

A Coalgebraic Foundation for Linear Time Semantics

John Power¹ and Daniele Turi²

LFCS, University of Edinburgh

Abstract

We present a coalgebraic approach to trace equivalence semantics based on lifting behaviour endofunctors for *deterministic action* to Kleisli categories of monads for *non-deterministic choice*. In *Set*, this gives a category with ordinary transition systems as objects and with morphisms characterised in terms of a linear notion of bisimulation. The final object in this category is the canonical abstract model for trace equivalence and can be obtained by extending the final coalgebra of the deterministic action behaviour to the Kleisli category of the non-empty powerset monad. The corresponding final coalgebra semantics is fully abstract with respect to trace equivalence.

Introduction

The most used models for operational semantics are Plotkin's (*labelled transition systems*) [23]. In recent years, their representation as *coalgebras* [1] – together with associated studies on final coalgebras (eg, [3,4,2,28]) – has proved particularly successful in providing a categorical foundation for much work on operational semantics, creating a widespread interest for coalgebras in computer science (cf [16,17]).

So far, the representation of transition systems as coalgebras has mostly been limited to a particular semantics of (non-deterministic) computation, namely to *branching time (BT)* semantics – and the corresponding *bisimulation* equivalence. Curiously, the simpler *linear time (LT)* semantics and the corresponding (*completed trace equivalence*), where non-deterministic computations are

¹ Research supported by EPSRC grant R34723.

² Research supported by the EU TMR programme and by EPSRC grant R34723.

reduced to sets of deterministic computations, have so far escaped a satisfactory coalgebraic treatment. In this paper we argue that the key to a coalgebraic foundation for LT is a *distributive law* between a behaviour endofunctor for determinism and a monad for non-determinism.

Coalgebras for BT

As mentioned above, the computational interest in coalgebras began with the observation that transition systems are in 1-1 correspondence with the coalgebras of a suitable behaviour endofunctor. The corresponding coalgebra homomorphisms are those functions whose graph is a bisimulation [24] and, crucially, the final object in the resulting category is the intended model for BT [1]. Such a final coalgebra can be used for coinductively defining, for every transition system, a coalgebra homomorphism which abstracts from the name of the states while respecting bisimilarity. When the transition system is relative to a set of programs this is called *final coalgebra semantics* [28].

In [27], coalgebras are shown to play a prominent, conceptual rôle in the most successful approach to operational semantics to date, namely *structural operational semantics (SOS)* [23], where the behaviour of programs is given as a transition system derived from some rules by induction on the structure of the programs. Over the years, various syntactic rule formats for SOS have been isolated which guarantee a good behaviour, typically that bisimulation is a congruence. (See, eg, [6,14,11].) The main contribution of [27] was to bring to light the mathematics underlying such formats through a suitable combination of initial algebra [13] and final coalgebra semantics, but, again, it has so far been restricted to the BT setting. It is our aim to apply it also to the LT setting, therefore we regard the present paper, with its coalgebraic characterization of LT, as a first step in that direction.

Coalgebras for LT

Our coalgebraic treatment of LT starts with a decomposition of the behaviour for BT into a behaviour B for *deterministic action* and a monad \mathcal{P} for *non-deterministic choice*. Lifting B to the Kleisli category of \mathcal{P} gives us a new behaviour \overline{B} whose coalgebras are the same as those for BT, but whose homomorphisms are very different, involving a distributive law between action and choice. We characterise these homomorphisms in terms of a linear notion of bisimulation. This is ordinary bisimulation between a linearization of the transition systems into deterministic transition systems; such a linearization corresponds to the view of the Kleisli category of the monad \mathcal{P} as a category of free semi-lattices.

The crucial observation is that the intended model for LT is a final coalgebra

of our lifted behaviour \overline{B} . Moreover, this final coalgebra is obtained by extending the intended model for deterministic action, a final B -coalgebra, along the canonical functor into the Kleisli category. The associated final coalgebra semantics makes use of the distributive law of B over \mathcal{P} to unfold and abstract non-deterministic computations in a linear way, identifying two programs if and only if they are trace equivalent. Thus it is fully abstract with respect to trace equivalence, rather than with respect to the finer notion of linear bisimulation which is used to characterize the \overline{B} -coalgebra homomorphisms. This is because in the final \overline{B} -coalgebra trace equivalence and linear bisimulation coincide.

The above can all be done using coalgebras over sets, but our final coalgebra theorem depends on the particular behaviour chosen and fails for finite and countable forms of non-determinism. In order to deal also with these important cases, we then move to an algebraically compact, order enriched setting, where we are able to state a much more general result for final coalgebras of liftings of endofunctors to Kleisli category.

Previous and Related Work

Previous coalgebraic work on LT [26,24,29] was based on interpreting the behaviour B for deterministic action in categories of semi-lattices, ie in Eilenberg-Moore rather than in Kleisli categories of monads for non-determinism. (Cf [15].) This yielded the same semantics as here, but the associated category of coalgebras contained extra objects not corresponding to transition systems; moreover, a characterisation of the coalgebra homomorphisms was missing.

Linearized transition systems were already used in [7] for applying the *open maps* approach of [19] to trace equivalence. One of the contributions of the present paper is to show that such a linearization stems from the distributive law between B and \mathcal{P} .

(For more details on the BT and LT models of non-determinism see early work on concurrency, such as, eg, [8].)

