URL: http://www.elsevier.nl/locate/entcs/volume19.html 22 pages

A Coequational Approach to Specifying Behaviours

Corina Cîrstea [1]

Computing Laboratory
University of Oxford
Oxford, UK

Abstract

A coalgebraic, equational approach to the specification of observational structures allowing for a choice in the result type of observations is presented. Coalgebraic operations whose result type is structured as a coproduct of basic types are considered, and notions of *coterm*, *covariable* and *coequation*, dual to the algebraic notions of term, variable and equation, are used to specify structures observable through such operations and to constrain their behaviour. A sound and complete deduction calculus for reasoning about observational properties expressed by coequations is then formulated.

1 Introduction

Recent developments in the theory of coalgebras have demonstrated the suitability of coalgebraic techniques for the specification of state-based, dynamical systems [5,7,8]. Such techniques have proved particularly fruitful in specifying observational properties of objects, with final/cofree coalgebras providing appropriate denotations for object specifications [3,2].

Various approaches to reasoning about state observation have also been proposed: in [4,6], ideas from modal logic have been applied to coalgebras, yielding logics whose sentences constrain single state observations, while in [1] equational sentences have been used to relate different observations of the same state. On the one hand, approaches stemming from modal logic can provide characterisability results for coalgebras, at the expense of using infinitary sentences, see [4]; but the formulation of completeness results in such approaches requires a restriction to finitary sentences, as well as the satisfaction of some rather restrictive finiteness conditions by the endofunctors in question, see [6]. On the other hand, equational sentences are not expressive enough to yield

[1] Research supported by an ORS Award and an Oxford Bursary.

similar characterisability results; however, equational approaches do not require any additional assumptions in order to derive completeness results, see [1]. Furthermore, equational sentences appear to be better suited for specifying observational properties quantified over the entire state space of coalgebras (whereas modal logic formulae seem more suitable for characterising single states). Since our aim is to reason effectively about state observations, we shall concentrate on equational approaches.

In [1], suitably restricted algebraic terms are used to denote particular state observations, and equations are used to relate such observations. A sound and complete deduction calculus for equations is then formulated. However, the use in [1] of an algebraic syntax prevents operations with structured result type to be accommodated by the approach, restricting the class of behaviours specifiable in this formalism to behaviours which can be regarded as both algebraic and coalgebraic. Operations with structured result type turn out to be essential for capturing termination, as well as for specifying systems whose structure is variable; in particular, the absence of certain subsystems in some of the system states can be captured naturally by such operations. The present paper extends the approach in [1] in order to accommodate operations whose result type is structured as a coproduct of basic types.

By moving from an essentially algebraic framework to a coalgebraic one, certain algebraic features such as the use of data values as (constant) observations or the use of data arguments to operations are discarded. Our approach could be easily adapted to include such features. However, we believe that their integration should take place at a different level, where it should be possible to specify arbitrary algebraic structures over coalgebraically specified state spaces.

The use in [1] of algebraic terms to denote state observations can not be carried over to our formalism, the reason being the presence of choice in the result type of operations. We therefore introduce a notion of *coterm* which provides alternatives for proceeding with an observation, depending on the type of the result yielded by the operation most recently evaluated. Equational sentences are then used to constrain observations, and a sound and complete deduction calculus for reasoning about the associated behaviours is formulated.

We assume familiarity with basic notions of algebraic specification, as well as with the coalgebraic approach to the specification of state-based systems. (The reader is referred to [8] for a comprehensive introduction to the theory of coalgebras.) The paper is structured as follows. Section 2 introduces a syntax for specifying behaviours observable through operations that allow for a choice in their result type: *coterms* over *destructor signatures* are used to denote observations consisting of successive applications of such operations, with *covariables* being used in coterms as place-holders for their possible results. After defining the models of destructor signatures as coalgebras of endofunctors induced by such signatures, Section 3 provides a concrete description of the elements of the final coalgebra of a destructor signature, as well as of the associated notion of bisimilarity. Section 4 introduces an equational framework for constraining state observations (by relating different such observations) and illustrates the kinds of constraints specifiable in this framework, while Section 5 presents a sound and complete deduction calculus for coequations. Section 6 relates the approach presented here to the ones in [1,6]. Finally, Section 7 summarises the results presented and outlines future work.

2 Cosignatures, Covariables, Coterms and Substitution

A fixed data universe is required by all the forthcoming definitions. We therefore let V denote a set (of visible sorts) and let D denote a V-sorted set (of data values), with $D_v \neq \emptyset$ for each $v \in V$.

Definition 2.1 A **destructor signature (over D)** is a pair (H, Δ) with H a set of **hidden sorts** and Δ an $H \times S^+$-sorted set of **destructor symbols** (where $S = V \cup H$ contains all the sorts, while S^+ denotes the set of finite, non-empty sequences of sorts). We write $\delta : h \to s_1 \ldots s_n$ for $\delta \in \Delta_{h,s_1 \ldots s_n}$.

We only assume that the set Δ of destructors is enumerable. In practice however, Δ is, in most cases, finite.

Destructor symbols specify basic ways of observing system states. Arbitrary state observations are then formalised by the notion of *coterm*, which provides alternatives for all possible result types of destructors. *Covariables* are used in coterms as place-holders for their potential outputs, in a manner similar to the use of variables as place-holders for the inputs of algebraic terms.

Definition 2.2 Let (H, Δ) denote a destructor signature and let $\mathcal{C}$ denote an S-sorted set (of **covariables**). The (S-sorted) set $T_\Delta[\mathcal{C}]$ of **Δ-coterms with covariables from $\mathcal{C}$** is the least S-sorted set satisfying:

- $Z \in T_\Delta[\mathcal{C}]_s$ for $Z \in \mathcal{C}_s$ and $s \in S$
- $[t_1, \ldots, t_n]\delta \in T_\Delta[\mathcal{C}]_h$ for $\delta \in \Delta_{h,s_1 \ldots s_n}$ and $t_i \in T_\Delta[\mathcal{C}]_{s_i}$, $i = 1, \ldots, n$

Coterms of sort $s \in S$ (elements of $T_\Delta[\mathcal{C}]_s$) specify ways of observing states of type s. The result type of a coterm is determined by the sorts of the covariables appearing in it. We note that there are no coterms over an empty set of covariables.

Notation 2.3 For a set $\mathcal{C}$ of covariables, we write $Z : s$ if $Z \in \mathcal{C}_s$. Also, the (S-sorted) set of covariables actually appearing in a coterm $t \in T_\Delta[\mathcal{C}]$ (in general, a subset of $\mathcal{C}$) is denoted $covar(t)$.

Example 2.4 Lists (finite or infinite) are specified using visible sorts `1` (interpreted by D as a one-point set) and `Elt` (denoting the type of the list elements), hidden sorts `List` and `NeList`, and operation symbols: ? : `List` → `1 NeList` (used to classify lists into empty and respectively non-empty ones), `head` :

$\mathtt{NeList} \to \mathtt{Elt}$ (yielding the head of a non-empty list) and $\mathtt{tail} : \mathtt{NeList} \to \mathtt{List}$ (yielding the tail of a non-empty list). The following are coterms of sort $\mathtt{List}$: `[F,N]?` (used to observe whether a list is empty or not), `[F,[E]head]?` (used to observe the first element of a list), `[F,[[F,[E]head]?]tail]?` (used to observe the second element), and so on. The result types of these coterms are: $\mathtt{1} + \mathtt{NeList}$, $\mathtt{1} + \mathtt{Elt}$, and respectively $\mathtt{1} + \mathtt{Elt}$. (The reason for the last coterm having result type $\mathtt{1}+\mathtt{Elt}$, rather than $\mathtt{1}+\mathtt{1}+\mathtt{Elt}$, is its use of two occurrences of the same covariable, rather than of two distinct covariables, of sort $\mathtt{1}$.)

Example 2.5 Binary trees are specified using a visible sort $\mathtt{Leaf}$, hidden sorts $\mathtt{Tree}$ and $\mathtt{NLeaf}$, and operation symbols $? : \mathtt{Tree} \to \mathtt{Leaf}\ \mathtt{NLeaf}$ (classifying trees into leaves and non-leaves) and $\mathtt{left}, \mathtt{right} : \mathtt{NLeaf} \to \mathtt{Tree}$ (yielding the left, respectively right subtree of a tree other than a leaf). It is worth noting the way in which the standard destructor on trees, i.e. $\mathtt{d} : \mathtt{Tree} \to \mathtt{Leaf} + (\mathtt{Tree} \times \mathtt{Tree})$, has been decomposed into three destructors of the form required by destructor signatures.

Substitution of coterms for covariables is now defined as follows.

Definition 2.6 If $t \in T_\Delta[\{Z_1, \ldots, Z_n\}]_h$ with $Z_i : s_i$ for $i = 1, \ldots, n$, and if $t_i \in T_\Delta[\mathcal{C}]_{s_i}$ for $i = 1, \ldots, n$, then the coterm obtained by **substituting** $t_1, \ldots, t_n$ **for** $Z_1, \ldots, Z_n$ **in** t, denoted $[t_1/Z_1, \ldots, t_n/Z_n]t$ ($[\overline{t}/\overline{Z}]t$ for short) is defined inductively on the structure of t as follows:

- $[\overline{t}/\overline{Z}]Z_i = t_i$, for $i = 1, \ldots, n$
- $[\overline{t}/\overline{Z}]([t'_1, \ldots, t'_m]\delta) = [[\overline{t}/\overline{Z}]t'_1, \ldots, [\overline{t}/\overline{Z}]t'_m]\delta$, for $\delta \in \Delta_{h,s'_1 \ldots s'_m}$ and $t'_j \in T_\Delta[\{Z_1, \ldots, Z_n\}]_{s'_j}$, $j = 1, \ldots, m$.

