon_\alpha\colon X \longrightarrow S^{A^*}$ given by, for $q \in X$ and $w \in A^*$,

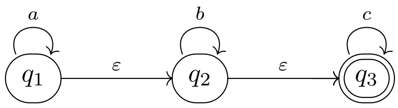
$$[\![q]\!]^\varepsilon_\alpha(w) \ =\ \min\ \big\{\ [\![q]\!]_\alpha(\tilde{w})\colon\ \tilde{w} \in (A + \{\varepsilon\})^* \text{ and } \tilde{w}\backslash\varepsilon = w\ \big\}. \tag{7}$$

In the maze analogy, $[\![q]\!]^\varepsilon_\alpha(w)$ is the length of a shortest w-labelled path from q to an exit when ε-moves are not counted.

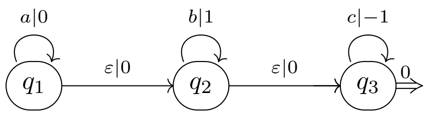
However, note that Equation (7) is not a sound definition for all α since the minimum might not exist. We will return to this problem shortly.

[1] The appropriate semiring structure on $\mathbb{R} \cup \{+\infty\}$ will become clear later on.

But first, we will further illustrate the semantics of ε-transitions. Recall that state q_1 in the following NDA accepts the language denoted by $a^*b^*c^*$:



Instead of talking about acceptance we now want to assign to each word in the language $a^*b^*c^*$ the difference between the number of b's and c's occurring in the word. In order to do that, we modify the above automaton into a weighted automaton over the tropical semiring $\mathscr{T}_{\min}$.



Note that for $w \in \{a, b, c\}^*$ the weight $[\![q_1]\!]^\varepsilon_\alpha(w)$ is precisely the number of b's occuring in w minus the number of c's occuring in w.

Inspired by Equation (7) we would like to define the semantics of a weighted automaton α with ε-transitions over any semiring S with states X and alphabet A to be the map $[\![-]\!]^\varepsilon_\alpha \colon X \longrightarrow S^{A^*}$ given by, for $q \in X$ and $w \in A^*$,

$$[\![q]\!]^\varepsilon_\alpha(w) \;=\; \sum \big\{\, [\![q]\!]_\alpha(\tilde{w}) \colon\; \tilde{w} \in (A + \{\varepsilon\})^* \text{ and } \tilde{w}\backslash\varepsilon = w \,\big\}. \tag{8}$$

However, without further information this is only a valid definition if the set

$$\big\{\, [\![q]\!]_\alpha(\tilde{w}) \colon \tilde{w} \in (A + \{\varepsilon\})^* \text{ and } \tilde{w}\backslash\varepsilon = w \,\big\}. \tag{9}$$

is finite. Otherwise, we do not know how we should interpret the symbol "$\sum$".

The problem is quite subtle. For example, consider the following weighted automata with ε-transitions over $\mathbb{R}$ (with the normal "+" and "·").

Writing $\square$ for the empty word, one sees using Equation (8), that

$$\begin{aligned}
[\![q_1]\!]^\varepsilon(\square) &\;=\; 1 + 0.5 + (0.5)^2 + \cdots \;=\; 2,\\
[\![q_2]\!]^\varepsilon(\square) &\;=\; 1 - 0.5 + (0.5)^2 - \cdots \;=\; {}^2\!/\!_3,
\end{aligned}$$

and in a daring mood we can compute,

$$[\![q_3]\!]^\varepsilon(\square) \;=\; 1 + 2 + 4 + 8 + \cdots \;=\; +\infty,$$

but what should we make of the following?

$$[\![q_4]\!]^\varepsilon(\square) \;=\; 1 - 1 + 1 - 1 + \cdots$$

To give proper meaning to weighted automata with ε-transitions it seems necessary to require that the semiring is equipped with a notion of summation for some sequences, and we must restrict ourselves to a class of weighted automata with ε-transitions for which the set in Expression (9) is summable.

Possibly due to this problem, the formal semantics of weighted automata with ε-transitions has not yet been settled in the literature.

In a recent proposal by Lombardy and Sakarovitch [12], semantics is given to a certain class of '*valid*' weighted automata with ε-transitions over *topological* semirings using a sophisticated ε-elimination *algorithm*. The automata with states q_1 and q_2 are valid, and the other two are not valid.

In this paper, the abstract view on automata gives rise to semantics to *all* weighted automata over certain semirings, namely *positive partial σ-semirings*. The semiring $[0, +\infty)$ is such a semiring, while $\mathbb{R}$ is not. So the general theory yields semantics for the automata with states q_1 and q_3, but not for the automata with states q_2 and q_4.

We will return to the example of weighted automata is Section 4.

3 Generalised ε-Elimination

Let us now turn to ε-elimination in a more general setting.

3.1 Automata in General

Setting 1 *Let* $\mathbf{C}$ *be a category, and assume that* $\mathbf{C}$ *has all finite limits and all countable colimits. Let* F *be a functor on* $\mathbf{C}$*, and let* T *be a monad on* $\mathbf{C}$*, with Kleisli category* $\mathcal{K}\ell(T)$*.*

In this setting, we abstractly define an automaton, parametrized by a functor F and a monad T, as follows.

Definition 2 *Let* X *be an object from* $\mathbf{C}$*. An* ***automaton*** *of type* T, F *with states* X *is a morphism* $\alpha\colon X \longrightarrow TFX$*.*

In other words, an automaton of type T, F is a morphism in $\mathcal{K}\ell(T)$ of the form

$$\alpha\colon X \longrightarrow\!\!\!\!\!\circ\;\; FX.$$

Examples 3 *Let* $\mathbf{C} = \mathbf{Sets}$*, and* $F = A \times - + 1$ *for some object* A *of* $\mathbf{C}$*.*

(i) Let $T = \wp$ *be the powerset monad. Then the automata of type* T, F *are non-deterministic automata with alphabet* A*.*
(ii) Let S *be a semiring. Let* $T := \mathscr{M}$ *be the multiset monad over* S*. Then the automata of type* T, F *are weighted automata with alphabet* A *over* S*.*

Example 4 *Let* $\mathbf{C} = \mathbf{Meas}$*. Let* $F = A\times-+1$*. Let* $T = \mathscr{G}$ *be the sub-probability monad (see [11]). Then the automata of type* T, F *are sub-probabalistic automata.*

3.2 Semantics of Automata

Setting 5 *All conditions from Setting 1 and, in addition, assume that F is lifted to a functor $\overline{F}$ on $\mathcal{K}\ell(T)$, via a distributive law $\lambda\colon FT \longrightarrow TF$, and that $\mathcal{K}\ell(T)$ has a final $\overline{F}$-coalgebra, $\omega\colon \Omega \multimapdotbothB\!\!\longrightarrow F\Omega$.*

Note that the $\overline{F}$-coalgebras in $\mathcal{K}\ell(T)$ are precisely the automata (of type T, F). The final $\overline{F}$-coalgebra is what we will use in order to abstractly define the semantics for F, T automata:

Definition 6 *Let $\alpha\colon X \longrightarrow\!\!\circ\!\!\longrightarrow \overline{F}X$ in $\mathcal{K}\ell(T)$ be given. We call unique homomorphism into the final coalgebra $[\![-]\!]_\alpha\colon X \longrightarrow\!\!\circ\!\!\longrightarrow \Omega$ the **semantics** of α.*

3.3 Trace and Iterate in General

Before we turn to the study of ε-elimination for these general automata, we present some theory on the assignment $\alpha \mapsto \alpha^{\#}$. The material is a slight simplification of the work by Hasuo in [7].

Setting 7 *Let $\mathbf{K}$ be a category that has all countable coproducts. Moreover, assume that for each object B from $\mathbf{K}$, there is a final $-+B$-coalgebra,*

$$\xi_B\colon N_B \longrightarrow N_B + B.$$

This setting is equivalent to require that the functor $-+B$ is iteratable [13]. In the sequel we instantiate $\mathbf{K}$ to the Kleisli category of a given monad.

Recall that since ξ_B is final, there is a unique homomorphism from each $-+B$-coalgebra to ξ_B. We call this homomorphism **trace**.

Definition 8 *Let $\beta\colon X \to X + B$ be a morphism in $\mathbf{K}$. The **trace** of β is the unique morphism $\mathrm{tr}_\beta\colon X \to N_B$ such that the following diagram commutes.*

$$\begin{tikzcd} X \arrow[r, "\mathrm{tr}_\beta"] \arrow[d, "\beta"'] & N_B \arrow[d, "\xi_B"] \\ X + B \arrow[r, "\mathrm{tr}_\beta + B"] & N_B + B \end{tikzcd}$$

Setting 9 *Let $\mathbf{K}$ be a category that has all countable coproducts. Let B be an object from $\mathbf{K}$. Denote the initial $-+B$-algebra by $\iota_B\colon \mathbb{N}\cdot B + B \longrightarrow \mathbb{N}\cdot B$. Assume also that $\xi_B := \iota_B^{-1}$ is the final $-+B$-coalgebra. So we have*

$$\xi_B\colon \mathbb{N}\cdot B \longrightarrow \mathbb{N}\cdot B + B.$$

Before we define the **iterate** operator, we need two additional definitions.

Definition 10 *Let $g\colon A \to B$ be a morphism in $\mathbf{K}$. Let $\mathbb{N}\cdot g\colon \mathbb{N}\cdot A \longrightarrow \mathbb{N}\cdot B$ be given by, for all $n \in \mathbb{N}$, $(\mathbb{N}\cdot g) \circ \kappa_n \;=\; \kappa_n \circ g$. Equivalently, $\mathbb{N}\cdot g$ is the unique morphism such that*

$$\begin{tikzcd} \mathbb{N}\cdot A \arrow[r, "\mathbb{N}\cdot g"] \arrow[d, "\xi_A"] & \mathbb{N}\cdot B \arrow[d, "\xi_B"] \\ \mathbb{N}\cdot A + A \arrow[r, "\mathbb{N}\cdot g + g"] & \mathbb{N}\cdot B + B \end{tikzcd} \qquad \textit{commutes.}$$

Definition 11 *Let B be an object of* $\mathbf{K}$*. The* ***codiagonal*** *is the morphism* $\nabla_B \colon \mathbb{N} \cdot B \longrightarrow B$ *given by* $\nabla_B \circ \kappa_n \;=\; \mathrm{id}_B$*, where* $n \in \mathbb{N}$*, and* $\kappa_n \colon B \longrightarrow \mathbb{N} \cdot B$ *is the* n*-th coprojection.*

Definition 12 *Let X and A be objects from* $\mathbf{K}$*, and let* $\alpha \colon X \longrightarrow X + A$ *be a morphism. Then the* ***iterate*** *of α is the morphism* $\alpha^{\#} \colon X \longrightarrow A$ *given by* $\alpha^{\#} := \nabla_A \circ \mathrm{tr}_\alpha$.

Proposition 13 *Suppose we have a commuting diagram in* $\mathbf{K}$ *of the form*

$$\begin{tikzcd} X \arrow[r, "f"] \arrow[d, "\alpha"] & Y \arrow[d, "\beta"] \\ X + A \arrow[r, "f+g"'] & Y + B \end{tikzcd}$$

where $g \colon A \to B$. Then the following square commutes.

$$\begin{tikzcd} X \arrow[r, "f"] \arrow[d, "\alpha^{\#}"] & Y \arrow[d, "\beta^{\#}"] \\ A \arrow[r, "g"] & B \end{tikzcd}$$

3.4 ε-Elimination in General

First, we define what an abstract automaton with ε-transitions is. (Since our general automata do not explicitly contain an alphabet this is not immediately clear.) Recall that in the case of non-deterministic automata, an automaton with ε-transitions is a map $\alpha \colon X \longrightarrow \wp(\,(A + \{\varepsilon\}) \times X + 1\,)$, and this map gives rise to a second map,

$$\alpha' \colon X \longrightarrow \wp(\,X \;+\; (A \times X + 1)\,).$$

We base our definition on the second map, α', instead of α.

Definition 14 *Let X be an object from* $\mathbf{C}$*. An* ***ε-automaton*** *of type T, F with states X is a morphism* $\alpha \colon X \longrightarrow T(\,X + FX\,)$*. In other words, α is an automaton of type* T, F_ε*, where* F_ε *is the functor with*

$$F_\varepsilon X \;=\; X + FX.$$

To provide the semantics of ε-automata, we need some assumptions.