Contents In Section 1 we recall some basic facts about transition systems and coalgebras. In Section 2 we define the category of coalgebras of the endofunctor \overline{B} obtained by lifting the deterministic behaviour endofunctor B to the Kleisli category of the non-empty powerset monad \mathcal{P} and we discuss the notion of linear bisimulation. In Section 3 we prove that the final B -coalgebra lifts to the final coalgebra of the lifted endofunctor \overline{B} . In Section 4 we deal with finitary non-determinism by moving to an order enriched setting.

1 Background and Notation

A *coalgebra* $\langle X, h \rangle$ of an endofunctor H on a category \mathcal{C} is given by an object X and a map $h : X \rightarrow H(X)$ of \mathcal{C} ; we shall often refer to a coalgebra $\langle X, h \rangle$ simply by its carrier X or by its structure $h : X \rightarrow H(X)$. The H -coalgebras form a category $H\text{-Coalg}$ with morphisms $f : \langle X, h \rangle \rightarrow \langle X', h' \rangle$ given by maps $f : X \rightarrow X'$ in \mathcal{C} such that $h' \circ f = H(f) \circ h$. (Cf [18].)

We write Id for the identity endofunctor and we denote by

$$\mathcal{P} = \langle \mathcal{P}, \{-\}, \cup \rangle$$

the *non-empty powerset monad*, mapping a set X to the set $\mathcal{P}X$ of its non-empty subsets, having as unit the singleton map $\{-\} : Id \rightarrow \mathcal{P}$ and as multiplication the big union operation $\cup : \mathcal{P}^2 \rightarrow \mathcal{P}$.

We write $A \cdot X$ for the copower of A copies of an object X , in_a for the a -th injection $X \rightarrow A \cdot X$, and $a \cdot x$ for its application to an element x of X . In the sequel we shall often omit writing the left and right injections inl and inr in a coproduct when clear from the context. The notation $a \cdot w$ will also be used for the concatenation of a with a word w over a set A .

Recall that a transition system $X = \langle X, A, \rightarrow \rangle$ consists of a set X of *states*, a set A of *actions* or labels, and a transition relation $\rightarrow \subseteq X \times A \times X$ whose elements are usually written as $x \xrightarrow{a} x'$ [23]. In these transition systems a state x terminates when it cannot make any transitions. In the sequel, for technical reasons explained in Remark 3.2, we shall consider the more general case of *transition systems* $X = \langle X, A, \rightarrow, \downarrow \rangle$ with an explicit termination predicate ' \downarrow ', where transitions and termination can coexist in the sense that both $x \xrightarrow{a} x'$ and $x \downarrow$ can hold. Without loss of generality, we shall consider a fixed set of actions A .

A transition system X is *deterministic* if for all states x in X and actions a in A there exists at most one state x' such that $x \xrightarrow{a} x'$. Every transition system $X = \langle X, A, \rightarrow, \downarrow \rangle$ can be *linearized* into a deterministic transition system $\mathcal{P}X = \langle \mathcal{P}X, A, \dashv, \downarrow \rangle$ with states given by non-empty sets α of states of X ; the transition relation and the termination predicate of $\mathcal{P}X$ are defined using those of X as follows:

$$\begin{aligned} \alpha \dashv \alpha' &\iff \alpha' = \bigcup_{x \in \alpha} \{x' \mid x \xrightarrow{a} x'\} \\ \alpha \downarrow &\iff \exists x \in \alpha. x \downarrow \end{aligned}$$

(Cf [7].)

From a coalgebraic point of view, it is convenient to consider also a stronger

form of deterministic transition systems X where the transition relation and the termination predicate form a function $X \rightarrow 1 + A \times X$. We shall call them *strongly deterministic transition systems*. Since in *Set* the isomorphism $A \times X \cong A \cdot X$ holds, the strongly deterministic transition systems are in 1-1 correspondence with coalgebras of the following *deterministic-action behaviour* endofunctor on *Set*:

$$B = 1 + A \cdot Id \quad . \quad (1)$$

Indeed, given a B -coalgebra structure $h : X \rightarrow 1 + A \cdot X$, one can set

$$\begin{aligned} x \xrightarrow{a} x' &\iff h(x) = a \cdot x' \\ x \downarrow &\iff h(x) = * \end{aligned}$$

and vice versa, where $*$ denotes the only element of the singleton set 1 . The following (folklore) proposition characterizes the final object in the category of B -coalgebras.

Proposition 1.1 The set $A^\infty \cong 1 + A \cdot A^\infty$ of finite and infinite words on A is a final B -coalgebra. \square

Therefore, for every B -coalgebra $X = \langle X, h \rangle$ the unique coalgebra homomorphism $h^b : X \rightarrow A^\infty$ is coinductively defined as follows: for every x in X ,

$$h^b(x) = \begin{cases} a \cdot h^b(x') & \text{if } h(x) = a \cdot x' \\ * & \text{if } h(x) = * \end{cases} \quad (2)$$

The above final coalgebra is the intended model for the deterministic-action behaviour. When the coalgebra $X = \langle X, h \rangle$ corresponds to a strongly deterministic transition system $P = \langle P, A, \rightarrow, \downarrow \rangle$ for a set P of programs one usually writes $\llbracket - \rrbracket : P \rightarrow A^\infty$ instead of h^b ; hence

$$\llbracket p \rrbracket = \begin{cases} a \cdot \llbracket p' \rrbracket & \text{if } p \xrightarrow{a} p' \\ * & \text{if } p \downarrow \end{cases} \quad (3)$$

for all programs p in P . This (final coalgebra) semantics unfolds computations and abstracts from the name of the states involved.