Notation 2.7 Given $t \in T_\Delta[\{Z_1, \ldots, Z_n\}]_h$ with $Z_i : s_i$, $i = 1, \ldots, n$, and given $t' \in T_\Delta[\mathcal{C}]_h$, we write $t \leq t'$ if and only if there exist $t_1 \in T_\Delta[\mathcal{C}]_{s_1}, \ldots, t_n \in T_\Delta[\mathcal{C}]_{s_n}$ such that $t' = [t_1/Z_1, \ldots, t_n/Z_n]t$.

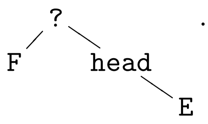
Notation 2.8 For $t \in T_\Delta[\{Z_1, \ldots, Z_n\}]$, we write $\underline{t}$ for a coterm with the following properties:

- $\underline{t} \in T_\Delta[\{X_1, \ldots, X_m\}]$ contains only one occurrence of each covariable
- $t = [Z_{i_1}/X_1, \ldots, Z_{i_m}/X_m]\underline{t}$, with $Z_{i_1}, \ldots, Z_{i_m} \in \{Z_1, \ldots, Z_n\}$.

That is, t is obtained from $\underline{t}$ by renaming (and possibly identifying) some of its covariables. (We note that $\underline{t}$ is only defined up to a bijective renaming of its covariables.)

Remark 2.9 Coterms can be represented as trees having the leaves labelled by covariables and the internal nodes labelled by operation symbols:

- covariables Z are represented as trees having one node, labelled by Z
- coterms of form $[t_1, \ldots, t_n]\delta$ are represented as trees having the root labelled by δ and its subtrees given by the trees associated to $t_1, \ldots, t_n$.

In Example 2.4, the tree associated to `[F,[E]head]?` is 

.

3 Coalgebras, Finality and Bisimilarity

The models of a destructor signature provide particular interpretations for its sorts and operation symbols.

Definition 3.1 Let (H, Δ) denote a destructor signature over D. A **Δ-coalgebra (over D)** is given by an S-sorted set C such that $C_v = D_v$ for each $v \in V$, together with, for each $\delta : h \to s_1 \ldots s_n$ in Δ, a function $\delta_C : C_h \to C_{s_1} + \ldots + C_{s_n}$ (with $+$ denoting the coproduct in Set). Given Δ-coalgebras A and C, a **Δ-homomorphism** $f : A \to C$ is given by an S-sorted function $(f_s)_{s \in S}$ with $f_s : A_s \to C_s$ for $s \in S$, additionally satisfying:

- $f_v = 1_{D_v}$ for each $v \in V$
- $[\iota_1 \circ f_{s_1}, \ldots, \iota_n \circ f_{s_n}](\delta_A(a)) = \delta_C(f_h(a))$ for each $\delta : h \to s_1 \ldots s_n$ in Δ and each $a \in A_h$ (where $\iota_j : C_{s_j} \to C_{s_1} + \ldots + C_{s_n}$ for $j = 1, \ldots, n$ are the coproduct injections).

We let $\mathsf{Coalg}(\Delta)$ denote the category whose objects are Δ-coalgebras and whose arrows are Δ-homomorphisms.

Notation 3.2 Given a Δ-coalgebra C, a set $\{Z_1, \ldots, Z_n\}$ of covariables with $Z_i : s_i$ for $i = 1, \ldots, n$ and a covariable $Z \in \{Z_1, \ldots, Z_n\}$ with $Z : s$, we write $\iota_Z : C_s \to C_{s_1} + \ldots + C_{s_n}$ for the corresponding coproduct injection.

Definition 3.3 Let Δ denote a destructor signature. The **interpretation** of a coterm $t \in T_\Delta[\mathcal{C}]$ in a Δ-coalgebra C, denoted t_C, is defined as follows:

- $Z_C = \iota_Z$ for $Z \in \mathcal{C}_s$ and $s \in S$
- $([t_1, \ldots, t_n]\delta)_C = [(t_1)_C, \ldots, (t_n)_C] \circ \delta_C$ for $\delta \in \Delta_{h, s_1 \ldots s_n}$ and $t_i \in T_\Delta[\mathcal{C}]_{s_i}$, $i = 1, \ldots, n$

with $[(t_1)_C, \ldots, (t_n)_C] : C_{s_1} + \ldots + C_{s_n} \to \coprod\limits_{s \in S} \coprod\limits_{Z \in \mathcal{C}_s} C_s$ denoting the unique Set-arrow induced by $(t_i)_C : C_{s_i} \to \coprod\limits_{s \in S} \coprod\limits_{Z \in \mathcal{C}_s} C_s$, $i = 1, \ldots, n$.

The next result relates models of destructor signatures with coalgebras of endofunctors induced by such signatures.

Proposition 3.4 *Let (H, Δ) denote a destructor signature and let $S = V \cup H$. Also, let Set^S_D denote the category of S-sorted sets whose visible-sorted components are given by D, and S-sorted functions whose visible-sorted components are given by 1_D. Finally, let $\mathsf{G}_\Delta : \mathsf{Set}^S_D \to \mathsf{Set}^S_D$ be given by:*

- $\mathsf{G}_\Delta(X)_v = D_v$, *for $v \in V$*
- $\mathsf{G}_\Delta(X)_h = \prod\limits_{\delta \in \Delta_{h, s_1 \ldots s_n}} (X_{s_1} + \ldots + X_{s_n})$, *for $h \in H$*

Then, the categories $\mathsf{Coalg}(\Delta)$ *and* $\mathsf{Coalg}(\mathsf{G}_\Delta)$ *are isomorphic.*

Proof (Sketch) Δ-coalgebras C uniquely determine Set^S_D-arrows $\gamma : C \to \mathsf{G}_\Delta(C)$ (whose h-component maps $c \in C_h$ to $(\delta_C(c))_{\delta \in \Delta_{h,s_1 \ldots s_n}}$ for each $h \in H$) and conversely, any such Set^S_D-arrow uniquely induces a Δ-coalgebra structure on its domain. □

A characterisation of the abstract behaviours observable using a given set of destructors is provided by (the elements of) final coalgebras. Existence of final coalgebras of destructor signatures is an immediate consequence of Proposition 3.4 and of a general result regarding the existence of final coalgebras of ω^{op}-continuous endofunctors (see e.g. [8]). Here we give an alternative proof of the existence of such final coalgebras which, in addition, provides a concrete description of their elements.

Theorem 3.5 (Final coalgebra) *Any destructor signature* Δ *admits a final coalgebra.*

Proof. For $h \in H$ and a set $\mathcal{C}$ of covariables, we let $T^1_\Delta[\mathcal{C}]_h \subseteq T_\Delta[\mathcal{C}]_h$ consist of those coterms containing exactly one occurrence of each covariable in $\mathcal{C}$. We also let $T^1_{\Delta,h} = (\bigcup\limits_{\mathcal{C}} T^1_\Delta[\mathcal{C}]_h)/_{\leq \cap \geq}$. (Quotienting by $\leq \cap \geq$ identifies those coterms which are the same up to a bijective renaming of their covariables.) That is, $T^1_{\Delta,h}$ contains equivalence classes of coterms in which a covariable may only occur once. For simplicity of notation, we shall refer to such an equivalence class by using an arbitrarily chosen representative. Finally, for each $h \in H$, we let $D_h = \{*\}$. We now define a Δ-coalgebra F by:

$$\begin{aligned} F_h = \{\ & \langle Z_t, d_t \rangle_{t \in T^1_{\Delta,h}} \mid Z_t \in \mathit{covar}(t)_s,\ d_t \in D_s \text{ with } s \in S, \\ & ((t, t' \in T^1_{\Delta,h},\ t' = [t_1/Z_1, \ldots, t_n/Z_n]t,\ Z_t = Z_k) \Rightarrow \\ & (Z_{t'} \in \mathit{covar}(t_k) \text{ and } (t_k = Z_{t'} \Rightarrow d_t = d_{t'}))\ \} \end{aligned}$$

for $h \in H$, and:

$$\delta_F(\varphi) = \begin{cases} d \text{ if } \varphi \text{ at } [Z_1, \ldots, Z_n]\delta \text{ has value } \langle Z, d \rangle \text{ with } Z : v,\ d \in D_v,\ v \in V \\ \langle Z_{[Z_1, .., t', .., Z_n]\delta}, d_{[Z_1, .., t', .., Z_n]\delta} \rangle_{t' \in T^1_{\Delta,h'}} \text{ if } \varphi \text{ at } [Z_1, \ldots, Z_n]\delta \text{ has value} \\ \hfill \langle Z, d \rangle \text{ with } Z : h',\ d \in D_{h'},\ h' \in H \end{cases}$$

for $\varphi \in F_h$, $\delta \in \Delta_{h,s_1 \ldots s_n}$, $h \in H$ and $s_i \in S$, $i = 1, \ldots, n$. That is, the elements of F are observation-indexed pairs $\langle$covariable,value$\rangle$ that are compatible under coterm substitution.