Setting 15 *In addition to the assumptions in Setting 5, we assume that* $\mathcal{K}\ell(T)$ *has a final* $\overline{F}_\varepsilon$*-coalgebra* $\omega_\varepsilon \colon \Omega_\varepsilon \;\longrightarrow\!\!\circ\; \Omega_\varepsilon + F\Omega_\varepsilon$*. Here,* $\overline{F}_\varepsilon$ *is the lifting of* F_ε *to* $\mathcal{K}\ell(T)$*, via the distributive law* λ_ε *given by* $(\lambda_\varepsilon)_X \;=\; [T\kappa_\ell, T\kappa_r \circ \lambda_X]$*, where X is an object from* $\mathbf{C}$*. Moreover, let B be an object from* $\mathbf{C}$*. We denote the initial* $- + B$*-algebra in* $\mathcal{K}\ell(T)$ *by* $\iota_B \colon \mathbb{N} \cdot B + B \;\longrightarrow\!\!\circ\; \mathbb{N} \cdot B$*. Assume that* $\xi_B := \iota_B^{-1}$ *is the* final $- + B$*-coalgebra in* $\mathcal{K}\ell(T)$*.*

We need a last definition, before providing semantics to ε-automata.

Definition 16 *Let $-\backslash\varepsilon$ be the unique morphism in* $\mathbf{C}$ *such that*

$$\begin{tikzcd}
\Omega_\varepsilon \arrow[rr, "\cdot\backslash\varepsilon"] \arrow[rr, "\circ" marking] \arrow[d, "\omega_\varepsilon^{\#}"'] \arrow[d, "\circ" marking] & & \Omega \arrow[d, "\omega"] \arrow[d, "\circ" marking] \\
F\Omega_\varepsilon \arrow[rr, "F(\cdot\backslash\varepsilon)"'] \arrow[rr, "\circ" marking] & & F\Omega
\end{tikzcd}$$

commutes. That is, $\cdot\backslash\varepsilon$ is the semantics of the automaton $\omega_\varepsilon^{\#}$, $\cdot\backslash\varepsilon \;=\; [\![-]\!]_{\omega_\varepsilon^{\#}}$.

Definition 17 *Let $\alpha\colon X \;\mathrel{-\!\!\circ\!\!\rightarrow}\; X + FX$ be an ε-automaton of type T, F. The* ***semantics*** *of α is the map $[\![-]\!]_\alpha^\varepsilon\colon X \;\mathrel{-\!\!\circ\!\!\rightarrow}\; \Omega$ such that*

$$\begin{tikzcd}
X \arrow[r, "{[\![-]\!]_\alpha^\varepsilon}"] \arrow[r, "\circ" marking] \arrow[dr, "{[\![-]\!]_\alpha}"'] \arrow[dr, "\circ" marking] & \Omega \\
& \Omega_\varepsilon \arrow[u, "-\backslash\varepsilon"'] \arrow[u, "\circ" marking]
\end{tikzcd}$$

commutes, where $[\![-]\!]_\alpha$ is the semantics of α seen as automaton of type T, F_ε.

We can now present one of the main results of this paper, showing that (language) semantics is preserved by the abstract ε-elimination procedure.

Theorem 18 (ε-Elimination) *Let X be from* $\mathbf{C}$. *Let $\alpha\colon X \;\mathrel{-\!\!\circ\!\!\rightarrow}\; X + FX$ be an ε-automaton of type T, F. Then the iterate $\alpha^{\#}\colon X \;\mathrel{-\!\!\circ\!\!\rightarrow}\; FX$ is an automaton of type T, F with the same semantics as α. That is,*

$$[\![-]\!]_{\alpha^{\#}} \;=\; [\![-]\!]_\alpha^\varepsilon.$$

4 Weighted Automata and the $\mathscr{M}$ Monad

We now briefly return to the case of the weighted automata. Due to space constraints, we leave most details to the reader. Recall that a weighted automaton over a semiring S with states X and alphabet A is a map $\alpha\colon X \longrightarrow \mathscr{M}FX$, where $F = A \times - + 1$. So α is an automaton of type $\mathscr{M}, F$.

Unfortunately, the type $\mathscr{M}, F$ does not fit our general framework for automata (see Setting 15), since the inverse

$$\iota_B^{-1}\colon \mathbb{N}\cdot B \;\mathrel{-\!\!\circ\!\!\rightarrow}\; \mathbb{N}\cdot B + B$$

of the initial $- + B$-algebra ι_B in $\mathcal{K}\ell(\mathscr{M})$ is not the final $- + B$-coalgebra.

Indeed, this follows from the following example.

Example 19 *Let $B := \{b\}$ and let $\alpha\colon \{*\} \;\mathrel{-\!\!\circ\!\!\rightarrow}\; \{*\} + B$ be given by*

$$\alpha(*)(\kappa_\ell(*)) = \alpha(*)(\kappa_r(b)) = 1.$$

Suppose $\tau\colon \{*\} \;\mathrel{-\!\!\circ\!\!\rightarrow}\; \mathbb{N}\cdot B$ is a homomorphism from α to ι_B^{-1}. Then $\operatorname{supp}\tau(*)$ is finite by definition of $\mathscr{M}$. However, the reader can verify that $\tau(*)(n, b) = 1$ for all $n \in \mathbb{N}$. So we see that $\operatorname{supp}\tau(*)$ must be infinite as well. No such τ exists. Hence, ι_B^{-1} is not the final $- + B$-coalgebra in $\mathcal{K}\ell(\mathscr{M})$.

In order to study weighted automata in the general framework, we use

$$\mathscr{M}X \;:=\; \{\, \varphi\colon X \to S \mid \operatorname{supp}\varphi \text{ is at most countable} \,\}$$

instead of $\mathscr{M}X$. To turn $\mathscr{M}$ in a monad we need to assume that S is equipped with a notion of countable sums. For more details, see [**?**].

Note that an automaton of type $\mathscr{M}, F$ represents a weighted automaton that is allowed to have infinitely many (proper) transitions from a given state, while an automaton of type $\mathscr{M}, F$ is a weighted automaton with only finitely many transitions from a given state.

Fortunately, the automata of type $\mathscr{M}, F$ do fit nicely in our framework. That is, Setting 15 applies to them.

Proposition 1. *Given a set B, the inverse $\iota_B^{-1}\colon \mathbb{N}\cdot B \multimap \mathbb{N}\cdot B + B$ of the initial $-+B$-algebra ι_B in $\mathcal{K}\ell(\mathscr{M})$ is the final $-+B$-coalgebra. Similarly, the inverse $\xi\colon A^* \multimap A\times A^* + 1$ of the initial $\overline{F}$-algebra is the final $\overline{F}$-coalgebra in $\mathcal{K}\ell(\mathscr{M})$.*

Moreover, given a set A, and $\alpha\colon X \multimap X + A$ in $\mathcal{K}\ell(\mathscr{M})$, the iterate $\alpha^{\#}$ of α is given by, for all $q_0 \in X$ and $a \in A$,

$$\alpha^{\#}(q_0)(a) \;=\; \sum_{n\in\mathbb{N}} \sum_{q_1\in X} \cdots \sum_{q_{n+1}\in X} \Big(\prod_{i=1}^{n} \alpha(q_i)(q_{i+1})\Big) \cdot \alpha(a).$$

So we see that the abstract theory gives the expected results: the semantics $[\![-]\!]_\alpha$ of an automaton of type $\mathscr{M}, F$ turns out to be precisely the same as the semantics that we discussed before (see Equation (6)).

4.1 Valid Semirings

There is, however, a catch. The monad $\mathscr{M}$ is only defined over a *σ-semiring*, that is, a semiring S equipped with a summation operation that assigns to each family $(x_i)_{i\in I}$ of elements of S a sum $\sum_{i\in I} x_i$.

Usually, a semiring S is only equipped with a sum for some families of elements, which are then called *summable*. This idea is formalised in the notion *partial σ-semiring*. An example is the semiring of non-negative reals, $[0,\infty)$, equipped with a sum for all absolutely summable sequences. There are many examples of such partial σ-semirings. In fact, any semiring S is a partial σ-semiring in which only the *finite* families are summable.

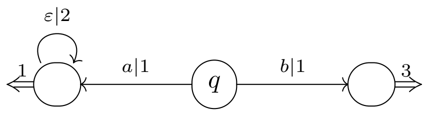
It is often possible to extend a partial σ-semiring S to a σ-semiring by adding one element $*$ to S and declaring that the sum of a family of element $(x_i)_{i\in I}$ of $S\cup\{*\}$ is the sum in S when $(x_i)_{i\in I}$ was summable in S and otherwise $*$.

Indeed, the above construction is possible if the partial σ-semiring has the following property: for all $a, b \in S$, $a + b = 0 \implies a = 0$ and $b = 0$. We call such semirings **positive** (using the terminology Gumm introduced for monoids [6]). In fact, any σ-semiring must be positive. So we see that only the positive partial

σ-semirings can be extended to a σ-semiring. A typical example of a semiring that is not positive is $\mathbb{R}$.

Let S be a positive partial σ-semiring. Then S can be extended to a σ-semiring $S \cup \{*\}$, and hence the abstract framework for automata is applicable to weighted automata over the semiring $S \cup \{*\}$.

The object $*$ acts as an "undefined" element. Consider the following weighted automaton α with ε-transitions over the semiring $\mathbb{R}$ with alphabet $\{a, b\}$.



Let us compute the semantics of α with Equation (8). We see that $[\![q]\!]^{\varepsilon}_{\alpha}(b) = 3$, but there is a difficulty when computing $[\![q]\!]^{\varepsilon}_{\alpha}(a) = 1 + 2 + 4 + \cdots$. However, if we consider α as a weighted automaton over the semiring $S \cup \{*\}$, then we simply get $[\![q]\!]^{\varepsilon}_{\alpha}(a) = *$, while still $[\![q]\!]^{\varepsilon}_{\alpha}(b) = 3$.

According to the Lombardy and Sakarovitch [12] the entire Automaton (**??**) is *invalid* because of the loop at state q_2. Note that our use of $*$ gives a finer notion of validity: only some combinations of a word w and a state q are invalid, namely those for which $[\![q]\!]^{\varepsilon}_{\alpha}(w) = *$.

All in all, the abstract framework applies to, and hence given us semantics, ε-elimination, and so on, for *all* weighted automata over *positive* semirings (possibly equipped with a partial summation).

5 Discussion

We have presented a framework where ϵ-elimination can be thought of in an abstract manner. The framework yields procedures for non-deterministic automata and, notably, for weighted automata. What we presented here can be seen as the beginning of a larger quest to understand multi-step behavior, which is still a challenge coalgebraically. There are several directions we would like to explore further and which we discuss briefly next.

Kleisli versus Eilenberg–Moore Recovering coalgebraic definitions of language equivalence has been done in two different settings. The one we used in this paper, based on Kleisli categories, and the one presented in [18,17,10], based on Eilenberg-Moore categories and a generalized powerset construction. The definition of iterate is natural in Kleisli and hence we have taken the first approach. We want to explore if it is possible to define similar notions in the Eilenberg-Moore setting and enlarge the examples the framework covers. For instance, the generalized powerset construction works for every weighted automaton, without having to resort to changes in the monad.

Weak Bisimilarity The notion of *bisimilarity* which was originally formulated for labelled transition systems (i.e. NDAs without final states) has been beautifully generalised to coalgebras. However the closely related notion of weak bisimilarity which has to do with ε-transitions has yet to receive a satisfactory general treatment. As we will see, the abstract framework in this paper does give us a 'weak' variant of bisimilarity, but this variant, let us call it *poor bisimilarity*, does not coincide with weak bisimilarity.

Before we give the definitions of weak bisimilarity and poor bisimilarity it is useful to spend some words on language equivalence. Two states x and y of an NDA α are *language equivalent* if they accept the same words, that is, if $[\![x]\!]_\alpha = [\![y]\!]_\alpha$. Similarly, two states of an ε-NDA α are *weakly language equivalent* if they accept the same words, that is, if $[\![x]\!]^\varepsilon_\alpha = [\![y]\!]^\varepsilon_\alpha$.

Since $[\![-]\!]^\varepsilon_\alpha = [\![-]\!]_{\alpha^\#}$ we can obtain weak language equivalence from language equivalence in the following way. let x and y be states of an ε-NDA α, then

$$[\![x]\!]^\varepsilon_\alpha = [\![y]\!]^\varepsilon_\alpha \qquad \iff \qquad [\![x]\!]_{\alpha^\#} = [\![y]\!]_{\alpha^\#}.$$

Bisimilarity is a variation on language equivalence for NDA. Indeed, two states x and y of an NDA α are *bisimilar* if $\{\!\{x\}\!\}_\alpha = \{\!\{y\}\!\}_\alpha$, where

$$\{\!\{-\}\!\}_\alpha \colon X \longrightarrow \Theta$$

is the unique homomorphism from α to the final coalgebra $\vartheta\colon \Theta \longrightarrow \wp(A\times\Theta+1)$ of type $\wp(A \times - + 1)$ in **Sets**.[2]

Now, taking inspiration from the way we could obtain weak language equivalence from language equivalence, let us define that two states x and y of an ε-NDA are *poorly bisimilar* if $\{\!\{x\}\!\}_{\alpha^\#} = \{\!\{y\}\!\}_{\alpha^\#}$.