Next, consider the behaviour endofunctor for BT

$$B^\dagger = \mathcal{P}B = \mathcal{P}(1 + A \times Id) \quad . \quad (4)$$

Its coalgebras $h : X \rightarrow \mathcal{P}(1 + A \times X)$ are in 1-1 correspondence with transition systems:

$$\begin{aligned} x \xrightarrow{a} x' &\iff a \cdot x' \in h(x) \\ x \downarrow &\iff * \in h(x) \end{aligned} \tag{5}$$

If we replace \mathcal{P} by its finitary version, the non-empty *finite* powerset, this correspondence cuts down to a correspondence with *finitely branching* transition systems, where each state can perform at most a finite set of transitions. Also, transition systems with a set E of predicates correspond to coalgebras of the endofunctor obtained replacing 1 by E in (4).

Next, we are going to lift the behaviour B to an endofunctor \overline{B} on the Kleisli category of the monad \mathcal{P} ; a \overline{B} -coalgebra will be the same as a B^\dagger -coalgebra, but the coalgebra homomorphisms will be different.

2 Transition systems as coalgebras in a Kleisli category

We write $Kl(T)$ for the *Kleisli category* of a monad $T = \langle T, \eta, \mu \rangle$ on \mathcal{C} and

$$J : \mathcal{C} \rightarrow Kl(T) \tag{6}$$

for the canonical functor associated to it. The latter is the identity on objects and precomposes maps with the unit: $(f : X \rightarrow Y) \mapsto (\eta_Y \circ f : X \rightarrow TY)$.

An endofunctor H on \mathcal{C} *lifts* to an endofunctor \overline{H} on $Kl(T)$ if $J \circ H = \overline{H} \circ J$. As shown in the following proposition, such liftings are equivalent to *distributive laws* [5] of H over $T = \langle T, \eta, \mu \rangle$, ie natural transformations $\lambda : HT \Rightarrow TH$ such that

$$\lambda \circ H\eta = \eta_H \quad \text{and} \quad \lambda \circ H\mu = \mu_H \circ T\lambda \circ \lambda_T \quad .$$

Proposition 2.1 (Cf [21].) For any category \mathcal{C} , any monad T on \mathcal{C} and any endofunctor H on \mathcal{C} , to give a lifting \overline{H} of H to $Kl(T)$ is equivalent to giving a distributive law of H over T .

Proof. Given the distributive law, the action of \overline{H} on objects of $Kl(T)$ is determined. For homs, \overline{H} is given by applying H and then composing with the distributive law. For the converse construction, consider the behaviour of the lifting \overline{H} on the counit $\epsilon : T \Rightarrow Id$ of the Kleisli adjunction. It is routine to verify that these operations are mutually inverse. \square

Proposition 2.2 Let $T = \langle T, \eta, \mu \rangle$ be a monad on a category \mathcal{C} .

- (1) If \mathcal{C} has binary products, then for every object E in \mathcal{C} the natural transformation

$$[T(\text{inl}) \circ \eta_E, T(\text{inr})] : E + T \Longrightarrow T(E + Id)$$

is a distributive law of the endofunctor $E + Id$ over the monad T .

- (2) If \mathcal{C} has A -copowers, for a set A , then the A -copairing of the maps

$$T(\text{in}_a) : TId \Longrightarrow T(A \cdot Id)$$

for a in A , is a distributive law of the endofunctor $A \cdot Id$ over the monad T .

Proof. Decompose the relevant diagrams using the injections and then use the naturality of the operations of the monad. \square

We can then apply the above proposition to $T = \mathcal{P}$ and $E = 1$ to obtain a distributive law

$$\lambda : 1 + A \cdot \mathcal{P} \Longrightarrow \mathcal{P}(1 + A \cdot Id) \quad (7)$$

mapping $*$ to $\{*\}$ and $a \cdot \alpha$ to $\{a \cdot x \mid x \in \alpha\}$.

Corollary 2.3 The endofunctor $B = 1 + A \cdot Id$ on *Set* lifts to an endofunctor \overline{B} on $Kl(\mathcal{P})$. \square

The lifted endofunctor \overline{B} acts on objects as B , while its action on morphisms $f : X \rightarrow \mathcal{P}Y$ is given by

$$\overline{B}(f) = \lambda_Y \circ B(f) : BX \rightarrow \mathcal{P}BY \quad .$$

2.1 Linear Time Coalgebras

Clearly, the \overline{B} -coalgebras are the same as coalgebras of the branching time behaviour B^\dagger . The interest lies in the coalgebra homomorphisms, which are very different from the B^\dagger -coalgebras homomorphisms. Indeed, a \overline{B} -coalgebra homomorphism $f : \langle X, h \rangle \rightarrow \langle Y, k \rangle$ is a function $f : X \rightarrow \mathcal{P}Y$ such that the

diagram

$$\begin{array}{ccc}
X & \xrightarrow{f} & \mathcal{P}Y \\
\downarrow h & & \downarrow \mathcal{P}k \\
\mathcal{P}BX & \xrightarrow{\mathcal{P}Bf} \mathcal{P}B\mathcal{P}Y & \xrightarrow{\mathcal{P}\lambda} \mathcal{P}^2BY & \xrightarrow{\cup} & \mathcal{P}BY \\
& & & & \downarrow \cup \\
& & & & \mathcal{P}^2BY \\
& & & & \downarrow \mathcal{P}k \\
& & & & \mathcal{P}BY
\end{array}
\qquad
\begin{array}{c}
Y \\
\downarrow k \\
\mathcal{P}BY
\end{array}$$

commutes. This commutativity can be characterised in terms of transition systems as follows.