Then, F is a final Δ-coalgebra. For, given an arbitrary Δ-coalgebra C, one can define a Δ-homomorphism $f : C \to F$ by mapping states $c \in C_h$ to $\varphi_c = \langle Z_t, d_t \rangle_{t \in T^1_{\Delta,h}} \in F_h$, with Z_t and d_t being uniquely determined by $t_C(c)$, for each $t \in T^1_{\Delta,h}$: if $t \in T^1_\Delta[\{Z_1, \ldots, Z_n\}]_h$ with $Z_1 : s_1, \ldots, Z_n : s_n$ and if $t_C(c) \in$

$\iota_{Z_i}(C_{s_i})$ for some $i \in \{1, \ldots, n\}$, then $Z_t = Z_i$ and $d_t = \begin{cases} t_C(c) & \text{if } s_i \in V \\ * & \text{if } s_i \in H \end{cases}$. The above definition ensures that f is a Δ-homomorphism. Moreover, any Δ-homomorphism from C to F is necessarily defined in this way. □

Remark 3.6 If $C \in |\mathsf{Set}_D^S|$, a cofree Δ-coalgebra over C is obtained by letting $D_h = C_h$ (instead of $D_h = \{*\}$) for $h \in H$ in the proof of Theorem 3.5.

The next result gives a characterisation of the notion of bisimilarity associated to destructor signatures.

Theorem 3.7 (Bisimilarity) *Let Δ denote a destructor signature and let C denote a Δ-coalgebra. Then, two states $c, c' \in C_h$ with $h \in H$ are bisimilar if and only if for any set $\mathcal{C}$ of covariables and any $t \in T_\Delta[\mathcal{C}]_h$, there exist $s \in S$ and $Z \in \mathcal{C}_s$ such that $t_C(c), t_C(c') \in \iota_Z(C_s)$ and moreover, $t_C(c) = t_C(c')$ if $s \in V$.*

Proof. We show that the relation $\sim$ defined by:

- $c \sim_v c'$ if and only if $c = c'$, for $v \in V$
- $c \sim_h c'$ if and only if for any $t \in T_\Delta[\mathcal{C}]_h$, $t_C(c), t_C(c') \in \iota_Z(C_s)$ for some $Z \in \mathcal{C}_s$ and $s \in S$, and moreover, $t_C(c) = t_C(c')$ if $s \in V$, for $h \in H$

is a bisimulation on C, and that any bisimulation on C is contained in $\sim$.

To show that $\sim$ is a bisimulation, we let $c, c' \in C_h$ with $c \sim_h c'$ and $\delta \in \Delta_{h, s_1 \ldots s_n}$, and show that $\delta_C(c) \sim \delta_C(c')$. Taking $t = [Z_1, \ldots, Z_n]\delta$ in the definition of $\sim$ gives $\delta_C(c), \delta_C(c') \in \iota_{Z_i}(C_{s_i})$ for some $i \in \{1, \ldots, n\}$ with $Z_i : s_i$, and moreover, $\delta_C(c) = \delta_C(c')$ if $s_i \in V$. We distinguish two cases:

(i) $s_i = v \in V$. The fact that $\delta_C(c) \sim_v \delta_C(c')$ then follows immediately from $\delta_C(c) = \delta_C(c')$.

(ii) $s_i = h' \in H$. To show that $\delta_C(c) \sim_{h'} \delta_C(c')$, we let $t' \in T_\Delta[\mathcal{C}']_{h'}$ for some set $\mathcal{C}'$ of covariables, and then let $t = [Z_1, \ldots, t', \ldots, Z_n]\delta$. Since $c \sim_h c'$, it follows that $t_C(c), t_C(c') \in \iota_Z(C_s)$ for some $Z \in (\mathcal{C}' \cup \{Z_1, \ldots, Z_{i-1}, Z_{i+1}, \ldots, Z_n\})_s$, with $s \in S$; furthermore, if $s \in V$ then $t_C(c) = t_C(c')$. Then, $\delta_C(c), \delta_C(c') \in \iota_Z(C_{h'})$ gives $t'_C(\delta_C(c)), t'_C(\delta_C(c')) \in \iota_Z(C_s)$ for some $Z \in \mathcal{C}'_s$ and $s \in S$, and moreover, $t'_C(\delta_C(c)) = t_C(c) = t_C(c') = t'_C(\delta_C(c'))$ if $s \in V$. That is, $\delta_C(c) \sim_{h'} \delta_C(c')$.

Hence, $\sim$ is a bisimulation. To show that $\sim$ is the greatest bisimulation on C, let $\sim'$ denote an arbitrary bisimulation on C. If $h \in H$ and if $c, c' \in C_h$ are such that $c \sim'_h c'$, then one can show by induction on the depth of $t \in T_\Delta[\mathcal{C}]_h$ that $t_C(c)$ and $t_C(c')$ are both in $\iota_Z(C_s)$ for some $Z \in \mathcal{C}_s$ and $s \in S$ and moreover, if $s \in V$ then they coincide. (The fact that $\sim'$ carries coalgebraic structure is used here.) Hence, $c \sim_h c'$. It then follows that $\sim' \subseteq \sim$. This concludes the proof. □

Corollary 3.8 *Let Δ denote a destructor signature, let F denote a final Δ-coalgebra and let $l, r \in T_\Delta[\{Z_1, \ldots, Z_n\}]_h$ with $Z_1 : s_1, \ldots, Z_n : s_n$ and $h \in H$.*

Then, for $\varphi \in F_h$, $l_F(\varphi) = r_F(\varphi)$ if and only if for any set $\mathcal{C}$ of covariables and any $t_i \in T_\Delta[\mathcal{C}]_{s_i}$ for $i = 1, \ldots, n$, $([t_1/Z_1, \ldots, t_n/Z_n]l)_F(\varphi)$ and $([t_1/Z_1, \ldots, t_n/Z_n]r)_F(\varphi)$ are both in $\iota_Z(F_s)$ for some $Z \in \mathcal{C}_s$ and $s \in S$ and moreover, $([t_1/Z_1, \ldots, t_n/Z_n]l)_F(\varphi) = ([t_1/Z_1, \ldots, t_n/Z_n]r)_F(\varphi)$ if $s \in V$.

Proof. The **only if** direction is straightforward. For the **if** direction, it suffices to show that $l_F(\varphi)$ and $r_F(\varphi)$ are bisimilar. Taking $t_i = Z_i$ for $i = 1, \ldots, n$ gives $l_F(\varphi), r_F(\varphi) \in \iota_{Z_i}(F_{s_i})$ for some $i \in \{1, \ldots, n\}$. Then, for any $t \in T_\Delta[\mathcal{C}']_{s_i}$ for some set $\mathcal{C}'$ of covariables, taking $t_j = Z_j$ for $j \in \{1, \ldots, i-1, i+1, \ldots, n\}$ and $t_i = t$ in the hypothesis gives $t_F(l_F(\varphi)) = t_F(r_F(\varphi))$. Hence, $l_F(\varphi) \sim_{s_i} r_F(\varphi)$, which then yields $l_F(\varphi) = r_F(\varphi)$. □

4 Coequational Specification

In algebraic specification, one uses equations to constrain the interpretation of terms by algebras. A similar approach might prove suitable for constraining state observations, as long as one is only interested in relating different observations of the same state. This section formally defines *coequations* and their satisfaction by coalgebras, and illustrates the kinds of constraints they are able to capture.

A first approximation of the notion of coequation is given by a pair of coterms of the same sort. Satisfaction of a coequation by a coalgebra then corresponds to the coalgebra providing similar interpretations for the two coterms. For instance, a coequation of form:

```
[[F,[E]head]?]tail = [E]head
```

constrains the interpretation of `NeList` in any coalgebra C satisfying it to constant, infinite lists (as it requires $\texttt{[[F,[E]head]?]tail}_C$, $\iota_{\texttt{E}} \circ \texttt{head}_C : C_{\texttt{NeList}} \to C_1 + C_{\texttt{Elt}}$ to yield similar results on any non-empty list).

However, due to the presence of choice in the result type of operations, one expects reasoning with coequations to involve some form of case analysis on the result type of coterms. For instance, in order to derive the coequation:

```
[[F,N]?]tail = [[F',N]?]tail
```

(constraining $\texttt{tail}_C$ to always yield a non-empty list, by requiring $[\iota_{\texttt{F}}, \iota_{\texttt{N}}] \circ \texttt{[[F,N]?]tail}_C, [\iota_{\texttt{F'}}, \iota_{\texttt{N}}] \circ \texttt{[[F',N]?]tail}_C : C_{\texttt{NeList}} \to C_1 + C_1 + C_{\texttt{NeList}}$ to yield similar results) from the previous coequation, a case analysis on the result type of `[[F,N]?]tail` should be performed. Specifically, the satisfaction of the above coequation would follow by considering all possible result types for `[[F,N]?]tail` (in this case, `1` and `NeList`), and showing that the assumption that the result type is different from `NeList` together with the first coequation yield a contradiction. It turns out that in order to obtain a complete deduction calculus for coequations, this form of case analysis should be incorporated in the notion of coequation. This justifies the following definition.

