Killing Epsilons with a Dagger: A Coalgebraic Study of Systems with Algebraic Label Structure

Filippo Bonchi[a], Stefan Milius[b,1], Alexandra Silva[c,d,e], Fabio Zanasi[a]

[a] *ENS Lyon, U. de Lyon, CNRS, INRIA, UCBL, France*
[b] *Lehrstuhl für Theoretische Informatik, FAU Erlangen-Nürnberg*
[c] *Institute for Computing and Information Sciences, Radboud University Nijmegen*
[d] *Centrum Wiskunde & Informatica (Amsterdam, The Netherlands)*
[e] *HASLab / INESC TEC, Universidade do Minho (Braga, Portugal)*

Abstract

We propose an abstract framework for modeling state-based systems with internal behaviour as e.g. given by silent or ϵ-transitions. Our approach employs monads with a parametrized fixpoint operator † to give a semantics to those systems and implement a sound procedure of abstraction of the internal transitions, whose labels are seen as the unit of a free monoid. More broadly, our approach extends the standard coalgebraic framework for state-based systems by taking into account the algebraic structure of the labels of their transitions. This allows to consider a wide range of other examples, including Mazurkiewicz traces for concurrent systems and non-deterministic transducers.

Keywords: coalgebras on Kleisli categories, parametrized fixpoint operator, trace semantics, epsilon transitions, Mazurkiewicz traces, non-deterministic transducers

1. Introduction

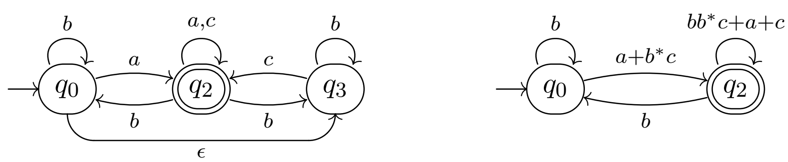
The theory of coalgebras provides an elegant mathematical framework to express the semantics of computing devices: the operational semantics, which

Email addresses: filippo.bonchi@ens-lyon.fr (Filippo Bonchi), mail@stefan-milius.eu (Stefan Milius), alexandra@cs.ru.nl (Alexandra Silva), fabio.zanasi@ens-lyon.fr (Fabio Zanasi)

[1]Stefan Milius acknowledges support by the Deutsche Forschungsgemeinschaft (DFG) under project MI 717/5-1

is usually given as a state machine, is modeled as a coalgebra for a functor; the denotational semantics as the unique map into the final coalgebra of that functor. While the denotational semantics is often *compositional* (as, for instance, ensured by the bialgebraic approach of [34]), it is sometimes not *fully-abstract*, i.e., it discriminates systems that are equal from the point of view of an external observer. This is due to the presence of internal transitions (also called ϵ-transitions) that are not observable but that are not abstracted away by the usual coalgebraic semantics using the unique homomorphism into the final coalgebra.

In this paper, we focus on the problem of giving trace semantics to systems with internal transitions. Our approach stems from an elementary observation (pointed out in previous work, e.g. [39]): the labels of transitions form a monoid and the internal transitions are those labeled by the unit of the monoid. Thus, there is an *algebraic structure* on the labels that needs to be taken into account when modeling the denotational semantics of those systems. To illustrate this point, consider the following two non-deterministic automata (NDA).



The one on the left (that we call $\mathbb{A}$) is an NDA with ϵ-transitions: its transitions are labeled either by the symbols of the alphabet $A = \{a, b, c\}$ or by the empty word $\epsilon \in A^*$. The one on the right (that we call $\mathbb{B}$) has transitions labeled by languages in $\mathcal{P}(A^*)$, here represented as regular expressions. The monoid structure on the labels is explicit on $\mathbb{B}$, while it is less evident in $\mathbb{A}$ since the set of labels $A \cup \{\epsilon\}$ does not form a monoid. However, this set can be trivially embedded into $\mathcal{P}(A^*)$ by looking at each symbols as the corresponding singleton language. For this reason each automaton with ϵ-transitions, like $\mathbb{A}$, can be regarded as an automaton with transitions labeled by languages, like $\mathbb{B}$. Furthermore, we can define the semantics of NDA with ϵ-transitions by defining the semantics of NDA with transitions labeled by languages: a word w is accepted by a state q if there is a path $q \xrightarrow{L_1} \cdots \xrightarrow{L_n} p$ where p is a final state, and there exist a decomposition $w = w_1 \cdots w_n$ such that $w_i \in L_i$. Observe that, with this definition, $\mathbb{A}$ and $\mathbb{B}$ accept the same language: all words over A that end with a or c. In fact, $\mathbb{B}$ was obtained from $\mathbb{A}$ in a well-known process to compute the regular expression denoting the language accepted by a given automaton [25].

We propose to define the semantics of systems with internal transitions following the same idea as in the above example. Given some transition type (i.e. an endofunctor) F, one first defines an embedding of F-systems with internal transitions into F^*-system, where F^* has been derived from F by making explicit the algebraic structure on the labels. Next one models the semantics of an F-system as the one of the corresponding F^*-system e. Naively, one could think of defining the semantics of e as the unique map $!_e$ into the final coalgebra for F^*. However, this approach turns out to be too fine grained, essentially because it ignores the underlying algebraic structure on the labels of e. The same problem can be observed in the example above: $\mathbb{B}$ and the representation of $\mathbb{A}$ as an automaton with languages as labels have different final semantics—they accept the same language only modulo the equations of monoids.

Thus we need to extend the standard coalgebraic framework by taking into account the algebraic structure on labels. To this end, we develop our theory for systems whose transition type F^* has a *canonical fixpoint*, i.e. its initial algebra and final coalgebra coincide. This is the case for many relevant examples, as observed in [23]. Our *canonical fixpoint semantics* will be given as the composite $¡ \circ !_e$, where $!_e$ is a coalgebra morphism given by finality and $¡$ is an algebra morphism given by initiality. The target of $¡$ will be an algebra for F^* encoding the equational theory associated with the labels of F^*-systems. Intuitively, $¡$ being an *algebra* morphism, will take the quotient of the semantics given by $!_e$ modulo those equations. Therefore the extension provided by $¡$ is the technical feature allowing us to take into account the algebraic structure on labels.

It were Simpson and Plotkin [38, Section 6] who realized that the above composites $¡ \circ !_e$ yield a *parametrized fixpoint operator* $e \mapsto e^\dagger$. This operator can be understood as assigning to systems of mutally recursive equations a certain *solution*, and the properties of $e \mapsto e^\dagger$ will be crucial for our canonical fixpoint semantics.

Within the same perspective we also consider a different fixpoint operator $e \mapsto e^\ddagger$ turning any system e with internal transitions into one $e^\ddagger$ where those have been abstracted away. By comparing the operators $e \mapsto e^\dagger$ and $e \mapsto e^\ddagger$, we are then able to show that such a procedure (also called *ϵ-elimination*) is sound with respect to the canonical fixpoint semantics.

We further explore the flexibility of our framework by modelling the case in which the algebraic structure of the labels is quotiented under some equations, resulting in a coarser equivalence than the one given by the canonical fixpoint semantics. As a relevant example of this phenomenon, we give the first coalgebraic account of Mazurkiewicz traces.

As our last application, we model non-deterministic transducers (with and without ϵ-transitions). This is a pleasing case study: on the one hand, it was known to be a hard problem to solve in the coalgebraic framework [21]; on the other hand, it follows as a simple application of our approach, thereby illustrating its power. In fact, as we observe, the only difference between transducers and non-deterministic automata is a change in the monad capturing the branching structure. In the NDA case, this is just non-determinism ($\mathcal{P}$, the powerset monad) whereas in the transducer case the monad needs to also capture the fact that transitions can output words ($\mathcal{P}(B^* \times \mathrm{Id})$, composition of the powerset and monoid action monads).

This paper is an extended and improved version of our CMCS'14 paper [10]. We have included all the proofs and the new example of non-deterministic transducers. We were also able to weaken the assumptions of our framework. In the conference version, Assumption 5.1 required the base category **C** to be **Cppo**-enriched and the monad T to be locally continuous. These assumptions ensure (a) initial algebra-final coalgebra coincidence for the functors $T(\mathrm{Id} + Y)$ and (b) that the canonical fixpoint operator $e \mapsto e^\dagger$ satisfies the so-called *double dagger law*. The latter is instrumental in our framework to correctly capture the semantics of systems with internal behaviour. Fortunately, it follows from the results of Simpson and Plotkin [38] that (a) and (b) hold whenever T has *enough canonical fixpoints*, in particular, no **Cppo**-enrichment and local continuity of T is needed.

Synopsis. After recalling the necessary background in Section 2, we discuss our motivating examples—automata with ϵ-transitions and automata on words—in Section 3. Section 4 and 5 are devoted to present the canonical fixpoint semantics and the sound procedure of ϵ-elimination. This framework is then instantiated to the examples of Section 3. In Section 6 we show how a quotient of the algebra on labels induces a coarser canonical fixpoint semantics. We propose Mazurkiewicz traces as a motivating example for such a construction. Finally, in Section 7 we apply our theory to give a coalgebraic modeling of non-deterministic transducers.

2. Preliminaries

In this section we introduce the basic notions we need for our abstract framework. We assume some familiarity with category theory. We will use boldface capitals $\mathbf{C}$ to denote categories, $X, Y, \ldots$ for objects and $f, g, \ldots$ for morphisms. We use Greek letters and double arrows, e.g. $\eta \colon F \Rightarrow G$, for natural transformations and monad morphisms. If $\mathbf{C}$ has coproducts we will denote them by $X + Y$ and use $\mathsf{inl}, \mathsf{inr}$ for the coproduct injections.

2.1. Monads

We recall the basics of the theory of monads, as needed here. For more information, see e.g. [30]. A monad is a functor $T \colon \mathbf{C} \to \mathbf{C}$ together with two natural transformations, a *unit* $\eta \colon \mathrm{id}_{\mathbf{C}} \Rightarrow T$ and a *multiplication* $\mu \colon T^2 \Rightarrow T$, which are required to satisfy the following equations, for every $X \in \mathbf{C}$: $\mu_X \circ \eta_{TX} = \mu_X \circ T\eta_X = \mathrm{id}_{TX}$ and $\mu_X \circ \mu_{TX} = \mu_X \circ T\mu_X$.

A *morphism of monads* from (T, η^T, μ^T) to (S, η^S, μ^S) is a natural transformation $\gamma \colon T \Rightarrow S$ that preserves unit and multiplication: $\gamma_X \circ \eta^T_X = \eta^S_X$ and $\gamma_X \circ \mu^T_X = \mu^S_X \circ \gamma_{SX} \circ T\gamma_X$. A *quotient of monads* is a morphism of monads with epimorphic components.

Example 2.1. We briefly describe the examples of monads that we use in this paper.

1. Let $\mathbf{C} = \mathbf{Sets}$. The powerset monad $\mathcal{P}$ maps a set X to the set $\mathcal{P}X$ of subsets of X, and a function $f \colon X \to Y$ to $\mathcal{P}f \colon \mathcal{P}X \to \mathcal{P}Y$ given by direct image. The unit is given by the singleton set map $\eta_X(x) = \{x\}$ and multiplication by union $\mu_X(U) = \bigcup_{S \in U} S$.

2. For later reference we introduce another monad on $\mathbf{Sets}$, namely $B^* \times \mathrm{Id}$. Its value on a set X is $B^* \times X$, where B^* is the set of words on a fixed B. The unit η_X maps $x \in X$ into $(\epsilon, x) \in B^* \times X$ and the multiplication μ_X maps $(w, v, x) \in B^* \times B^* \times X$ to $(wv, x) \in B^* \times X$.

3. Let $\mathbf{C}$ be a category with coproducts and E an object of $\mathbf{C}$. The exception monad $\mathcal{E}$ is defined on objects as $\mathcal{E}X = E + X$ and on arrows $f \colon X \to Y$ as $\mathcal{E}f = \mathrm{id}_E + f$. Its unit and multiplication are given on $X \in \mathbf{C}$ respectively as $\mathsf{inr}_X \colon X \to E + X$ and $\nabla_E + \mathrm{id}_X : E + E + X \to E + X$, where $\nabla_E = [\mathrm{id}_E, \mathrm{id}_E]$ is the codiagonal. When $\mathbf{C} = \mathbf{Sets}$, E can be thought as a set of *exceptions* and this monad is often used

to encode computations that might fail throwing an exception chosen from the set E.

4. Let H be an endofunctor on a category $\mathbf{C}$ such that for every object X there exists a free H-algebra H^*X on X (equivalently, an initial $H(-)+X$-algebra) with the structure $\tau_X : HH^*X \to H^*X$ and universal morphism $\eta_X : X \to H^*X$. Then as proved by Barr [8] (see also Kelly [28]) H^* is the object mapping of a *free monad* on H with the unit given by the above η_X and the multiplication given by the freeness of H^*H^*X: μ_X is the unique H-algebra homomorphism from (H^*H^*X, τ_{H^*X}) to (H^*X, τ_X) such that $\mu_X \circ \eta_{H^*X} = \mathrm{id}_{H^*X}$. Also notice that for a cocomplete category every free monad arises in this way. Finally, for later use we fix the notation $\kappa = \tau \circ H\eta \colon H \Rightarrow H^*$ for the universal natural transformation of the free monad.

Given a monad $M\colon \mathbf{C} \to \mathbf{C}$, its *Kleisli category* $\mathcal{K}\ell(M)$ has the same objects as $\mathbf{C}$, but morphisms $X \to Y$ in $\mathcal{K}\ell(M)$ are morphisms $X \to MY$ in $\mathbf{C}$. The identity map $X \to X$ in $\mathcal{K}\ell(M)$ is M's unit $\eta_X\colon X \to MX$; and composition $g \circ f$ in $\mathcal{K}\ell(M)$ uses M's multiplication: $g \circ f = \mu \circ Mg \circ f$. There is a forgetful functor $\mathcal{U}\colon \mathcal{K}\ell(T) \to \mathbf{C}$, sending X to TX and f to $\mu \circ Tf$. This functor has a left adjoint $\mathcal{J}$, given by $\mathcal{J}X = X$ and $\mathcal{J}f = \eta \circ f$. The Kleisli category $\mathcal{K}\ell(M)$ inherits coproducts from the underlying category $\mathbf{C}$. More precisely, for every objects X and Y their coproduct $X+Y$ in $\mathbf{C}$ is also a coproduct in $\mathcal{K}\ell(M)$ with the injections $\mathcal{J}\mathsf{inl}$ and $\mathcal{J}\mathsf{inr}$.

2.2. Distributive Laws and Liftings

The most interesting examples of the theory that we will present in Section 4 concern coalgebras for functors $\widehat{H}\colon \mathcal{K}\ell(M) \to \mathcal{K}\ell(M)$ that are obtained as liftings of endofunctors H on $\mathbf{Sets}$. Formally, given a monad $M\colon \mathbf{C} \to \mathbf{C}$, a *lifting* of $H\colon \mathbf{C} \to \mathbf{C}$ to $\mathcal{K}\ell(M)$ is an endofunctor $\widehat{H}\colon \mathcal{K}\ell(M) \to \mathcal{K}\ell(M)$ such that $\mathcal{J} \circ H = \widehat{H} \circ \mathcal{J}$. The lifting of a monad (T, η, μ) is a monad $(\widehat{T}, \widehat{\eta}, \widehat{\mu})$ such that $\widehat{T}$ is a lifting of T and $\widehat{\eta}$, $\widehat{\mu}$ are given on $X \in \mathcal{K}\ell(M)$ (i.e. $X \in \mathbf{Sets}$) respectively as $\mathcal{J}(\eta_X)$ and $\mathcal{J}(\mu_X)$.

A natural way of lifting functors and monads is by means of a distributive law. A *distributive law* of a monad (T, η^T, μ^T) over a monad (M, η^M, μ^M) is a natural transformation $\lambda\colon TM \Rightarrow MT$, that commutes appropriately with the unit and multiplication of both monads; more precisely, the diagrams below commute:

$$
\begin{array}{ccc}
TX & = & TX \\
{\scriptstyle T\eta^M_X}\downarrow & & \downarrow{\scriptstyle \eta^M_{TX}} \\
TMX & \xrightarrow{\lambda_X} & MTX \\
{\scriptstyle \eta^T_{MX}}\uparrow & & \uparrow{\scriptstyle M\eta^T_X} \\
MX & = & MX
\end{array}
\qquad
\begin{array}{ccccc}
TM^2X & \xrightarrow{\lambda_{MX}} & MTMX & \xrightarrow{M\lambda_X} & M^2TX \\
{\scriptstyle T\mu^M_X}\downarrow & & & & \downarrow{\scriptstyle \mu^M_{TX}} \\
TMX & & \xrightarrow{\lambda_X} & & MTX \\
{\scriptstyle \mu^T_{MX}}\uparrow & & & & \uparrow{\scriptstyle M\mu^T_X} \\
T^2MX & \xrightarrow{T\lambda_{MX}} & TMTX & \xrightarrow{\lambda_{TX}} & T^2MX
\end{array}
$$

A distributive law of a *functor* T over a *monad* (M, η^M, μ^M) is a natural transformation $\lambda\colon TM \Rightarrow MT$ such that only the two topmost squares above commute.

The following "folklore" result gives an alternative description of distributive laws in terms of liftings to Kleisli categories, see also [26], [33] or [7].

Proposition 2.2 ([33]). *Let (M, η^M, μ^M) be a monad on a category $\mathbf{C}$. Then the following holds:*

1. *For every endofunctor T on $\mathbf{C}$, there is a bijective correspondence between liftings of T to $\mathcal{K}\ell(M)$ and distributive laws of T over M.*

2. *For every monad (T, η^T, μ^T) on $\mathbf{C}$, there is a bijective correspondence between liftings of (T, η^T, μ^T) to $\mathcal{K}\ell(M)$ and distributive laws of T over M.*

In what follows we shall simply write $\widehat{H}\colon \mathcal{K}\ell(M) \to \mathcal{K}\ell(M)$ for the lifting of an endofunctor H for some distributive law $\lambda\colon HM \Rightarrow MH$. This can be explicitly given as $\widehat{H}(X) = HX$ for an object X and $\widehat{H}(f\colon X \to MY) = \lambda \circ Hf\colon HX \to MHX$ for a morphism $f\colon X \to Y$ in $\mathcal{K}\ell(M)$.