Let us now work towards the definition of weak bisimilarity. Let α be an ε-NDA with states X and alphabet A and assume α has no final states. We need some notation. Let " $\Longrightarrow$ " be the relation on X defined by, for $q, r \in X$,

$$q \Longrightarrow r \qquad \text{iff} \qquad \left[\begin{array}{l} q \xrightarrow{\varepsilon}_\alpha q_1 \xrightarrow{\varepsilon}_\alpha \cdots \xrightarrow{\varepsilon}_\alpha q_n \xrightarrow{\varepsilon}_\alpha r \\ \quad \text{for some } n \in \mathbb{N} \text{ and } q_1, \ldots, q_n \in X. \end{array}\right.$$

According to Sangiorgi [16, Def. 4.2.1], two states $x, y \in X$ are *weakly bisimilar* if there is a relation $R \subseteq X \times X$ such that $(x, y) \in R$, and for all $(p, q) \in R$:

(i) if $p \Longrightarrow \tilde{p} \xrightarrow{a}_\alpha \tilde{p}' \Longrightarrow p'$ for some $a \in A$ and $\tilde{p}, \tilde{p}', p' \in X$,
then there are $\tilde{q}, \tilde{q}', q' \in X$ with $q \Longrightarrow \tilde{q} \xrightarrow{a}_\alpha \tilde{q}' \Longrightarrow q'$ and $(p', q') \in R$;
(ii) if $p \Longrightarrow p'$ for some $p' \in X$,
then there is a $q' \in X$ with $q \Longrightarrow q'$ and $(p', q') \in R$;
(iii) if $q \Longrightarrow \tilde{q} \xrightarrow{a}_\alpha \tilde{q}' \Longrightarrow q'$ for some $a \in A$ and $\tilde{q}, \tilde{q}', q' \in X$,
then there are $\tilde{p}, \tilde{p}', p' \in X$ with $p \Longrightarrow \tilde{p} \xrightarrow{a}_\alpha \tilde{p}' \Longrightarrow p'$ and $(p', q') \in R$;
(iv) if $q \Longrightarrow q'$ for some $q' \in X$,
then there is a $p' \in X$ with $p \Longrightarrow p'$ and $(p', q') \in R$.

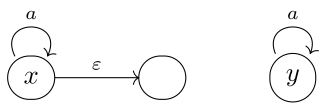
---

[2] In comparision, $[\![-]\!]_\alpha \colon X \multimap\!\!\!\longrightarrow \Omega$ is the unique homomorphism from α to the final coalgebra $\xi\colon \Omega \multimap\!\!\!\longrightarrow A \times \Omega + 1$ of type $A \times - + 1$ in $\mathcal{K}\ell(\wp)$.

Let us compare weak bisimilarity with poor bisimilarity. It turns out that two states $x, y \in X$ are poorly bisimilar if and only if there is a relation $R \subseteq X \times X$ such that $(x, y) \in R$, and for all $(p, q) \in R$:

(i) if $p \implies \tilde{p} \xrightarrow{a}_\alpha p'$ for some $a \in A$ and $\tilde{p}, p' \in X$,
then there are $\tilde{q}, q' \in X$ with $q \implies \tilde{q} \xrightarrow{a}_\alpha q'$ and $(p', q') \in R$;
(ii) if $q \implies \tilde{q} \xrightarrow{a}_\alpha q'$ for some $a \in A$ and $\tilde{q}, q' \in X$,
then there are $\tilde{p}, p' \in X$ with $p \implies \tilde{p} \xrightarrow{a}_\alpha p'$ and $(p', q') \in R$.

It is easy to see that in the ε-NDA depicted below, x and y are poorly bisimilar, but not weakly bisimilar. Hence poor and weak bisimilarity do not coincide.



Poor bisimilarity is one of many modications of weak bisimilarity. Sangiorgi [16] lists more than ten variations on weak bisimilarity, including dynamic bisimilarity, rooted weak bisimilarity, (rooted) delay bisimilarity and (rooted) branching bisimilarity. None of these coincide with poor bisimilarity.

It should be noted that poor bisimilarity might not behave well in the eyes of the concurrency theorist as it is not preserved by parallel composition "|".[3]

Only time will tell if poor bisimilarity bears any fruit. A different approach to the generalisation of weak bisimilarity is made by Brengos [4].

References

1. F. Bonchi, M. Bonsangue, M. Boreale, J. Rutten, and A. Silva. A coalgebraic perspective on linear weighted automata. *Inf. Comput.*, 211:77–105, 2012.
2. F. Bonchi, M. Bonsangue, J. Rutten, and A. Silva. Brzozowski's algorithm (co)algebraically. In *Logic and Program Semantics*, vol. 7230 of *LNCS*, pp. 12–23. Springer, 2012.
3. F. Bonchi and D. Pous. Checking nfa equivalence with bisimulations up to congruence. In *POPL*, pp. 457–468. ACM, 2013.
4. T. Brengos. Weak bisimulations for coalgebras over ordered functors. In *IFIP TCS*, vol. 7604 of *LNCS*, pp. 87–103. Springer, 2012.
5. M. Droste, W. Kuich, and W. Vogler. *Handbook of Weighted Automata*. Springer-Verlag, 2009.
6. H. Peter Gumm, Tobias Schröder. Monoid-labeled transition systems. *ENTCS* 44(1):185–204, 2001.
7. I. Hasuo, B. Jacobs, and A. Sokolova. Generic Forward and Backward Simulations. (Partly in Japanese) Proceedings of *JSSST Annual Meeting*, 2006.
8. Ichiro Hasuo, Bart Jacobs, and Ana Sokolova. Generic Trace Semantics via Coinduction. *Logical Methods in Computer Science* 3(4) (2007).
9. B. Jacobs. From coalgebraic to monoidal traces. *ENTCS*, 264(2):125–140, 2010.

[3] Nevertheless we believe it is possible to modify the parallel composition operator in a natural way so that it preserves poor bisimilarity. We will not pursue this.

10. B. Jacobs, A. Silva, and A. Sokolova. Trace semantics via determinization. In *CMCS*, volume 7399 of *LNCS*, pp. 109–129. Springer, 2012.
11. H. Kerstan and B. König. Coalgebraic trace semantics for probabilistic transition systems based on measure theory. *CONCUR 2012*, pp. 410–424, 2012.
12. S. Lombardy and J. Sakarovitch. The Removal of Weighted ε-Transitions. In *CIAA*, vol. 7381 of *LNCS*, pp. 345–352. Springer, 2012.
13. Stefan Milius. On Iteratable Endofunctors. In *CTCS*, vol. 69 of *ENTCS*, pp. 287–304, Elsevier, 2002.
14. M. Mohri. Generic ε-removal algorithm for weighted automata. In *CIAA*, vol. 2088 of *LNCS*, pp. 230–242. Springer, 2000.
15. Jan Rutten. Automata and coinduction (an exercise in coalgebra). In *CONCUR*, vol. 1466 of *LNCS*, pp. 194–218. Springer, 1998.
16. Davide Sangiorgi. An introduction to bisimulation and coinduction. Cambridge University Press, 2012.
17. A. Silva, F. Bonchi, M. Bonsangue, and J. Rutten. Generalizing the powerset construction, coalgebraically. In *FSTTCS*, volume 8 of *LIPIcs*, pp. 272–283, 2010.
18. A. Silva, F. Bonchi, M. Bonsangue, and J. Rutten. Generalizing determinization from automata to coalgebras. *LMCS*, 9(1), 2013.
19. A. Sokolova, E. de Vink, and H. Woracek. Coalgebraic weak bisimulation for action-type systems. *Sci. Ann. Comp. Sci.*, 19:93–144, 2009.

A Semirings

Definition 20 *A* ***semiring*** S *is a set endowed with a binary operation* $+$ *called* addition*; a constant* 0*, called* zero*; a binary operation* $\cdot$ *called* multiplication*; and a constant* 1*, called* one*, which satisfy the following axioms. The operations* $+$ *and* $\cdot$ *are associative;* 0 *is an identity element for* $+$*;* $+$ *is commutative;* 1 *is an identity element for* $\cdot$*;* $\cdot$ *distributes over* $+$*;* 0 *is an absorbing element for* $\cdot$*.*

Examples 21 *(i) The reals* $\mathbb{R}$ *for a semiring with the normal "+" and "·".*

(ii) Similarly, the non-negative reals $[0,\infty)$*, and the natural numbers* $\mathbb{N}$ *form semirings with the normal operations "+" and "·".*

(iii) The extended non-negative reals $[0,\infty]$ *form a semiring under one of the normal interpretation of "+" and "·". In particular, for* $a \in (0,\infty]$*,*

$$\infty + a \,=\, \infty \,=\, \infty \cdot a \qquad \text{and} \qquad 0\cdot\infty = 0.$$

(iv) The ***tropical semiring*** $\mathscr{T}_{\min}$ *on* $\mathbb{R}\cup\{+\infty\}$ *has as* multiplication *the normal operation "+" and as* addition *the operation "min".*
Confusingly, the one of $\mathscr{T}_{\min}$ *is* 0 *while the zero of* $\mathscr{T}_{\min}$ *is* $+\infty$*.*

(v) Similarly, the ***tropical semiring*** $\mathscr{T}_{\max}$ *on* $\mathbb{R}\cup\{-\infty\}$ *has as* multiplication *the normal operation "+" and as* addition *the operation "max".*
The one of $\mathscr{T}_{\max}$ *is* 0 *while the zero of* $\mathscr{T}_{\max}$ *is* $-\infty$*.*

(vi) The ***Boolean semiring*** $\mathbf{2}$ *is the semiring on* $2=\{0,1\}$ *given by*

$$a+b \,=\, \max\{a,b\} a\cdot b \,=\, \min\{a,b\}.$$

The zero of $\mathbf{2}$ *is* 0 *and the one of* $\mathbf{2}$ *is* 1*.*

B Monads and Kleisli categories

Coalgebraic (finite) trace semantics has been developed for coalgebras of the form $X \to TF(X)$ where T is a suitable monad and F a suitable functor, see [8]. Essential for coalgebraic trace semantics is the Kleisli category of a monad. A monad (T, η, μ), which we will frequently denote by T, on **Sets** consists of an endofunctor T on **Sets** and two natural transformations, the unit $\eta\colon id \Rightarrow T$ and the multiplication $\mu\colon TT \Rightarrow T$, that is, functions $\eta_X : X \to T(X)$ and $\mu_X : TT(X) \to T(X)$ for each set X satisfying a naturality condition. The unit and multiplication satisfy the compatibility conditions $\mu_X \circ \eta_{TX} = \mu_X \circ T\eta_X = id$ and $\mu_X \circ T\mu_X = \mu_X \circ \mu_{TX}$.

The monad structures provide a perfect way of modelling "branching". Intuitively, the unit η embeds a non-branching behavior as a trivial branching (with a single branch) whereas the multiplication μ "flattens" two successive branchings into one branching, abstracting away internal branchings.
We briefly describe the examples of monads on **Set** that we use in this paper.

(i) The powerset monad $\wp$ maps a set X to the set $\wp(X)$ of subsets of X, and a function $f\colon X \to Y$ to $\wp(f)\colon \wp(X) \to \wp(Y)$ given by direct image. Its unit is given by singleton $\eta(x) = \{x\}$ and multiplication by union

$$\mu(\{X_i \in \wp(X) \mid i \in I\}) = \bigcup_{i \in I} X_i$$

.

(ii) For a semiring S, the multiset monad $\mathscr{M}_S$ is defined on a set X as:

$$\mathscr{M}_S X = \{\varphi\colon X \to S \mid \mathrm{supp}(\varphi) \text{ is finite }\}.$$

The support set of a multiset $\varphi \in \mathscr{M}_S X$ is defined as

$$\mathrm{supp}(\varphi) = \{x \in X \mid \varphi(x) \neq 0\}.$$

This monad captures multisets $\varphi \in \mathscr{M}_S X$, where the value $\varphi(x) \in S$ gives the multiplicity of the element $x \in X$. When $S = \mathbb{N}$, this is sometimes called the bag monad.
On functions $f\colon X \to Y$, $\mathscr{M}_S$ is defined as $\mathscr{M}_S f(\varphi)(y) = \sum_{x \in f^{-1}(y)} \varphi(x)$. The finite support requirement is necessary for $\mathscr{M}$ to be a monad. The unit η and multiplication μ of $\mathscr{M}_S$ are defined as $\eta(x) = (x \mapsto 1)$ for $x \in X$ and the multiplication by $\mu(\Phi)(x) = \sum_{\varphi \in \mathrm{supp}(\Phi)} \Phi(\varphi) \cdot \varphi(x)$ for $\Phi \in \mathscr{M}_S \mathscr{M}_S X$.