Definition 4.1 Let Δ denote a destructor signature. A **Δ-coequation** is a tuple $((l,r),(t_1,\mathcal{C}'_1),\ldots,(t_n,\mathcal{C}'_n))$, also denoted $l = r$ if $(t_1,\mathcal{C}'_1),\ldots,(t_n,\mathcal{C}'_n)$, with $l,r \in T_\Delta[\mathcal{C}]_h$ and $t_i \in T_\Delta[\mathcal{C}_i]_h$, $\mathcal{C}'_i \subseteq \mathcal{C}_i$ for $i = 1,\ldots,n$. A Δ-coalgebra C **satisfies** a Δ-coequation e of the above form (written $C \models_\Delta e$) if and only if $l_C(c) = r_C(c)$ holds whenever $c \in C_h$ is such that $(t_i)_C(c) \in \iota_{Z_i}(C_{s_i})$ for some $Z_i \in (\mathcal{C}'_i)_{s_i}$, for $i = 1,\ldots,n$ (case in which c is said to **satisfy** the conditions $(t_1,\mathcal{C}'_1),\ldots,(t_n,\mathcal{C}'_n)$).

Notation 4.2 If $\mathcal{C}'_i = \{Z_i\}$, we sometimes write (t_i, Z_i) for $(t_i, \mathcal{C}'_i)$.

Example 4.3 Given the destructor signature in Example 2.4, the coequation:

```
[[F,[E]head]?]tail = [E]head if ([[F,N]?]tail,N)
```

constrains the interpretation of `NeList` to constant, finite or infinite lists. Similarly, the coequation:

```
[[F,[[F',N]?]tail]?]tail = N if ([[F,[[F',N]?]tail]?]tail,N)
```

constrains the interpretation of `NeList` to alternating lists.

Example 4.4 Connections consisting of a number of adjacent directed segments are specified using hidden sorts: `Point`, `Segment` and `Connection`, and operation symbols: `x, y : Point` → `Int`, `source, target : Segment` → `Point`, `first : Connection` → `Segment`, `rest : Connection` → `1 Connection`, further constrained by the following coequation:

```
[[P]target]first = [F,[[P]source]first]rest if ([F,C]rest,C)
```

capturing a sharing condition on the segments constituting a connection.

Satisfaction of coequations is both preserved and reflected by coalgebra homomorphisms.

Proposition 4.5 *Let C, D denote Δ-coalgebras, let $f : C \to D$ denote a Δ-homomorphism, and let e denote a Δ-coequation. Then:*

(i) *$C \models_\Delta e$ implies $\mathsf{Im}(f) \models_\Delta e$*

(ii) *$D \models_\Delta e$ implies $C \models_\Delta e$.*

Proof (Sketch) The fact that $t_{\mathsf{Im}(f)}(f_h(c)) = [\iota_1 \circ f_{s_1},\ldots,\iota_n \circ f_{s_n}](t_C(c))$ for each $h \in H$, $t \in T_\Delta[\{Z_1,\ldots,Z_n\}]_h$ with $Z_1 : s_1,\ldots,Z_n : s_n$ and $c \in C_h$ is used. (This is a consequence of the definition of a Δ-homomorphism.) □

Hence, the class of coalgebras satisfying a set of coequations is closed under sub-coalgebras, homomorphic images and finite coproducts, i.e. it is a *covariety* (see [8]).

Corollary 4.6 *The Δ-coalgebras satisfying a set of Δ-coequations form a covariety.*

Notation 4.7 Given $\delta \in \Delta_{h,s_1 \ldots s_n}$, $i \in \{1, \ldots, n\}$ and conditions C of form $(t_j, \mathcal{C}'_j)_{j=1,\ldots,m}$ for sort s_i, we write $[Z_1, \ldots, C, \ldots, Z_n]\delta$ as a shorthand for $([Z_1, \ldots, t_j, \ldots, Z_n]\delta, \mathcal{C}'_j \cup \{Z_1, \ldots, Z_{i-1}, Z_{i+1}, \ldots, Z_n\})_{j=1,\ldots,m}$, with $covar(t_j) \cap \{Z_1, \ldots, Z_{i-1}, Z_{i+1}, \ldots, Z_n\} = \emptyset$ for $j = 1, \ldots, m$. Similarly, given $t \in T_\Delta[\{Z_1, \ldots, Z_n\}]_h$ and i, C as before, we write $[Z_1/Z_1, \ldots, C/Z_i, \ldots, Z_n/Z_n]t$ for $([Z_1/Z_1, \ldots, t_j/Z_i, \ldots, Z_n/Z_n]t, \mathcal{C}'_j \cup \{Z_1, \ldots, Z_{i-1}, Z_{i+1}, \ldots, Z_n\})_{j=1,\ldots,m}$.

Definition 4.8 A **destructor specification** is a pair (Δ, E) with Δ a destructor signature and E a set of Δ-coequations. A Δ-coalgebra C **satisfies** a destructor specification (Δ, E) (written $C \models_\Delta E$) if and only if $C \models_\Delta e$ for each $e \in E$. A destructor specification (Δ, E) is said to be **inconsistent** w.r.t. a hidden sort $h \in H$ if and only if $C_h = \emptyset$ whenever $C \models_\Delta E$, for any Δ-coalgebra C. A set E of Δ-coequations **semantically entails** a Δ-coequation e (written $E \models_\Delta e$) if and only if $C \models_\Delta E$ implies $C \models_\Delta e$ for any Δ-coalgebra C.

Notation 4.9 Given a set E of Δ-coequations together with $h \in H$, we write E_h for the subset of E consisting of coequations for sort h.

Existence of final coalgebras generalises from destructor signatures to destructor specifications, with the final coalgebra of a destructor specification being a sub-coalgebra of the final coalgebra of the underlying signature.

Proposition 4.10 *Let (Δ, E) denote a destructor specification. There exists a final (Δ, E)-coalgebra.*

Proof. Let F denote a final Δ-coalgebra and let:

$$\begin{aligned}
F_{E,h} = \{\ \varphi \in F_h \mid\ & (l_F(t_F(\varphi)) = r_F(t_F(\varphi)) \text{ whenever} \\
& t \in T^1_\Delta[\{Z_1, \ldots, Z_n\}]_h, i \in \{1, \ldots, n\} \text{ and } (l = r \text{ if } C) \in E_{s_i} \\
& \text{are such that } t_F(\varphi) \in \iota_{Z_i}(F_{s_i}) \text{ and } C \text{ holds in } t_F(\varphi)\ \}, \quad h \in H \\
F_{E,v} = D_v, \quad & v \in V
\end{aligned}$$

Then $(F_{E,s})_{s \in S}$ defines a Δ-subcoalgebra of F. For, given $\varphi \in F_{E,h}$ and $\delta \in \Delta_{h,s_1 \ldots s_m}$, say $\delta_F(\varphi) \in \iota_{s_j}(F_{s_j})$ with $j \in \{1, \ldots, m\}$, $l_F(t'_F(\delta_F(\varphi))) = r_F(t'_F(\delta_F(\varphi)))$ holds for any $t' \in T^1_\Delta[\{Z_1, \ldots, Z_n\}]_{s_j}$ (with $Z_k : s'_k$ for $k = 1, \ldots, n$) and any coequation $(l = r \text{ if } C) \in E$ such that C holds in $t'_F(\delta_F(\varphi))$. This follows from $\varphi \in F_{E,h}$ by taking $t = [X_1, \ldots, X_{j-1}, t', X_{j+1}, \ldots, X_m]\delta$.

Also, given an arbitrary (Δ, E)-coalgebra C, the unique Δ-homomorphism $!_C$ from C to F factors through the inclusion of F_E into F. (Proposition 4.5 gives $Im(!_C) \models_\Delta E$, and F_E is, by definition, the greatest subcoalgebra of F which satisfies the coequations in E.) Hence, $!_C : C \to F$ defines a Δ-homomorphism from C to F_E. Uniqueness of such a homomorphism then follows from uniqueness of a Δ-homomorphism from C to F. □

The suitability of final coalgebras as denotations for destructor specifications is further justified by the following result.

Theorem 4.11 *Let (Δ, E) denote a destructor specification, let e denote a Δ-coequation and let F_E denote a final (Δ, E)-algebra. Then, $E \models_\Delta e$ if and only if $F_E \models_\Delta e$.*

Proof. The **if** direction follows from Proposition 4.5, while the **only if** direction follows from $F_E \models_\Delta E$. □

As opposed to algebra, where equations of form $X = X'$ are only satisfied by algebras whose corresponding carrier is a one-point set, in coalgebra, coequations of form $Z = Z'$ are only satisfied by coalgebras whose corresponding carrier is empty. More generally, coequations of form $l = r$ with $covar(l) \neq covar(r)$ constrain the result type of l and r to the type of a covariable appearing in both l and r. Among such coequations, of particular interest are those with l and r being the same up to a renaming of their covariables.