Example 2.3. Consider the powerset monad $\mathcal{P}$ (see Example 2.1.1) and the functor $HX = A \times X + 1$ on **Sets** (with $1 = \{\checkmark\}$). We can lift H to $\widehat{H}\colon \mathcal{K}\ell(\mathcal{P}) \to \mathcal{K}\ell(\mathcal{P})$ via the distributive law $\varphi\colon H\mathcal{P} \Rightarrow \mathcal{P}H$ defined as

$$
\begin{aligned}
\varphi_X\colon A \times \mathcal{P}X + 1 &\to \mathcal{P}(A \times X + 1) \\
\checkmark &\mapsto \{\checkmark\} \\
(a, Y) &\mapsto \{(a, y) \mid y \in Y\}
\end{aligned}
$$

More explicitly, the functor H lifts to $\widehat{H}$ on $\mathcal{K}\ell(\mathcal{P})$ as follows: for any $f\colon X \to Y$ in $\mathcal{K}\ell(\mathcal{P})$ (that is $f\colon X \to \mathcal{P}(Y)$ in **Sets**), $\widehat{H}f\colon A \times X + 1 \to A \times Y + 1$ is given by $\widehat{H}f(\checkmark) = \{\checkmark\}$ and $\widehat{H}f(\langle a, x\rangle) = \{\langle a, y\rangle \mid y \in f(x)\}$.

Proposition 2.4 ([23])**.** *Let $M\colon \mathbf{C} \to \mathbf{C}$ be a monad and $H\colon \mathbf{C} \to \mathbf{C}$ be a functor with a lifting $\widehat{H}\colon \mathcal{K}\ell(M) \to \mathcal{K}\ell(M)$. If H has an initial algebra $\iota\colon HI \stackrel{\cong}{\Rightarrow} I$ (in $\mathbf{C}$), then $\mathcal{J}\iota\colon \widehat{H}I \to I$ is an initial algebra for $\widehat{H}$ (in $\mathcal{K}\ell(M)$).*

In our examples, we will often consider the free monad $\widehat{H}^*$ generated by a lifted functor $\widehat{H}$ (cf. Example 2.1.4). From now on, whenever we write H^*, we will implicitly assume that for every object X a free H-algebra H^*X on X exists. This can be assured under mild assumptions on $\mathbf{C}$ and H, e.g. assuming that $\mathbf{C}$ is a locally presentable category and H an accessible functor (see e.g. Adámek and Rosický [5]). The following result (also appearing in [12, 13]) will be pivotal.

Proposition 2.5. *Let $H\colon \mathbf{C} \to \mathbf{C}$ be a functor and $M\colon \mathbf{C} \to \mathbf{C}$ be a monad such that there is a lifting $\widehat{H}\colon \mathcal{K}\ell(M) \to \mathcal{K}\ell(M)$. Then the free monad $H^*\colon \mathbf{C} \to \mathbf{C}$ lifts to a monad $\widehat{H^*}\colon \mathcal{K}\ell(M) \to \mathcal{K}\ell(M)$. Moreover, $\widehat{H^*} = \widehat{H}^*$.*

Proof. Let $\varphi\colon HM \to MH$ be the distributive law of the functor H over the monad M corresponding to the lifting $\widehat{H}$ (see Proposition 2.2). For an object X, we define $\lambda_X\colon H^*M \to H^*M$ by the universal property of the initial $H(-) + MX$-algebra $H^*(MX)$.

$$\begin{array}{ccccc} HH^*MX & \xrightarrow{\qquad\tau_{MX}\qquad} & H^*MX & \xleftarrow{\eta_{MX}} & MX \\ {\scriptstyle H\lambda_X}\downarrow & & \vdots\,{\scriptstyle \lambda_X} & \swarrow{\scriptstyle M\eta_X} & \\ HMH^*X \xrightarrow[\varphi_{TX}]{} MHH^*X & \xrightarrow[M\tau_X]{} & MH^*X & & \end{array} \tag{1}$$

By diagram chasing, one can show that $\lambda\colon H^*M \Rightarrow MH^*$ is a distributive law of the monad H^* over the monad M and, by Proposition 2.2, we have a lifting $\widehat{H^*}\colon \mathcal{K}\ell(M) \to \mathcal{K}\ell(M)$.

For proving $\widehat{H^*} = \widehat{H}^*$, take $\alpha_X\colon HH^*X + X \to H^*X$ to be the initial $H(-) + X$-algebra and observe that $\mathcal{J}(\alpha)$ is the initial $\widehat{H}(-) + X$-algebra (by Proposition 2.4). The fact that the units and the multiplications of $\widehat{H^*}$ and $\widehat{H}^*$ coincide is immediately proved by functoriality of $\mathcal{J}$. □

Remark 2.6. Recall from [23] that for every polynomial endofunctor H on **Sets** there exists a canonical distributive law of H over any *commutative* monad M (equivalently, a canonical lifting of H to $\mathcal{K}\ell(M)$); this result was later extended to so-called analytic endofunctors of **Sets** (see [31]). This provides a number of examples in which Propositions 2.4 and 2.5 apply and it can be used in our applications since the powerset functor $\mathcal{P}$ is commutative, and so is the exception monad $\mathcal{E}$ iff $E = 1$.

Example 2.7. Continuing Example 2.3 we see that the free monad on $HX = A \times X + 1$ is given by $H^*X = A^* \times X + A^*$. It is not difficult to verify that the distributive law $\lambda : H^*\mathcal{P} \Rightarrow \mathcal{P}H^*$ acts as follows

$$\begin{aligned} \lambda_X \colon A^* \times \mathcal{P}X + A^* &\to \mathcal{P}(A^* \times X + A^*) \\ w &\mapsto \{w\} \\ (w, Y) &\mapsto \{(w, y) \mid y \in Y\} \end{aligned}$$

for any $w \in A^*$ and $Y \in \mathcal{P}X$. Indeed, one readily verifies that this morphism λ_X makes diagram (1) commutative. Note that both H and H^* are polynomial functors; both φ and λ are the canonical distributive laws obtained from the results in [23] (see Remark 2.6).

2.3. **Cppo**-*Enriched Categories*

For our applications in $\mathcal{K}\ell(M)$ we are going to assume that the hom-sets of that category carry a cpo structure. Recall that a *cpo* is a partially ordered set in which all ω-chains have a join. A cpo with bottom is a cpo with a least element $\bot$. A function between cpos is called *continuous* if it preserves joins of ω-chains. Cpos with bottom and continuous maps form a category that we denote by **Cppo**.

A **Cppo**-*enriched category* $\mathbf{C}$ is a category where (a) each hom-set $\mathbf{C}(X, Y)$ is a cpo with a bottom element $\bot_{X,Y} \colon X \to Y$ and (b) composition is continuous, that is:

$$g \circ \left(\bigsqcup_{n<\omega} f_n \right) = \bigsqcup_{n<\omega} (g \circ f_n) \qquad \text{and} \qquad \left(\bigsqcup_{n<\omega} f_n \right) \circ g = \bigsqcup_{n<\omega} (f_n \circ g).$$

The composition is called *left strict* if $\bot_{Y,Z} \circ f = \bot_{X,Z}$ for all arrows $f \colon X \to Y$. An endofunctor $H \colon \mathbf{C} \to \mathbf{C}$ on a **Cppo**-enriched category $\mathbf{C}$ is said to be *locally continuous* if for any ω-chain $f_n \colon X \to Y$, $n < \omega$ in $\mathbf{C}(X, Y)$ we have:

$$H \left(\bigsqcup_{n<\omega} f_n \right) = \bigsqcup_{n<\omega} H(f_n).$$

We are going to make use of the fact that a locally continuous endofunctor H on $\mathbf{C}$ has a *canonical fixpoint*, i.e. whenever its initial algebra exists it is also its final coalgebra:

Theorem 2.8 ([18])**.** *Let $H\colon \mathbf{C} \to \mathbf{C}$ be a locally continuous endofunctor on the* **Cppo***-enriched category* **C** *whose composition is left-strict. If an initial H-algebra $\iota\colon HI \overset{\cong}{\Rightarrow} I$ exists, then $\iota^{-1}\colon I \overset{\cong}{\Rightarrow} HI$ is a final H-coalgebra.*

In the sequel, we will be interested in free algebras for a functor H on $\mathbf{C}$ and the free monad H^* (cf. Example 2.1.4). For this observe that coproducts in $\mathbf{C}$ are always **Cppo**-enriched, i.e. all copairing maps $[-,-] : \mathbf{C}(X,Y) \times \mathbf{C}(X',Y) \to \mathbf{C}(X+X',Y)$ are continuous; in fact, it is easy to show that this map is continuous in both of its arguments using that composition with the coproduct injections is continuous.

Proposition 2.9. *Let* **C** *be* **Cppo***-enriched with composition left-strict. Furthermore, let $H : \mathbf{C} \to \mathbf{C}$ be locally continuous and assume that all free H-algebras exist. Then the free monad H^* is locally continuous.*

Proof. First recall that H^*X is a free H-algebra with the structure τ_X and the universal morphism η_X (cf. Example 2.1.4). Equivalently, $\alpha_X = [\tau_X, \eta_X] : H(H^*X) + X \to H^*X$ is an initial algebra for $H(-)+X$. Given a morphism $f : X \to Y$, H^*f is defined by initiality; more precisely, H^*f is the unique morphism such that the following diagram commutes:

$$\begin{array}{ccccc}
H(H^*X)+X & & \xrightarrow{\quad\alpha_X\quad} & & H^*X \\
{\scriptstyle H(H^*f)+\mathrm{id}}\downarrow & & & & \downarrow{\scriptstyle H^*f} \\
H(H^*Y)+X & \xrightarrow[\mathrm{id}+f]{} & H(H^*Y)+Y & \xrightarrow[\alpha_Y]{} & H^*Y
\end{array}$$

Now recall that α_X is an isomorphism by Lambek's lemma and consider the following function

$$\Phi : \mathbf{C}(X,Y) \times \mathbf{C}(H^*X, H^*Y) \to \mathbf{C}(H^*X, H^*Y)$$

with

$$\Phi(f,h) = \alpha_Y \cdot (Hh + f) \cdot \alpha_X^{-1}.$$

Since H is locally continuous, we see that Φ is continuous (in both arguments). Clearly, H^*f is the unique fixpoint of $\Phi(f,-)$. To see that H^*f is locally continuous let $f_n : X \to Y$ be an ω-chain in $\mathbf{C}(X,Y)$. It is easy to see that $\bigsqcup_{n<\omega} H^*f_n$ is a fixpoint of $\Phi\left(\bigsqcup_{n<\omega} f_n, -\right)$; indeed we have (using continuity of Φ):

$$\begin{aligned}\bigsqcup_{n<\omega} H^* f_n &= \bigsqcup_{n<\omega} \Phi(f_n, H^* f_n) \\ &= \Phi\left(\bigsqcup_{n<\omega} f_n, \bigsqcup_{n<\omega} H^* f_n\right).\end{aligned}$$

Thus, by the uniqueness of the fixpoint $H^*\left(\bigsqcup_{n<\omega} f_n\right)$ we have

$$H^*\left(\bigsqcup_{n<\omega} f_n\right) = \bigsqcup_{n<\omega} H^* f_n$$

as desired. □

2.4. Final Coalgebras in Kleisli categories

As we mentioned already, in our applications the **Cppo**-enriched category will be the Kleisli category $\mathbf{C} = \mathcal{K}\ell(M)$ of a monad on **Sets** and the endofunctors of interest are liftings $\widehat{H}$ of endofunctors H on **Sets**. It is known that in this setting a final coalgebra for the lifting $\widehat{H}$ can be obtained as a lifting of an initial H-algebra (see Hasuo et al. [23]). Recall e.g. from [5] that an endofunctor H on **Sets** is *accessible* if it preserves λ-filtered colimits for some cardinal λ; equivalently, H is *bounded*, i.e. there exists a cardinal λ such that for every $x \in HX$ there exists a subset $m : Y \hookrightarrow X$ and $y \in Y$ with $|Y| < \lambda$ and such that $x = Hm(y)$ (see Adámek and Porst [4]). Many endofunctors of interest are accessible: basic examples are constant functors, the identity functor and the finite power-set functor; moreover, accessible functors are closed under coproducts, finite products and composition. The (full) power-set functor is a notable example of an endofunctor that is not accessible.

The following result is a variation of Theorem 3.3 in [23]:

Theorem 2.10. *Let M :* **Sets** $\to$ **Sets** *be a monad and H :* **Sets** $\to$ **Sets** *be a functor such that*

(a) $\mathcal{K}\ell(M)$ is **Cppo**-*enriched with composition left strict;*

(b) H is accessible and has a lifting $\widehat{H} : \mathcal{K}\ell(M) \to \mathcal{K}\ell(M)$ which is locally continuous.

If $\iota\colon HI \stackrel{\cong}{\Rightarrow} I$ is the initial algebra for the functor H, then

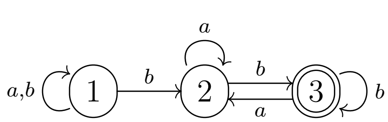
1. *$\mathcal{J}\iota\colon \widehat{H}I \to I$ is the initial algebra for the functor $\widehat{H}$;*
2. *$\mathcal{J}\iota^{-1}\colon I \to \widehat{H}I$ is the final coalgebra for the functor $\widehat{H}$.*

The first item follows from Proposition 2.4 and the second one follows from Theorem 2.8. There are two differences with Theorem 3.3 in [23]:

(1) There the functor $H\colon \mathbf{Sets} \to \mathbf{Sets}$ is supposed to preserve ω-colimits rather than being accessible. We use the assumption of accessibility because it guarantees the existence of all free algebras for H and for $\widehat{H}$, which implies also that for all $Y \in \mathcal{K}\ell(M)$ an initial $\widehat{H}^*(\mathrm{Id} + Y)$-algebra exists. This property of $\widehat{H}^*$ will be needed for applying our framework of Section 4.

(2) We assume that the lifting $\widehat{H}\colon \mathcal{K}\ell(M) \to \mathcal{K}\ell(M)$ is locally continuous rather than locally monotone. This assumption is not really restrictive since, as explained in Section 3.3.1 of [23], in all the meaningful examples where $\widehat{H}$ is locally monotone, it is also locally continuous. On the other hand this stronger assumption allows us to replace preservation of ω-colimits by accessibility of H.

Example 2.11 (NDA)**.** Non-deterministic automata (NDA) over the input alphabet A can be regarded as coalgebras for the functor $\widehat{H}\colon \mathcal{K}\ell(\mathcal{P}) \to \mathcal{K}\ell(\mathcal{P})$ in Example 2.3. Consider, on the left, a 3-state NDA, where the only final state is marked by a double circle.



$$X = \{1,2,3\} \quad A = \{a,b\}$$
$$e(1) = \{\langle a,1\rangle, \langle b,1\rangle, \langle b,2\rangle\}$$
$$e(2) = \{\langle a,2\rangle, \langle b,3\rangle\} \quad e(3) = \{\checkmark, \langle a,2\rangle, \langle b,3\rangle\}$$

It can be represented as a coalgebra $e\colon X \to \widehat{H}X$, that is a function $e\colon X \to \mathcal{P}(A \times X + 1)$, given above on the right, which assigns to each state $x \in X$ a set which: contains $\checkmark$ if x is final; and $\langle a, y\rangle$ for all transitions $x \stackrel{a}{\to} y$.

It is easy to see that $M = \mathcal{P}$ and H above satisfy the conditions of Theorem 2.10 and therefore both the final $\widehat{H}$-coalgebra and the initial $\widehat{H}$-algebra are the lifting of the initial algebra for the functor $HX = A \times X + 1$, given by A^* with structure $\iota\colon A \times A^* + 1 \to A^*$ which maps $\langle a, w\rangle$ to aw and ✓ to ϵ.

For an NDA (X, e), the final coalgebra homomorphism $!_e\colon X \to A^*$ is the function $X \to \mathcal{P}A^*$ that maps every state in X to the language that it accepts. In $\mathcal{K}\ell(\mathcal{P})$:

$$
\begin{array}{ccc}
X & \overset{!_e}{\dashrightarrow} & A^* \\
{\scriptstyle e}\big\downarrow & \begin{array}{rcl} \epsilon \in\, !_e(x) & \Leftrightarrow & \checkmark \in e(x) \\ aw \in\, !_e(x) & \Leftrightarrow & \text{for some } y \in X,\ (a, y) \in e(x) \text{ and } w \in\, !_e(y) \end{array} & \big\downarrow{\scriptstyle \mathcal{J}\iota^{-1}} \\
A \times X + 1 & \underset{\widehat{A \times !_e + 1}}{\dashrightarrow} & A \times A^* + 1
\end{array}
\qquad (2)
$$

2.5. Monads with Fixpoint Operators

In order to develop our theory of systems with internal behaviour, we will adopt an equational perspective on coalgebras. In the sequel we recall some preliminaries on this viewpoint.