Let X and Y be two transition systems and $\mathcal{P}Y$ be the linearization of Y as defined in Section 1. Then a function $f : X \rightarrow \mathcal{P}Y$ is a \overline{B} -coalgebra homomorphism if and only if for all x in X and a in A the following two clauses hold:

- $f(x) \xrightarrow{a} \beta \iff \beta = \cup \{f(x') \mid \exists x'. x \xrightarrow{a} x'\}$,
- $f(x) \downarrow \iff x \downarrow$.

An equivalent characterisation, wholly in terms of the linearized transition systems, can be obtained by using the following linear transposition of f :

$$f^\sharp : \mathcal{P}X \rightarrow \mathcal{P}Y \quad f^\sharp(\alpha) \stackrel{\text{def}}{=} \bigcup_{x \in \alpha} f(x) \quad . \quad (8)$$

Then a function $f : X \rightarrow \mathcal{P}Y$ is a \overline{B} -coalgebra homomorphism if and only if for all α in $\mathcal{P}X$

- $f^\sharp(\alpha) \xrightarrow{a} \beta \iff \alpha \xrightarrow{a} \alpha' \wedge \beta = f^\sharp(\alpha')$,
- $f^\sharp(\alpha) \downarrow \iff \alpha \downarrow$.

Let $Casl$ be the category of *complete affine semi-lattices* (ie semi-lattices with joins of arbitrary non-empty sets of elements) and join-preserving maps. The monad \mathcal{P} arises from the familiar adjunction

$$F \dashv U : Set \rightarrow Casl \quad .$$

This allows us to regard $Kl(\mathcal{P})$ as the full subcategory of free objects in $Casl$ and underlies both the linear transposition of (8) and the alternative presentation of \overline{B} -coalgebra homomorphisms given above.

Since left adjoint preserve colimits and since the above left adjoint F preserves the final object $F1 \cong 1$, we can extend the structure of the deterministic-action

behaviour B along F and obtain the endofunctor

$$\tilde{B} \stackrel{\text{def}}{=} 1 + A \cdot Id : Casl \longrightarrow Casl$$

with the property that $FB \cong \tilde{B}F$. The restriction of this endofunctor \tilde{B} to the subcategory of *finite* affine semi-lattices is used in previous coalgebraic approaches to LT [26,24,29].

The following proposition relates \tilde{B} to our lifted behaviour \overline{B} .

Proposition 2.4 Let $K : Kl(\mathcal{P}) \longrightarrow Casl$ be the canonical comparison functor. Then \tilde{B} extends \overline{B} along K in the sense that:

$$K\overline{B} \cong \tilde{B}K \quad .$$

□

Corollary 2.5 Given two \overline{B} -coalgebras $\langle X, h \rangle$ and $\langle Y, k \rangle$, a function $f : X \longrightarrow \mathcal{P}Y$ is a \overline{B} -coalgebra homomorphism if and only if the diagram

$$\begin{array}{ccc} FX & \xrightarrow{f^\sharp} & FY \\ h^\sharp \downarrow & & \downarrow k^\sharp \\ \tilde{B}FX & \xrightarrow{\tilde{B}f^\sharp} & \tilde{B}FY \end{array}$$

commutes.

□

Therefore, $\overline{B}\text{-Coalg}$ is a full subcategory of $\tilde{B}\text{-Coalg}$.

2.2 Linear Bisimulation

A function between two transition systems is a B^\dagger -coalgebra homomorphism if and only if its graph is a (strong) bisimulation. (See, eg, [24].) Here we show that a similar characterisation holds for the \overline{B} -coalgebra homomorphisms. The difference between the two is that the latter is with respect to the linearization of the transition systems.

Definition 2.6 A relation $R \subseteq \mathcal{P}X \times \mathcal{P}Y$ is *linear* if it is a subsemilattice of $\mathcal{P}X \times \mathcal{P}Y$, ie for every set I , if $\alpha_i R \beta_i$ for all i in I then $(\bigcup \alpha_i) R (\bigcup \beta_i)$.

A relation R is a *linear bisimulation* between two transition systems X and Y if it is an ordinary bisimulation between the corresponding linearizations $\mathcal{P}X$ and $\mathcal{P}Y$ and, moreover, it is linear. That is, $R \subseteq \mathcal{P}X \times \mathcal{P}Y$ is a linear relation such that for all non-empty subsets α of X and β of Y , $\alpha R \beta$ implies that:

- (1) $\alpha \xrightarrow{a} \alpha' \implies \beta \xrightarrow{a} \beta' \wedge \alpha' R \beta'$,
- (2) $\beta \xrightarrow{a} \beta' \implies \alpha \xrightarrow{a} \alpha' \wedge \alpha' R \beta'$,
- (3) $\alpha \downarrow \iff \beta \downarrow$. □

Note that a linear bisimulation gives a span of homomorphisms between the \overline{B} -coalgebras corresponding to the transition systems, hence it is a coalgebraic \overline{B} -bisimulation in the sense of [27].

Definition 2.7 Given two sets X and Y , the *multigraph* of a function $f : X \rightarrow \mathcal{P}Y$ is the graph R_f of its linear transpose $f^\sharp : \mathcal{P}X \rightarrow \mathcal{P}Y$:

$$R_f \stackrel{\text{def}}{=} \{(\alpha, \beta) \in \mathcal{P}X \times \mathcal{P}Y \mid \beta = f^\sharp(\alpha)\} .$$

□

Note that the linearity of f^\sharp implies that R_f is linear.