Definition 4.12 Let Δ denote a destructor signature, let $t \in T_\Delta[\mathcal{C}]_h$ for some set $\mathcal{C}$ of covariables and some $h \in H$, and let $\mathcal{C}' \subseteq \mathcal{C}$. The coequation:

$$\underline{t} = [y_1/X_1, \ldots , y_m/X_m]\underline{t}$$

where:

- $t = [Z_{i_1}/X_1, \ldots , Z_{i_m}/X_m]\underline{t}$ (see Notation 2.8)
- $y_j = \begin{cases} X_j & \text{if } Z_{i_j} \in \mathcal{C}' \\ Y_j \neq X_j & \text{if } Z_{i_j} \notin \mathcal{C}' \end{cases}$ for $j = 1, \ldots , m$

is called a **type constraint** for t and is denoted $c(t, \mathcal{C}')$.

A type constraint for t constrains the result type of t to the type of one of the covariables in $\mathcal{C}'$: given a Δ-coalgebra C, $c(t, \mathcal{C}')$ holds in the state $c \in C_h$ if and only if $t_C(c) \in \iota_Z(C_s)$ for some $Z \in \mathcal{C}'_s$.

Notation 4.13 We sometimes write $c(t, Z)$ for $c(t, \mathcal{C}')$ with $\mathcal{C}' = \{Z\}$.

Remark 4.14 If $t \in T^1_\Delta[\{Z_1, \ldots , Z_n\}]$ and $i \in \{1, \ldots , n\}$, then $c(t, Z_i)$ has the form $t = [Y_1/Z_1, \ldots , Z_i/Z_i, \ldots , Y_n/Z_n]t$.

Example 4.15 Given the destructor signature in Example 2.4, the type constraint $c($`[[F,N]?]tail`$,$ `N`$)$ has the form:

```
[[F,N]?]tail = [[F',N]?]tail
```

This coequation constrains the interpretation of `NeList` in coalgebras satisfying it to infinite lists (as it requires the interpretation of `tail` in any such coalgebra to always yield a non-empty list).

5 Coequational Deduction

We are now in the position to formulate a sound and complete deduction calculus for coequations. We consider the following deduction rules:

$$[\,\mathbf{base}\,]\quad \frac{}{E \vdash e} \quad e \in E$$

$$[\,\mathbf{cond\text{-}base}\,]\quad \frac{}{E \vdash c(t,\mathcal{C}) \text{ if } (t,\mathcal{C})}$$

$$[\,\mathbf{weakening}\,]\quad \frac{E \vdash t = t' \text{ if } C}{E \vdash t = t' \text{ if } C, C'}$$

$$[\,\mathbf{reflexivity}\,]\quad \frac{}{E \vdash t = t}$$

$$[\,\mathbf{symmetry}\,]\quad \frac{E \vdash t = t' \text{ if } C}{E \vdash t' = t \text{ if } C}$$

$$[\,\mathbf{transitivity}\,]\quad \frac{E \vdash t = t' \text{ if } C, \quad E \vdash t' = t'' \text{ if } C}{E \vdash t = t'' \text{ if } C}$$

$$[\,\mathbf{closure}\,]\quad \frac{E \vdash t_1 = t_1' \text{ if } C_1, \ldots, E \vdash t_n = t_n' \text{ if } C_n}{E \vdash [t_1, \ldots, t_n]\delta = [t_1', \ldots, t_n']\delta \text{ if } [C_1, \ldots, Z_n]\delta, \ldots, [Z_1, \ldots, C_n]\delta}$$

$$[\,\mathbf{substitution}\,]\quad \frac{E \vdash t = t' \text{ if } C}{E \vdash [t_1/Z_1, \ldots, t_n/Z_n]t = [t_1/Z_1, \ldots, t_n/Z_n]t' \text{ if } C}$$

$$[\,\mathbf{contradiction}\,]\quad \frac{E \vdash t = t' \text{ if } C}{E \vdash l = r \text{ if } C} \quad t, t' \in T_\Delta[\mathcal{C}]_h,\ l, r \in T_\Delta[\mathcal{C}']_h$$

$$\mathit{covar}(t) \cap \mathit{covar}(t') = \emptyset$$

$$[\,\mathbf{unity}\,]\quad \frac{}{E \vdash t = t' \text{ if } (t, Z), (t', Z)} \quad Z : v,\ |D_v| = 1$$

$$[\,\mathbf{case\text{-}analysis}\,]\quad \frac{E \vdash t = t' \text{ if } C, (t_0, \mathcal{C}_1)\ , \ldots ,\ E \vdash t = t' \text{ if } C, (t_0, \mathcal{C}_n)}{E \vdash t = t' \text{ if } C}$$

$$t, t' \in T_\Delta[\mathcal{C}']_h,\ t_0 \in T_\Delta[\mathcal{C}]_h,\ \mathcal{C} = \mathcal{C}_1 \cup \ldots \cup \mathcal{C}_n$$

Theorem 5.1 (Soundness) *Let (Δ, E) denote a destructor specification and let e denote a Δ-coequation. Then, $E \vdash e$ implies $E \models_\Delta e$.*

Proof. We use induction on the structure of the proof of $E \vdash e$ to show that $E \vdash e$ implies $E \models_\Delta e$.

If the last rule applied is **base**, then $E \models_\Delta e$ follows from the definition of $A \models_\Delta E$ for a Δ-coalgebra A. If the last rule applied is **weakening**, then $E \models_\Delta t = t'$ if C, C' follows from the fact that if C and C' hold in a state $a \in A_h$, then C holds in a. If the last rule applied is **cond-base** or **reflexivity**, then $E \models_\Delta e$ follows by any Δ-coalgebra (and hence any (Δ, E)-coalgebra) satisfying any coequation of form $c(t, \mathcal{C})$ if $(t, \mathcal{C})$, respectively $t = t$. If the last rule applied is one of **symmetry**, **transitivity** or **substitution**, then $E \models_\Delta e$ follows from the induction hypothesis by using properties of equality. If the last rule applied is **closure**, then for any (Δ, E)-coalgebra A and any $a \in A_h$ satisfying $[C_1, \ldots, Z_n]\delta, \ldots, [Z_1, \ldots, C_n]\delta$, say $\delta_A(a) \in \iota_{Z_i}(A_i)$ with $i \in \{1, \ldots, n\}$, the satisfaction of $[Z_1, \ldots, C_i, \ldots, Z_n]\delta$ by a implies the satisfaction of C_i by $\delta_A(a)$, which yields $(t_i)_A(\delta_A(a)) = (t'_i)_A(\delta_A(a))$ (by the induction hypothesis); that is, $([t_1, \ldots, t_n]\delta)_A(a) = ([t'_1, \ldots, t'_n]\delta)_A(a)$. If the last rule applied is **contradiction**, then $E \models_\Delta l = r$ if C follows from the fact that for a (Δ, E)-coalgebra A, there are no states $a \in A_h$ satisfying C (as they would then have to satisfy $t_A(a) = t'_A(a)$). If the last rule applied is **unity**, $t_A(a), t'_A(a) \in \iota_Z(D_v)$ together with $|D_v| = 1$ yield $t_A(a) = t'_A(a)$, for any Δ-coalgebra A and any state $a \in A_h$. Finally, if the last rule applied is **case-analysis**, $E \models_\Delta t = t'$ if C follows from one of $(t_0, \mathcal{C}_1), \ldots, (t_0, \mathcal{C}_n)$ holding in any state $a \in A_h$ satisfying C, for any (Δ, E)-coalgebra A. □

To prove *completeness* of the deduction calculus (namely that $E \models_\Delta e$ implies $E \vdash e$ for any E and e), we first need some preliminary results.

Lemma 5.2 *Let Δ denote a destructor signature and let E denote a set of Δ-coequations. If $E \vdash l = r$ if $C, (t, \mathcal{C})$ and $E \vdash c(t, \mathcal{C})$ if C, C', then $E \vdash l = r$ if C, C'.*

Proof (Sketch) The conclusion follows by **case-analysis** and **contradiction** from $E \vdash l = r$ if $C, C', (t, \mathcal{C})$ and $E \vdash l = r$ if $C, C', (t, \mathcal{C}')$, with $\mathcal{C}' = covar(t) \setminus \mathcal{C}$. □

Lemma 5.3 *Let (Δ, E) denote a destructor specification and let F_E denote a final (Δ, E)-coalgebra. Also, let $h \in H$ and let C denote some conditions for sort h. If $E \not\vdash l = r$ if C for any $l, r \in T_\Delta[\mathcal{C}]_h$ with $covar(l) \cap covar(r) = \emptyset$, then $F^C_{E,h} = \{\varphi \in F_{E,h} \mid C$ holds in $\varphi\} \neq \emptyset$.*

Proof. We define an ω^{op}-chain in Set with the following properties:

(a) $F^C_{E,h}$ projects to the limit object L of this ω^{op}-chain

(b) if $L = \emptyset$ then $E \vdash l = r$ if C with $l, r \in T_\Delta[\mathcal{C}]_h$, $covar(l) \cap covar(r) = \emptyset$.

Then, $F^C_{E,h} = \emptyset$ implies $L = \emptyset$ (since there exists a surjective mapping of $F^C_{E,h}$ into L), which, in turn, implies that $E \vdash l = r$ if C for some $l, r \in T_\Delta[\mathcal{C}]_h$ with $covar(l) \cap covar(r) = \emptyset$, yielding a contradiction.