Let $T : \mathbf{C} \to \mathbf{C}$ be a monad on any category $\mathbf{C}$. Any morphism $e : X \to T(X+Y)$ (i.e. a coalgebra for the functor $T(\mathrm{Id}+Y)$) may be understood as a system of mutually recursive equations. In our applications we are interested in the case where $\mathbf{C} = \mathcal{K}\ell(M)$ and $T = \widehat{H^*}$ is a (lifted) free monad. As in the example of NDA (Example 2.11) take $M = \mathcal{P}$ and $HX = 1 + A \times X$, meaning that $TX = \widehat{H^*}X = A^* + A^* \times X$. Now, consider the following system of mutually recursive equations

$$x_0 \approx \{c, (ab, x_1)\}, \qquad x_1 \approx \{d, (a, x_0), (\epsilon, y)\},$$

where $x_0, x_1 \in X$ are *recursion variables*, $y \in Y$ is a *parameter* and a, b, c, d are elements of A. The right hand side of each equation is an element of $\mathcal{P}(T(X + Y))$. A *solution* assigns to each of the two variables x_0, x_1 an element of $\mathcal{P}(TY)$ such that the two equations hold true:

$$
\begin{aligned}
x_0 &\mapsto (aba)^*c \cup (aba)^*abd \cup \{(w, y) \mid w \in (aba)^*ab\} \\
x_1 &\mapsto (aab)^*d \cup (aab)^*ac \cup \{(w, y) \mid w \in (aab)^*\}.
\end{aligned}
$$

The above system of equations corresponds to an *equation morphism* $e\colon X \to T(X + Y)$ and the solution to a morphism $e^\dagger\colon X \to TY$, both in $\mathcal{K}\ell(M)$.

The property that $e^\dagger$ is a solution for e is expressed by the following equality in $\mathcal{K}\ell(M)$:

$$e^\dagger = (X \xrightarrow{e} T(X+Y) \xrightarrow{T[e^\dagger, \eta_Y^T]} TTY \xrightarrow{\mu_Y^T} TY). \tag{3}$$

Definition 2.12 (Following Simpson and Plotkin [38])**.** Given any monad T on $\mathbf{C}$ a *parametrized fixpoint operator* is a family of maps $\mathbf{C}(X, T(X+Y)) \to \mathbf{C}(X, TY)$, $e \mapsto e^\dagger$, indexed by parameter objects Y such that for every $e : X \to T(X+Y)$, (3) hold.

Remark 2.13. In our applications we shall need a certain equational property of the operator $e \mapsto e^\dagger$ called *double dagger law*: for all $Y \in \mathbf{C}$ and equation morphism $e : X \to T(X+X+Y)$,

$$e^{\dagger\dagger} = (X \xrightarrow{e} T(X+X+Y) \xrightarrow{T(\nabla_X + Y)} T(X+Y))^\dagger.$$

This and other laws of parametrized fixpoint operators have been studied by Bloom and Ésik in the context of *iteration theories* [9].A closely related notion is that of *Elgot monads* [2, 3]. The parametrized fixpoint operators that we introduce in Section 4 will satisfy the double dagger law by construction (Theorem 4.3).

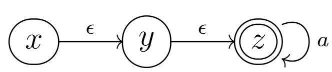
Example 2.14 (Least fixpoint solutions)**.** Observe that in the above example of NDA every equation morphism e has a least solution $e^\dagger$. More generally, let $T : \mathbf{C} \to \mathbf{C}$ be a locally continuous monad on the **Cppo**-enriched category $\mathbf{C}$. Then T is equipped with a parametrized fixpoint operator obtained by taking least fixpoints: given a morphism $e : X \to T(X+Y)$ consider the function Φ_e on $\mathbf{C}(X, TY)$ given by $\Phi_e(s) = \mu_Y^T \circ T[s, \eta_Y^T] \circ e$. Then Φ_e is continuous and we take $e^\dagger$ to be the least fixpoint of Φ_e. Since $e^\dagger = \Phi_e(e^\dagger)$, equation (3) holds, and hence we have a parametrized fixpoint operator. Moreover it follows from the argument in Theorem 8.2.15 and Exercise 8.2.17 in [9] that the operator $e \mapsto e^\dagger$ satisfies the axioms of iteration theories (or Elgot monads, respectively). In particular the double dagger law holds for the least fixpoint operator $e \mapsto e^\dagger$.

3. Motivating examples

The work of [23] bridged a gap in the theory of coalgebras: for certain functors, taking the final coalgebra directly in **Sets** does not give the right notion of equivalence on states of coalgebras. For instance, for NDA, one would obtain bisimilarity instead of language equivalence. The change to Kleisli categories allowed the recovery of the usual language semantics for NDA and, more generally, led to the development of *coalgebraic trace semantics*.

In the Introduction we argued that there are relevant examples for which this approach still does not yield the right notion of equivalence, the problem being that it does not consider the extra algebraic structure on the label set. In the sequel, we motivate the reader for the generic theory we will develop by detailing two case studies in which this phenomenon can be observed: NDA with ϵ-transitions and NDA with word transitions. Later on, in Example 6.7, we will also consider Mazurkiewicz traces [29].

NDA with ϵ-transitions. In the world of automata, ϵ-transitions are considered in order to enable easy composition of automata and compact representations of languages. These transitions are to be interpreted as the empty word when computing the language accepted by a state. Consider, on the left, the following simple example of an NDA with ϵ-transitions, where states x and y just make ϵ transitions. The intended semantics in this example is that all states accept words in a^*.

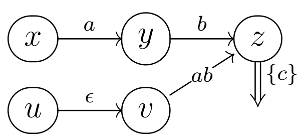


$$\begin{aligned} e(x) &= \{(\epsilon, y)\} & \quad !_e(x) &= \epsilon\epsilon a^* \\ e(y) &= \{(\epsilon, z)\} & \quad !_e(y) &= \epsilon a^* \\ e(z) &= \{(a, z), \checkmark\} & \quad !_e(z) &= a^* \end{aligned}$$

Note that, more explicitly, these are just NDA where the alphabet has a distinguished symbol ϵ. So, they are coalgebras for the functor $\widehat{H + \mathrm{Id}} \colon \mathcal{K}\ell(\mathcal{P}) \to \mathcal{K}\ell(\mathcal{P})$ (where H is the functor of Example 2.11), i.e. functions $e \colon X \to \mathcal{P}((A \times X + 1) + X) \cong \mathcal{P}((A + 1) \times X + 1)$, as made explicit for the above automaton in the middle.

The final coalgebra for $\widehat{H + \mathrm{Id}}$ is simply $(A + 1)^*$ and the final map $!_e \colon X \to (A + 1)^*$ assigns to each state the language in $(A + 1)^*$ that it accepts. However, the equivalence induced by $!_e$ is too fine grained: for the automata above, $!_e$ maps x, y and z to three different languages (on the right), where the number of ϵ plays an explicit role, but the intended semantics should disregard ϵ's.

NDA with word transitions. This is a variation on the motivating example of the introduction: instead of languages, transitions are labeled by words[2]. Formally, consider again the functor H from Example 2.11. Then NDA with word transitions are coalgebras for the functor $\widehat{H^*}\colon \mathcal{K}\ell(\mathcal{P}) \to \mathcal{K}\ell(\mathcal{P})$, that is, functions $e\colon X \to \mathcal{P}(A^* \times X + A^*) \cong \mathcal{P}(A^* \times (X + 1))$. We observe that they are like NDA but (1) transitions are labeled by words in A^*, rather than just symbols of the alphabet A, and (2) states have associated output languages, rather than just ✓. We will draw them as ordinary automata plus an arrow $\overset{L}{\Rightarrow}$ to denote the output language of a state (no $\Rightarrow$ stands for the empty language). For example, consider the following word automaton and associated transition function e.



$$e(x) = \{(a, y)\} \quad e(y) = \{(b, z)\} \quad e(z) = \{c\}$$
$$e(u) = \{(\epsilon, v)\} \quad e(v) = \{(ab, z)\}$$

The semantics of NDA with word transitions is given by languages over A, obtained by concatenating the words in the transitions and ending with a word from the output language. For instance, x above accepts word abc but not ab.

However, if we consider the final coalgebra semantics we again have a mismatch. The initial H^*-algebra has carrier $(A^*)^* \times A^*$ that can be represented as the set of non-empty lists of words over A^*, where $(A^*)^*$ indicates possibly empty lists of words. Its structure $\iota\colon A^* \times ((A^*)^* \times A^*) + A^* \to (A^*)^* \times A^*$ maps w into $(\langle\rangle, w)$ and $(w', (l, w))$ into $(w' :: l, w)$. Here, we use $\langle\rangle$ to denote the empty list and $::$ is the append operation. By Theorem 2.10, the final $\widehat{H^*}$-coalgebra has the same carrier and structure $\mathcal{J}\iota^{-1}$. The final map, as a function $!_e\colon X \to \mathcal{P}((A^*)^* \times A^*)$, is then defined by commutativity of the following square (in $\mathcal{K}\ell(\mathcal{P})$):

$$\begin{array}{ccc}
X & \xrightarrow{\quad !_e \quad} & (A^*)^* \times A^* \\
{\scriptstyle e}\big\downarrow & \begin{array}{c} (\langle\rangle, w) \in !_e(x) \;\Leftrightarrow\; w \in e(x) \\ (w :: l, w') \in !_e(x) \;\Leftrightarrow\; \exists y\; (w, y) \in e(x) \text{ and } (l, w') \in !_e(y) \end{array} & \big\downarrow{\scriptstyle \mathcal{J}\iota^{-1}} \\
A^* \times X + A^* & \xrightarrow[\mathrm{id}_{A^*} \times !_e + \mathrm{id}_{A^*}]{} & A^* \times ((A^*)^* \times A^*) + A^*
\end{array} \qquad (4)$$

[2]More generally, one could consider labels from an arbitrary monoid.

Once more, the semantics given by $!_e$ is too fine grained: in the above example, $!_e(x) = \{([a,b],c)\}$ and $!_e(u) = \{([\epsilon, ab], c)\}$ whereas the intended semantics would equate both x and u, since they both accept the language $\{abc\}$.

Note that any NDA can be regarded as word automaton. Recall the natural transformation $\kappa\colon \widehat{H} \Rightarrow \widehat{H^*}$ defined in Example 2.1.4: for the functor $\widehat{H}$ of NDA,

$$\kappa_X\colon A \times X + 1 \to A^* \times X + A^*$$

maps any pair $(a,x) \in A \times X$ into $\{(a,x)\} \in \mathcal{P}(A^* \times X + A^*)$ and $\checkmark \in 1$ into $\{\epsilon\} \in \mathcal{P}(A^* \times X + A^*)$. Composing an NDA $e\colon X \to \widehat{H}X$ with $\kappa_X\colon \widehat{H}X \to \widehat{H^*}X$, one obtains the word automaton $\kappa_X \circ e$.

In the same way, every NDA with ϵ-transitions can also be seen as a word automaton by postcomposing with the natural transformation

$$[\kappa, \eta]\colon \widehat{H + \mathrm{Id}} \Rightarrow \widehat{H^*}.$$

Here, $\eta\colon \mathrm{Id} \Rightarrow \widehat{H^*}$ is the unit of the free monad $\widehat{H^*}$ defined on a given set X below (the multiplication $\mu\colon \widehat{H^*}\widehat{H^*} \Rightarrow \widehat{H^*}$ is shown on the right).

$$\begin{array}{ll} \eta_X\colon X \to A^* \times X + A^* & \mu_X\colon A^* \times ((A^* \times X + A^*) + A^* \to A^* \times X + A^* \\ x \mapsto \{(\epsilon, x)\} & (w,(w',x)) \mapsto \{(w \cdot w', x)\} \quad (w, w') \mapsto \{w \cdot w'\} \\ & w \mapsto \{w\} \end{array}$$

In the next section, we propose to define the semantics of $\widehat{H^*}$-coalgebras via a canonical fixpoint operator rather than with the final map which as we saw above might yield unwanted semantics. Then, using the observation above, the semantics of $\widehat{H}$-coalgebras and $\widehat{H + \mathrm{Id}}$-coalgebras will be defined by embedding them into $\widehat{H^*}$-coalgebras via the natural transformations κ and $[\kappa, \eta]$ described above.

4. Canonical Fixpoint Solutions

In this section we present the foundations of our approach. We recall here a construction assigning canonical solutions to coalgebras seen as equation morphisms (*cf.* Section 2.5) which (in the dual setting) is due to Simpson and Plotkin [38]. A functor T on a category $\mathbf{C}$ is said to have *sufficiently many canonical fixpoints* if for every object Y all functors $T(\mathrm{Id} + Y)$, $T(T(\mathrm{Id} + Y) + Y)$ and $T(\mathrm{Id} + \mathrm{Id} + Y)$ have canonical fixpoints. We will be working under the following assumptions.

Assumption 4.1. Let $\mathbf{C}$ be a category with coproducts and let T be a monad on $\mathbf{C}$ having sufficiently many canonical fixpoints.

In Section 5 we will see that these assumptions are satisfied for certain monads arising from the lifted functor $\widehat{H}$ in Theorem 2.10. This will allow us to apply the theory developed in this section to the motivating examples of Section 3 and the non-deterministic transducer in Section 7.

Remark 4.2. Recall that, by Freyd's iterated square theorem [17], for an endofunctor $H : \mathbf{C} \to \mathbf{C}$ an initial algebra for HH yields one for H. Conversely, one can show that if $\mathbf{C}$ has binary products then an initial algebra for H yields one for HH. Thus, assuming the existence of binary products and coproducts in $\mathbf{C}$, we see that canonical fixpoints for $T(\mathrm{Id} + X)$ exist iff they exist for $T(T(\mathrm{Id} + X) + X)$ (and moreover, these canonical fixpoints are carried by the same object). However, to our knowledge it is not known whether existence of canonical fixpoints for $T(\mathrm{Id} + \mathrm{Id} + X)$ can be obtained from their existence for $T(\mathrm{Id} + X)$ (under some reasonable assumptions).

For every parameter object $Y \in \mathbf{C}$, denote the initial algebra and final coalgebra for $T(\mathrm{Id} + Y)$-algebra by

$$\iota_Y : T(I_Y + Y) \xrightarrow{\cong} I_Y \qquad \text{and} \qquad \iota_Y^{-1} : I_Y \xrightarrow{\cong} T(I_Y + Y).$$

Then to any equation morphism $e : X \to T(X + Y)$ we associate a canonical morphism of type $X \to TY$ as in the following diagram.

$$\begin{array}{ccccc}
X & \overset{!_e}{\dashrightarrow} & I_Y & \overset{¡}{\dashrightarrow} & TY \\
 & & & & \uparrow{\scriptstyle \mu_Y^T} \\
{\scriptstyle e}\downarrow & & {\scriptstyle \iota_Y^{-1}}\downarrow\ \uparrow{\scriptstyle \iota_Y} & & TTY \\
 & & & & \uparrow{\scriptstyle T[\mathrm{id}_{TY}, \eta_Y^T]} \\
T(X+Y) & \underset{T(!_e + \mathrm{id}_Y)}{\dashrightarrow} & T(I_Y + Y) & \underset{T(¡ + \mathrm{id}_Y)}{\dashrightarrow} & T(TY + Y)
\end{array} \tag{5}$$

In (5), the map $!_e \colon X \to I_Y$ is the unique morphism of $T(\mathrm{Id} + Y)$-coalgebras given by finality of $\iota_Y^{-1} \colon I_Y \to T(I_Y + Y)$, whereas $¡ : I_Y \to TY$ is the unique morphism of $T(\mathrm{Id} + Y)$-algebras given by initiality of $\iota_Y \colon T(I_Y + Y) \to I_Y$.

We call the composite $¡ \circ !_e \colon X \to TY$ the *canonical fixpoint solution* of e, and we write $e^\dagger$ for this solution.

Theorem 4.3 ([38], Theorem 3)**.** *The operation $e \mapsto e^\dagger$ of assigning a canonical fixpoint solution yields a parametrized fixpoint operator satisfying the double dagger law.*

Remark 4.4. (1) Actually, [38, Theorem 3] is a stronger result; Simpson and Plotkin prove that the above operator † is a unique *parametrically uniform* parametrized fixpoint operator, and it satisfies the *Conway identities*. Parametric uniformity is the following property of †: given $e : X \to T(X + Y)$, $f : X' \to T(X' + Y)$ and $h : X \to X'$, then

$$\begin{array}{ccc} X & \xrightarrow{e} & T(X+Y) \\ {\scriptstyle h}\downarrow & & \downarrow{\scriptstyle T(h+Y)} \\ X' & \xrightarrow[f]{} & T(X'+Y) \end{array} \quad \Longrightarrow \quad \begin{array}{ccc} X & & \\ {\scriptstyle h}\downarrow & \overset{e^\dagger}{\searrow} & \\ & & TY \\ & \underset{f^\dagger}{\nearrow} & \\ X' & & \end{array}$$

And the Conway identities are equational properties of † which together with parametrized uniformity imply the axioms of iteration theories for † (*cf.* Example 2.14). However, the double dagger law from Remark 2.13 (which is one of the Conway identities) is the only property that we need here, and so we do not recall the other properties.

(2) From the previous item it immediately follows that in the case where our category **C** is **Cppo**-enriched and T is locally continuous the parametrized fixpoint operator of Example 2.14 given by taking least fixpoints is the one described here assigning canonical fixpoint solutions. In fact, the former satisfies parametric uniformity (see [9, Exercise 8.2.17]), and so it agrees with the latter by the uniqueness mentioned in item (1).

The following result shows that monad morphisms between monads having sufficiently many canonical fixpoints preserve canonical fixpoint solutions. This is useful for comparing solutions provided by different monads, and it is a direct consequence of [38, Lemma 6.6].

Proposition 4.5 (†-preservation)**.** *Suppose that T and T' are monads on* **C** *satisfying Assumption 4.1 and $\gamma\colon T \Rightarrow T'$ is a monad morphism. For any morphism $e : X \to T(X + Y)$:*

$$\gamma_Y \circ e^\dagger = (\gamma_{X+Y} \circ e)^\dagger : X \to T'Y,$$

where $e^\dagger$ is provided by the canonical fixpoint solution for T and $(\gamma_{X+Y} \circ e)^\dagger$ by the one for T'.

5. A Theory of Systems with Internal Behaviour

We now use canonical fixpoint solutions to provide an abstract theory of systems with internal behaviour, that we will later instantiate to the motivating examples of Section 3. Throughout this section, we will develop our framework for the following ingredients.