The following proposition can be regarded as the coalgebraic equivalent of [7, Lemma 11].

Proposition 2.8 A function $f : X \rightarrow \mathcal{P}Y$ is a \overline{B} -coalgebra homomorphism if and only if its multigraph R_f is a linear bisimulation.

Proof. Assume f is a coalgebra homomorphism. Then $f^\sharp(\alpha) \xrightarrow{a} \beta$ if and only if $\alpha \xrightarrow{a} \alpha'$ and $\beta = f^\sharp(\alpha')$ which ensures that R_f verifies the first two clauses of the definition of linear bisimulation. The third clause is trivial.

Conversely, assume that R_f is a linear bisimulation. Then $\alpha \xrightarrow{a} \alpha'$ implies that $f^\sharp(\alpha) \xrightarrow{a} \beta$ and $\alpha' R_f \beta$. This implies that $\beta = f^\sharp(\alpha')$. Similarly, if $f^\sharp(\alpha) \xrightarrow{a} \beta$ then necessarily $\alpha \xrightarrow{a} \alpha'$ and $\beta = f^\sharp(\alpha')$. Finally, the clause about termination is trivial. □

3 Final coalgebra semantics for linear time

The intended model for LT is the set $\mathcal{P}(A^\infty)$ of non-empty sets of finite and infinite words over A . Given a transition system for a set P of non-deterministic programs, one would like to define a semantic mapping

$$\llbracket - \rrbracket : P \rightarrow \mathcal{P}(A^\infty) \tag{9}$$

such that

$$\llbracket p \rrbracket = \begin{cases} \{a \cdot w \mid p \xrightarrow{a} p', w \in \llbracket p' \rrbracket\} \cup \{*\} & \text{if } p \downarrow \\ \{a \cdot w \mid p \xrightarrow{a} p', w \in \llbracket p' \rrbracket\} & \text{otherwise} \end{cases} \quad (10)$$

for every program p in P . The following theorem provides a coalgebraic foundation for such a coinductive definition.

Theorem 3.1 The final B -coalgebra extends along the canonical functor $J : \text{Set} \rightarrow \text{Kl}(\mathcal{P})$ to the final \overline{B} -coalgebra.

Proof. The claim is that

$$A^\infty \cong 1 + A \cdot A^\infty \xrightarrow{\{-\}} \mathcal{P}(A^\infty)$$

is a final \overline{B} -coalgebra, ie for every \overline{B} -coalgebra $h : X \rightarrow \mathcal{P}BX$ there is a unique \overline{B} -coalgebra homomorphism from X to $\mathcal{P}(A^\infty)$. This amounts to the existence of a unique $h^b : X \rightarrow \mathcal{P}(A^\infty)$ such that the following diagram commutes, where we have used the monad law $\cup \circ \mathcal{P}\{-\} = id_{\mathcal{P}}$ to simplify the right hand side of the rectangle.

$$\begin{array}{ccc} X & \xrightarrow{h^b} & \mathcal{P}A^\infty & & A^\infty \\ \downarrow h & & \downarrow \cong & & \downarrow \cong \\ \mathcal{P}BX & \xrightarrow{\mathcal{P}Bh^b} & \mathcal{P}B\mathcal{P}A^\infty & \xrightarrow{\mathcal{P}\lambda} & \mathcal{P}^2BA^\infty & \xrightarrow{\cup} & \mathcal{P}BA^\infty & & BA^\infty \end{array}$$

Equationally, for every x in X ,

$$h^b(x) = \begin{cases} \{a \cdot w \mid a \cdot x' \in h(x), w \in h^b(x')\} \cup \{*\} & \text{if } * \in h(x) \\ \{a \cdot w \mid a \cdot x' \in h(x), w \in h^b(x')\} & \text{otherwise} \end{cases} \quad (11)$$

Since $h^b(x)$ cannot be empty by assumption, this uniquely determines the set of traces (ie words) that $h^b(x)$ must be. \square

Note that this result generalizes to the case where we use an arbitrary set E instead of 1 in B .

Remark 3.2 We can now explain why we have used transition systems with an explicit termination predicate instead of the more traditional ones with no such predicate. The latter are in 1-1 correspondence with the coalgebras of the behaviour endofunctor

$$B^\ddagger = \mathcal{P}_0(A \cdot Id)$$

where $\mathcal{P}_0(X)$ is the set of all subsets of X , thus including the empty set. If we decompose the behaviour B^\dagger into $A \cdot Id$ and \mathcal{P}_0 we still have a distributive law between the two components, but the above theorem would not hold because of the empty set. Moreover, the final coalgebra for $A \cdot Id$ does not contain the finite words on A . \square

When $\langle X, h \rangle$ corresponds to a transition system for a set P of programs we have that h^\flat is the semantic mapping $\llbracket - \rrbracket$ in (9). Such semantics identifies to programs if and only if they are trace equivalent; in general, given two coalgebras $h : X \rightarrow \mathcal{P}BX$ and $k : Y \rightarrow \mathcal{P}BY$, we call two sets $\alpha \in \mathcal{P}X$ and $\beta \in \mathcal{P}Y$ *trace equivalent* if

$$\bigcup_{x \in \alpha} h^\flat(x) = \bigcup_{y \in \beta} k^\flat(y) \quad .$$

By taking α and β to be the singletons $\{x\}$ and $\{y\}$ we recover the notion of trace equivalence between two states x and y . However, since the image of a state x under a \overline{B} -coalgebra homomorphism f is a set $f(x)$ of states rather than a single state, it is natural to consider trace equivalence between sets of states.