Further examples of monads on **Set** include the sub distribution monad $\mathcal{D}$, the lift monad $1 + -$, and the quantale monad $\mathscr{Q}$. The sub distribution monad also is defined in the category **Meas** of measure spaces [11].

A monad T on a category **C** allows for a definition of a Kleisli category $\mathcal{K}\ell(T)$ whose objects are sets, and a morphism $f\colon\ X \longrightarrow\!\!\!\!\!\circ\;\; Y$ is a function

$f\colon X \to TY$. The identity morphism on X is η_X, and composition of morphisms is defined as

$$f \odot g = \mu \circ Tf \circ g = \left(X \xrightarrow{\ g\ } TY \xrightarrow{\ Tf\ } TTZ \xrightarrow{\ \mu\ } TZ \right).$$

There is a canonical lifting functor $J\colon \mathbf{Sets} \to \mathcal{K}\ell(T)$ which is the identity on objects, and maps a function $f\colon X \to Y$ to the function $J(f) = \eta \circ f\colon X \to TY$.

In Kleisli categories, coproducts are defined the same way as in the base category $\mathbf{C}$.

The coalgebraic trace result of [8] applies to TF-coalgebras in $\mathbf{Sets}$ if T and F satisfy a number requirements:

(i) There exists a distributive law $\lambda\colon FT \Rightarrow TF$. As a consequence, F lifts to a functor $\overline{F}$ on $\mathcal{K}\ell(T)$, with $\overline{F}(X) = F(X)$ and for a Kleisli arrow $f\colon X \multimap Y$, i.e., a map $f\colon X \to TY$, $\overline{F}(f) = \lambda \circ F(f)$. Hence TF-coalgebras in $\mathbf{Sets}$ are $\overline{F}$-coalgebras in $\mathcal{K}\ell(T)$.
(ii) The Kleisli category $\mathcal{K}\ell(T)$ is suitably order-enriched, with order $\sqsubseteq$ on Kleisli homsets, bottom element $\perp$ and suprema of directed subsets.
(iii) The lifting $\overline{F}\colon \mathcal{K}\ell(T) \to \mathcal{K}\ell(T)$ is locally monotone.

The requirements are explained in detail in [8]. The main result of the generic trace theory [8] is:

> If T and F satisfy the requirements of the generic trace theory and there exists an initial F-algebra $\iota\colon F(I) \xrightarrow{\cong} I$ in $\mathbf{Sets}$, then the lifted coalgebra $J(\iota^{-1}) = \eta \circ \iota^{-1}\colon I \to TF(I)$ is final $\overline{F}$-coalgebra in $\mathcal{K}\ell(T)$.

This enables defining trace semantics for TF-coalgebras in $\mathbf{Sets}$ as the final coalgebra semantics for $\overline{F}$-coalgebras in $\mathcal{K}\ell(T)$. More precisely, for a coalgebra $\alpha\colon X \to TFX$ in $\mathbf{Sets}$, i.e., $\alpha\colon X \multimap \overline{F}Y$ in $\mathcal{K}\ell(T)$, we denote by tr_α the final coalgebra map in $\mathcal{K}\ell(T)$, called the trace map. The trace of a state $x \in X$ is given by the image $\mathrm{tr}_\alpha(x)$. Trace equivalence is defined by $x \sim_{\mathrm{tr}} y \Leftrightarrow \mathrm{tr}_\alpha(x) = \mathrm{tr}_\alpha(y)$.

The requirements of the generic trace theory hold for the powerset monad $\mathcal{P}$, the subdistribution monad $\mathcal{D}$, and the lift monad $1 + -$, together with the inductively defined class of all "shapely functors" [8].

Unfortunately, the requirements do not hold for the monad $\mathscr{M}$ which prevents us from getting from the general framework the notion of trace for weighted automata. To overcome this, we define a new monad, $\mathscr{M}$, which still does not fit into the framework of trace semantics but does admit traces, fact which we prove explicitly in Section 34.

C Theory on Trace and Iterate

Setting 22 *See Setting 7 and Definition 8.*

Proposition 23 *Let $g\colon A \to B$ be a morphism in $\mathbf{K}$.*
Then there is a unique $N_g\colon N_A \to N_B$ such that the following diagram commutes.

$$\begin{tikzcd}
N_A \arrow[r, "N_g"] \arrow[d, "\xi_A"] & N_B \arrow[d, "\xi_B"] \\
N_A + A \arrow[r, "N_g+g"] & N_B + B
\end{tikzcd}$$

Moreover, if we have a commuting diagram in $\mathbf{K}$ of the form

$$\begin{tikzcd}
X \arrow[r, "f"] \arrow[d, "\alpha"] & Y \arrow[d, "\beta"] \\
X + A \arrow[r, "f+g"'] & Y + B
\end{tikzcd}$$

then the following cube commutes.

$$\begin{tikzcd}
& N_A \arrow[rr, "N_g"] \arrow[dd, "\xi_A"] & & N_B \arrow[dd, "\xi_B"] \\
X \arrow[ur, "\mathrm{tr}_\alpha"] \arrow[rr, "f"] \arrow[dd, "\alpha"] & & Y \arrow[ur, "\mathrm{tr}_\beta"] \arrow[dd, "\beta"] & \\
& N_A + A \arrow[rr, "N_g+g"] & & N_B + B \\
X + A \arrow[ur, "\mathrm{tr}_\alpha + A"] \arrow[rr, "f+g"'] & & X + B \arrow[ur, "\mathrm{tr}_\beta + B"'] &
\end{tikzcd} \tag{10}$$

Proof. Let us first prove that there is a unique $h\colon N_A \to N_B$ such that

$$\xi_B \circ h \;=\; (h + g) \circ \xi_A. \tag{11}$$

Let $h\colon N_A \to N_B$ be any morphism. Note that $h + g = (h + B) \circ (N_A + g)$.
So we see that Statement (11) is equivalent to

$$\xi_B \circ h \;=\; (h + B) \circ (N_A + g) \circ \xi_A. \tag{12}$$

Note that the morphism $(N_A + g) \circ \xi_A$ is a $- + B$-coalgebra. Because ξ_B is the final $- + B$-coalgebra, there is a unique $h\colon N_A \to N_B$ such that Statement (12) holds. Hence there is a unique $h\colon N_A \to N_B$ such that Statement (11) holds.

$$\begin{tikzcd}
N_A \arrow[rr, "h"] \arrow[d, "\xi_A"'] & & N_B \arrow[dd, "\xi_B"] \\
N_A + A \arrow[d, "N_A+g"'] \arrow[drr, "h+g"] & & \\
N_A + B \arrow[rr, "h+B"'] & & N_B + B
\end{tikzcd} \tag{13}$$

Recall that we denote this unique h by N_g.

Let us now prove that the cube commutes (see Diagram (10)).

That is, we must show that all six face of the cube commute. It is easy to see that four of them commute: the face on the front commutes by assumption; the face on the back commutes as it is the unique property of N_g; and the faces on the sides commute as tr_α, and of tr_β are homomorphisms.

It remains to be shown that the faces on the top and bottom commute.

Let us prove that the top face commutes. We must show that

$$N_g \circ \mathrm{tr}_\alpha \;=\; \mathrm{tr}_\beta \circ f. \tag{14}$$

Note that $(X+g) \circ \alpha$ is a $-+B$-coalgebra on X. So to prove that Statement (14) holds, it suffices to show that both $N_g \circ \mathrm{tr}_\alpha$ and $\mathrm{tr}_\beta \circ f$ are homomorphisms from $(X+g) \circ \alpha$ to ξ_B. (Recall ξ_B is the final $-+B$-coalgebra.)

Let us prove that $N_g \circ \mathrm{tr}_\alpha$ is a homomorphism. We have

$$\begin{array}{rcll}
\xi_B \circ N_g \circ \mathrm{tr}_\alpha & = & (N_g + g) \circ \xi_A \circ \mathrm{tr}_\alpha & \text{by definition of } N_g \\
& = & (N_g + g) \circ (\mathrm{tr}_\alpha + A) \circ \alpha & \text{as } \mathrm{tr}_\alpha \text{ is a hom.} \\
& = & (\,(N_g \circ \mathrm{tr}_\alpha) + B\,) \circ (X+g) \circ \alpha. &
\end{array}$$

$$\begin{tikzcd}
X \arrow[rr, "\mathrm{tr}_\alpha"] \arrow[d, "\alpha"'] & & N_A \arrow[d, "\xi_A"'] \arrow[dr, "N_g"] & \\
X+A \arrow[rr, "\mathrm{tr}_\alpha + A"] \arrow[dr, "X+g"'] & & N_A+A \arrow[dr, "N_g+g"] & N_B \arrow[d, "\xi_B"] \\
& X+B \arrow[rr, "(N_g \circ \mathrm{tr}_\alpha)+B"'] & & N_B+B
\end{tikzcd}$$

Hence $\mathbb{N}_g \circ \mathrm{tr}_\alpha$ is a homomorphism from $(X+g) \circ \alpha$ to ξ_B.

We now prove that $\mathrm{tr}_\beta \circ f$ is a homomorphism from $(X+g) \circ \alpha$ to ξ_B.

$$\begin{array}{rcll}
\xi_B \circ \mathrm{tr}_\beta \circ f & = & (\mathrm{tr}_\beta + B) \circ \beta \circ f & \text{as } \mathrm{tr}_\beta \text{ is a hom.} \\
& = & (\mathrm{tr}_\beta + B) \circ (f+g) \circ \alpha & \text{by assumption} \\
& = & (\,(\mathrm{tr}_\beta \circ f) + B\,) \circ (X+g) \circ \alpha. &
\end{array}$$

$$\begin{tikzcd}
X \arrow[rr, "f"] \arrow[d, "\alpha"'] & & Y \arrow[d, "\beta"'] \arrow[dr, "\mathrm{tr}_\beta"] & \\
X+A \arrow[rr, "f+g"] \arrow[dr, "X+g"'] & & Y+B \arrow[dr, "\mathrm{tr}_\beta + B"] & N_B \arrow[d, "\xi_B"] \\
& X+B \arrow[rr, "(\mathrm{tr}_\beta \circ f)+B"'] & & N_B+B
\end{tikzcd}$$

So both $\mathrm{tr}_\beta \circ f$ and $N_g \circ \mathrm{tr}_\alpha$ are homomorphisms to ξ_B, and hence equal. Thus the top face of the cube commutes.

It follows easily that the bottom face of the cube commutes as well,

$$\begin{array}{rcll}
(N_g + g) \circ (\mathrm{tr}_\alpha + A) & = & (\,N_g \circ \mathrm{tr}_\alpha\,) + g & \\
& = & (\,\mathrm{tr}_\beta \circ f\,) + g & \text{the top face commutes} \\
& = & (\mathrm{tr}_\beta + B) \circ (f+g). &
\end{array}$$

Hence all faces of the cube commute.

Setting 24 *See Setting 9 and Definition 12.*

Lemma 25 *Let $g\colon A \to B$ be a morphism in $\mathbf{K}$. Then the diagram*

$$\begin{tikzcd}
\mathbb{N}\cdot A \arrow[r, "\mathbb{N}\cdot g"] \arrow[d, "\nabla_A"] & \mathbb{N}\cdot B \arrow[d, "\nabla_B"] \\
A \arrow[r, "g"] & B
\end{tikzcd}$$

commutes. In other words, The morphisms ∇_B form a natural transformation.

Proof. To prove $\nabla_B \circ (\mathbb{N}\cdot g) = g \circ \nabla_A$, it suffices to show that, for all $n \in \mathbb{N}$,

$$\nabla_B \circ (\mathbb{N}\cdot g) \circ \kappa_n \;=\; g \circ \nabla_A \circ \kappa_n. \tag{15}$$

Let $n \in \mathbb{N}$ be given. Then we have:

$$\begin{aligned}
\nabla_B \circ (\mathbb{N}\cdot g) \circ \kappa_n &= \nabla_B \circ \kappa_n \circ g && \text{by Definition 10} \\
&= \mathrm{id}_B \circ g && \text{by Definition 11} \\
&= g \circ \mathrm{id}_A && \\
&= g \circ \nabla_A \circ \kappa_n && \text{by Definition 11}
\end{aligned}$$

So we have proven Statement (15).