Assumption 5.1. Let $\mathbf{C}$ be a category with coproducts and let $F\colon \mathbf{C} \to \mathbf{C}$ be an endofunctor for which all free F-algebras exist. The following monads derived from F are assumed to have sufficiently many canonical fixpoints:

- the free monad $F^*\colon \mathbf{C} \to \mathbf{C}$ (*cf.* Example 2.1.4);
- for every fixed $X \in \mathbf{C}$, the exception monad $FX + \mathrm{Id}\colon \mathbf{C} \to \mathbf{C}$ (*cf.* Example 2.1.3).

In other words the monads F^* and $FX + \mathrm{Id}$ satisfy Assumption 4.1; thus they have canonical fixpoint solutions (which satisfy the double dagger law by Theorem 4.3). We shall see in Theorem 5.5 that Assumption 5.1 is satisfied for F being the lifted functor $\widehat{H}$ of Theorem 2.10.

To avoid ambiguity, we denote with $e \mapsto e^\dagger$ the canonical fixpoint operator associated with F^* and with $e \mapsto e^\ddagger$ the one associated with $FX + \mathrm{Id}$.

We will employ the additional structure of those two monads for the analysis of *F-systems with internal transitions*. An F-system is simply an F-coalgebra $e\colon X \to FX$, where we take the operational point of view of seeing X as a space of states and F as the transition type of e. An F-system with internal transitions is an $(F+\mathrm{Id})$-coalgebra $e\colon X \to FX+X$, where the component X of the codomain is targeted by those transitions representing the internal (non-interacting) behaviour of system e.

A key observation for our analysis is that F-systems—with or without internal transitions—enjoy a standard representation as F^*-systems, that is, coalgebras of the form $e\colon X \to F^*X$.

Definition 5.2 (F-systems as F^*-systems)**.** Let $\kappa : F \Rightarrow F^*$ be as in Example 2.1.4. We introduce the following encoding $e \mapsto \bar{e}$ of F-systems and F-systems with internal transitions as F^*-systems.

- Given an F-system $e\colon X \to FX$, define $\bar{e}\colon X \to F^*X$ as

$$\bar{e} : X \xrightarrow{\;e\;} FX \xrightarrow{\;\kappa_X\;} F^*X.$$

- Given an F-system with internal transitions $e : X \to FX + X$, define $\bar{e} : X \to F^*X$ as

$$\bar{e} : X \xrightarrow{e} FX + X \xrightarrow{[\kappa_X, \eta_X^{F^*}]} F^*X .$$

Thus F-systems (with or without internal transitions) may be seen as equation morphisms $X \to F^*(X + 0)$ for the monad F^* (with the initial object $Y = 0$ as parameter), with canonical fixpoint solutions (*cf.* Section 4). This will allow us to achieve the following.

§1 We supply a uniform trace semantics for F-systems, possibly with internal transitions, and F^*-systems, based on the canonical fixpoint solution operator of F^*.

§2 We use the canonical fixpoint solution operator of $FX+\mathrm{Id}$ to transform any F-system $e : X \to FX + X$ with internal transitions into an F-system $e\backslash\epsilon : X \to FX$ without internal transitions.

§3 We prove that the transformation of **§2** is sound with respect to the semantics of **§1**.

5.1. Uniform Trace Semantics

The canonical fixpoint semantics of F-systems, with or without internal transitions, and F^*-systems is defined as follows.

Definition 5.3 (Canonical Fixpoint Semantics)**.**

- For an F^*-system $e\colon X \to F^*X$, its semantics $[\![e]\!]\colon X \to F^*0$ is defined as $e^\dagger$ (note that e can be seen as an equation morphism for F^* on parameter $Y = 0$).

- For an F-system $e : X \to FX$, its semantics $[\![e]\!]\colon X \to F^*0$ is defined as $\bar{e}^\dagger = (\kappa_X \circ e)^\dagger$.

- For an F-system with internal transitions $e : X \to FX + X$, its semantics $[\![e]\!]\colon X \to F^*0$ is defined as $\bar{e}^\dagger = ([\kappa_X, \eta_X^{F^*}] \circ e)^\dagger$.

The underlying intuition of Definition 5.3 is that canonical fixpoint solutions may be given an operational understanding. Given an F^*-system $e\colon X \to F^*X$, its solution $e^\dagger\colon X \to F^*0$ is formally defined as the composite $¡ \circ !_e$ (*cf.* (5)): we can see the coalgebra morphism $!_e$ as a map that gives the *behaviour* of system e without taking into account the structure of labels and the algebra morphism ¡ as evaluating this structure, e.g. flattening words of words, using the initial algebra $\mu_0^{F^*}\colon F^*F^*0 \to F^*0$ for the monad F^*. In particular, the action of ¡ is what makes our semantics suitable for modeling "algebraic" operations on internal transitions such as ϵ-elimination, as we will see in concrete instances of our framework.

Remark 5.4. The canonical fixpoint semantics of Definition 5.3 encompasses the framework for traces in [23], where the semantics of an F-system e : $X \to FX$—without internal transitions—is defined as the unique morphism $!_e$ from X into the final F-coalgebra F^*0. Indeed, using finality of F^*0, it can be shown that $!_e = [\![e]\!]$. Theorem 5.5 below guarantees compatibility with Assumption 5.1.

The following result is instrumental in our examples and in comparing our theory with the one developed in [23] for trace semantics in Kleisli categories.

Theorem 5.5. *Let* $M\colon \mathbf{Sets} \to \mathbf{Sets}$ *be a monad and* $H\colon \mathbf{Sets} \to \mathbf{Sets}$ *be a functor satisfying the assumptions of Theorem 2.10, that is:*

(a) $\mathcal{K}\ell(M)$ *is* **Cppo***-enriched and composition is left strict;*

(b) H *is accessible and has a locally continuous lifting* $\widehat{H}\colon \mathcal{K}\ell(M) \to \mathcal{K}\ell(M)$*.*

Then $\mathcal{K}\ell(M)$, $\widehat{H}$, $\widehat{H}^*$ *and* $\widehat{H}\mathcal{J}X + \mathrm{Id}$ *(for a given set* X*) satisfy Assumption 5.1.*

Before we proceed to prove the above theorem, we first show its relevance for the motivating examples of Section 3.

Example 5.6 (Semantics of NDA with word transitions)**.** In Section 3, we have modeled NDA with word transitions as $\widehat{H^*}$-coalgebras on $\mathcal{K}\ell(M)$, where H and M are defined as for NDA (see Example 2.11). By Proposition 2.5, $\widehat{H^*} = \widehat{H}^*$ and thus, by virtue of Theorem 5.5, $\widehat{H^*}$ satisfies Assumption 5.1. Therefore we can define the semantics of NDA with word transitions $e\colon X \to$

$\mathcal{P}(A^* \times X + A^*)$ via canonical fixpoint solutions as $[\![e]\!] = e^\dagger = ¡ \circ !_e$:

$$\begin{array}{ccccc}
X & \overset{!_e}{\dashrightarrow} & (A^*)^* \times A^* & \overset{¡}{\dashrightarrow} & A^* \\
\Big\downarrow e & & \cong \updownarrow \quad \begin{aligned} ¡(\langle\rangle, w) &= \{w\} \\ ¡(w :: l, w') &= \{wu \mid u \in ¡(l, w')\} \end{aligned} & & \Big\uparrow \mu_0 \\
A^* \times X + A^* & \underset{\mathrm{id}\times !_e + \mathrm{id}}{\dashrightarrow} & A^* \times ((A^*)^* \times A^*) + A^* & \underset{\mathrm{id}\times ¡ + \mathrm{id}}{\dashrightarrow} & A^* \times A^* + A^*
\end{array}$$

Observe that the above diagram is just (5) instantiated with $T = \widehat{H^*}$ and $Y = 0$. Moreover, this diagram is in $\mathcal{K}\ell(\mathcal{P})$ and hence the explicit definition of $e^\dagger$ as a function $X \to \mathcal{P}(A^*)$ is given by $e^\dagger(x) = \bigcup \mathcal{P}¡(!_e(x))$.

Both $!_e$ and $¡$ can be defined uniquely by the commutativity of the above diagram. We have already defined $!_e$ in (4) and the definition of $¡$ is given in the right-hand square of the above diagram. The isomorphism in the middle and μ_0 were defined in Section 3.

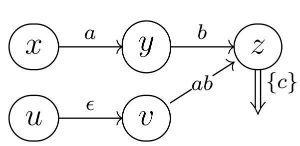
Using the above formula $e^\dagger(x) = \bigcup \mathcal{P}(¡)(!_e(x))$ we now have the semantics of e:

$$\begin{aligned} w \in e^\dagger(x) \Leftrightarrow \; & w \in e(x) \;\; \textbf{or} \;\; \exists y \in X, w_1, w_2 \in A^* \\ & (w_1, y) \in e(x), w_2 \in e^\dagger(y) \text{ and } w = w_1 w_2. \end{aligned} \tag{7}$$

This definition is precisely the language semantics: a word w is accepted by a state x if there exists a decomposition $w = w_1 \cdots w_n$ such that

$$x \xrightarrow{w_1} y_1 \xrightarrow{w_2} \cdots \xrightarrow{w_{n-1}} y_{n-1} \overset{w_n}{\Longrightarrow} \;.$$

Take again the automaton of the motivating example. We can calculate the semantics and observe that we now get exactly what was expected: $e^\dagger(u) = e^\dagger(v) = e^\dagger(x)$.

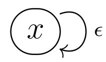


$$\begin{array}{rclcrcl}
!_e(x) &=& \{([a,b],c)\} & \quad & e^\dagger(x) &=& \{abc\} \\
!_e(y) &=& \{([b],c)\} & & e^\dagger(y) &=& \{bc\} \\
!_e(z) &=& \{(\langle\rangle,c)\} & & e^\dagger(z) &=& \{c\} \\
!_e(u) &=& \{([\epsilon,ab],c)\} & & e^\dagger(u) &=& \{abc\} \\
!_e(v) &=& \{([ab],c)\} & & e^\dagger(v) &=& \{abc\}
\end{array}$$

The key role played by the monad structure on A^* can be appreciated by comparing the graphs of $!_e$ and $e^\dagger = ¡ \circ !_e$ as in the example above. The algebra morphism $¡\colon (A^*)^* \times A^* \to A^*$ maps values from the initial algebra

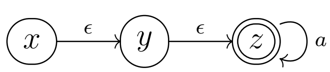
$(A^*)^* \times A^*$ for the *endofunctor* $\widehat{H^*}$ into the initial algebra A^* for the *monad* $\widehat{H^*}$: its action is precisely to take into account the additional equations encoded by the algebraic theory of the monad $\widehat{H^*}$. For instance, we can see the mapping of $!_e(u) = \{([\epsilon, ab], c)\}$ into the word abc as the result of concatenating the words ϵ, ab, c and then quotienting out of the equation $\epsilon abc = abc$ in the monoid A^*.

Remark 5.7 (Multiple Solutions)**.** The canonical solution $e^\dagger$ is not the unique solution. Indeed, the uniqueness of $!_e$ in the left-hand square and of ¡ in the right-hand square of the diagram above does not imply the uniqueness of $e^\dagger$. To see this, take for instance the automaton



Both $s(x) = \emptyset$ and $s'(x) = A^*$ are solutions. The canonical one is the least one, i.e., $e^\dagger(x) = s(x) = \emptyset$. Indeed, as discussed in Remark 4.4(2), whenever the underlying category is **Cppo**-enriched and the monad locally continuous, the canonical solution coincides with the least fixpoint solution introduced in Example 2.14

Example 5.8 (Semantics of NDA with ϵ-transitions)**.** NDA with ϵ-transitions are modeled as $\widehat{H + \mathrm{Id}}$-coalgebras on $\mathcal{K}\ell(M)$, where H and M are defined as for NDA (see Example 2.11). We can define the semantics of NDA with ϵ-transitions via canonical fixpoint solutions as $[\![e]\!] = \bar{e}^\dagger$, where $\bar{e}$ is the automaton with word transitions corresponding to e (see Definition 5.2). The first example in Section 3 would be represented as follows,



$$\begin{aligned} \bar{e}(x) = [\kappa_X, \eta_X] \circ e(x) &= \{(\epsilon, y)\} \\ \bar{e}(y) = [\kappa_X, \eta_X] \circ e(y) &= \{(\epsilon, z)\} \\ \bar{e}(z) = [\kappa_X, \eta_X] \circ e(z) &= \{(a, z), \epsilon\} \end{aligned}$$

where η and κ are defined as at the end of Section 3. By using (7), it can be easily checked that the semantics $[\![e]\!] = \bar{e}^\dagger \colon X \to \mathcal{P}A^*$ maps x, y and z into a^*.

We now proceed to prove Theorem 5.5. First, we give more details on accessible endofunctors and how they yield a canonical free algebra construction.

Remark 5.9. (1) Adámek and Porst [4] showed that an endofunctor H on **Sets** is accessible if and only if it is bounded in the following sense: there exists a cardinal λ such that for every set A, every element of HA lies in the image of Hb for some $b : B \hookrightarrow A$ of less than λ elements.

(2) Recall from [1] that for an accessible endofunctor H on a cocomplete category $\mathbf{C}$ (not only the initial but) all *free* H-algebras exist and are obtained from an inductive construction. More precisely, for every object X of $\mathbf{C}$ define the following ordinal indexed *free-algebra-chain*:

$$\begin{aligned} H_0X &= X, \\ H_{i+1}X &= HH_iX + X, \\ H_jX &= \operatorname*{colim}_{i<j} H_iX \qquad \text{for a limit ordinal } j. \end{aligned}$$

Its connecting morphisms $u_{i,j} : H_iX \to H_jX$ are uniquely determined by

$$\begin{aligned} u_{0,1} &= (X \xrightarrow{\mathsf{inr}} HX + X), \\ u_{i+1,j+1} &= (HH_iX + X \xrightarrow{Hu_{i,j}+X} HH_jX + X), \\ & u_{i,j}\ (i<j) \text{ is the colimit cocone for limit ordinals } j. \end{aligned}$$

Indeed, this defines an ordinal indexed chain uniquely (up to isomorphism). The "missing" connecting maps are determined by the universal property of colimits, e.g. $u_{\omega,\omega+1}$ is unique such that $u_{\omega,\omega+1} \cdot u_{i+1,\omega} = u_{i+1,\omega+1} = Hu_{i,\omega}$ for all $i < \omega$.

Now suppose that H preserves λ-filtered colimits. Then $u_{\lambda,\lambda+1}$ is an isomorphism and one can show that $H_\lambda X$ is a free H-algebra on X with the structure and universal morphism given by $u^{-1}_{\lambda,\lambda+1}$.

(3) As we saw previously, the assignment of a free H-algebra on X to any object X yields a free monad on H; thus, in item (2) above we have $H^* = H_\lambda$. Now notice that the construction in the previous point can be written object free; we obtain H^* after λ steps of the following chain in the category of endofunctors on $\mathbf{C}$:

$$\begin{aligned} H_0 &= \mathrm{Id}, \\ H_{i+1} &= HH_i + \mathrm{Id}, \\ H_j &= \operatorname*{colim}_{i<j} H_j \qquad \text{for limit ordinals } i. \end{aligned}$$

The connecting natural transformations $H_i \Rightarrow H_j$ have the components described as connecting morphisms in item (2).

As a consequence we see that if H is accessible then so is H^*; indeed, all H_i preserve λ-filtered colimits if H does.

The next proposition is instrumental in relating accessiblity of an endofunctor with the existence of initial algebras for its lifting.

Proposition 5.10. *Let* $\mathbf{C}$ *be a cocomplete category,* $M\colon \mathbf{C} \to \mathbf{C}$ *be a monad and* $G\colon \mathbf{C} \to \mathbf{C}$ *be an accessible endofunctor with a lifting* $\widehat{G}\colon \mathcal{K}\ell(M) \to \mathcal{K}\ell(M)$*. Then for all* $Y \in \mathcal{K}\ell(M)$ *both the initial* $\widehat{G}(\mathrm{Id}+Y)$*-algebra and the initial* $\widehat{G}(\mathrm{Id}) + Y$*-algebra exist.*

Proof. As the left adjoint $\mathcal{J}\colon \mathbf{C} \to \mathcal{K}\ell(M)$ is defined as the identity on objects, without loss of generality we can prove our statement for an object $\mathcal{J}Y \in \mathcal{K}\ell(M)$, where $Y \in \mathbf{C}$.

First we observe that the endofunctor $Y+\mathrm{Id}\colon \mathbf{C} \to \mathbf{C}$ (*cf.* Example 2.1.3) always has a lifting to $\mathcal{K}\ell(M)$. Indeed, because the left adjoint $\mathcal{J}\colon \mathbf{C} \to \mathcal{K}\ell(M)$ preserves coproducts, we have

$$\mathcal{J} \circ (\mathrm{Id} + Y) = \mathcal{J}(\mathrm{Id}) + \mathcal{J}Y = (\mathrm{Id} + \mathcal{J}Y) \circ \mathcal{J}$$

implying that $\mathrm{Id} + \mathcal{J}Y\colon \mathcal{K}\ell(M) \to \mathcal{K}\ell(M)$ is a lifting of $\mathrm{Id} + Y\colon \mathbf{C} \to \mathbf{C}$.

Now we can compose the $\mathbf{C}$-endofunctors G and $\mathrm{Id} + Y$ in two different ways, obtaining $G(\mathrm{Id})+Y\colon \mathbf{C} \to \mathbf{C}$ and $G(\mathrm{Id}+Y)\colon \mathbf{C} \to \mathbf{C}$. It is straightforward to check that the composite of two liftings is a lifting of the composite functor. This means that we have liftings $\widehat{G}(\mathrm{Id}) + \mathcal{J}Y\colon \mathcal{K}\ell(M) \to \mathcal{K}\ell(M)$ and $\widehat{G}(\mathrm{Id} + \mathcal{J}Y)\colon \mathcal{K}\ell(M) \to \mathcal{K}\ell(M)$ respectively of $G(\mathrm{Id}) + Y\colon \mathbf{C} \to \mathbf{C}$ and $G(\mathrm{Id} + Y)\colon \mathbf{C} \to \mathbf{C}$.