Corollary 3.3 If $f : X \rightarrow \mathcal{P}Y$ is a \overline{B} -coalgebra homomorphism then, for all x in X , $\{x\}$ is trace equivalent to $f(x)$.

Proof. If f is a coalgebra homomorphism then the unique coalgebra homomorphism from X to $\mathcal{P}A^\infty$ factorizes through f , hence:

$$h^\flat(x) = \bigcup_{y \in f(x)} k^\flat(y) \quad .$$

\square

Note that the converse is not true:

Counterexample 3.4 Consider the transition systems

$$X = \{x \xrightarrow{a} x' \downarrow\} \quad \text{and} \quad Y = \{y \xrightarrow{a} y' \downarrow, y \xrightarrow{a} y'' \downarrow\}$$

and the function $f : X \rightarrow \mathcal{P}Y$ with values $f(x) = \{y\}$, $f(x') = \{y'\}$. Then $\{x\}$ and $\{x'\}$ are trace equivalent to $f(x)$ and $f(x')$ respectively, but f is not a coalgebra homomorphism since $f(x) \xrightarrow{a} \{y', y''\} \neq f(x')$. \square

However:

Corollary 3.5 In the final \overline{B} -coalgebra equality, trace equivalence, and linear bisimulation coincide. \square

4 Finite Non-Determinism

Theorem 3.1 can be routinely generalized a little: it holds not only for \mathcal{P} as we have defined it but, given any uncountable cardinal κ (and assuming the continuum hypothesis) also for the monad taking a set to the set of non-empty subsets of cardinality no greater than κ . However, we have:

Counterexample 4.1 If one replaces \mathcal{P} by the *finite, non-empty* powerset monad \mathcal{P}_f in the above, the final B -coalgebra does not extend to a final \overline{B} -coalgebra. The reason is that one routinely follows the proof of the theorem, then discovers that one may have more than a finite set of traces. More specifically, taking A to be a two element set, say $A = \{0, 1\}$, it follows that $h^b(x)$ could be any stream of 0's and 1's, and there are uncountably many such. \square

As a consequence there are probably no reasonable conditions on a finitary monad T on Set such that a final B -coalgebra lifts to $Kl(T)$: the example of the finitary monad given by freely adding a binary operator subject to associativity, commutativity, and idempotence laws is a leading example.

This situation is unattractive for several reasons. In particular, finitely branching transition systems arise most naturally in practice, so a theory that excludes them is very narrow. We address that by passing from the category Set to algebraically compact enriched categories, for which the leading example is the ω -cpo enriched category of ω -cpo's with least element, and structure preserving maps.

Bipolar Endofunctors

Let \mathcal{V} be a symmetric monoidal closed category. A \mathcal{V} -endofunctor is called *bipolar* or *algebraically compact* if it has both an initial algebra and a final coalgebra such that the canonical map from the former to the latter is an isomorphism. (See [12].) A \mathcal{V} -category is called *bipolar* if every \mathcal{V} -endofunctor on it is bipolar. Let $H-Alg$ be the category of algebras $HX \rightarrow X$ of a \mathcal{V} -endofunctor H .

The definition of lifting and distributive law and the relationship between them extends from categories as in Proposition 2.1 to \mathcal{V} -categories as we shall study in this section. We use the same notation as that of Section 2; in particular, \overline{H} denotes the lifting of a \mathcal{V} -endofunctor H .

Proposition 4.2 For every lifting \overline{H} of a \mathcal{V} -endofunctor H , the Kleisli adjunction lifts to an adjunction between $H-Alg$ and $\overline{H}-Alg$.

Proof. This can be proved directly by the universal property of the construction $K\text{-Alg}$ applied to each of H and \overline{H} . It holds in any 2-category that admits the construction of algebras, in particular in any 2-category of the form $\mathcal{V}\text{-Cat}$ for symmetric monoidal closed complete \mathcal{V} . \square

Corollary 4.3 Let \mathcal{C} be a \mathcal{V} -category, let T be a \mathcal{V} -monad on \mathcal{C} and let H be a bipolar \mathcal{V} -endofunctor on \mathcal{C} which lifts to a \mathcal{V} -endofunctor \overline{H} on $Kl(T)$. If \overline{H} is bipolar then a final H -coalgebra extends along the canonical functor $J : \mathcal{C} \rightarrow Kl(T)$ to a final \overline{H} -coalgebra.

Proof. A final H -coalgebra is, by bipolarity, an initial H -algebra. By Proposition 4.2, it is sent to an initial \overline{H} -algebra, which, in turn, is a final \overline{H} -coalgebra. \square

Note that the above proof is fundamentally different to our proof of the previous section: the category Set is not bipolar, and in particular, the endofunctor $1 + A \cdot Id$ does not satisfy the bipolarity condition either, as the initial algebra is given by the finite words A^* on A , while the final coalgebra A^∞ also contains infinite words.

Let $Cppo_\perp$ be the category of $(\omega\text{-})\text{cpos}$ with a least element and strict continuous maps, and with pointwise partial order on the homs. $Cppo_\perp$ is a full reflective subcategory of Cpo , the symmetric monoidal closed category of cpos and continuous functions. In the sequel, the *lifting* functor $(-)_\perp$ will denote both the left adjoint, the induced monad (on Cpo) and the induced comonad (on $Cppo_\perp$) of the reflection.