We begin by noting that the set $T^1_{\Delta,h}$ is enumerable (since Δ is); say $T^1_{\Delta,h} = \{t_1, t_2, \ldots\}$. We consider the following ω^{op}-chain:

$$\begin{tikzcd}
C_1 & C_2 \arrow[l, "p_1"'] & C_3 \arrow[l, "p_2"'] & \cdots \arrow[l, "p_3"']
\end{tikzcd}$$

where:

$$\begin{aligned}
C_n = \{\ (Z_{t_1}, \ldots, Z_{t_n}) \mid\ & Z_{t_i} \in covar(t_i) \text{ for } i \in \{1, \ldots, n\} \\
& E \not\vdash l = r \text{ if } C, (t_1, Z_{t_1}), \ldots, (t_n, Z_{t_n}) \\
& \text{for any } l, r \in T_\Delta[\mathcal{C}]_h \text{ with } covar(l) \cap covar(r) = \emptyset\ \}
\end{aligned}$$

and:

$$p_n(Z_{t_1}, \dots, Z_{t_{n+1}}) = (Z_{t_1}, \dots, Z_{t_n})$$

for $n = 1, 2, \dots$. A limit object L for this ω^{op}-chain is then given by:

$$\begin{aligned} L = \{\ & (Z_{t_i})_{i \in \{1,2,\dots\}} \mid E \not\vdash l = r \text{ if } C, (t_1, Z_{t_1}), \dots, (t_n, Z_{t_n}) \\ & \qquad \text{for any } l, r \in T_\Delta[\mathcal{C}]_h \text{ with } \mathit{covar}(l) \cap \mathit{covar}(r) = \emptyset \text{ and any } n\ \} \end{aligned}$$

To show (a), we show that $\varphi = \langle Z_{t_i}, d_{t_i} \rangle_{i \in \{1,2,\dots\}} \in F^C_{E,h} \mapsto (Z_{t_i})_{i \in \{1,2,\dots\}} \in L$ defines a surjective mapping from $F^C_{E,h}$ to L.

For this mapping to be correctly defined, we must show that $(Z_{t_i})_{i \in \{1,2,\dots\}} \in L$ for each $\varphi \in F^C_{E,h}$. But $E \vdash l = r$ if $C, (t_1, Z_{t_1}), \dots, (t_n, Z_{t_n})$ for some $\varphi \in F^C_{E,h}$, some $l, r \in T_\Delta[\mathcal{C}]_h$ with $\mathit{covar}(l) \cap \mathit{covar}(r) = \emptyset$ and some $n \in \{1, 2, \dots\}$ together with soundness of deduction would yield a contradiction (as both C and each (t_i, Z_i), with $i = 1, \dots, n$, hold in $\varphi \in F^C_{E,h}$, while $l = r$ does not).

We now show that the mapping $\varphi \mapsto (Z_{t_i})_{i \in \{1,2,\dots\}}$ is surjective. For this, we first fix $d_s \in D_s$ for each $s \in S$. Then, given $(Z_{t_i})_{i \in \{1,2,\dots\}} \in L$, we construct $\varphi \in F^C_{E,h}$ by letting $\varphi = \langle Z_{t_i}, d_{t_i} \rangle_{i \in \{1,2,\dots\}}$, where $d_{t_i} = d_{s_i}$ if $Z_{t_i} : s_i$ with $s_i \in S$.

To show that $\varphi \in F_h$, let $t_i, t_j \in T^1_{\Delta,h}$ be such that $t_j = [t'_1/Z_1, \dots, t'_n/Z_n]t_i$. If $Z_{t_i} = Z_k$, we must show that $Z_{t_j} \in \mathit{covar}(t'_k)$ and moreover, if $t'_k = Z_{t_j}$ then $d_{t_i} = d_{t_j}$. Suppose $Z_{t_j} \notin \mathit{covar}(t'_k)$. Then:

$$E \vdash t_j = [t'_1/Z_1, \dots, t'_n/Z_n]t_i$$

(following by **reflexivity**) together with:

$$E \vdash t_i = [U_1/Z_1, \dots, Z_k/Z_k, \dots, U_n/Z_n]t_i \text{ if } C, (t_1, Z_{t_1}), \dots, (t_i, Z_{t_i})$$

and:

$$E \vdash t_j = [V_1/Z'_1, \dots, Z_{t_j}/Z_{t_j}, \dots, V_m/Z'_m]t_j \text{ if } C, (t_1, Z_{t_1}), \dots, (t_j, Z_{t_j})$$

(both following by **cond-base** and **weakening**) yield (by **substitution** followed by **weakening** and then by **transitivity**):

$$\begin{aligned} E \vdash\ & [V_1/Z'_1, \dots, Z_{t_j}/Z_{t_j}, \dots, V_m/Z'_m]t_j = [U_1/Z_1, \dots, t'_k/Z_k, \dots, U_n/Z_n]t_i \\ & \text{if } C, (t_1, Z_{t_1}), \dots, (t_N, Z_{t_N}) \end{aligned}$$

with $N = \mathit{max}(i, j)$. But the lhs and rhs of the last coequation have no covariable in common, thus contradicting the definition of L. Hence, $Z_{t_j} \in \mathit{covar}(t'_k)$. If, in addition, $t'_k = Z_{t_j}$, then also $s_i = s_j$, which gives $d_{t_i} = d_{t_j}$.

To show that $\varphi \in F_{E,h}$, let $t \in T^1_\Delta[\{Z_1, \dots, Z_n\}]_h$, $i \in \{1, \dots, n\}$ and $(t_i = t'_i \text{ if } C_i) \in E$ be such that $t_F(\varphi) \in \iota_{Z_i}(F_{s_i})$, $t_i, t'_i \in T_\Delta[\mathcal{C}_i]_{s_i}$ and C_i holds in $t_F(\varphi)$. We must show that $(t_i)_F(t_F(\varphi)) = (t'_i)_F(t_F(\varphi))$. According to Corollary 3.8, it suffices to show that for any coterms $u_1, \dots, u_q$ of suitable sort, either $l_F(\varphi), r_F(\varphi) \in \iota_Z(F_h)$ for some $Z : h$, or $l_F(\varphi), r_F(\varphi) \in \iota_Z(D_v)$ for some $Z : v$, and in the last case $l_F(\varphi) = r_F(\varphi)$, where:

$$\begin{aligned} l &= [u_1/U_1, \ldots, u_q/U_q][Z_1/Z_1, \ldots, t_i/Z_i, \ldots, Z_n/Z_n]t \\ r &= [u_1/U_1, \ldots, u_q/U_q][Z_1/Z_1, \ldots, t'_i/Z_i, \ldots, Z_n/Z_n]t \end{aligned}$$

with $\{U_1, \ldots, U_q\} = \mathcal{C}_i \cup \{Z_1, \ldots, Z_{i-1}, Z_{i+1}, \ldots, Z_n\}$.

We let $l = [V_{i_1}/X_1, \ldots, V_{i_m}/X_m]\underline{l}$ with $\underline{l} \in T^1_\Delta[\{X_1, \ldots, X_m\}]_h$, and $r = [V_{j_1}/Y_1, \ldots, V_{j_p}/Y_p]\underline{r}$ with $\underline{r} \in T^1_\Delta[\{Y_1, \ldots, Y_p\}]_h$ (see Notation 2.8). From $E \vdash t_i = t'_i$ if C_i we can infer, by successive applications of the **closure** rule followed by **substitution**:

$$E \vdash l = r \text{ if } [Z_1/Z_1, \ldots, C_i/Z_i, \ldots, Z_n/Z_n]t$$

We claim that if $Z_{\underline{l}} = X_k$ and $Z_{\underline{r}} = Y_l$, then $V_{i_k} = V_{j_l}$. For, if this was not the case, **cond-base** together with **substitution** would yield:

$$\begin{aligned} E \vdash {} & [S_1/V_1, \ldots, V_{i_k}/V_{i_k}, \ldots, S_o/V_o]l = [T_1/V_1, \ldots, V_{j_l}/V_{j_l}, \ldots, T_o/V_o]r \\ & \text{if } [Z_1/Z_1, \ldots, C_i/Z_i, \ldots, Z_n/Z_n]t, (\underline{l}, X_k), (\underline{r}, Y_l) \end{aligned}$$

with $V_{i_k} \neq V_{j_l}$, which would then yield:

$$\begin{aligned} E \vdash {} & [S_1/V_1, \ldots, V_{i_k}/V_{i_k}, \ldots, S_o/V_o]l = [T_1/V_1, \ldots, V_{j_l}/V_{j_l}, \ldots, T_o/V_o]r \\ & \text{if } (t_1, Z_{t_1}), \ldots, (t_N, Z_{t_N}) \end{aligned}$$

for N sufficiently large (the fact that $[Z_1/Z_1, \ldots, C_i/Z_i, \ldots, Z_n/Z_n]t$ holds in φ together with Lemma 5.2 are used here). But this would contradict the definition of L. Hence, $V_{i_k} = V_{j_l} = Z : s$ and $l_F(\varphi), r_F(\varphi) \in \iota_Z(F_s)$. Furthermore, if $s = v$ for some $v \in V$, $l_F(\varphi) = r_F(\varphi)$ follows from both $l_F(\varphi)$ and $r_F(\varphi)$ being equal to d_v. Hence, $\varphi \in F_{E,h}$.