The next step is to use accessibility to get initial algebras in $\mathbf{C}$ that will be then lifted to $\mathcal{K}\ell(M)$. In fact, we observe that both functors $G(\mathrm{Id})+Y\colon \mathbf{C} \to \mathbf{C}$ and $G(\mathrm{Id}+Y)\colon \mathbf{C} \to \mathbf{C}$ are accessible, because the functor $Y+\mathrm{Id}$ is clearly accessible and G is assumed to have this property.

Thus as observed in Remark 5.9.(2) both an initial $G(\mathrm{Id})+Y$-algebra and an initial $G(\mathrm{Id}+Y)$-algebra exist. Then Proposition 2.4 yields the existence both of an initial $\widehat{G}(\mathrm{Id}) + \mathcal{J}Y$-algebra and an initial $\widehat{G}(\mathrm{Id} + \mathcal{J}Y)$-algebra. □

We are now ready to supply a proof of Theorem 5.5.

Proof of Theorem 5.5. Since $\mathcal{K}\ell(M)$ inherits coproducts from **Sets**, we only need to check the following properties:

1. all free $\widehat{H}$-algebras exist;

2. for all $Y \in \mathcal{K}\ell(M)$, the initial algebras for $\widehat{H}^*(\mathrm{Id} + Y)$, $\widehat{H}^*(\widehat{H}^*(\mathrm{Id} + Y) + Y)$ and $\widehat{H}^*(\mathrm{Id} + \mathrm{Id} + Y)$ exist;

3. for all $Y \in \mathcal{K}\ell(M)$, the initial algebras for $\widehat{H}\mathcal{J}X + \mathrm{Id} + Y$, $\widehat{H}\mathcal{J}X + \widehat{H}\mathcal{J}X + \mathrm{Id} + Y + Y$ and $\widehat{H}\mathcal{J}X + \mathrm{Id} + \mathrm{Id} + Y$ exist.

Then it follows from Proposition 2.9, Theorem 2.8 and the fact the coproducts in $\mathcal{K}\ell(M)$ are **Cppo**-enriched that $\widehat{H}^*$ and $\widehat{H}\mathcal{J}X + \mathrm{Id}$ have sufficiently many canonical fixpoints.

By virtue of Proposition 5.10, the three properties above are implied respectively by the following statements:

1. the functor $H\colon \mathbf{Sets} \to \mathbf{Sets}$ is accessible;

2. the functor $\widehat{H}^*\colon \mathcal{K}\ell(M) \to \mathcal{K}\ell(M)$ is the lifting of $H^*\colon \mathbf{Sets} \to \mathbf{Sets}$ and H^* is accessible;

3. the functor $\widehat{H}\mathcal{J}X+\mathrm{Id}\colon \mathcal{K}\ell(M) \to \mathcal{K}\ell(M)$ is the lifting of $HX+\mathrm{Id}\colon \mathbf{Sets} \to \mathbf{Sets}$ and $HX + \mathrm{Id}$ is accessible.

The first point is given by assumption. For the second point, H^* is accessible by Remark 5.9.(3) and $\widehat{H}^*\colon \mathcal{K}\ell(M) \to \mathcal{K}\ell(M)$ is its lifting by Proposition 2.5. For the third point, since the identity $\mathrm{Id}\colon \mathbf{Sets} \to \mathbf{Sets}$ and the constant functor $HX\colon \mathbf{Sets} \to \mathbf{Sets}$ are clearly accessible and coproducts preserve this property, then $HX + \mathrm{Id}\colon \mathbf{Sets} \to \mathbf{Sets}$ is also accessible. As the left adjoint $\mathcal{J}\colon \mathbf{Sets} \to \mathcal{K}\ell(M)$ preserves coproducts, it is immediate to check that $\widehat{H}\mathcal{J}X + \mathrm{Id}\colon \mathcal{K}\ell(M) \to \mathcal{K}\ell(M)$ is the lifting of $HX + \mathrm{Id}\colon \mathbf{Sets} \to \mathbf{Sets}$. Indeed:

$$\mathcal{J} \circ (HX + \mathrm{Id}) = \mathcal{J}HX + \mathcal{J}(\mathrm{Id}) = \widehat{H}\mathcal{J}X + \mathcal{J}(\mathrm{Id}) = (\widehat{H}\mathcal{J}X + \mathrm{Id}) \circ \mathcal{J}.$$

This concludes the proof of the three properties above. □

5.2. Elimination of Internal Transitions

We view an F-system $e\colon X \to FX + X$ with internal transitions as an equation morphism for the monad $FX + \mathrm{Id}$, with parameter $Y = 0$. Thus we can use the canonical fixpoint solution of $FX + \mathrm{Id}$ to obtain an F-system

$e^{\ddagger}\colon X \to FX + 0 = FX$, which we denote by $e\backslash\epsilon$. The construction is depicted below.

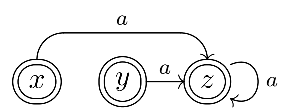
$$
\begin{array}{ccccc}
 & & \overset{e\backslash\epsilon \overset{\text{def}}{=} e^{\ddagger}}{\frown} & & \\
X & \overset{!_e}{\dashrightarrow} & \mathbb{N} \times FX & \overset{\text{¡}}{\dashrightarrow} & FX \\
{\scriptstyle e}\downarrow & & \uparrow\cong\downarrow & & \uparrow{\scriptstyle \mu_0 = \nabla_{FX}} \\
FX + X & \xrightarrow[\mathrm{id}_{FX} + !_e]{} & FX + \mathbb{N} \times FX & \xrightarrow[\mathrm{id}_{FX} + \text{¡}]{} & FX + FX
\end{array}
\tag{8}
$$

Example 5.11 (ϵ-elimination)**.** Using the automaton of Example 5.8, we can perform ϵ-elimination, as defined in (8), using the canonical solution for the monad $\widehat{H}\mathcal{J}X + \mathrm{Id}$:

$$
\begin{array}{ccccc}
X & \overset{!_e}{\dashrightarrow} & \mathbb{N} \times (A \times X + 1) & \overset{\text{¡}}{\dashrightarrow} & (A \times X + 1) \\
{\scriptstyle e}\downarrow & & \uparrow\cong\downarrow & & \uparrow{\scriptstyle \mu_0 = \nabla} \\
(A \times X + 1) + X & \xrightarrow[\mathrm{id} + !_e]{} & (A \times X + 1) + \mathbb{N} \times (A \times X + 1) & \xrightarrow[\mathrm{id} + \text{¡}]{} & (A \times X + 1) + (A \times X + 1)
\end{array}
$$

We obtain the following NDA $e\backslash\epsilon \overset{\text{def}}{=} \text{¡} \circ !_e \colon X \to A \times X + 1$.

$$
\begin{array}{rclrcl}
!_e(x) & = & \{(2, a, z), (2, \checkmark)\} & e\backslash\epsilon(x) & = & \{(a, z), \checkmark\} \\
!_e(y) & = & \{(1, a, z), (1, \checkmark)\} & e\backslash\epsilon(y) & = & \{(a, z), \checkmark\} \\
!_e(z) & = & \{(0, a, z), (0, \checkmark)\} & e\backslash\epsilon(x) & = & \{(a, z), \checkmark\}
\end{array}
$$



The semantics $[\![e\backslash\epsilon]\!]$ is defined as $\overline{e\backslash\epsilon}^{\dagger}$, where $\overline{e\backslash\epsilon} = \kappa_X \circ e\backslash\epsilon$ is the representation of the NDA $e\backslash\epsilon$ as an automaton with word transitions (Definition 5.2). It is immediate to see, in this case, that $[\![e\backslash\epsilon]\!] = [\![e]\!]$. This fact is an instance of Theorem 5.14 below.

Remark 5.12. Note that ϵ-elimination was recently defined using a trace operator on a Kleisli category [22, 37, 6]. These works are based on the trace semantics of Hasuo et al. [23] and tailored for ϵ-elimination. They do not take into account any algebraic structure of the labels and are hence not applicable to the other examples we consider in this paper.

5.3. Soundness of ϵ-Elimination

We now formally prove that the canonical fixpoint semantics of e and $e\backslash\epsilon$ coincide. To this end, we first show how the construction $e \mapsto e\backslash\epsilon$ can be expressed in terms of the canonical fixpoint solution of F^*. This turns out to be an application of †-preservation (Proposition 4.5), for which we introduce the natural transformation $\pi\colon FX + \mathrm{Id} \Rightarrow F^*(X + \mathrm{Id})$ defined at $Y \in \mathbf{C}$ by

$$\pi_Y\colon\ FX + Y \xrightarrow{[\kappa_X,\ \eta_Y^{F^*}]} F^*X + F^*Y \xrightarrow{[F^*\mathsf{inl}, F^*\mathsf{inr}]} F^*(X + Y)\ .$$

Since F^* is a monad with sufficiently many canonical fixpoints, it follows that so is $F^*(X + \mathrm{Id})$. Moreover, π is a monad morphism between $FX + \mathrm{Id}$ and $F^*(X + \mathrm{Id})$. These observations allow us to prove the following.

Proposition 5.13 (Factorisation property of $e \mapsto e\backslash\epsilon$)**.** *For any F-system $e : X \to FX + X$ with internal transitions, consider the equation morphism $\pi_X \circ e : X \to F^*(X + X)$. Then:*

$$\pi_0 \circ e\backslash\epsilon = (\pi_X \circ e)^\dagger : X \to F^*X.$$

Proof. Let us use the notation $e \mapsto e^\bullet$ for the canonical fixpoint solution operator of $F^*(X+\mathrm{Id})$. We now apply Proposition 4.5 to show that solutions of $F^*(X+\mathrm{Id})$ factorize through the ones of $FX+\mathrm{Id}$. The connecting monad morphism is $\pi : FX + \mathrm{Id} \to F^*(X + \mathrm{Id})$, defined above. Proposition 4.5 yields the following factorisation property:

($*$) for any $Y, Z \in \mathbf{C}$ and equation morphism $e\colon Z \to FX+Z+Y$, consider $\pi_{Z+Y} \circ e\colon Z \to F^*(X+Z+Y)$. The solution $(\pi_{Z+Y} \circ e)^\bullet\colon Z \to F^*(X+Y)$ provided by $F^*(X+\mathrm{Id})$ factorises as $\pi_Y \circ e^\ddagger$, where $e^\ddagger\colon Z \to FX+Y$ is the solution of e provided by $FX + \mathrm{Id}$.

If we fix $Z = X$ and $Y = 0$, then ($*$) says: for any F-system $e : X \to FX+X$ with internal computation, consider the equation morphism $(\pi_{X+0} \circ e : X) \to F^*(X + X + 0)$ for $F^*(X + \mathrm{Id})$ with parameter $Y = 0$. Then the following diagram commutes:

$$\begin{array}{ccc} X & \xrightarrow{(\pi_X \circ e)^\bullet} & F^*X \\ & \searrow^{e^\ddagger} & \uparrow{\scriptstyle \pi_0} \\ & & FX \end{array} \qquad (9)$$

To conclude our argument, we observe that the system $\pi_{X+0} \circ e \colon X \to F^*(X+X+0)$ can be also seen as an equation for F^* with parameter $Y = X+0$. This means that also F^* provides a solution to such equation, which can be checked to coincide with the one given by $F^*(X + \mathrm{Id})$, that is, $(\pi_X \circ e)^\bullet = (\pi_X \circ e)^\dagger$. Then the main statement is proven by the following derivation:

$$\begin{aligned} \pi_0 \circ e \backslash \epsilon &= \pi_0 \circ e^\ddagger && \text{(Definition of } e\backslash\epsilon) \\ &= (\pi_X \circ e)^\bullet && \text{(commutativity of (9))} \\ &= (\pi_X \circ e)^\dagger. && \text{(observation above)} \end{aligned}$$

□

We are now ready to show **§3** (see page 21): soundness of ϵ-elimination.

Theorem 5.14 (Eliminating internal transitions is sound)**.** *For any F-system $e : X \to FX + X$ with internal transitions,*

$$[\![e\backslash\epsilon]\!] = [\![e]\!].$$

Proof. The statement is shown by the following derivation.

$$\begin{aligned} [\![e\backslash\epsilon]\!] &= [\![e^\ddagger]\!] && \text{Definition of } e\backslash\epsilon \\ &= (\kappa_X \circ e^\ddagger)^\dagger && \text{Definition of } [\![-]\!] \text{ (Def. 5.3)} \\ &= (\pi_0 \circ e^\ddagger)^\dagger && \text{Definition of } \pi_0 \\ &= (\pi_X \circ e)^{\dagger\dagger} && \text{Proposition 5.13} \\ &= (F^*(\nabla_X) \circ (\pi_X \circ e))^\dagger && \text{double dagger law} \\ &= \bar{e}^\dagger && \text{Definition of } \bar{e} \text{ (Def. 5.2) and } \pi_X \\ &= [\![e]\!] && \text{Definition of } [\![-]\!]. \end{aligned}$$

□

6. Quotient Semantics

When considering behaviour of systems it is common to encounter spectrums of successively coarser equivalences. For instance, in basic process algebra trace equivalence can be obtained by quotienting bisimilarity with an axiom stating the distributivity of action prefixing by non-determinism [35]. There are many more examples of this phenomenon, including Mazurkiewicz traces, which we will describe below.

In this section we develop a variant of the canonical fixpoint semantics, where we can encompass in a uniform manner behaviours which are quotients of the canonical behaviours of the previous section (that is, the object F^*0).

Assumption 6.1. Let $\mathbf{C}$, F, F^* and $FX + \mathrm{Id}$ be as in Assumption 5.1 and $\gamma\colon F^* \Rightarrow Q$ a monad quotient for some monad Q, i.e., the natural transformation γ has epimorphic components. Moreover, suppose that Q has sufficiently many canonical fixpoints.

Observe that, as Assumption 6.1 subsumes Assumption 5.1, we are within the framework of the previous section, with the canonical fixpoint solution of F^* providing semantics for F^*- and F-systems. For our extension, one is interested in $Q0$ as a semantic domain coarser than F^*0 and we aim at defining an interpretation for F-systems in $Q0$. We first note that according to Theorem 4.3, Q is a monad with canonical fixpoint solutions, which satisfy the double dagger law. We use the notation $e \mapsto e^{\sim}$ for the canonical fixpoint operator of Q. This allows us to define the semantics of Q-systems, analogously to what we did for F^*-systems in Definition 5.3. Moreover, the connecting monad morphism $\gamma\colon F^* \Rightarrow Q$ yields an extension of this semantics to include also systems of transition type F^* and F.

Definition 6.2 (Quotient Semantics)**.** The quotient semantics of F-systems, with or without internal transitions, F^*-systems and Q-systems is defined as follows.

- For a Q-system $e\colon X \to QX$, its semantics $[\![e]\!]_\sim\colon X \to Q0$ is defined as $e^{\sim}$ (note that e can be regarded as an equation morphism for Q with $Y = 0$).
- For an F^*-system $e\colon X \to F^*X$, its semantics $[\![e]\!]_\sim\colon X \to Q0$ is defined as $(\gamma_X \circ e)^{\sim}$.
- For an F-system e—with or without internal transitions—its semantics $[\![e]\!]_\sim\colon X \to Q0$ is defined as $(\gamma_X \circ \bar{e})^{\sim}$, where $\bar{e}$ is as in Definition 5.2.

Proposition 4.5 allows us to establish a link between the canonical fixpoint semantics $[\![-]\!]$ and the quotient semantics $[\![-]\!]_\sim$.

Proposition 6.3 (Factorisation for the quotient semantics)**.** *Let e be either an F^*-system or an F-system (with or without internal transitions). Then:*

$$[\![e]\!]_\sim = \gamma_0 \circ [\![e]\!]. \tag{10}$$

Proof. We instantiate the statement of Proposition 4.5 to the monads F^*, Q and the monad morphism $\gamma\colon F^* \Rightarrow Q$. It amounts to commutativity of the following diagram for a given F^*-system $e\colon X \to F^*X$ and the parameter $Y = 0$:

$$\begin{array}{ccc} X & \xrightarrow{(\gamma_X \circ e)^{\sim}} & Q0 \\ & {\scriptstyle e^{\dagger}} \searrow & \uparrow {\scriptstyle \gamma_0} \\ & & F^*0 \end{array} \tag{11}$$

Thus for F^*-systems the equality (10) is immediate, because $[\![e]\!]_{\sim} = (\gamma_X \circ e)^{\sim}$ by Definition 6.2 and $(\gamma_X \circ e)^{\sim} = \gamma_0 \circ e^{\dagger} = \gamma_0 \circ [\![e]\!]$ by commutativity of (11).

Starting instead from an F-system e' based on state space X, with or without internal computations, consider the following chain of equalities:

$$[\![e']\!]_{\sim} = (\gamma_X \circ \overline{e'})^{\sim} = \ \gamma_0 \circ \overline{e'}^{\dagger} = \gamma_0 \circ [\![e']\!].$$

The first and third equalities are given by unfolding the definition of $[\![-]\!]_{\sim}$ and $[\![-]\!]$, respectively, whereas the second one is due to commutativity of (11) applied to the F^*-system $\overline{e'}\colon X \to F^*X$ in lieu of e. ☐

As a corollary we obtain that eliminating internal transitions is sound also for quotient semantics.

Corollary 6.4. *For any F-system $e : X \to FX{+}X$ with internal transitions,*

$$[\![e]\!]_{\sim} = [\![e\backslash\epsilon]\!]_{\sim}.$$

Proof. The statement is immediately given by the following derivation

$$[\![e]\!]_{\sim} = \gamma_0 \circ [\![e]\!] = \gamma_0 \circ [\![e\backslash\epsilon]\!] = [\![e\backslash\epsilon]\!]_{\sim}$$

where the first and third equalities hold by Proposition 6.3 and the second equality by Theorem 5.14. ☐

The quotient semantics can be formulated in a Kleisli category $\mathcal{K}\ell(M)$ by further assuming ($\diamondsuit$) below. This is needed to lift a quotient of monads from **Sets** to $\mathcal{K}\ell(M)$.