Proposition 26 *Suppose we have a commuting diagram in $\mathbf{K}$ of the form*

$$\begin{tikzcd}
X \arrow[r, "f"'] \arrow[d, "\alpha"] & Y \arrow[d, "\beta"] \\
X + A \arrow[r, "f+g"'] & Y + B
\end{tikzcd}$$

where $g\colon A \to B$. Then the following square commutes.

$$\begin{tikzcd}
X \arrow[r, "f"] \arrow[d, "\alpha^{\#}"] & Y \arrow[d, "\beta^{\#}"] \\
A \arrow[r, "g"] & B
\end{tikzcd}$$

Proof. We must prove that $g \circ \alpha^{\#} = \beta^{\#} \circ f$. We have

$$\begin{aligned}
g \circ \alpha^{\#} &= g \circ \nabla_A \circ \mathrm{tr}_\alpha && \text{by definition of } \alpha^{\#} \\
&= \nabla_B \circ (\mathbb{N}\cdot g) \circ \mathrm{tr}_\alpha && \text{by Lemma 25} \\
&= \nabla_B \circ \mathrm{tr}_\beta \circ f && \text{by Proposition 23} \\
&= \beta^{\#} \circ f && \text{by definition of } \beta^{\#}.
\end{aligned}$$

$$\begin{tikzcd}
A \arrow[rr, "g"] & & B & \\
& \mathbb{N}\cdot A \arrow[ul, "\nabla_A"] \arrow[rr, "\mathbb{N}\cdot g"] & & \mathbb{N}\cdot B \arrow[ul, "\nabla_B"] \\
X \arrow[uu, "\alpha^{\#}"] \arrow[ur, "\mathrm{tr}_\alpha"'] \arrow[rr, "f"'] & & Y \arrow[uu, "\beta^{\#}"] \arrow[ur, "\mathrm{tr}_\beta"'] &
\end{tikzcd}$$

D The Monad $\mathscr{M}$

Let S be a semiring. Recall that the multiset monad $\mathscr{M}$ assigns to each set X,

$$\mathscr{M}X \;=\; \{\,\varphi \in X \to S\colon\;\; \operatorname{supp}\varphi \text{ is finite}\,\}.$$

One can think of $\mathscr{M}X$ as the set of *finite* multisets on X over the semiring S.

Informally, the monad $\mathscr{M}$ is variation on $\mathscr{M}$ that assigns to X the *at most countable* multisets on X over the semiring S. To define $\mathscr{M}$, we need a notion of countable sums on S. Formally, we assume S is a σ-semiring (see Appendix E).

Definition 27 *Let S be some semiring. Let X be a set.*

$$\mathscr{M}_S X \;:=\; \{\;\varphi\colon X \to S \mid \operatorname{supp}\varphi \textit{ is at most countable}\;\}.$$

Notation 28 *We will write $\mathscr{M}X$ instead of $\mathscr{M}_S X$ when we believe that it is clear from the context which semiring S is meant.*

Let S be a semiring. We want extend the assignment $X \mapsto \mathscr{M}X$ to a monad on **Sets** similar to the multiset monad. To do this, we want to have suitable notion of countable summation on S at our disposal. Thus we assume that S is a σ-semiring (see Definition 36).

Let us now introduce the pieces that will turn $X \mapsto \mathscr{M}X$ into a monad.

Definition 29 *Let C be a σ-semiring.*

*(i) Let X and Y be sets, and let $f\colon X \to Y$ be a function.
Let $\mathscr{M}f\colon \mathscr{M}X \to \mathscr{M}Y$ be given by: for all $\varphi \in \mathscr{M}X$, and $y \in Y$,*

$$(\mathscr{M}f)(\varphi)(y) \;:=\; \sum_{f(x)=y} \varphi(x).$$

To see the above equation does indeed give a map into $\mathscr{M}Y$, note that

$$\operatorname{supp}(\mathscr{M}f)(\varphi) \;\subseteq\; f(\operatorname{supp}\varphi).$$

(ii) Let X be a set. Let $\eta_X\colon X \to \mathscr{M}X$ be defined by: for all $x, \tilde{x} \in X$,

$$\eta_X(x)(\tilde{x}) \;=\; \begin{cases} 1 & \textit{if } x = \tilde{x}, \\ 0 & \textit{otherwise.} \end{cases}$$

(iii) Let X be a set. Let $\mu_X\colon \mathscr{M}\mathscr{M}X \to \mathscr{M}X$ be given by

$$\mu_X(\Phi)(x) \;=\; \sum_{\varphi \in \mathscr{M}X} \Phi(\varphi)\,\cdot\,\varphi(x).$$

where $\Phi \in \mathscr{M}X$ and $x \in X$.

The proof that the above data give a monad involves some calculation.
To make the work easier, we make some remarks.

Remarks 30 *Let C be a σ-semiring.*

(i) Let X and Y be sets, and let $f\colon X \to Y$. We have

$$(\mathscr{M} f)(\varphi)(y) \;=\; \sum_{x\in X} \eta_Y(\,f(x)\,)(y) \,\cdot\, \varphi(x)$$

for all $\varphi \in \mathscr{M} X$ and $y \in Y$.

(ii) Let X be a set. Let $x \in X$. Let $A, B\colon X \to C$. Then we have

$$\begin{aligned} A(x)\cdot B(x) \;&=\; \sum_{\tilde{x}\in X} A(\tilde{x})\cdot \eta_X(x)(\tilde{x})\cdot B(\tilde{x}) \\ &=\; \sum_{\tilde{x}\in X} A(\tilde{x})\cdot \eta_X(\tilde{x})(x)\cdot B(\tilde{x}). \end{aligned}$$

Lemma 31 *Let C be a σ-semiring.*

*(i) Definition 29(i) gives a functor $\mathscr{M}\colon$ **Sets** $\to$ **Sets**.*
(ii) Definition 29(ii) gives a natural transformation $\eta\colon 1 \to \mathscr{M}$.
(iii) Definition 29(iii) gives a natural transformation $\mu\colon \mathscr{M}\mathscr{M} \to \mathscr{M}$.

Proof. (i) To prove that Def. 29(i) gives us a functor $\mathscr{M}$ we must show that

$$\mathscr{M}\mathrm{id}_W \;=\; \mathrm{id}_{\mathscr{M}W} \qquad \text{and} \qquad \mathscr{M}(g\circ f) \;=\; \mathscr{M} g \circ \mathscr{M} f \tag{16}$$

for every set W, and for all maps $f\colon X \to Y$ and $g\colon Y \to Z$.

Let W be a set, let $\varphi \in \mathscr{M}W$, and let $w \in W$.
Then we have (see Remarks 30(i)),

$$\begin{aligned} (\mathscr{M}\mathrm{id}_W)(\varphi)(w) \;&=\; \sum_{\tilde{w}\in W} \eta_W(\tilde{w})(w)\cdot\varphi(\tilde{w}) \;=\; \varphi(w) \\ &=\; \mathrm{id}_{\mathscr{M}W}(\varphi)(w). \end{aligned}$$

So we see that $\mathscr{M}\mathrm{id}_W = \mathrm{id}_{\mathscr{M}W}$.

Let $f\colon X \to Y$ and $g\colon Y \to Z$ be functions. Let $\varphi \in \mathscr{M}X$ and $z \in Z$. Then

$$\begin{aligned} (\mathscr{M} g \circ \mathscr{M} f\,)(\varphi)(z) \;&=\; (\mathscr{M} g)(\,(\mathscr{M} f)(\varphi)\,)(z) \\ &=\; \sum_{y\in Y} \eta_Z(\,g(y)\,)(z) \,\cdot\, (\mathscr{M} f)(\varphi)(y) \\ &=\; \sum_{y\in Y} \eta_Z(\,g(y)\,)(z) \,\cdot\, \sum_{x\in X} \eta_Y(\,f(x)\,)(y) \,\cdot\, \varphi(x) \\ &=\; \sum_{y\in Y}\sum_{x\in X} \eta_Z(\,g(y)\,)(z) \,\cdot\, \eta_Y(\,f(x)\,)(y) \,\cdot\, \varphi(x) \\ &=\; \sum_{x\in X}\sum_{y\in Y} \eta_Z(\,g(y)\,)(z) \,\cdot\, \eta_Y(\,f(x)\,)(y) \,\cdot\, \varphi(x) \end{aligned}$$

By Remarks 30(ii) we get the following equality.

$$
\begin{aligned}
&= \sum_{x \in X} \eta_Z(\, g(f(x))\,)(z) \cdot \varphi(x) \\
&= \mathscr{M}(g \circ f)(\varphi)(z).
\end{aligned}
$$

So we see that $\mathscr{M} g \circ \mathscr{M} f = \mathscr{M}(g \circ f)$.

Hence we have obtained a functor $\mathscr{M}$: **Sets** $\to$ **Sets**.

(ii) To prove that Def. 29(ii) gives a natural transformation $\eta\colon 1 \to \mathscr{M}$, we must show that for every function $f\colon X \to Y$ the following diagram commutes.

$$
\begin{array}{ccc}
X & \xrightarrow{\eta_X} & \mathscr{M}X \\
{\scriptstyle f}\downarrow & & \downarrow{\scriptstyle \mathscr{M}f} \\
Y & \xrightarrow{\eta_Y} & \mathscr{M}Y
\end{array}
$$

That is, given $x \in X$ and $y \in Y$, we must show that

$$
(\mathscr{M} f)(\,\eta_X(x)\,)(y) \;=\; \eta_Y(\,f(x)\,)(y). \tag{17}
$$

Note that by Remarks 30(i) and Remarks 30(ii), we have,

$$
\begin{aligned}
(\mathscr{M} f)(\,\eta_X(x)\,)(y) &= \sum_{\tilde{x} \in X} \eta_Y(\,f(\tilde{x})\,)(y) \cdot \eta_X(x)(\tilde{x}) \\
&= \eta_Y(\,f(x)\,)(y).
\end{aligned}
$$

So Equation (17) holds. Hence we obtain a natural transformation $\eta\colon 1 \to \mathscr{M}$.

(iii) To prove that Def. 29(iii) gives a natural transf. $\mu\colon \mathscr{M}\mathscr{M} \to \mathscr{M}$, we must show that for every function $f\colon X \to Y$ the following diagram commutes.

$$
\begin{array}{ccc}
\mathscr{M}\mathscr{M}X & \xrightarrow{\mu_X} & \mathscr{M}X \\
{\scriptstyle \mathscr{M}\mathscr{M}f}\downarrow & & \downarrow{\scriptstyle \mathscr{M}f} \\
\mathscr{M}\mathscr{M}Y & \xrightarrow{\mu_Y} & \mathscr{M}Y
\end{array}
$$

That is, given $\Phi \in \mathscr{M}\mathscr{M}X$ and $y \in Y$, we must show that

$$
\mu_Y(\,(\mathscr{M}\mathscr{M} f)(\Phi)\,)(y) \;=\; (\mathscr{M} f)(\,\mu_X(\Phi)\,)(y). \tag{18}
$$

Let us is simplify the left-hand side of the above equation. We have

$$
\begin{aligned}
\mu_Y(\,(\mathscr{M}\mathscr{M} f)(\Phi)\,)(y) &= \sum_{\psi \in \mathscr{M}Y} (\mathscr{M}\mathscr{M} f)(\Phi)(\psi) \cdot \psi(y) \\
&= \sum_{\psi \in \mathscr{M}Y} \Big(\sum_{\varphi \in \mathscr{M}X} \eta_{\mathscr{M}Y}(\,(\mathscr{M} f)(\varphi)\,)(\psi) \cdot \Phi(\varphi) \Big) \cdot \psi(y) \\
&= \sum_{\varphi \in \mathscr{M}X} \sum_{\psi \in \mathscr{M}Y} \eta_{\mathscr{M}Y}(\,(\mathscr{M} f)(\varphi)\,)(\psi) \cdot \Phi(\varphi) \cdot \psi(y) \\
&= \sum_{\varphi \in \mathscr{M}X} \Phi(\varphi) \cdot (\mathscr{M} f)(\varphi)(y).
\end{aligned}
$$

On the other hand we have the following equalities.

$$\begin{aligned}
(\mathscr{M}f)(\,\mu_X(\Phi)\,)(y) &= \sum_{x\in X} \eta_Y(\,f(x)\,)(y)\,\cdot\,\mu_X(\Phi)(x) \\
&= \sum_{x\in X} \eta_Y(\,f(x)\,)(y)\,\cdot \sum_{\varphi\in\mathscr{M}X} \Phi(\varphi)\,\cdot\,\varphi(x) \\
&= \sum_{\varphi\in\mathscr{M}X}\sum_{x\in X} \eta_Y(\,f(x)\,)(y)\,\cdot\,\Phi(\varphi)\,\cdot\,\varphi(x) \\
&= \sum_{\varphi\in\mathscr{M}X}\sum_{x\in X} \Phi(\varphi)\,\cdot\,\eta_Y(\,f(x)\,)(y)\,\cdot\,\varphi(x) \\
&= \sum_{\varphi\in\mathscr{M}X} \Phi(\varphi)\,\cdot \sum_{x\in X} \eta_Y(\,f(x)\,)(y)\,\cdot\,\varphi(x) \\
&= \sum_{\varphi\in\mathscr{M}X} \Phi(\varphi)\,\cdot\,(\mathscr{M}f)(\varphi)(y).
\end{aligned}$$

So we see that both sides simplify to the same expression. Hence Equation (18) holds, and we have obtained a natural transformation $\mu\colon \mathscr{M}\mathscr{M} \to \mathscr{M}$.