In addition, we have $\varphi \in F^C_{E,h}$. For, if this was not the case, the conditions in C would contradict $(t_1, Z_{t_1}), \ldots, (t_N, Z_{t_N})$ for N sufficiently large (by Lemma 5.2), and hence we could infer $E \vdash l = r$ if $C, (t_1, Z_{t_1}), \ldots, (t_N, Z_{t_N})$ with $covar(l) \cap covar(r) = \emptyset$. This concludes the proof of the fact that $F^C_{E,h}$ has a surjective mapping into L.

To show (b), assume $L = \emptyset$. Then, for any $Z \in C_1$, there exists $n_Z \in \{2, \ldots\}$ such that $Z \notin Im(p_1 \circ \ldots \circ p_{n_Z})$. For, if $Z \in C_1$ was such that $Z \in Im(p_1 \circ \ldots \circ p_n)$ for any $n \in \{2, \ldots\}$, then also $Z \in Im(l_1)$ (with $l_1 : L \to C_1$ denoting the corresponding arrow of the limiting cone), which would contradict the assumption that $L = \emptyset$. We now let $n' = max\{n_Z \mid Z \in C_1\}$. It follows by **weakening** and **contradiction** that $E \vdash l = r$ if $C, (t_1, Z_1), \ldots, c(t_{n'}, Z_{n'})$ for any choice of $Z_1 \in covar(t_1), \ldots, Z_{n'} \in covar(t_{n'})$, with $l, r \in T_\Delta[\mathcal{C}]_h$ being such that $covar(l) \cap covar(r) = \emptyset$. Then, successive applications of the **case-analysis** rule yield $E \vdash l = r$ if C, which contradicts the hypothesis. Hence, $L \neq \emptyset$. This concludes the proof. □

Lemma 5.4 *Let (Δ, E) denote a destructor specification, let $l, r \in T_\Delta[\mathcal{C}]_h$ for some set $\mathcal{C}$ of covariables and some $h \in H$, and let $Z \in \mathcal{C}_v$ for some $v \in V$. Also, let F_E denote a final (Δ, E)-coalgebra. If $E \not\vdash l = r$ if $C, (l, Z), (r, Z)$, then there exists $\varphi' \in F_{E,h}^{C,(l,Z),(r,Z)}$ such that $l_{F_E}(\varphi') \neq r_{F_E}(\varphi')$.*

Proof. One can immediately infer that $|D_v| > 1$ (otherwise **unity** together with **weakening** would yield $E \vdash l = r$ if $C, (l, Z), (r, Z)$). We let $d_v, d'_v \in D_v$ be such that $d'_v \neq d_v$.

The proof is similar to the proof of Lemma 5.3. We construct an ω^{op}-chain in Set whose limit object is the empty set only if $E \vdash l = r$ if $C, (l, Z), (r, Z)$, and use an element of the limit object to construct $\varphi' \in F_{E,h}^{C,(l,Z),(r,Z)}$ with $l_{F_E}(\varphi') \neq r_{F_E}(\varphi')$.

We consider the following ω^{op}-chain:

$$\begin{tikzcd} S_1 & S_2 \arrow[l, "p_1"'] & S_3 \arrow[l, "p_2"'] & \cdots \arrow[l, "p_3"'] \end{tikzcd}$$

where:

$$\begin{aligned} S_n = \{\ & (Z_{t_1}, \ldots, Z_{t_n}) \mid Z_{t_i} \in covar(t_i) \text{ for } i \in \{1, \ldots, n\}, \\ & E \not\vdash l = r \text{ if } C, (l, Z), (r, Z), (t_1, Z_{t_1}), \ldots, (t_n, Z_{t_n})\ \} \end{aligned}$$

and:

$$p_n(Z_{t_1}, \ldots, Z_{t_{n+1}}) = (Z_{t_1}, \ldots, Z_{t_n})$$

for $n = 1, 2, \ldots$. A limit object L for this ω^{op}-chain is then given by:

$$\begin{aligned} L = \{\ & (Z_{t_i})_{i \in \{1,2,\ldots\}} \mid Z_{t_i} \in covar(t_i) \text{ for } i \in \{1, 2, \ldots\}, \\ & E \not\vdash l = r \text{ if } C, (l, Z), (r, Z), (t_1, Z_{t_1}), \ldots, (t_n, Z_{t_n}) \text{ for any } n\ \} \end{aligned}$$

We claim that:

(a) $S_n \neq \emptyset$ for any $n \in \{1, 2, \ldots\}$

(b) $L \neq \emptyset$

To show (a), we assume $S_n = \emptyset$ for some $n \in \{1, 2, \ldots\}$. Hence, for any $Z_{t_i} \in covar(t_i)$ with $i = 1, \ldots, n$, $E \vdash l = r$ if $C, (l, Z), (r, Z), (t_1, Z_{t_1}), \ldots, (t_n, Z_{t_n})$. But then **case-analysis** yields $E \vdash l = r$ if $C, (l, Z), (r, Z)$, which contradicts the hypothesis. Therefore, $S_n \neq \emptyset$ for any $n \in \{1, 2, \ldots\}$.

To show (b), assume $L = \emptyset$. Hence, for any $Z \in S_1$, there exists $n_Z \in \{2, \ldots\}$ such that $Z \notin Im(p_1 \circ \ldots \circ p_n)$. If $n' = max\{n_Z \mid Z \in S_1\}$, it follows by **weakening** that $E \vdash l = r$ if $C, (l, Z), (r, Z), (t_1, Z_{t_1}), \ldots, (t_{n'}, Z_{t_{n'}})$ for any choice of $Z_{t_1} \in covar(t_1), \ldots, Z_{t_{n'}} \in covar(t_{n'})$. Then, **case-analysis** yields $E \vdash l = r$ if $C, (l, Z), (r, Z)$, which contradicts the hypothesis. Hence, $L \neq \emptyset$.

We now fix $(Z_{t_i})_{i=1,2,\ldots} \in L$ and use it to define $\varphi' \in F_{E,h}^{C,(l,Z),(r,Z)}$ such that $l_{F_E}(\varphi') \neq r_{F_E}(\varphi')$. We let $\mathcal{C} = \{Z_1, \ldots, Z_n\}$, $l = [Z_{i_1}/X_1, \ldots, Z_{i_m}/X_m]\underline{l}$,

$r = [Z_{j_1}/Y_1, \ldots, Z_{j_p}/Y_p]\underline{r}$ with $\underline{l} \in T^1_\Delta[\{X_1, \ldots, X_m\}]_h$, $\underline{r} \in T^1_\Delta[\{Y_1, \ldots, Y_p\}]_h$ (see Notation 2.8). We also let $k \in \{1, \ldots, m\}$ be such that $X_k = Z_{\underline{l}}$, and $l \in \{1, \ldots, p\}$ be such that $Y_l = Z_{\underline{r}}$. It follows immediately that $Z_{i_k} = Z$ and $Z_{j_l} = Z$; for if, say $Z_{i_k} \neq Z$, then the conditions (l, Z) and $(\underline{l}, X_k)$ would contradict each other, and we would be able to infer:

$$E \vdash l' = r' \text{ if } C, (l, Z), (r, Z), (t_1, Z_{t_1}), \ldots, (t_N, Z_{t_N})$$

with $covar(l') \cap covar(r') = \emptyset$ for N sufficiently large. Finally, for $n \in \{1, 2 \ldots\}$, we let C_n stand for $(t_1, Z_{t_1}), \ldots, (t_n, Z_{t_n})$.

We now define:

$$\begin{aligned} T = \{\ t \in T^1_{\Delta,h} \mid\ & \text{there exists } n \in \{1, 2, \ldots\} \text{ such that} \\ & E \vdash t = [Y_1/Y_1, \ldots, Z_t/Y_l, \ldots, Y_p/Y_p]\underline{r} \text{ if } C, (l, Z), (r, Z), C_n\ \} \end{aligned}$$

(with $\{Y_1, \ldots, Y_p\} \cap covar(t) = \emptyset$ for any $t \in T^1_{\Delta,h}$). That is, T consists of coterms whose interpretation must agree with that of $\underline{r}$ on any state $\varphi \in F^{C,(l,Z),(r,Z)}_{E,h}$ which, in addition, satisfies each of (t_n, Z_n) with $n = 1, 2, \ldots$. We then let $\varphi' = \langle Z_t, d_t \rangle_{t \in T^1_{\Delta,h}}$, where:

$$d_t = \begin{cases} d_s & \text{if } t \notin T,\ Z_t : s \\ d'_v \neq d_t & \text{if } t \in T,\ Z_t : v \end{cases}$$

We note that $t \in T$ implies $Z_t : v$, since $Z_{\underline{r}} = Y_l : v$. Also, we recall that $d_v, d'_v \in D_v$ were chosen so that $d_v \neq d'_v$. We now claim that:

(c) $\varphi' \in F^{C,(l,Z),(r,Z)}_{E,h}$

(d) $r_{F_E}(\varphi') \neq l_{F_E}(\varphi')$

Proving (c) reduces to proving that $\varphi' \in F_{E,h}$ and that each of C, (l, Z), (r, Z) hold in φ'.