In the following theorem we will work with a monad $M\colon$ **Sets** $\to$ **Sets** and and an accessible endofunctor $H\colon$ **Sets** $\to$ **Sets** satisfying the assumptions of Theorem 2.10. So H is assumed to have a lifting $\widehat{H}$ on $\mathcal{K}\ell(M)$; equivalently we have a distributive law $\varphi : HM \to MH$ (see Proposition 2.2). As shown in Proposition 2.5, φ induces the distributive law $\lambda : H^*M \to MH^*$ so that the free monad H^* on H lifts to a monad $\widehat{H^*} \colon \mathcal{K}\ell(M) \to \mathcal{K}\ell(M)$ via λ and $\widehat{H^*} = \widehat{H}^*$.

Theorem 6.5. *Let $M \colon \mathbf{Sets} \to \mathbf{Sets}$ be a monad and $H \colon \mathbf{Sets} \to \mathbf{Sets}$ be an accessible functor satisfying the assumptions of Theorem 2.10, and let $\lambda \colon H^*M \Rightarrow MH^*$ be the induced distributive law yielding $\widehat{H^*} = \widehat{H}^*$. Let $R \colon \mathbf{Sets} \to \mathbf{Sets}$ be a monad and $\xi \colon H^* \Rightarrow R$ a monad quotient such that*

($\diamondsuit$) *for each set X, there is a map $\lambda'_X \colon RMX \to MRX$ making the following commute.*

$$\begin{array}{ccc} H^*MX & \xrightarrow{\lambda_X} & MH^*X \\ \downarrow{\scriptstyle \xi_{MX}} & & \downarrow{\scriptstyle M\xi_X} \\ RMX & \xrightarrow[\lambda'_X]{} & MRX \end{array}$$

Then the following hold:

1. *there is a monad $\widehat{R} \colon \mathcal{K}\ell(M) \to \mathcal{K}\ell(M)$ lifting R and a monad morphism $\widehat{\xi} \colon \widehat{H}^* \Rightarrow \widehat{R}$ defined as $\widehat{\xi_X} = \mathcal{J}(\xi_X)$;*
2. *$\mathcal{K}\ell(M)$, $\widehat{H}$, $\widehat{H}^*$, $\widehat{H}\mathcal{J}X + \mathrm{Id}$ (for a given set X), $\widehat{R}$ and $\widehat{\xi} \colon \widehat{H}^* \Rightarrow \widehat{R}$ satisfy Assumption 6.1.*

Before we proceed to the proof of the theorem we state and prove the following lemma that provides sufficient conditions for lifting the quotient of an endofunctor to $\mathcal{K}\ell(M)$.

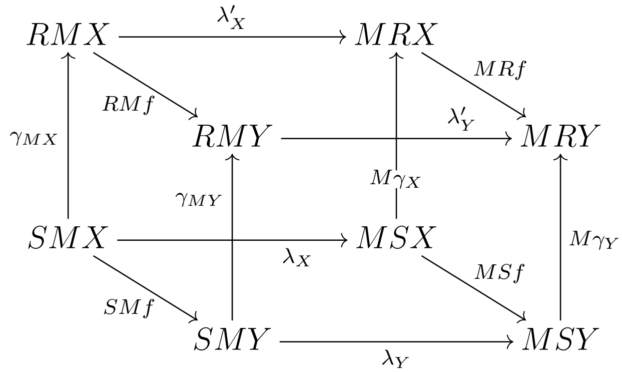
Proposition 6.6. *Let $M, S \colon \mathbf{C} \to \mathbf{C}$ be monads such that there exists a distributive law $\lambda \colon SM \Rightarrow MS$ and let $\widehat{S} \colon \mathcal{K}\ell(M) \to \mathcal{K}\ell(M)$ be the corresponding lifting. Let $\gamma \colon S \Rightarrow R$ be a monad quotient such that*

($\triangle$) *for each X, there is a map $\lambda'_X \colon RMX \to MRX$ making the following commute.*

$$\begin{array}{ccc} SMX & \xrightarrow{\lambda_X} & MSX \\ \downarrow{\scriptstyle \gamma_{MX}} & & \downarrow{\scriptstyle M\gamma_X} \\ RMX & \xrightarrow[\lambda'_X]{} & MRX \end{array}$$

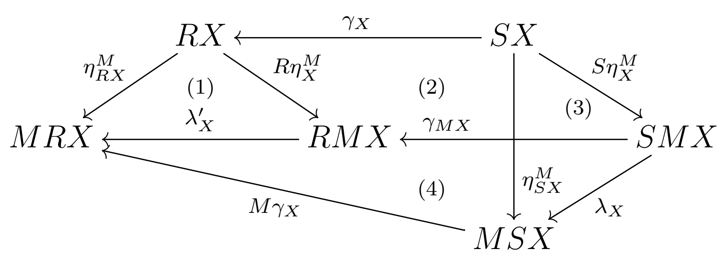
Then R lifts to a monad $\widehat{R} \colon \mathcal{K}\ell(M) \to \mathcal{K}\ell(M)$ and $\widehat{q} \colon \widehat{S} \Rightarrow \widehat{R}$ defined as $\widehat{\gamma_X} = \mathcal{J}(\gamma_X)$ is a monad quotient.

Proof. We first prove that $\lambda' \colon RM \Rightarrow MR$ with the components λ'_X is a natural transformation. Let $f \colon X \to Y$ be a morphism in $\mathbf{C}$. We construct the following cube.



The bottom face commutes by naturality of λ; the leftmost and the righmost faces commute by naturality of γ; the backward and the front faces commute because of $(\triangle)$. It is therefore easy to see that $MRf \circ \lambda'_X \circ \gamma_{MX} = \lambda'_Y \circ RMf \circ \gamma_{MX}$. As each γ-component is epi, it follows that $MRf \circ \lambda'_X = \lambda'_Y \circ RMf$.

Now, we prove that $\lambda' \colon RM \Rightarrow MR$ is a distributive law of monads. The argument for the four diagrams is analogous, so we just show the one for η_M, depicted in the triangle (1), below.



Observe that (2) commutes by naturality of γ, (3) commutes since λ is a distributive law of monads and (4) commute by $(\triangle)$. Therefore the first equality of the following equation holds

$$\lambda'_X \circ R\eta^M_X \circ \gamma_X = M\gamma_X \circ \eta^M_{SX} = \eta^M_{RX} \circ \gamma_X$$

and the second equality holds by naturality of η^M. The commutativity of (1) follows since γ_X is epi.

By Proposition 2.2, and the fact that $\lambda' \colon RM \Rightarrow MR$ is a distributive law of the monad R over the monad M, we obtain that R has a monad lifting $\widehat{R} \colon \mathcal{K}\ell(M) \to \mathcal{K}\ell(M)$.

We now prove that $\widehat{q}\colon \widehat{S} \Rightarrow \widehat{R}$ is a monad morphism. First, we need to check that it is a natural transformation, that is for all morphisms $f\colon X \to Y$ in $\mathcal{K}\ell(M)$, the following diagram commutes.

$$\begin{array}{ccc} \widehat{S}X & \xrightarrow{\mathcal{J}(\gamma_X)} & \widehat{R}X \\ {\scriptstyle \widehat{S}f}\downarrow & & \downarrow{\scriptstyle \widehat{R}f} \\ \widehat{S}Y & \xrightarrow[\mathcal{J}(\gamma_Y)]{} & \widehat{R}Y \end{array}$$

By spelling out the definitions of $\mathcal{J}$ and $\widehat{S}$, the above diagram corresponds to the following in $\mathbf{C}$.

$$\begin{array}{ccccccc} SX & \xrightarrow{\eta^M_{SX}} & MSS & \xrightarrow{M\gamma_X} & MRX & & \\ {\scriptstyle Sf}\downarrow & (1) & \downarrow{\scriptstyle MSf} & (2) & \downarrow{\scriptstyle MRf} & & \\ SMY & \xrightarrow{\eta^M_{SY}} & MSMY & \xrightarrow{M\gamma_{MY}} & MRMY & & \\ {\scriptstyle \lambda_Y}\downarrow & (3) & \downarrow{\scriptstyle M\lambda_Y} & (4) & \downarrow{\scriptstyle M\lambda'_Y} & & \\ MSY & \xrightarrow[\eta^M_{MSY}]{} & MMSY & \xrightarrow[MM\gamma_Y]{} & MMRY & \xrightarrow{\mu^M_{RY}} & MRY \end{array}$$

Observe that (1) and (3) commute by naturality of η^M, (2) commutes by naturality of γ and (4) commutes by ($\triangle$).

Verifying that $\widehat{q}$ is a also morphism of monads is immediate: $\widehat{q} \circ \eta^{\widehat{S}} = \mathcal{J}(q) \circ \mathcal{J}(\eta^S) = \mathcal{J}(\eta^R) = \eta^{\widehat{R}}$ and $\widehat{q} \circ \mu^{\widehat{S}} = \mathcal{J}(q) \circ \mathcal{J}(\mu^S) = \mathcal{J}(\mu^R) \circ \mathcal{J}(Rq \circ \gamma_S) = \mu^{\widehat{S}} \circ \widehat{R}\widehat{q} \circ \widehat{\gamma_S}$.

All its components are epi since $\mathcal{J}$ is a left adjoint and thus preserves epis. □

Proof of Theorem 6.5. Point 1 holds by Proposition 6.6. In particular, the morphism $\widehat{\xi}\colon \widehat{H}^* \Rightarrow \widehat{R}$ is of the right type because $\widehat{H^*} = \widehat{H}^*$ by Proposition 2.5. For point 2 we observe that, for $\mathcal{K}\ell(M)$, $\widehat{H}$, $\widehat{H}^*$ and $\widehat{H}\mathcal{J}X + \mathrm{Id}$, proving Assumption 6.1 amounts to showing Assumption 5.1, which we already did in Theorem 5.5.

Thus it only remains to prove that $\widehat{R}$ has sufficiently many canonical fixpoints. Using Proposition 2.9 and Theorem 2.8 this follows if we show that for all $Y \in \mathcal{K}\ell(M)$ initial algebras for $\widehat{R}(\mathrm{Id} + Y)$, $\widehat{R}(\widehat{R}(\mathrm{Id} + Y) + Y)$ and $\widehat{R}(\mathrm{Id} + \mathrm{Id} + Y)$ exist. By virtue of Proposition 5.10, it suffices to show that $R\colon \mathbf{Sets} \to \mathbf{Sets}$ is accessible. The accessibility of the quotient R of

$H^*\colon \mathbf{Sets} \to \mathbf{Sets}$ is guaranteed from the fact that $H^*\colon \mathbf{Sets} \to \mathbf{Sets}$ is accessible (Remark 5.9(3)) and thus bounded (Remark 5.9(1)) and that the quotients of bounded functors are also bounded. □

Notice that condition (△) and the first part of Statement 1 are related to [11, Theorem 1]; however, that paper treats distributive laws of monads over endofunctors. Observe also, that [11, Example 3] gives a counterexample of a monad quotient and distributive law involving the functor $FX = \mathbb{R} \times X$ that does not satisfy (△). This distributive law is easily seen to be a distributive law between monads, if we regard F as a monad (with the monad structure arising from the monoid $(\mathbb{R}, +, 0)$). Thus, this yields a counterexample to Proposition 6.6.

Example 6.7 (Mazurkiewicz traces)**.** This example, using a known equivalence in concurrency theory, illustrates the use of the quotient semantics developed in this section.

The trace semantics proposed by Mazurkiewicz [29] accounts for concurrent actions. Intuitively, let A be the action alphabet and $a, b \in A$. We will call a and b concurrent, and write $a \equiv b$, if the order in which these actions occur is not relevant. This means that we equate words that only differ in the order of these two actions, e.g. $uabv$ and $ubav$ denote the same Mazurkiewicz trace.

To obtain the intended semantics of Mazurkiewicz traces we use the quotient semantics defined above[3]. In particular, for Mazurkiewisz traces one considers a symmetric and irreflexive "independence" relation I on the label set A. Let $\equiv$ be the least congruence relation on the free monoid A^* such that

$$(a, b) \in I \Rightarrow ab \equiv ba.$$

We now have two monads on **Sets**, namely $H^*X = A^* \times X + A^*$ and $RX = A^*/_{\equiv} \times X + A^*/_{\equiv}$. There is the canonical quotient of monads $\xi\colon H^* \Rightarrow R$ given by identifying words of the same $\equiv$-equivalence class. We now verify that those data satisfy the assumptions of Theorem 6.5.

Proposition 6.8. *The monads* $\mathcal{P}\colon \mathbf{Sets} \to \mathbf{Sets}$ *and* $R\colon \mathbf{Sets} \to \mathbf{Sets}$*, the*

[3]Mazurkiewicz traces were defined over labelled transition systems which are similar to NDA but where every state is final. For simplicity, we consider LTS here immediately as NDA.

*functor $H\colon$ **Sets** $\to$ **Sets** and the quotient of monads $\xi\colon H^* \Rightarrow R$ satisfy the assumptions of Theorem 6.5.*

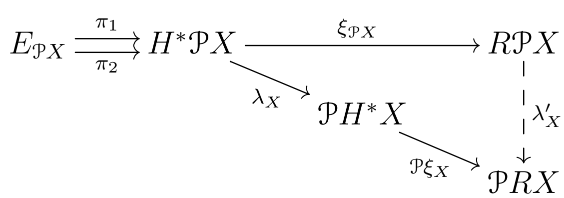
Proof. Clearly the functor $H\colon$ **Sets** $\to$ **Sets** is accessible. The remaining properties of H and of the monad $\mathcal{P}\colon$ **Sets** $\to$ **Sets** are as in Theorem 2.10 and have been already verified in [23]. Thus it remains to show that the quotient $\xi\colon H^* \Rightarrow R$ satisfies condition ($\diamondsuit$) of Theorem 6.5. For this purpose, fix a set X. The desired morphism $\lambda'_X\colon R\mathcal{P}X \to \mathcal{P}RX$ will be given by the universal property of a standard coequalizer diagram induced by the congruence relation $\equiv\, \subseteq A^* \times A^*$. First we define the set $E_{\mathcal{P}X} \subseteq (H^*\mathcal{P}X \times H^*\mathcal{P}X)$ as

$$E_{\mathcal{P}X} \;:=\; \{\big((w,Y)(v,Y)\big) \mid w \equiv v\} \cup \{(w,v) \mid w \equiv v\}.$$

Intuitively, $E_{\mathcal{P}X}$ is the set of equations on $H^*\mathcal{P}X$ induced by $\equiv$. There are evident projection maps $\pi_1, \pi_2\colon E_{\mathcal{P}X} \to H^*\mathcal{P}X$. It is immediate to verify that the following is a coequalizer diagram.

$$E_{\mathcal{P}X} \overset{\pi_1}{\underset{\pi_2}{\rightrightarrows}} H^*\mathcal{P}X \xrightarrow{\xi_{\mathcal{P}X}} R\mathcal{P}X$$

Also one can check that the morphism $\mathcal{P}\xi_X \circ \lambda_X\colon H^*\mathcal{P}X \to \mathcal{P}RX$ (where $\lambda\colon H^*\mathcal{P} \Rightarrow \mathcal{P}H^*$ is the distributive law from Example 2.7) gives the same values when precomposed with π_1 or with π_2. Thus the universal property of coequalizer yields a unique morphism λ'_X making the following commute:



Commutativity of the above diagram yields condition ($\diamondsuit$) of Theorem 6.5. $\square$

Continuing with Example 6.7 we are thus entitled to apply the quotient semantics $[\![-]\!]_\sim$. This will be given on an NDA $e\colon X \to \widehat{H}X$ by first embedding it into $\widehat{H}^*$ as $\bar{e} = \kappa_X \circ e\colon X \to \widehat{H}^*X$ and then into $\widehat{R}$ as $\widehat{\xi}_X \circ \bar{e}\colon X \to \widehat{R}X$. To this morphism we apply the canonical fixpoint operator of $\widehat{R}$ to obtain

$(\widehat{\xi}_X \circ \bar{e})^{\sim}$, that is, the semantics $[\![e]\!]_\sim \colon X \to R0 = A^*/\equiv$. It is easy to see that this definition captures the intended semantics: for all states $x \in X$

$$[\![e]\!]_\sim(x) = \{[w]_\equiv \mid w \in [\![e]\!](x)\}.$$

Indeed, by Proposition 6.3, $[\![e]\!]_\sim = \widehat{\xi_0} \circ [\![e]\!]$ and $\widehat{\xi_0} \colon \widehat{H}^*0 \to \widehat{R}0$ is just $\mathcal{J}\xi_0$ where $\xi_0 \colon A^* \to A^*/_\equiv$ maps every word w into its equivalence class $[w]_\equiv$.

7. Non-Deterministic Transducers

We now consider another application of our theory, namely to non-deterministic transducers. Introduced by Schützenberger [36], these systems are a generalisation of classical automata by allowing each transition to produce an output word. They have been employed in various areas of computational linguistics [27, 32]. The question of whether transducers could be modelled coalgebraically was tackled by Hansen [21], though her results were only about deterministic transducers and did not capture the semantics in a fully satisfactory way.

In this section, we consider the more general case of non-deterministic transducers and show how their semantics can be correctly modeled in a Kleisli category. Later, we shall also extend our approach to transducers with internal behaviour.

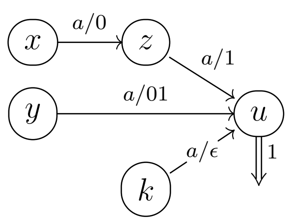
Formally, a *non-deterministic transducer* with inputs in A and outputs in B is a tuple (X, δ, o) where X is a set of states, $\delta \subseteq X \times A \times B^* \times X$ is a transition relation and $o \colon X \to \mathcal{P}(B^*)$ is a terminal output function associating to each state a language over B. To avoid confusion amongst the words in A^* and those in B^*, we use $w, w_1, w_2, \ldots$ for the former and $v, v_1, v_2, \ldots$ for the latter. Moreover, we write $x \xrightarrow{a/v} y$ for $(x, a, v, y) \in \delta$ and $x{\Downarrow_v}$ for $v \in o(x)$. We shall simply speak of transducers dropping "non-deterministic" whenever we feel like it.