Proposition 32 *The objects* $1 \xrightarrow{\eta} \mathscr{M} \xleftarrow{\mu} \mathscr{M}\mathscr{M}$ *form a monad in* **Sets**.

Proof. Let X be a set. We must show that the following diagrams commute.

$$\begin{tikzcd}
\mathscr{M}\mathscr{M}\mathscr{M}X \arrow[r, "\mathscr{M}\mu_X"] \arrow[d, "\mu_{\mathscr{M}X}"'] & \mathscr{M}\mathscr{M}X \arrow[d, "\mu_X"] \\
\mathscr{M}\mathscr{M}X \arrow[r, "\mu_X"'] & \mathscr{M}X
\end{tikzcd}
\qquad
\begin{tikzcd}
\mathscr{M}X \arrow[r, "\eta_{\mathscr{M}X}"] \arrow[d, "\mathscr{M}\eta_X"'] \arrow[dr, equal] & \mathscr{M}\mathscr{M}X \arrow[d, "\mu_X"] \\
\mathscr{M}\mathscr{M}X \arrow[r, "\mu_X"'] & \mathscr{M}X
\end{tikzcd}
\tag{19}$$

Let us first prove that the rightmost diagram of Display (19) commutes. That is, we must prove that the following two equalities hold.

$$\mu_X \circ \eta_{\mathscr{M}X} = \mathrm{id}_{\mathscr{M}X} \tag{20}$$

$$\mu_X \circ \mathscr{M}\eta_X = \mathrm{id}_{\mathscr{M}X} \tag{21}$$

We prove that Eq. (20) holds. Let $\varphi \in \mathscr{M}X$ and $x \in X$ be given. We have

$$\begin{aligned}
(\mu_X \circ \eta_{\mathscr{M}X})(\varphi)(x) &= \mu_X(\,\eta_{\mathscr{M}X}(\varphi)\,)(x) \\
&= \sum_{\tilde{\varphi}\in\mathscr{M}X} \eta_{\mathscr{M}X}(\varphi)(\tilde{\varphi})\,\cdot\,\tilde{\varphi}(x) \\
&= \varphi(x) \;=\; \mathrm{id}_{\mathscr{M}X}(\varphi)(x).
\end{aligned}$$

We now prove that Eq. (21) hold. Let $\varphi \in \mathscr{M}X$ and $x \in X$ be given. We have

$$\begin{aligned}
(\mu_X \circ \mathscr{M}\eta_X)(\varphi)(x) &= \mu_X(\,(\mathscr{M}\eta_X)(\varphi)\,)(x) \\
&= \sum_{\tilde{\varphi} \in \mathscr{M}X} (\mathscr{M}\eta_X)(\varphi)(\tilde{\varphi}) \cdot \tilde{\varphi}(x) \\
&= \sum_{\tilde{\varphi} \in \mathscr{M}X} \Big(\sum_{\tilde{x} \in X} \eta_{\mathscr{M}X}(\,\eta_X(\tilde{x})\,)(\tilde{\varphi}) \cdot \varphi(x) \Big) \cdot \tilde{\varphi}(x) \\
&= \sum_{\tilde{x} \in X} \sum_{\tilde{\varphi} \in \mathscr{M}X} \eta_{\mathscr{M}X}(\,\eta_X(\tilde{x})\,)(\tilde{\varphi}) \cdot \varphi(x) \cdot \tilde{\varphi}(x) \\
&= \sum_{\tilde{x} \in X} \varphi(x) \cdot \eta_X(\tilde{x})(x) \\
&= \varphi(x) \;=\; \mathrm{id}_{\mathscr{M}X}(\varphi)(x).
\end{aligned}$$

We have shown that the rightmost diagram of Display (19) commutes.

To prove that the leftmost diagram of (19) commutes, we must show that

$$\mu_X \circ \mu_{\mathscr{M}X} \;=\; \mu_X \circ (\mathscr{M}\mu_X). \tag{22}$$

Let $\Omega \in \mathscr{M}\mathscr{M}\mathscr{M}X$ and $x \in X$ be given. On the one hand, we have

$$\begin{aligned}
(\mu_X \circ \mu_{\mathscr{M}X})(\Omega)(x) &= \mu_X(\,\mu_{\mathscr{M}X}(\Omega)\,)(x) \\
&= \sum_{\varphi \in \mathscr{M}X} \mu_{\mathscr{M}X}(\Omega)(\varphi) \cdot \varphi(x) \\
&= \sum_{\varphi \in \mathscr{M}X} \Big(\sum_{\Phi \in \mathscr{M}\mathscr{M}X} \Omega(\Phi) \cdot \Phi(\varphi) \Big) \cdot \varphi(x) \\
&= \sum_{\Phi \in \mathscr{M}\mathscr{M}X} \sum_{\varphi \in \mathscr{M}X} \Omega(\Phi) \cdot \Phi(\varphi) \cdot \varphi(x).
\end{aligned}$$

On the other hand, we have

$$\begin{aligned}
(\mu_X \circ (\mathscr{M}\mu_X))(\Omega)(x) &= \mu_X(\,(\mathscr{M}\mu_X)(\Omega)\,)(x) \\
&= \sum_{\varphi \in \mathscr{M}X} (\mathscr{M}\mu_X)(\Omega)(\varphi) \cdot \varphi(x) \\
&= \sum_{\varphi \in \mathscr{M}X} \Big(\sum_{\Phi \in \mathscr{M}\mathscr{M}X} \eta_{\mathscr{M}X}(\,\mu_X(\Phi)\,)(\varphi) \cdot \Omega(\Phi) \Big) \cdot \varphi(x) \\
&= \sum_{\Phi \in \mathscr{M}\mathscr{M}X} \sum_{\varphi \in \mathscr{M}X} \eta_{\mathscr{M}X}(\,\mu_X(\Phi)\,)(\varphi) \cdot \Omega(\Phi) \cdot \varphi(x) \\
&= \sum_{\Phi \in \mathscr{M}\mathscr{M}X} \Omega(\Phi) \cdot \mu_X(\Phi)(x) \\
&= \sum_{\Phi \in \mathscr{M}\mathscr{M}X} \sum_{\varphi \in \mathscr{M}X} \Omega(\Phi) \cdot \Phi(\varphi) \cdot \varphi(x).
\end{aligned}$$

Hence we see that Equation (22) holds.

We have proven that together $\mathscr{M}$, μ and η are a monad.

D.1 Traces in the Kleisli Category of $\mathscr{M}$

Remark 33 *One can prove that the category $\mathcal{K}\ell(\mathscr{M})$ is cocomplete. In particular, $\mathcal{K}\ell(\mathscr{M})$ has all finite coproducts. We describe the finite coproducts below, since we will need this knowledge lateron.*

Let X and Y be sets. We will describe the coproduct of X and Y in $\mathcal{K}\ell(\mathscr{M})$.

But first, let $X + Y$ denote the coproduct of X and Y in **Sets**, *and let*

$$X \xrightarrow{\quad\kappa_\ell\quad} X + Y \xleftarrow{\quad\kappa_r\quad} Y$$

be the corresponding coprojections. Recall that for every set S and maps $f\colon X \to S$ and $g\colon Y \to S$, there is a unique map $[f, g]\colon X + Y \to S$, such that

$$[f, g] \circ \kappa_\ell \;=\; f \qquad \text{and} \qquad [f, g] \circ \kappa_r \;=\; g.$$

The coproduct of X and Y in $\mathcal{K}\ell(\mathscr{M})$ is simply $X + Y$, with coprojections,

$$X \xrightarrow{\quad\dot\kappa_\ell\quad} X + Y \xleftarrow{\quad\dot\kappa_r\quad} Y\ ,$$

where $\dot\kappa_\ell = \eta_{X+Y} \circ \kappa_\ell$ and $\dot\kappa_r = \eta_{X+Y} \circ \kappa_r$.

Let $f\colon X \longrightarrow S$ and $g\colon Y \longrightarrow S$. Then the unique map $h\colon X + Y \longrightarrow S$ such that $h \odot \dot\kappa_\ell = f$ and $h \odot \dot\kappa_r = g$ is simply $[f, g]\colon X + Y \to \mathscr{M} S$.

Recall that we get a functor $+$: **Sets** $\times$ **Sets** $\to$ **Sets** *if we set $f + g = [\kappa_\ell \circ f,\ \kappa_r \circ g]$, where f and g are maps between sets.*

Similarly, we can get a functor on $\mathcal{K}\ell(\mathscr{M})$ by defining $f \dot{+} g = [\dot\kappa_\ell \odot f,\ \dot\kappa_r \odot g]$, for given $f\colon X_1 \longrightarrow Y_1$ and $g\colon X_2 \longrightarrow Y_2$. One can prove that

$$(\, f \dot{+} g\,)(p)(q) \;=\; \begin{cases} f(x_1)(y_1) & \text{if } p \equiv \kappa_\ell(x_1) \text{ and } q \equiv \kappa_\ell(y_1) \\ g(x_2)(y_2) & \text{if } p \equiv \kappa_r(x_2) \text{ and } q \equiv \kappa_r(y_2) \\ 0 & \text{otherwise,} \end{cases} \tag{23}$$

where $p \in X_1 + X_2$ and $q \in Y_1 + Y_2$,

The following proposition shows that $\mathcal{K}\ell(\mathscr{M})$ 'has iterates', or more precisely, that $\mathcal{K}\ell(\mathscr{M})$ fits Setting 9.

Proposition 34 *Let $\omega\colon \mathbb{N} \cdot Y \to \mathscr{M}(\mathbb{N} \cdot Y + Y)$ be given by*

$$\omega(0, y)(\beta) \;=\; \begin{cases} 1 & \text{if } \beta = \kappa_r(y) \\ 0 & \text{otherwise} \end{cases} \qquad \omega(n+1, y)(\beta) \;=\; \begin{cases} 1 & \text{if } \beta = \kappa_\ell(n, y) \\ 0 & \text{otherwise} \end{cases}$$

where $y \in Y$, $n \in \mathbb{N}$ and $\beta \in Y \times \mathbb{N} \cdot Y$. Then, $\omega\colon \mathbb{N} \cdot Y \longrightarrow \mathbb{N} \cdot Y + Y$ is the final coalgebra in $\mathcal{K}\ell(\mathscr{M})$ for the functor

$$X \mapsto X + Y.$$

Proof. Let $c\colon X \longrightarrow\!\!\!\!\circ\;\; X + Y$ be given. To prove that ω is the final coalgebra for the functor $X \mapsto X + Y$ we must show that there is a unique $t\colon X \longrightarrow\!\!\!\!\circ\;\; \mathbb{N} \cdot Y$ such that the following diagram commutes.

$$\begin{tikzcd} X \arrow[r, "t"] \arrow[d, "c"'] & \mathbb{N} \cdot Y \arrow[d, "\omega"] \\ X + Y \arrow[r, "t \dot{+} \eta_Y"'] & \mathbb{N} \cdot Y + Y \end{tikzcd} \tag{24}$$

Let $t\colon X \longrightarrow\!\!\!\!\circ\;\; \mathbb{N} \cdot Y$ be given. We claim that Diagram (24) commutes if and only we have the following equalities.

$$\begin{aligned} t(x)(0, y) \;&=\; c(x)(\kappa_r(y)) \\ t(x)(n+1, y) \;&=\; \sum_{\tilde{x} \in X} t(\tilde{x})(n, y) \,\cdot\, c(x)(\kappa_\ell(\tilde{x})). \end{aligned} \tag{25}$$

To prove this claim we simply unfold the definitions.

To begin, note that Diagram (24) commutes if and only if we have

$$\omega \odot t \;=\; (t \dot{+} \eta_Y) \odot c.$$

Let $x \in X$ and $\beta \in \mathbb{N} \cdot Y + Y$ be given. Diagram (24) commutes if and only if

$$(\,\omega \odot t\,)(x)(\beta) \;=\; (\,(t \dot{+} \eta_Y\,) \odot c)(x)(\beta). \tag{26}$$

So to prove the claim it suffices to prove: Eq. (26) holds iff St. (25) holds.