Since $Cppo_\perp$ is a bipolar Cpo -category [10] we can try to apply Theorem 4.3 to its endofunctors, and to a Cpo -enriched version of B in particular. There are several endofunctors which one can associate to (strong) deterministic behaviour depending on the use that one makes of the lifting endofunctor. The following proposition takes care of this extra data.

Proposition 4.4 Let up be the unit of the lifting monad $(-)_\perp$. For every monad T on $Cppo_\perp$ the map

$$(TId)_\perp \rightarrow T(Id_\perp)$$

obtained by transposing

$$T(up) : TId \rightarrow T(Id_\perp)$$

across the reflection is a distributive law of $(-)_\perp$ over T . \square

Our main theorem does not hold only in $Cppo_\perp$, or even only for bipolar Cpo -categories. More generally, we consider a bipolar Cpo -endofunctor on a Cpo -category with some limiting and colimiting properties. The crucial result

we use here is the following, where \mathcal{C}_E denotes the subcategory of a *Cpo*-category \mathcal{C} , with the same objects and with maps given by the *embeddings* in \mathcal{C} , ie maps f with a right adjoint retract f^R .

Theorem 4.5 (Limit/Colimit Coincidence [25]) In a *Cpo*-category \mathcal{C} , given an ω -chain $\Gamma : \omega \rightarrow \mathcal{C}$ and a cone $\gamma : \Gamma \Rightarrow C$ for it, the following are equivalent:

- (1) γ is a colimiting cone.
- (2) γ is a cone in \mathcal{C}_E with $\bigvee \gamma_n \circ (\gamma_n)^R = id_C$.
- (3) γ is *Cpo*-colimiting in \mathcal{C} .
- (4) γ^R is a limiting cone.
- (5) γ^R is *Cpo*-limiting in \mathcal{C} .

Proof. See [10, Thm. 5.4.5]. □

Using this result we deduce:

Theorem 4.6 Let H be a *Cpo*-endofunctor on a *Cpo*-category \mathcal{C} with a zero object and colimits of ω -chains. Assume that for every object X in \mathcal{C} the composite morphism $X \rightarrow 1 = 0 \rightarrow X$ is the least element of $\mathcal{C}(X, X)$. Let T be a *Cpo*-monad on \mathcal{C} such that the canonical functor $J : \mathcal{C} \rightarrow Kl(T)$ preserves the zero object. If H is bipolar and lifts to a *Cpo*-endofunctor \overline{H} on $Kl(T)$, then \overline{H} is bipolar.

Proof. As H is *Cpo*-enriched, an initial H -algebra may be given as a colimit of an ω -chain. By our assumptions that ω -chain lives in \mathcal{C}_E , hence, by the limit-colimit coincidence, it may also be given equationally by use of the adjoints in the maps of the colimiting diagram. The data and equations are preserved by any *Cpo*-endofunctor, so in particular by T . It follows that the presentation of the initial H -algebra as a colimit is sent by J to a presentation of the initial \overline{H} -algebra as a colimit. Moreover, the assumption that J preserves the zero object implies that also the presentation of the final H -coalgebra as a limit is sent by J to a presentation of the final \overline{H} -coalgebra as a limit. Thus the limit-colimit coincidence yields the initial \overline{H} -algebra, the final \overline{H} -coalgebra, and the isomorphism between them. □

The equivalent in $Cppo_{\perp}$ of the non-empty finite powerset is the convex powerdomain. This satisfies the conditions of the above theorem, and so do its two variations, the lower and the upper powerdomains. (Cf [22].)

5 Conclusions and Future Work

We have established a coalgebraic foundation for a linear form of bisimulation as needed in linear time semantics. From here we would like to move in three directions. Firstly, we would like to consolidate our results by exploring in more detail the relationship between linear bisimulation, trace equivalence, the coalgebraic equivalence corresponding to our behaviour endofunctor, and the work on open maps in [7].

Secondly, we would like to give an explicit description of the various final coalgebra semantics arising in the *Cpo*-enriched setting, understanding the effect that the choice of powerdomain has on the semantics.

Thirdly, we want to fit the categorical theory of structural operational semantics of [27] in the Kleisli setting. Preliminary results suggest that a syntactic format of structural operational rules for our behaviour \overline{B} would satisfy the expected linearity constraint that each premise in a rule can only be used once. This is a consequence of the use of a (lifted) deterministic-action behaviour rather than the usual non-deterministic behaviour.

We have not discussed here the possibility of working in categories of complete metric spaces. There, we expect that working with the closed and compact (non-empty) metric powerdomains should yield the metric version [9] of the semantics in (10) and its finitary restriction respectively.

On a more speculative level, the idea of having a basic notion of observation given by the endofunctor B and then adding non-determinism by lifting B to the Kleisli category of a monad for non-determinism seems to bring the coalgebraic approach to operational semantics a little closer to Moggi's work on computational monads [20].

Acknowledgements. Discussions with Julian Rathke and with Gordon Plotkin, Alex Simpson and Martin Wehr are gratefully acknowledged.

References

- [1] P. Aczel. *Non-well-founded sets*. Number 14 in Lecture Notes. CSLI, 1988.
- [2] J. Adamek and V. Koubek. On the greatest fixed point of a set functor. *Theoretical Computer Science*, 150(1):57–75, October 1995.
- [3] P. Aczel and N.P. Mendler. A final coalgebra theorem. In D. H. Pitt, D. E. Rydeheard, P. Dybjer, A. M. Pitts, and A. Poigné, editors, *Proceedings of the*

Conference on Category Theory and Computer Science, volume 389 of *LNCS*, pages 357–365, Berlin, September 1989. Springer.