Every state $x \in X$ induces a function $\|x\| \colon A^* \to \mathcal{P}(B^*)$ mapping a word $w \in A^*$ into the set

$$\begin{aligned}\|x\|(w) = \{v \mid \exists x_i \in X, a_i \in A, v_i \in B^* \text{ s.t. } w = a_1 \cdots a_n, \\ v = v_1 \cdots v_{n+1} \text{ and } x \xrightarrow{a_1/v_1} x_1 \cdots \xrightarrow{a_n/v_n} x_n{\Downarrow_{v_{n+1}}}\}.\end{aligned}$$

For example consider the transducer $\mathcal{A}$ (below on the left) with input alphabet $A = \{a\}$ and output alphabet $B = \{0, 1\}$. The function o has value $\{1\}$

on u and the empty language on the other states of $\mathcal{A}$. In the central column we show the function $\| \cdot \|$ for each state; the unspecified words of A^* are mapped to the empty language on B.



$\|x\|\colon aa \mapsto \{011\}$	$e(x) = \{(0, a, z)\}$
$\|z\|\colon a \mapsto \{11\}$	$e(z) = \{(1, a, u)\}$
$\|u\|\colon \epsilon \mapsto \{1\}$	$e(u) = \{(1, \checkmark)\}$
$\|y\|\colon aa \mapsto \{011\}$	$e(y) = \{(01, k)\}$
$\|k\|\colon a \mapsto \{1\}$	$e(k) = \{(\epsilon, a, u)\}$

We model non-deterministic transducers as coalgebras

$$e\colon X \to \mathcal{P}(B^* \times (A \times X + 1)),$$

letting $(v, a, y) \in e(x)$ if and only if $x \xrightarrow{a/v} y$ and $(v, \checkmark) \in e(x)$ if and only if $x{\Downarrow_v}$. The rightmost column above shows the definition of e for $\mathcal{A}$.

Analogously to the case of non-deterministic automata, in order to properly capture their semantics we want to formally distinguish between the branching and the transition type of transducers. To this aim, we split the functor $\mathcal{P}(B^* \times (A \times \mathrm{Id} + 1))$ as MH with $M = \mathcal{P}(B^* \times \mathrm{Id})$ and $H = A \times \mathrm{Id} + 1$ and, like for NFA, we consider coalgebras for some lifting $\widehat{H}$ of H to $\mathcal{K}\ell(M)$.

7.1. Distributive Laws, again.

Before defining $\widehat{H}$, we have to show that $\mathcal{P}(B^* \times \mathrm{Id})$ is a monad. For this purpose, we can compose the monads $\mathcal{P}$ and $B^* \times \mathrm{Id}$ (in Example 2.1) via the following distributive law:

$$\begin{aligned} \psi_X\colon B^* \times \mathcal{P}X &\to \mathcal{P}(B^* \times X) \\ (w, Y) &\mapsto \{(w, y) \mid y \in Y\} \end{aligned}$$

The unit $\eta_X\colon X \to \mathcal{P}(B^* \times X)$ of the composed monad $\mathcal{P}(B^* \times \mathrm{Id})$ maps $x \in X$ into the singleton $\{(\epsilon, x)\}$. Its multiplication $\mu_X\colon \mathcal{P}(B^* \times \mathcal{P}(B^* \times X)) \to \mathcal{P}(B^* \times X)$ assigns to each $Y \in \mathcal{P}(B^* \times \mathcal{P}(B^* \times X))$ the set

$$\{(v_1 v_2, x) \mid \exists Z \in \mathcal{P}(B^* \times X) \text{ s.t. } (v_1, Z) \in Y \text{ and } (v_2, x) \in Z\}.$$

For defining $\widehat{H}\colon \mathcal{K}\ell(M) \to \mathcal{K}\ell(M)$, we need to provide a distributive law θ of H over M. This is constructed by combining the distributive law

$\varphi\colon H\mathcal{P} \Rightarrow \mathcal{P}H$ (given in Example 2.3) and $\chi\colon H(B^* \times \mathrm{Id}) \Rightarrow B^* \times H(\mathrm{Id})$ with the following components:

$$\begin{aligned} \chi_X\colon A \times (B^* \times X) + 1 &\to B^* \times (A \times X + 1) \\ \checkmark &\mapsto (\epsilon, \checkmark) \\ (a, (w, x)) &\mapsto (w, (a, x)) \end{aligned}$$

Then, we can give $\theta\colon H\mathcal{P}(B^* \times \mathrm{Id}) \Rightarrow \mathcal{P}(B^* \times H(\mathrm{Id}))$ as:

$$\theta_X\colon\ H\mathcal{P}(B^* \times X) \xrightarrow{\varphi_{B^* \times X}} \mathcal{P}H(B^* \times X) \xrightarrow{\mathcal{P}\chi_X} \mathcal{P}(B^* \times HX).$$

By unfolding the definitions of φ and χ, this just means:

$$\begin{aligned} \theta_X\colon A \times \mathcal{P}(B^* \times X) + 1 &\to \mathcal{P}(B^* \times (A \times X + 1)) \\ \checkmark &\mapsto \{(\epsilon, \checkmark)\} \\ (a, Y) &\mapsto \{(w, (a, x)) \mid (w, x) \in Y\} \end{aligned}$$

For φ, ψ and χ, we do not report the proofs showing that these satisfy the equations of distributive laws, since they are straightforward calculations. For θ instead, we can provide a nicer proof relying on the following general result, which is a variation of [16, Theorem 2.1].

Proposition 7.1. *Let M, T be monads and F an endofunctor on some category* $\mathbf{C}$*. Also consider distributive laws $\alpha\colon TM \Rightarrow MT$ (of the monad T over the monad M), $\beta\colon FM \Rightarrow MF$ (of the functor F over the monad M) and $\gamma\colon FT \Rightarrow TF$ (of the functor F over the monad T). If the "Yang-Baxter equation" holds, that is, the following diagram commutes*

$$\begin{array}{ccccccc} & & FMT & \xrightarrow{\beta_T} & MFT & & \\ & \overset{F\alpha}{\nearrow} & & & & \overset{M\gamma}{\searrow} & \\ FTM & & & & & & MTF \\ & \underset{\gamma_M}{\searrow} & & & & \underset{\alpha_F}{\nearrow} & \\ & & TFM & \xrightarrow{T\beta} & TMF & & \end{array} \tag{YB}$$

then $\delta\colon FMT \Rightarrow MTF$ defined as $M\beta \circ \gamma_T$ is a distributive law of F over the monad MT.

Proof. See Appendix A. □

Proposition 7.2. *θ is a distributive law of H over the monad $\mathcal{P}(B^* \times \mathrm{Id})$.*

Proof. We instantiate Proposition 7.1 to the case where F, M, T, α, β, γ are H, $\mathcal{P}$, $B^* \times \mathrm{Id}$, ψ, φ, χ, respectively. In order to prove the statement, it suffices to check commutativity of the diagram

$$\begin{array}{ccccc} & H\mathcal{P}(B^* \times X) & \xrightarrow{\varphi_{B^* \times X}} & \mathcal{P}H(B^* \times X) & \\ {\scriptstyle H\psi_X} \nearrow & & & & \searrow {\scriptstyle \mathcal{P}\chi_X} \\ H(B^* \times \mathcal{P}X) & & & & \mathcal{P}(B^* \times HX) \\ {\scriptstyle \chi_{\mathcal{P}X}} \searrow & & & & \nearrow {\scriptstyle \psi_{HX}} \\ & B^* \times H\mathcal{P}X & \xrightarrow{B^* \times \varphi_X} & B^* \times \mathcal{P}HX & \end{array}$$

which is verified by considering the following two cases.

$$\begin{array}{ccccc} & \checkmark & \longmapsto & \{\checkmark\} & \\ \nearrow & & & & \searrow \\ \checkmark & & & & \{(\epsilon, \checkmark)\} \\ \searrow & & & & \nearrow \\ & (\epsilon, \checkmark) & \longmapsto & (\epsilon, \{\checkmark\}) & \end{array}$$

$$\begin{array}{ccccc} & (a, \{(w, y) \mid y \in Y\}) & \longmapsto & \{(a, w, y) \mid y \in Y\} & \\ \nearrow & & & & \searrow \\ (a, w, Y) & & & & \{(w, a, y) \mid y \in Y\} \\ \searrow & & & & \nearrow \\ & (w, a, Y) & \longmapsto & (w, \{(a, y) \mid y \in Y\}) & \end{array}$$

□

By Propositions 2.2 and 7.2 it follows that:

Corollary 7.3. *There exists a lifting* $\widehat{H}$ *of* H *to* $\mathcal{K}\ell(\mathcal{P}(B^* \times \mathrm{Id}))$.

This lifting acts on object X as $\widehat{H}(X) = HX$ and, on morphisms $f\colon X \to Y$ in $\mathcal{K}\ell(\mathcal{P}(B^* \times \mathrm{Id}))$, as $\widehat{H}f\colon A \times X + 1 \to A \times Y + 1$ mapping $\checkmark$ to $\{(\epsilon, \checkmark)\}$ and (a, x) to $\{(v, (a, y)) \mid (v, y) \in f(x)\}$.

7.2. Final Semantics for Transducers

We now have all the ingredients to model the trace semantics of transducers in $\mathcal{K}\ell(\mathcal{P}(B^* \times \mathrm{Id}))$, following the approach of Section 2.4. One can readily check that $\mathcal{K}\ell(\mathcal{P}(B^* \times \mathrm{Id}))$ is **Cppo**-enriched (with composition left strict) and that $\widehat{H}\colon \mathcal{K}\ell(\mathcal{P}(B^* \times \mathrm{Id})) \to \mathcal{K}\ell(\mathcal{P}(B^* \times \mathrm{Id}))$ is locally continuous. Therefore, by Theorem 2.10, the final $\widehat{H}$-coalgebra is $\mathcal{J}\iota^{-1}$ where $\iota\colon A \times A^* + 1 \to A^*$ is, like for NDA (Example 2.11), the initial H-algebra.

For a transducer $e\colon X \to \mathcal{P}(B^* \times (A \times X + 1))$, the final coalgebra homomorphism $!_e\colon X \to A^*$ is the function $X \to \mathcal{P}(B^* \times A^*)$ which maps every state x into the set $\{(v, w) \mid v \in \|x\|(w)\}$. This is displayed in the following diagram, which is the same as (2) for NDA but in $\mathcal{K}\ell(\mathcal{P}(B^* \times \mathrm{Id}))$:

$$
\begin{array}{ccc}
X & \overset{!_e}{\dashrightarrow} & A^* \\
{\scriptstyle e}\downarrow & \begin{array}{l} (v,\epsilon) \in\, !_e(x) \;\Leftrightarrow\; (v,\epsilon) \in e(x) \\ (v,aw) \in\, !_e(x) \;\Leftrightarrow\; \text{for some } v_1, v_2 \in B^*, y \in X \;\; v = v_1 v_2, \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad (v_1, a, y) \in e(x) \text{ and} \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad (v_2, w) \in\, !_e(y) \end{array} & \downarrow{\scriptstyle \mathcal{J}\iota^{-1}} \\
A \times X + 1 & \underset{A \times !_e + 1}{\dashrightarrow} & A \times A^* + 1
\end{array}
$$

In order to read the definition of $!_e$ from the commutativity, it is worth to spell out the various ingredients forming the diagram. First, the composition of two morphisms $f\colon X \to Y$ and $g\colon Y \to Z$ in $\mathcal{K}\ell(\mathcal{P}(B^* \times \mathrm{Id}))$ is the function $g \circ f\colon X \to \mathcal{P}(B^* \times Z)$

$$
\begin{aligned}
x \mapsto \{(v, z) \mid \exists v_1, v_2 \in B^*, y \in Y \text{ s.t. } & v = v_1 v_2, \\
& (v_1, y) \in f(x) \text{ and } (v_2, z) \in g(y)\}.
\end{aligned}
$$

Second, by unfolding the definitions of the unit of $\mathcal{P}(B^* \times \mathrm{Id})$, $\mathcal{J}$ and ι, we have that $\mathcal{J}\iota^{-1}$ maps ϵ into $\{(\epsilon, \checkmark)\}$ and aw into $\{(\epsilon, a, w)\}$. Therefore the morphism passing through the top-right corner is the function $\mathcal{J}\iota^{-1} \circ !_e\colon X \to \mathcal{P}(B^* \times (A \times A^* + 1))$ with

$$
x \mapsto \{(v, \checkmark) \mid (v, \epsilon) \in\, !_e(x)\} \cup \{(v, a, w) \mid (v, aw) \in\, !_e(x)\}.
$$

Third, the morphism through the bottom-left corner is the function $\widehat{H}(!_e) \circ e\colon X \to \mathcal{P}(B^* \times (A \times A^* + 1))$ with

$$
\begin{aligned}
x \mapsto \{(v, \checkmark) \mid (v, \epsilon) \in e(x)\} \cup \{(v, a, w) \mid \exists v_1, v_2 \in B^*, y \in X \text{s.t. } & v = v_1 v_2 \\
& (v_1, a, y) \in e(x), (v_2, w) \in\, !_e(y)\}.
\end{aligned}
$$

It is now easy to see that the equation $\widehat{H}(!_e) \circ e = \mathcal{J}\iota^{-1} \circ !_e$ is equivalent to the two conditions in the above diagram.

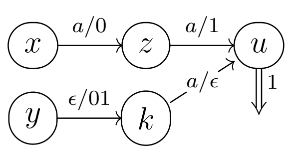
7.3. Transducers with Internal Behaviour

The above semantics of non-deterministic transducers already takes care correctly of the algebraic structure on the output labels in B^*—this is a "built-in" feature of the category $\mathcal{K}\ell(\mathcal{P}(B^* \times \mathrm{Id}))$. However, just as in the case of NDAs, it fails to detect the monoid structure on A^*. We now focus on a case in which this is relevant: transducers with internal behaviour—that is, with transitions labeled on $(A+\{\epsilon\}) \times B^*$ rather than just $A \times B^*$. In order to model their semantics, we shall apply the general framework of Section 5. In fact, the construction presented for NDAs in the previous sections suffices for that purpose: one simply needs to change the underlying category from $\mathcal{K}\ell(\mathcal{P})$ to $\mathcal{K}\ell(\mathcal{P}(B^* \times \mathrm{Id}))$.

As for an NDA, the semantics of a transducer with internal behaviour $e\colon X \to \mathcal{P}(B^* \times (A \times X + 1 + X))$ can be defined by transforming it into the $\widehat{H}^*$-coalgebra $\bar{e} = [\kappa_X, \eta_X] \circ e$ where the natural transformation $[\kappa, \eta]\colon \widehat{H + \mathrm{Id}} \Rightarrow \widehat{H}^*$ is defined for each set X as

$$\begin{aligned}
[\kappa_X, \eta_X]\colon A \times X + 1 + X &\to \mathcal{P}(B^* \times (A^* \times X + A^*)) \\
(a, x) &\mapsto \{(\epsilon, a, x)\} \\
\checkmark &\mapsto \{(\epsilon, \epsilon)\} \\
x &\mapsto \{(\epsilon, \epsilon, x)\}
\end{aligned}$$

For example, consider the following transducer $\mathcal{B}$ with internal behaviour, on input alphabet $A = \{a\}$ and output alphabet $B = \{0, 1\}$.



$$\begin{aligned}
e(x) &= \{(0, a, z)\} & \bar{e}(x) &= \{(0, a, z)\} \\
e(z) &= \{(1, a, u)\} & \bar{e}(z) &= \{(1, a, u)\} \\
e(u) &= \{(1, \checkmark)\} & \bar{e}(u) &= \{(1, \epsilon)\} \\
e(y) &= \{(01, k)\} & \bar{e}(y) &= \{(01, \epsilon, k)\} \\
e(k) &= \{(\epsilon, a, u)\} & \bar{e}(k) &= \{(\epsilon, a, u)\}.
\end{aligned}$$

The semantics of an $\widehat{H}^*$-coalgebra $e\colon X \to \mathcal{P}(B^* \times (A^* \times X + A^*))$ is given by canonical fixpoint as $[\![e]\!] = e^\dagger = ¡ \circ !_e$:

$$\begin{array}{ccccc}
X & \xrightarrow{\;!_e\;} & (A^*)^* \times A^* & \xrightarrow{\;¡\;} & A^* \\
{\scriptstyle e}\downarrow & & \cong & & \uparrow{\scriptstyle \mu_0} \\
A^* \times X + A^* & \xrightarrow[\mathrm{id} \times !_e + \mathrm{id}]{} & A^* \times ((A^*)^* \times A^*) + A^* & \xrightarrow[\mathrm{id} \times ¡ + \mathrm{id}]{} & A^* \times A^* + A^*
\end{array}$$

By following the same arguments as for an NDA, one can readily check that commutativity of the left-hand square (which is the same as (4), but in a different Kleisli category) uniquely makes $!_e$ the unique map such that

$$\begin{aligned}(v, \langle\rangle, w) \in\ !_e(x) &\Leftrightarrow (v, w) \in e(x)\\ (v, w_1 :: l, w) \in\ !_e(x) &\Leftrightarrow \exists v_1, v_2, y\ \ v = v_1 v_2, (v_1, w_1, y) \in e(x)\\ &\qquad\qquad\qquad \text{and } (v_2, l, w) \in\ !_e(y)\end{aligned}$$

Commutativity of the right-hand square makes ¡ the unique map such that

$$\begin{aligned}¡(\langle\rangle, w) &= \{(\epsilon, w)\}\\ ¡(w_1 :: l, w) &= \{(v_3, w_1 w_2) \mid (v_3, w_2) \in ¡(l, w)\}\end{aligned}$$

By composing the two morphisms in $\mathcal{K}\ell(\mathcal{P}(B^* \times \mathrm{Id}))$, one obtains that

$$\begin{aligned}[\![e]\!](x) &= \{(v, w) \mid (v, w) \in e(x)\} \cup\\ &\quad \{(v_1 v_2 v_3, w_1 w_2) \mid \exists y, (l, w) \text{ s.t. } (v_1, w_1, y) \in e(x), (v_2, l, w) \in\ !_e(y)\\ &\qquad\qquad\qquad\qquad \text{and } (v_3, w_2) \in ¡(l, w)\}.\end{aligned}$$

Observe that, by the definition of ¡, $(v_3, w_2) \in ¡(l, w)$ implies that $v_3 = \epsilon$ and that w_2 is just the concatenation of all the words in l and w. Thus, it is easy to see that $(v, w) \in [\![e]\!](x)$ if and only if either $(v, x) \in e(x)$ or there exist v_1, v_2, w_1, w_2 and y such that $(v, w) = (v_1 v_2, w_1 w_2)$, $(v_1, w_1, y) \in e(x)$ and $(v_2, w_2) \in [\![e]\!](y)$.