Let us unfold the left-hand side of Equation (26). We have

$$(\omega \odot t)(x)(\beta) \;=\; \sum_{\alpha \in \mathbb{N} \cdot Y} \omega(\alpha)(\beta) \,\cdot\, t(x)(\alpha).$$

Let us apply the definition of ω. Note that $\omega(\alpha)(\beta) \in \{0, 1\}$. Informally, we have that $\omega(\alpha)(\beta) = 1$ if and only if $\beta \in \mathbb{N} \cdot Y + Y$ is the 'decomposition' of $\alpha \in \mathbb{N} \cdot Y$. More precisely, we have that $\omega(\alpha)(\beta) = 1$ if and only if *either* $\alpha \equiv (0, y)$ and $\beta = \kappa_r(y)$ *or* $\alpha \equiv (n+1, y)$ and $\beta = \kappa_\ell(n, y)$. Hence we get

$$(\omega \odot t)(x)(\beta) \;=\; \begin{cases} t(x)(0, y) & \text{if } \beta \equiv \kappa_r(y) \\ t(x)(n+1, y) & \text{if } \beta \equiv \kappa_\ell(n, y) \end{cases} \tag{27}$$

Let us unfold the right-hand side of Equation (26). We have

$$(\,(t \dot{+} \eta_Y) \odot c\,)(x)(\beta) \;=\; \sum_{p \in X + Y} (t \dot{+} \eta_Y)(p)(\beta) \,\cdot\, c(x)(p). \tag{28}$$

Let us consider $(t \dot{+} \eta_Y)(p)(\beta)$ where $p \in X + Y$. By Statement (23) we have

$$(t \dot{+} \eta_Y)(p)(\beta) \;=\; \begin{cases} t(\tilde{x})(n, y) & \text{if } p \equiv \kappa_\ell(\tilde{x}) \text{ and } \beta \equiv \kappa_\ell(n, y) \\ \eta_Y(\tilde{y})(y) & \text{if } p \equiv \kappa_r(\tilde{y}) \text{ and } q \equiv \kappa_r(y) \\ 0 & \text{otherwise.} \end{cases}$$

So if we expand the definition of η_Y we get

$$(t\dot{+}\eta_Y)(p)(\beta) \;=\; \begin{cases} t(\tilde{x})(n,y) & \text{if } p \equiv \kappa_\ell(\tilde{x}) \text{ and } \beta \equiv \kappa_\ell(n,y) \\ 1 & \text{if } p \equiv \kappa_r(y) \text{ and } q \equiv \kappa_r(y) \\ 0 & \text{otherwise.} \end{cases}$$

If we apply this knowledge to Equation (28), we get

$$\begin{aligned} &(\,(t\dot{+}\eta_Y)\odot c\,)(x)(\beta) \;= \\ &\qquad = \begin{cases} c(x)(\kappa_r(y)) & \text{if } \beta \equiv \kappa_r(y) \\ \sum_{\tilde{x}\in X}\; t(\tilde{x})(n,y)\,\cdot\, c(x)(\kappa_\ell(\tilde{x})) & \text{if } \beta \equiv \kappa_\ell(n,y). \end{cases} \end{aligned} \tag{29}$$

If we combine Statement (27) and Statement (29), we see that Equation (26) holds if and only if Statement (25) holds. Hence we have proven the claim, that Diagram (24) commutes if and only if Statement (25) holds.

Recall that we must prove that there is a unique $t\colon X \multimap \mathbb{N}\cdot Y$ such that Diagram (24) commutes. That is, we must prove that there is a unique $t\colon X \multimap \mathbb{N}\cdot Y$ such that Statement (25) holds (for all x, y, and n).

Let us first prove that there can be at most one such t. Let $t_1, t_2\colon X \multimap \mathbb{N}\cdot Y$ be given, and suppose that Statement (25) holds for $t = t_i$. We prove that $t_1 = t_2$.

For $n \in \mathbb{N}$, let $P(n)$ be the following statement. For all $x \in X$ and $y \in Y$,

$$t_1(x)(n,y) \;=\; t_2(x)(n,y).$$

It suffices to prove that $P(n)$ holds for all $n \in \mathbb{N}$. We do this with induction.

Note that since Statement (25) holds for $t = t_i$, we have

$$t_1(x)(0,y) \;=\; c(x)(\kappa_r(y)) \;=\; t_2(x)(0,y)$$

for all $x \in X$ and $y \in Y$. Hence we have $P(0)$.

Now, suppose that $P(n)$ holds in order to prove that $P(n+1)$ holds. We must prove that $t_1(x)(n+1,y) \;=\; t_2(x)(n+1,y)$, where $x \in X$, $y \in Y$ and $n \in \mathbb{N}$. We have

$$\begin{aligned} t_1(x)(n+1,y) \;&=\; \sum_{\tilde{x}\in X} t_1(\tilde{x})(n,y)\,\cdot\, c(x)(\kappa_\ell(\tilde{x})) && \text{by St. (25) for } t_1 \\ &=\; \sum_{\tilde{x}\in X} t_2(\tilde{x})(n,y)\,\cdot\, c(x)(\kappa_\ell(\tilde{x})) && \text{by } P(n) \\ &=\; t_2(x)(n+1,y) && \text{by St. (25) for } t_2. \end{aligned}$$

So we see that $P(n)$ holds for all $n \in \mathbb{N}$. Hence $t_1 = t_2$.

To complete the proof, we must show that there is an $t\colon X \multimap \mathbb{N}\cdot Y$ such that Statement (25) holds (for all $x \in X$, $y \in Y$ and $n \in \mathbb{N}$).

But before we find such $t\colon X \longrightarrow\!\!\!\!\!\circ\;\; \mathbb{N}\cdot Y$, let us rewrite Statement (25). Define $c_X\colon X \multimap X$ and $c_Y\colon X \multimap Y$ by

$$c_X(x)(\tilde{x}) \;=\; c(x)(\kappa_\ell(\tilde{x})) \qquad c_Y(x)(y) \;=\; c(x)(\kappa_r(y)).$$

Given $t\colon X \multimap \mathbb{N}\cdot Y$, define $t_n\colon X \multimap Y$ for $n\in\mathbb{N}$ by

$$t_n(x)(y) \;=\; t(x)(n,y).$$

If we write Statement (25) using t_n, c_X and c_Y, it becomes,

$$t_0 \;=\; c_Y \qquad t_{n+1} \;=\; t_n \odot c_X. \tag{30}$$

So it suffices to find a $t\colon X \multimap \mathbb{N}\cdot Y$ such that Statement (30) holds.

By recursion define $\tau_n\colon X \multimap Y$ for each $n\in\mathbb{N}$ by

$$\tau_0 \;=\; c_Y \qquad \tau_{n+1} \;=\; \tau_n \odot c_X.$$

Now define $t\colon X \to S^{\mathbb{N}\cdot Y}$ by: for all $x\in X$, $y\in Y$ and $n\in\mathbb{N}$ we have

$$t(x)(n,y) \;=\; \tau_n(x)(y)$$

It is clear that t will satisfy Statement (25). However, to complete the proof we need to check that $t(x)\in\mathscr{M}(\mathbb{N}\cdot Y)$ for all $x\in X$, so that $t\colon X \multimap \mathbb{N}\cdot Y$.

Let $x\in X$ be given. To prove $t(x)\in\mathscr{M}(X)$, we must show that $\operatorname{supp} t(x)$ is at most countable (see Definition 27). Note that by definition of t we have

$$\operatorname{supp} t(x) \;=\; \bigcup\,\{\ \operatorname{supp}\tau_n(x)\colon\ n\in\mathbb{N}\ \}.$$

Since $\tau_n\colon X \to \mathscr{M}Y$, we know that $\operatorname{supp}\tau_n(x)$ is at most for each $n\in\mathbb{N}$. Hence we see that $\operatorname{supp} t(x)$ is at most countable.

This completes the proof that the map $\omega\colon \mathbb{N}\cdot Y \multimap \mathbb{N}\cdot Y + Y$ gives us the final coalgebra of the functor $X\mapsto X+Y$ in the Kleisli category $\mathcal{K}\ell(\mathscr{M})$ of $\mathscr{M}$.

Proposition 35 *Let A be a set. Let $\xi\colon A^* \to \mathscr{M}(A\times A^* + 1)$ be given by*

$$\xi(w)(\kappa_r(*)) \;=\; \begin{cases} 1 & \textit{if } w=\varepsilon \\ 0 & \textit{otherwise}\end{cases} \quad \xi(w)(\kappa_\ell(a,w')) \;=\; \begin{cases} 1 & \textit{if } w=aw' \\ 0 & \textit{otherwise}\end{cases}$$

where $w,w'\in A^*$ and $a\in A$. Then $\xi\colon A^* \multimap A\times A^*+1$ is the final coalgebra in $\mathcal{K}\ell(\mathscr{M})$ for the functor $X\mapsto A\times X+1$.

Proof. Similar to the proof of Proposition 34.

E σ-Semirings

Definition 36 *A σ-**semiring** C is a semiring endowed with an operation that assigns to every map $f\colon I \to C$ an element of C which is denoted by*

$$\sum_{i\in I} f(i)$$

obeying the following axioms.

(i) Let $f\colon I\times J \to C$ be a map. Then we have

$$\sum_{i\in I}\sum_{j\in J} f(i,j) \;=\; \sum_{j\in J}\sum_{i\in I} f(i,j) \;=\; \sum_{(i,j)\in I\times J} f(i,j).$$

(ii) Let $f\colon I\to C$ be a map, and let $a,b\in C$. Then we have

$$a\cdot\Big(\sum_{i\in I} f(i)\Big)\cdot b \;=\; \sum_{i\in I} a\cdot f(i)\cdot b.$$

(iii) Let I and J be sets, and let $\alpha\colon J\to I$ be a bijection. Then for every $f\colon I\to C$ we have the following equality.

$$\sum_{i\in I} f(i) \;=\; \sum_{j\in J} f(\alpha(j)).$$

(iv) Let $f\colon I\to C$. Then we have the following equality.

$$\sum_{i\in I} f(i) \;=\; \sum_{i\in \operatorname{supp} f} f(i).$$

(v) Let $a_1,\dots,a_N\in C$ be given. Writing $\underline{N}=\{1,\dots,N\}$, we have

$$\sum_{n\in\underline{N}} a_n \;=\; a_1+\cdots+a_N.$$

Examples 37 *(i) The semiring $[0,+\infty]$ (see Examples 21(iii)) endowed with the summation one would expect is a σ-semiring.*

(ii) Recall that the tropical semiring $\mathscr{T}_{\min}$ (see Examples 21(iv)) is the semiring on $(-\infty,+\infty]$ with addition "min" and multiplication "+". We would like to turn $\mathscr{T}_{\min}$ into a σ-semiring by endowing it with the summation given by,

$$\sum_{i\in I} a_i \;=\; \bigwedge_{i\in I} a_i. \tag{31}$$

However, the infimum $\bigwedge_{i\in I} a_i$ might not exist; indeed, when $(a_i)_{i\in I}$ has no lower bound in $(-\infty,+\infty]$.
So instead, consider the semiring on the set $[-\infty,+\infty]$ with addition "min" and multiplication "+" (where we opt for the identity $-\infty+\infty=+\infty$). This semiring is a σ-semiring if we use the summation given by Equation (31).

(iii) We will see that the usual semiring on $\mathbb{R}$ can not be turned into a σ-semiring, because $\mathbb{R}$ is not positive (see Proposition 39).

Proposition 38 *Let C be a σ-semiring. Let X be a set. Let $f\colon I \to C$ be given. Let $\mathscr{J}$ be a partition of I. Then we have*

$$\sum_{i\in I} f(i) \;=\; \sum_{J\in\mathscr{J}} \sum_{j\in J} f(j).$$

Proof. Let $g\colon I \times \mathscr{J} \to C$ be given by

$$g(i,J) \;=\; \begin{cases} f(i) & \text{if } i \in J \\ 0 & \text{otherwise} \end{cases}.$$

Now apply Definition 36(i), We leave the details to the reader.

Not every sermiring can be turned into a σ-semiring:

Proposition 39 *Let C be a σ-semiring. Let $a, b \in C$. Then $a + b = 0$ implies $a = 0$ and $b = 0$.*

Proof. Informally, the argument is as follows.

$$a \;=\; a + (b+a) + (b+a) + \cdots \;=\; (a+b) + (a+b) + \cdots \;=\; 0.$$

We leave the details to the reader.