- [4] M. Barr. Terminal coalgebras in well-founded set theory. *Theoretical Computer Science*, 114(2):299–315, June 1993.
- [5] J. Beck. Distributive laws. In B. Eckmann, editor, *Seminar on Triples and Categorical Homology Theory*, volume 80 of *Lecture Notes in Mathematics*, pages 119–140. Springer-Verlag, 1969.
- [6] B. Bloom, S. Istrail, and A.R. Meyer. Bisimulation can't be traced. *Journal of the ACM*, 42(1):232–268, jan 1995. A preliminary report appeared in *Proc. 3rd LICS*, pages 229–239, 1988.
- [7] A. Cheng and M. Nielsen. Open maps (at) work. Technical Report RS-95-23, BRICS, April 1995. Appears with the title *Observing Behaviour Categorically* in *LNCS* 1026, pages 263–278.
- [8] J.W. de Bakker, J.A. Bergstra, J.W. Klop, and J.-J. Meyer. Linear time and branching time semantics for recursion with merge. *Theoretical Computer Science*, 34(1-2):135–156, November 1984.
- [9] J.W. de Bakker and J.-J. Meyer. Metric semantics for concurrency. *BIT*, 28(3):504–529, 1988.
- [10] M.P. Fiore. *Axiomatic Domain Theory in Categories of Partial Maps*. Distinguished Dissertations Series. Cambridge University Press, 1996.
- [11] W. Fokkink and R. van Glabbeek. Ntyft/ntyxt rules reduce to ntree rules. *Information and Computation*, 126(1):1–10, 1996.
- [12] P.J. Freyd. Algebraically complete categories. In A. Carboni et al., editors, *Proc. 1990 Como Category Theory Conference*, pages 95–104, Berlin, 1991. Springer-Verlag. *Lecture Notes in Mathematics* Vol. 1488.
- [13] J.A. Goguen, J.W. Thatcher, and E.G. Wagner. An initial algebra approach to the specification, correctness and implementation of abstract data types. In R.T. Yeh, editor, *Current Trends in Programming Methodology*, volume IV, pages 80–149. Prentice Hall, 1978.
- [14] J.F. Groote and F. Vaandrager. Structured operational semantics and bisimulation as a congruence. *Information and Computation*, 100(2):202–260, 1992.
- [15] M.C.B. Hennessy and G.D. Plotkin. Full abstraction for a simple parallel programming language. In J. Bečvář, editor, *Proc. 8th MFCS*, volume 74 of *LNCS*, pages 108–120. Springer-Verlag, 1979.
- [16] B. Jacobs, L. Moss, H. Reichel, and J. Rutten, editors. *CMCS'98 First Workshop on Coalgebraic Methods in Computer Science*, volume 11. ENTCS, 1998.

- [17] B. Jacobs and J. Rutten, editors. *CMCS'99 Coalgebraic Methods in Computer Science*, volume 19. ENTCS, 1999.
- [18] P.T. Johnstone, A.J. Power, T. Tsujishita, H. Watanabe, and J. Worrell. An axiomatics for categories of transition systems as coalgebras. In *LICS: IEEE Symposium on Logic in Computer Science*, 1998.
- [19] A. Joyal, M. Nielsen, and G. Winskel. Bisimulation and open maps. In *Proc. Eighth IEEE Symp. on Logic In Computer Science*, pages 418–427, 1993.
- [20] E. Moggi. Notions of computation and monads. *Information and Computation*, 93:55–92, 1991.
- [21] P. S. Mulry. Lifting theorems for Kleisli categories. In *Mathematical Foundations of Programming Semantics, Proc. 9th Int. Conf.*, volume 802 of *LNCS*, pages 304–319. Springer-Verlag, 1994.
- [22] G.D. Plotkin. Post-graduate lecture notes in advanced domain theory (incorporating the “Pisa Notes”). Dept. of Computer Science, Univ. of Edinburgh, 1981.
- [23] G.D. Plotkin. A structural approach to operational semantics. Technical Report DAIMI FN-19, Computer Science Department, Aarhus University, 1981.
- [24] J. Rutten and D. Turi. Initial algebra and final coalgebra semantics for concurrency. In J. de Bakker et al., editors, *Proc. of the REX workshop A Decade of Concurrency – Reflections and Perspectives*, volume 803 of *LNCS*, pages 530–582. Springer-Verlag, 1994.
- [25] M. Smyth and G. Plotkin. The category-theoretic solution of recursive domain equations. *SIAM J. Comput.*, 11:761–783, 1982.
- [26] D. Turi and B. Jacobs. On final semantics for applicative and non-deterministic languages. Fifth Biennial Meeting on Category Theory and Computer Science, Amsterdam, September 1993.
- [27] D. Turi and G.D. Plotkin. Towards a mathematical operational semantics. In *Proc. 12th LICS Conf.*, pages 280–291. IEEE, Computer Society Press, 1997.
- [28] D. Turi and J. Rutten. On the foundations of final coalgebra semantics: non-well-founded sets, partial orders, metric spaces. *Mathematical Structures in Computer Science*, 8(5):481–540, 1998.
- [29] D. Turi. Categorical modelling of structural operational rules: case studies. In E. Moggi and G. Rosolini, editors, *Proc. 7th CTCS Conf.*, volume 1290 of *LNCS*, pages 127–146. Springer-Verlag, 1997.