For the transducer $\mathcal{B}$ given above, we display the value of $!_{\bar{e}}$ and $[\![\bar{e}]\!]$:

$$\begin{aligned}
&!_{\bar{e}}(x) = \{(011, [aa], \epsilon)\} && [\![\bar{e}]\!](x) = \{(011, aa)\}\\
&!_{\bar{e}}(z) = \{(11, [a], \epsilon)\} && [\![\bar{e}]\!](z) = \{(11, a)\}\\
&!_{\bar{e}}(u) = \{(1, \langle\rangle, \epsilon)\} && [\![\bar{e}]\!](u) = \{(1, \epsilon)\}\\
&!_{\bar{e}}(y) = \{(011, [\epsilon, a], \epsilon)\} && [\![\bar{e}]\!](y) = \{(011, a)\}\\
&!_{\bar{e}}(k) = \{(1, [a], \epsilon)\} && [\![\bar{e}]\!](k) = \{(1, a)\}.
\end{aligned}$$

Observe that $!_{\bar{e}}$ takes correctly into account the algebraic structure of B^* but not the one on A^*. The semantics $[\![\bar{e}]\!]$ instead works properly also for A^*, thanks to the quotient ¡.

Just as for the case of NDAs, we can instantiate the theory developed in Section 5.2 to obtain a sound ϵ-elimination procedure for transducers. The relevant diagram is the same as in Example 5.11, but considered in $\mathcal{K}\ell(\mathcal{P}(B^* \times$

Id)) in lieu of $\mathcal{K}\ell(\mathcal{P})$. Concretely, the procedure follows the same rules as the one for NDAs, provided that words of B^* are composed appropriately when creating new transitions. For instance, in transforming the transducer $\mathcal{B}$ of the example above, one just needs to replace the transition $y \xrightarrow{\epsilon/01} k$ with $y \xrightarrow{a/01} u$—observe that output ϵ's are not eliminated since they belong to B^*. As a result, one obtains the transducer $\mathcal{A}$ introduced at the beginning of this section.

8. Discussion

The framework introduced in this paper provides a uniform way to express the semantics of systems with internal behaviour via canonical fixpoint solutions. Moreover, these solutions are exploited to eliminate internal transitions in a sound way, i.e., preserving the semantics. We have shown our approach at work on NDA with ϵ-transitions but, by virtue of Theorem 5.5, it also covers all the examples in [23] (like probabilistic systems) and more (like the weighted automata on positive reals of [37]).

It is worth noticing that, in principle, our framework is applicable also to examples that do not arise from Kleisli categories. Indeed the theory of Section 4 is formulated for a general category $\mathbf{C}$: Assumption 5.1 only requires $\mathbf{C}$ to have coproducts and the monads F^* and $FX + \mathrm{Id}$ to have sufficiently many canonical fixpoints.

As a further remark, let us recall that our original question concerned the problem of modeling the semantics of systems where labels carry an algebraic structure. In this paper we have mostly been focusing on automata theory, but there are many other examples in which the information carried by the labels has relevance for the semantics of the systems under consideration: in logic programming labels are substitutions of terms; in (concurrent) constraint programming they are elements of a lattice; in process calculi they are actions representing syntactical contexts and in tile systems [19] they are morphisms in a category. We believe that our approach provides various insights towards a coalgebraic semantics for these computational models.

The case of process calculi is particularly challenging: internal transitions are typically abstracted away by a notion of observational equivalence called *weak bisimilarity*. Being particularly hard to model by coalgebras, weak bisimilarity has recently captured a renewed interest by the community [12, 13, 20, 14]. Some of these works adopt techniques that are also part of our methodology: for instance, the idea of using systems over a free monad also appeared in [12, 13]. However, the approach is substantially different: rather than eliminating ϵ-transitions, in weak bisimilarity one has to *saturate*, namely to add more transitions to the systems.

Acknowledgments. In between the CMCS'14 paper and this journal version we had the chance to discuss with Alex Simpson at the Bellair Workshops on Duality and Probabilistic Systems. Alex provided us important insights which eventually lead to weaken our assumptions by dropping **Cppo**-enrichement and local continuity. We would like also to thank Tomasz Brengos for the interesting discussion at ETAPS 2014. The work of Alexandra Silva is partially funded by the ERDF through the Programme COMPETE and by the Portuguese Foundation for Science and Technology, project ref. `FCOMP-01-0124-FEDER-020537` and `SFRH/BPD/71956/2010`. The first and the fourth author acknowledge support by project `ANR 12IS0 2001 PACE`.

References

[1] Adámek, J., 1974. Free algebras and automata realizations in the language of categories. Comment. Math. Univ. Carolin. 15, 589–602.

[2] Adámek, J., Milius, S., Velebil, J., 2010. Equational properties of iterative monads. Inform. and Comput. 208, 1306–1348, doi:10.1016/j.ic.2009.10.006.

[3] Adámek, J., Milius, S., Velebil, J., 2011. Elgot theories: a new perspective of the equational properties of iteration. Math. Structures Comput. Sci. 21 (2), 417–480.

[4] Adámek, J., Porst, H.-E., 2004. On tree coalgebras and coalgebra presentations. Theoret. Comput. Sci. 311, 257–283.

[5] Adámek, J., Rosický, J., 1994. Locally presentable and accessible categories. Cambridge University Press.

[6] Asada, K., Hidaka, S., Kato, H., Hu, Z., Nakano, K., 2013. A parameterized graph transformation calculus for finite graphs with monadic branches. In: Peña, R., Schrijvers, T. (Eds.), PPDP. ACM, pp. 73–84.

[7] Balan, A., Kurz, A., 2011. On coalgebras over algebras. Theoretical Comput. Sci. 412(38), 4989–5005.

[8] Barr, M., 1970. Coequalizers and free triples. Math. Z. 116, 307–322.

[9] Bloom, S. L., Ésik, Z., 1993. Iteration Theories: the equational logic of iterative processes. EATCS Monographs on Theoretical Computer Science. Springer.

[10] Bonchi, F., Milius, S., Silva, A., Zanasi, F., 2014. How to kill epsilons with a dagger - A coalgebraic take on systems with algebraic label structure. In: Coalgebraic Methods in Computer Science - 12th IFIP WG 1.3 International Workshop, CMCS 2014, Colocated with ETAPS 2014, Grenoble, France, April 5-6, 2014, Revised Selected Papers. pp. 53–74.

[11] Bonsangue, M. M., Hansen, H. H., Kurz, A., Rot, J., 2013. Presenting distributive laws. In: [24], pp. 95–109.

[12] Brengos, T., 2013. Weak bisimulation for coalgebras over order enriched monads. CoRR abs/1310.3656.
URL `http://arxiv.org/abs/1310.3656`

[13] Brengos, T., 2014. On coalgebras with internal moves. In: Bonsangue, M. M. (Ed.), Coalgebraic Methods in Computer Science - 12th IFIP WG 1.3 International Workshop, CMCS 2014, Colocated with ETAPS 2014, Grenoble, France, April 5-6, 2014, Revised Selected Papers. Vol. 8446 of Lecture Notes in Computer Science. Springer, pp. 75–97.

[14] Brengos, T., Miculan, M., Peressotti, M., 2014. Behavioural equivalences for coalgebras with unobservable moves. CoRR abs/1411.0090.
URL `http://arxiv.org/abs/1411.0090`

[15] Brookes, S. D., Main, M. G., Melton, A., Mislove, M. W., Schmidt, D. A. (Eds.), 1994. Mathematical Foundations of Programming Semantics, 9th International Conference, New Orleans, LA, USA, April 7-10, 1993, Proceedings. Vol. 802 of Lecture Notes in Computer Science. Springer.

[16] Cheng, E., 2011. Iterated distributive laws. Math. Proc. Camb. Philos. Soc. 150 (3), 459–487.

[17] Freyd, P., 1991. Algebraically complete categories. In: Carboni, A., Pedicchio, M. C., Rosolini, G. (Eds.), Category Theory: Proceedings of the International Conference held in Como. Vol. 1488 of Lecture Notes in Math. Springer, pp. 95–104.

[18] Freyd, P. J., 1992. Remarks on algebraically compact categories. In: Applications of Categories in Computer Science: Proceedings of the London Mathematical Society Symposium, Durham 1991. Vol. 177 of London Mathematical Society Lecture Notes Series. Cambridge University Press, pp. 95–106.

[19] Gadducci, F., Montanari, U., 2000. The tile model. In: Plotkin, G. D., Stirling, C., Tofte, M. (Eds.), Proof, Language, and Interaction. The MIT Press, pp. 133–166.

[20] Goncharov, S., Pattinson, D., 2014. Coalgebraic weak bisimulation from recursive equations over monads. In: Esparza, J., Fraigniaud, P., Husfeldt, T., Koutsoupias, E. (Eds.), Automata, Languages, and Programming - 41st International Colloquium, ICALP 2014, Copenhagen, Denmark, July 8-11, 2014, Proceedings, Part II. Vol. 8573 of Lecture Notes in Computer Science. Springer, pp. 196–207.
URL `http://dx.doi.org/10.1007/978-3-662-43951-7_17`

[21] Hansen, H. H., Dec. 2010. Subsequential transducers: A coalgebraic perspective. Inf. Comput. 208 (12), 1368–1397.

[22] Hasuo, I., 2006. Generic forward and backward simulations. In: Baier, C., Hermanns, H. (Eds.), Proc. International Conference on Concurrency Theory (CONCUR'06). Vol. 4137 of Lecture Notes in Comput. Sci. Springer, pp. 406–420.

[23] Hasuo, I., Jacobs, B., Sokolova, A., 2007. Generic trace semantics via coinduction. Log. Methods Comput. Sci. 3 (4:11), 1–36.

[24] Heckel, R., Milius, S. (Eds.), 2013. Algebra and Coalgebra in Computer Science - 5th International Conference, CALCO 2013, Warsaw, Poland, September 3-6, 2013. Proceedings. Vol. 8089 of Lecture Notes in Computer Science. Springer.

[25] Hopcroft, J., Motwani, R., Ullman, J., 2006. Introduction to Automata Theory, Languages, and Computation (3rd Edition). Wesley.

[26] Johnstone, P., 1975. Adjoint lifting theorems for categories of algebras. Bull. London Math. Soc. 7, 294–297.

[27] Jurafsky, D., Martin, J. H., 2009. Speech and Language Processing (2Nd Edition). Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

[28] Kelly, G. M., 1980. A unified treatment of transfinite constructions for free algebras, free monoids, colimits, associated sheaves, and so on. Bull. Austral. Math. Soc. 22, 1–83.

[29] Mazurkiewicz, A., July 1977. Concurrent Program Schemes and Their Interpretation. Aarhus University, Comp. Science Depart., DAIMI PB-78.

[30] Mac Lane, S., 1971. Categories for the Working Mathematician. Springer, Berlin.

[31] Milius, S., Palm, T., Schwencke, D., 2009. Complete iterativity for algebras with effects. In: Kurz, A., Lenisa, M., Tarlecki, A. (Eds.), Proc. Coalgebraic and Algebraic Methods in Computer Science (CALCO'09). Vol. 5728 of Lecture Notes Comput. Sci. Springer, pp. 34–48.

[32] Mohri, M., Jun. 1997. Finite-state transducers in language and speech processing. Comput. Linguist. 23 (2), 269–311.

[33] Mulry, P. S., 1993. Lifting theorems for kleisli categories. In: [15], pp. 304–319.

[34] Plotkin, G. D., Turi, D., 1997. Towards a mathematical operational semantics. In: Proc. Logic in Computer Science (LICS'97). pp. 280–291.

[35] Rabinovich, A. M., 1993. A complete axiomatisation for trace congruence of finite state behaviors. In: [15], pp. 530–543.

[36] Schützenberger, M. P., 1977. Sur une variante des fonctions sequentielles. Theor. Comput. Sci. 4 (1), 47–57.

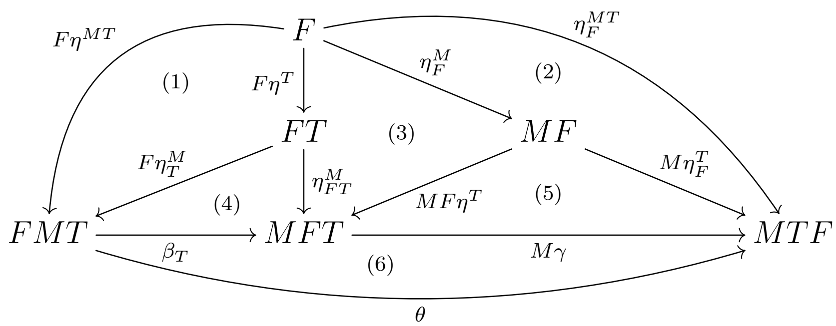
[37] Silva, A., Westerbaan, B., 2013. A coalgebraic view of -transitions. In: [24], pp. 267–281.

[38] Simpson, A., Plotkin, G. D., 2000. Complete axioms for categorical fixed-point operators. In: Proc. 15th Symposium on Logic in Computer Science (LICS'00). IEEE Computer Society, pp. 30–41.

[39] Sobociński, P., 2012. Relational presheaves as labelled transition systems. In: Coalgebraic Methods in Computer Science (CMCS '12). Vol. 7399 of LNCS. Springer, pp. 40–50.

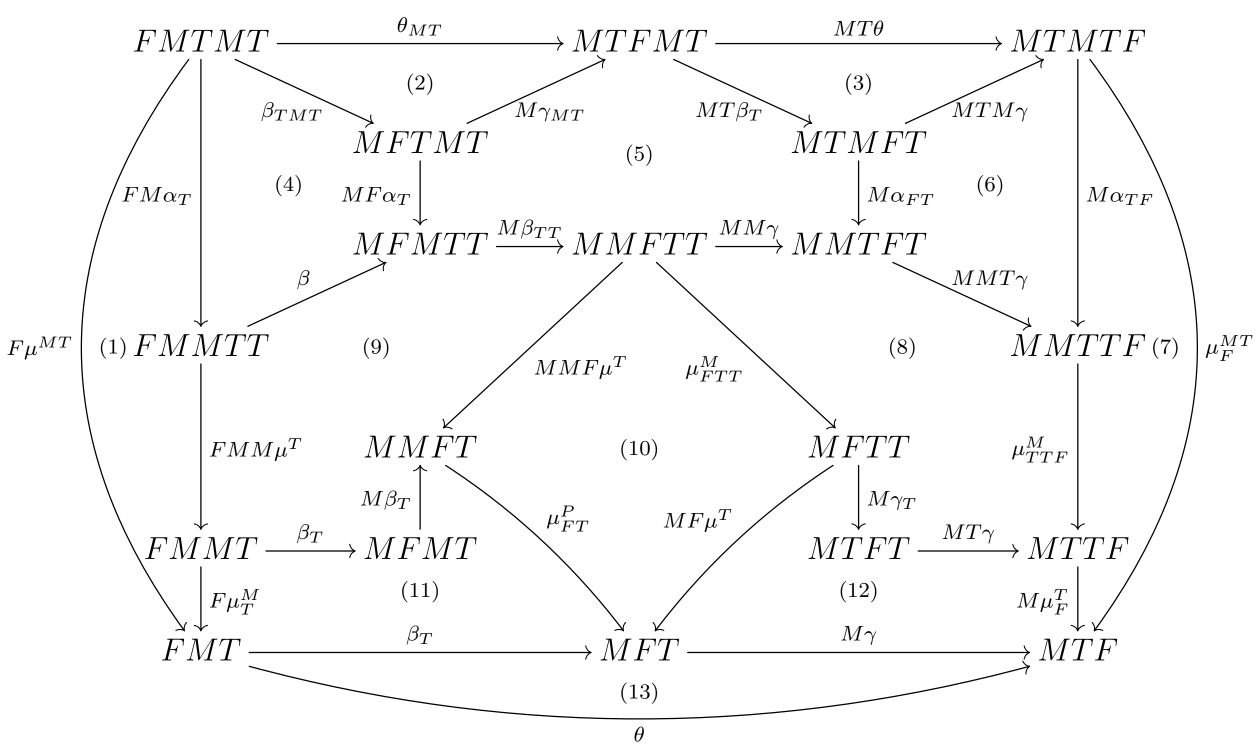
Appendix A. Composing Distributive Laws

Proof of Proposition 7.1. The unit law amounts to commutativity of the following diagram.



(1), (2) and (6) commute by definition of θ and of η^{MT}. (4) and (5) commute because β and γ, respectively, are distributive laws. Finally (3) commutes by naturality of η^M.

Here is the diagram for the multiplication law.



Diagrams (1) and (7) commute by definition of μ^{MT} and (2), (3) and (13) by definition of θ. (4) and (9) commute by naturality of β, (10) and (8) by naturality of μ^M and (6) by naturality of α. Commutativity of diagrams (11) and (12) is because of β and γ being distributive laws. Finally, (5) is just the application of M to diagram (YB), which commutes by assumption. $\square$