Definition 40 *A* ***positive semiring*** *S is a semiring such that, for all $a, b \in S$,*

$$a + b \;=\; 0 \quad\Longrightarrow\quad a = 0 \;\; \text{and} \;\; b = 0.$$

Examples 41 *(i) The usual semirings on $[0, +\infty)$ and on $[0, +\infty]$ are positive.*
(ii) The tropical semirings $\mathscr{T}_{\text{min}}$ and $\mathscr{T}_{\text{max}}$ are positive.
(iii) Any σ-semiring is positive (see Proposition 39).
(iv) The usual semiring on $\mathbb{R}$ is not *positive.*

Remark 42 *The term "positive" is taken from [6] by Gumm and Schröder in which they study "positive commutative monoids".*

Among other things, they prove that if the multiset functor $\mathscr{M}$ over a commutative moniod S preserves weak pullbacks, then S must be positive.

This gives some faith in the idea that a straightforward categorical approach to weighted automata is limited to positive semirings.

E.1 Sinks and Partial σ-Semirings

We have seen that any σ-semiring is positive. In particular, if a semiring S can be extended to a σ-semiring T (i.e. S is a subsemiring of T), then S is positive.

We will prove that the converse also holds. In fact, we will prove that if S is positive, and if we already have a sum $\sum_{i \in I} a_i$ for some families $(a_i)_{i \in I}$ (obeying some rules), then we can get a σ-semiring by adding one element, $*$.

More precisely, if S is a *partial σ-semiring* (see Def. 48) and if S is positive, then there is a σ-semiring on $S \cup \{*\}$ that extends S (see Prop. 50).

The element we add, $*$, will have the following property.

Definition 43 *Let S be a semiring.*
*A **sink** $*$ for S is an element of S such that for all $a \in S \backslash \{0\}$,*

$$* + a \;=\; * \;=\; a + * \qquad \textit{and} \qquad * \cdot a \;=\; * \;=\; a \cdot *. \tag{32}$$

Remark 44 *Note that in Def. 43 we do* not *require that St.* (32) *holds for $a = 0$.*

Examples 45 *(i) The element $+\infty$ in the usual semiring on $[0, +\infty]$ is a sink.*
(ii) $-\infty$ is a sink in the tropical semiring on $[-\infty, +\infty]$ (see Ex. 41(ii)).

Lemma 46 *Let S be a semiring with sink $*$. Then S is a positive semiring.*

Proof. Let $a, b \in S$ with $a + b = 0$ be given. We must prove that $a = 0$ and $b = 0$. Suppose that $a \neq 0$ and $b \neq 0$ in order to obtain a contradiction. We have

$$0 \;=\; * \cdot 0 \;=\; * \cdot (a + b) \;=\; * \cdot a + * \cdot b \;=\; * + * \;=\; *.$$

So we see that $0 = *$. Now, $a = a + 0 = a + * = * = 0$. Hence we see that $a = 0$, while we assumed that $a \neq 0$, which is absurd.

To a *positive* semiring we can add a sink.

Proposition 47 *Let S be a positive semiring. Let $*$ be some symbol.*
Set $S_* := S \cup \{*\}$. Then there is a unique way to extend the operations $+$ and $\cdot$ to S_* such that S_* is a semiring with sink $*$.

Proof. Extend the operations $+$ and $\cdot$ to S_* by the following rules.

$$\begin{aligned} * + 0 \;&=\; * \;=\; 0 + *, & * \cdot 0 \;&=\; 0 \;=\; 0 \cdot *, \\ * + * \;&=\; * \;=\; * + *, & * \cdot * \;&=\; * \;=\; * \cdot *, \\ * + a \;&=\; * \;= a + *, & * \cdot a \;&=\; * \;=\; a \cdot *, \end{aligned} \tag{33}$$

where $a \in S$ with $a \neq 0$. We will prove that then S_* is a semiring with sink $*$.

Before we prove that S_* is a semiring it is useful to note that we have the following equalities. (The equalities follow immediately from the definition.)

$$\begin{aligned} * + a \;&=\; * \;=\; a + * \qquad && (a \in S_*) \\ * \cdot a \;&=\; * \;=\; a \cdot * \qquad && (a \in S_* \backslash \{0\}) \\ 0 \cdot a \;&=\; 0 \;=\; a \cdot 0 \qquad && (a \in S_*). \end{aligned} \tag{34}$$

Let us prove that $+$ is associative. Given $a, b, c \in S_*$, we must show that

$$(a+b)+c \;=\; a+(b+c). \tag{35}$$

We have either $a, b, c \in S$ or $* \in \{a, b, c\}$.

Note that if $a, b, c \in S$, then Equation (35) holds because S is a semiring.

On the other hand, $* \in \{a, b, c\}$, then $(a+b)+c = *$. Similarly, $a+(b+c) = *$. Hence Equation (35) holds.

We have shown that $+$ is associative.

Let us prove that 0 is an identity element for $+$. Given $a \in S_*$ we must prove

$$0 + a \;=\; a \;= a + 0. \tag{36}$$

We have either $a = *$ or $a \in S$. If $a = *$, then Equation (36) holds because

$$* + 0 \;=\; * \;=\; 0 + *.$$

If $a \in S$, then Equation (36) holds, because S is a semiring.

Let us prove that $+$ is commutative. Let $a, b \in S_*$ be given. We must prove

$$a + b \;=\; b + a. \tag{37}$$

We have either $a, b \in S$ or $e = *$ for some $e \in \{a, b\}$.
If $a, b \in S$, then Equation (37) holds, because S is a semiring.
If $* \in \{a, b\}$, then $a + b = *$ and $b + a = *$, so Equation (37) holds.

Let us prove that $\cdot$ is associative. Let $a, b, c \in S_*$ be given. We must show

$$(a \cdot b) \cdot c \;=\; a \cdot (b \cdot c). \tag{38}$$

To prove this, we distinguish three (partially overlapping) cases.

(i) *Suppose* $a, b, c \in S$. Then Equation (38) holds because S is a semiring.

(ii) *Suppose* $0 \in \{a, b, c\}$. Then $(a \cdot b) \cdot c = 0$ and $a \cdot (b \cdot c) = 0$.
So we see that Equation (38) holds.

(iii) *Suppose* $* \in \{a, b, c\}$ *and* $\notin \{a, b, c\}$. Then Equation (38) holds, because

$$(a \cdot b) \cdot c \;=\; * \qquad \text{and} \qquad a \cdot (b \cdot c) \;=\; *.$$

Let us prove that 1 is an identity element for $\cdot$. Given $a \in S_*$ we must show

$$a \cdot 1 \;=\; a \;=\; 1 \cdot a. \tag{39}$$

Either $a \in S$ or $a = *$. If $a \in S$, then Equation (39) holds because S is a semiring. If $a = *$, then Equation (39) holds since $* \cdot 1 = * = 1 \cdot *$.

Let us prove that $\cdot$ distributes over $+$. Let $a, b, c \in S_*$ be given. To prove that $\cdot$ distributes over $+$ we must show that the following two equations hold.

$$a \cdot (b + c) \;=\; a \cdot b \;+\; a \cdot c \tag{40}$$

$$(b + c) \cdot a \;=\; b \cdot a \;+\; c \cdot a \tag{41}$$

We will only give a proof of Eq. (40) since the proof of Eq. (41) is similar.

To prove that Eq. (40) holds, we consider five (partially overlapping) cases.

(i) *Suppose* $a, b, c \in S$. Then Equation (40) holds because S is a semiring.

(ii) *Suppose* $a = 0$. Then Equation (40) holds because

$$0 \cdot (b + c) \;=\; 0 \;=\; 0 \cdot b \,+\, 0 \cdot c.$$

(iii) *Suppose* $b = 0$. Then Equation (40) holds since

$$a \cdot (0 + c) \;=\; a \cdot c \;=\; a \cdot 0 \,+\, a \cdot c.$$

(iv) *Suppose* $c = 0$. Then Equation (40) holds by a similar reasoning.

(v) *Suppose* $0 \notin \{a, b, c\}$ *and* $* \in \{a, b, c\}$. Then Equation (40) holds because

$$a \cdot (b + c) \;=\; * \;=\; a \cdot b \,+\, a \cdot c.$$

Note that $b + c \neq 0$, because $b \neq 0 \neq c$ and S is a positive semiring.

To prove that 0 is an absorbing element for $\cdot$ we must show that

$$0 \cdot a \;=\; 0 \;=\; 0 \cdot a \qquad (a \in S_*).$$

We have already proven this (see Equation (34)).

All in all, S_* is a semiring, and from Eq. (34) we see that $*$ is a sink for S_*.

So we have shown that there is a way to extend the operations $+$ and $\cdot$ of S to S_* in such a way that S_* is a semiring and $*$ is a sink for S_*.

It remains to be shown that there is *only one* way to extend $+$ and $\cdot$ to S_* such that S_* is a semiring and $*$ is a sink for S_*.

Let $+'$ and $\cdot'$ be extensions of the operations $+$ and $\cdot$ of S to $S \cup \{*\}$ such that $+'$ and $\cdot'$ turn $S \cup \{*\}$ into semiring, S'_*, and $*$ into a sink for S'_*.

To prove that there is only one way to extend the operations $+$ and $\cdot$ of S, etc., it suffices to show that $+'$ and $\cdot'$ coincide with the operations $+$ and $\cdot$ on $S \cup \{*\}$ we defined earlier using the rules from Statement (33).

It is easy to see that $+'$ and $\cdot'$ must also satisfy the rules of Statement (33), because S'_* is a semiring and $*$ is a sink for S'_*.

Hence $+'$ and $\cdot'$ coincide with $+$ and $\cdot$, respectively.

Definition 48 *A* ***partial*** $\boldsymbol{\sigma}$***-semiring*** C *is a semiring endowed with an operation which assigns to* some *of maps* $f \colon I \to C$ *an element of* C *denoted by*

$$\sum_{i \in I} f(i),$$

according to the axioms listed below. A map $f \colon I \to C$ *to which the operation assigns an element of* C *is called* ***summable***. *We will also say:* $\sum_{i \in I} f(i)$ *exists.*

(i) Let $f \colon I \times J \to C$ *be a map. Then* f *is summable if and only if:*

$$\sum_{j \in J} f(i,j) \;\; \text{exists for all } i, \qquad \text{and} \qquad \sum_{i \in I} \sum_{j \in J} f(i,j) \;\text{exists.}$$

Moreover, if f *is summable, then*

$$\sum_{i \in I} \sum_{j \in J} f(i,j) \;=\; \sum_{(i,j) \in I \times J} f(i,j).$$

(ii) *Let $f\colon I \to C$ be a map, and let $a, b \in C\backslash\{0\}$. Then*

$$\sum_{i\in I} f(i) \quad \textit{exists} \qquad \Longleftrightarrow \qquad \sum_{i\in I} a\cdot f(i)\cdot b \quad \textit{exists.}$$

Moreover, if f is summable, then

$$a\cdot\Big(\sum_{i\in I} f(i)\Big)\cdot b \;=\; \sum_{i\in I} a\cdot f(i)\cdot b.$$

(iii) *Let I and J be sets, and let $\alpha\colon J \to I$ be a bijection.*
Let $f\colon I \to C$ be summable. Then $f\circ\alpha\colon J \to C$ is summable, and

$$\sum_{i\in I} f(i) \;=\; \sum_{j\in J} f(\alpha(j)).$$

(iv) *Let I be a set, and let $f\colon I \to C$ be a function. Then*

$$\sum_{i\in I} f(i) \quad \textit{exists} \qquad \Longleftrightarrow \qquad \sum_{i\in \operatorname{supp} f} f(i) \quad \textit{exists.}$$

Moreover, if f is summable, then

$$\sum_{i\in I} f(i) \;=\; \sum_{i\in \operatorname{supp} f} f(i).$$

(v) *Let $a_1, \ldots, a_N \in C$ be given. Then $\sum_{n\in\underline{N}} a_n$ exists and*

$$\sum_{n\in\underline{N}} a_n \;=\; a_1 + \cdots + a_N,$$

where $\underline{N} := \{1, \ldots, N\}$.

Examples 49 *(i)* *The usual semiring on $[0, \infty)$ with the usual interpretation of "summable" and "$\sum_{i\in I} f(i)$" is a partial σ-semiring.*
(ii) *The tropical semiring $\mathscr{T}_{\min}$ on $(-\infty, +\infty]$ with $\sum_{i\in I}\ f(i) \;:=\; \bigwedge_{i\in I} f(i)$ whenever the infimum exists is a partial σ-semiring (see Examples 21(ii)).*

Proposition 50 *Let C be a* positive *partial σ-semiring.*
Then C_* (see Proposition 47) is a σ-semiring if we extend the $\sum$-operation on C to C_* as follows. Let $f\colon I \to C_*$ be given.

(i) *If f is summable (and $f(I) \subseteq C$), then we leave $\sum_{i\in I} f(i)$ as it is.*
(ii) *Otherwise, we set $\sum_{i\in I}\ f(i) = *$.*

Proof. Straightforward. We leave this to the reader.