The proof of $\varphi' \in F_h$ is similar to the proof of $\varphi \in F_h$ in Lemma 5.3. In addition, here we must show that if $t_i, t_j \in T^1_{\Delta,h}$ are such that $t_j = [t'_1/Z'_1, \ldots, t'_n/Z'_n]t_i$, $Z_{t_i} = Z'_k$ and $t'_k = Z_{t_j}$, then either t_i and t_j are both in T, or none of them is in T. We distinguish two cases:

(i) $t_i \in T$. This implies:

$$E \vdash t_i = [Y_1/Y_1, \ldots, Z'_k/Y_l, \ldots, Y_m/Y_m]\underline{r} \text{ if } C, (l, Z), (r, Z), C_{n_0}$$

for some $n_0 \in \{1, 2, \ldots\}$. Then, **substitution** yields:

$$\begin{aligned} E \vdash t_j = {} & [t'_1/Z'_1, \ldots, t'_n/Z'_n][Y_1/Y_1, \ldots, Z'_k/Y_l, \ldots, Y_m/Y_m]\underline{r} \\ & \text{if } C, (l, Z), (r, Z), C_{n_0} \end{aligned}$$

Hence, as $\{Z'_1, \ldots, Z'_{k-1}, Z'_{k+1}, \ldots, Z'_n\} \cap \{Y_1, \ldots, Y_{l-1}, Y_{l+1}, \ldots, Y_m\} = \emptyset$

and $t'_k = Z_{t_j}$, we obtain:

$$E \vdash t_j = [Y_1/Y_1, \ldots, Z_{t_j}/Y_l, \ldots, Y_m/Y_m]\underline{r} \text{ if } C, (l, Z), (r, Z), C_{n_0}$$

That is, $t_j \in T$.

(ii) $t_j \in T$. This implies:

$$E \vdash t_j = [Y_1/Y_1, \ldots, Z_{t_j}/Y_l, \ldots, Y_m/Y_m]\underline{r} \text{ if } C, (l, Z), (r, Z), C_{n_0}$$

for some $n_0 \in \{1, 2, \ldots\}$. Then, $t_j = [t'_1/Z'_1, \ldots, t'_n/Z'_n]t_i$ gives:

$$E \vdash [t'_1/Z'_1, \ldots, t'_n/Z'_n]t_i = [Y_1/Y_1, \ldots, Z_{t_j}/Y_l, \ldots, Y_m/Y_m]\underline{r}$$
$$\text{if } C, (l, Z), (r, Z), C_{n_0}$$

But $Z_{t_i} = Z'_k$ together with **substitution** and **cond-base** yield:

$$E \vdash [t'_1/Z'_1, \ldots, t'_n/Z'_n]t_i = [Z'_1/Z'_1, \ldots, t'_k/Z'_k, \ldots, Z'_n/Z'_n]t_i$$
$$\text{if } C, (l, Z), (r, Z), C_N$$

for N sufficiently large. Also, $t'_k = Z_{t_j}$. Hence, by **transitivity**:

$$E \vdash [Z'_1/Z'_1, \ldots, Z_{t_j}/Z'_k, \ldots, Z'_n/Z'_n]t_i = [Y_1/Y_1, \ldots, Z_{t_j}/Y_l, \ldots, Y_m/Y_m]\underline{r}$$
$$\text{if } C, (l, Z), (r, Z), C_N$$

Finally, substituting Z_{t_i} for Z_{t_j} yields:

$$E \vdash t_i = [Y_1/Y_1, \ldots, Z_{t_i}/Y_l, \ldots, Y_m/Y_m]\underline{r} \text{ if } C, (l, Z), (r, Z), C_N$$

That is, $t_i \in T$.

Hence, either both t_i and t_j belong to T, or neither of them does. This concludes the proof of $\varphi' \in F_h$.

The proof of $\varphi' \in F_{E,h}$ is, again, similar to the proof of $\varphi \in F_{E,h}$ in Lemma 5.3. In addition, here we must prove that given $t \in T^1_\Delta[\{Z'_1, \ldots, Z'_n\}]_h$, $i \in \{1, \ldots, n\}$ and $(t_i = t'_i \text{ if } C_i) \in E$ such that C_i holds in $t_F(\varphi)$, then either both $\underline{l'}$ and $\underline{r'}$ are in T, or none of them are (where l' and r' are defined similarly to l and r from Lemma 5.3).

Suppose $\underline{l'} \in T$. On the one hand,

$$E \vdash l' = r' \text{ if } [Z'_1/Z'_1, \ldots, C_i/Z'_i, \ldots, Z'_n/Z'_n]t$$

(following by successive applications of the **closure** rule) together with the fact that $[Z'_1/Z'_1, \ldots, C_i/Z'_i, \ldots, Z'_n/Z'_n]t$ holds in φ yield:

$$E \vdash l' = r' \text{ if } C_N$$

for N sufficiently large (Lemma 5.2 is used here). That is:

$$E \vdash [W_{i_1}/U_1, \ldots, W_{i_q}/U_q]\underline{l'} = [W_{j_1}/V_1, \ldots, W_{j_r}/V_r]\underline{r'} \text{ if } C_N$$

We immediately infer that if $q_0 \in \{1, \dots, q\}$ and $r_0 \in \{1, \dots, r\}$ are defined by $Z_{\underline{l'}} = U_{q_0}$ and respectively $Z_{\underline{r'}} = V_{r_0}$, then $W_{i_{q_0}} = W_{j_{r_0}}$.

On the other hand, $\underline{l'} \in T$ gives:

$$E \vdash \underline{l'} = [Y_1/Y_1, \dots, Z_{l'}/Y_l, \dots, Y_m/Y_m]\underline{r} \text{ if } C, (l, Z), (r, Z), C_{n_0}$$

for some $n_0 \in \{1, 2, \dots\}$.

The last two statements, together with:

$$E \vdash \underline{l'} = [W_{i_1}/U_1, \dots, U_{q_0}/U_{q_0}, \dots, W_{i_q}/U_q]\underline{l'} \text{ if } C_N$$

and:

$$E \vdash \underline{r'} = [W_{j_1}/V_1, \dots, V_{r_0}/V_{r_0}, \dots, W_{j_r}/V_r]\underline{r'} \text{ if } C_N$$

(following by **cond-base** for N sufficiently large) can then be used to infer:

$$E \vdash \underline{r'} = [Y_1/Y_1, \dots, Z_{r'}/Y_l, \dots, Y_m/Y_m]\underline{r} \text{ if } C, (l, Z), (r, Z), C_N$$

That is, $\underline{r'} \in T$. This concludes the proof of $\varphi \in F_{E,h}$.

It remains to prove that each of $C, (l, Z), (r, Z)$ holds in φ. If this was not the case, the condition $C, (l, Z), (r, Z), C_N$ would be contradictory for N sufficiently large, yielding $E \vdash l = r$ if $C, (l, Z), (r, Z), C_N$. But this would then contradict the definition of L. This concludes the proof of (c).

To prove (d), it suffices to show that $\underline{l} \notin T$. Then, since $\underline{r} \in T$, the claim follows from $d_v \neq d'_v$. We therefore assume that $\underline{l} \in T$ and show that this yields a contradiction. If $\underline{l} \in T$, then:

$$E \vdash \underline{l} = [Y_1/Y_1, \dots, X_k/Y_l, \dots, Y_m/Y_m]\underline{r} \text{ if } C, (l, Z), (r, Z), C_{n_0}$$

for some $n_0 \in \{1, 2, \dots\}$. This, together with:

$$E \vdash l = [X_1/X_1, \dots, Z/X_k, \dots, X_m/X_m]\underline{l} \text{ if } C_N$$

and:

$$E \vdash r = [Y_1/Y_1, \dots, Z/Y_l, \dots, Y_p/Y_p]\underline{r} \text{ if } C_N$$

for N sufficiently large (both following by **transitivity** and **cond-base**) can then be used to infer:

$$E \vdash l = r \text{ if } C, (l, Z), (r, Z), C_N$$

for N sufficiently large. But this contradicts the fact that $(Z_{t_i})_{i=1,2,\dots} \in L$. Hence, $\underline{l} \notin T$.

We have therefore constructed $\varphi' \in F_{E,h}$ such that each of C, (l, Z), (r, Z) hold in φ', but $r_{F_E}(\varphi') \neq l_{F_E}(\varphi')$. This concludes the proof. □

Theorem 5.5 (Completeness) *Let (Δ, E) denote a destructor specification and let e denote a Δ-coequation. Then, $E \models_\Delta e$ implies $E \vdash e$.*

Proof. Let e be of form $l = r$ if C, with $l, r \in T_\Delta[\mathcal{C}]_h$.

We first consider the case when all the covariables in $\mathcal{C}$ are visible-sorted. We let F_E denote a final (Δ, E)-coalgebra and distinguish the following cases:

(i) $F^C_{E,h} = \emptyset$.

In this case, $E \vdash e$ follows by Lemma 5.3.

(ii) $F^C_{E,h} \neq \emptyset$.

We assume that $E \nvdash e$ and show that this yields a contradiction. From $E \nvdash e$ it follows immediately that there exist $Z \in \mathcal{C}_v$ and $Z' \in \mathcal{C}_{v'}$ with $v, v' \in V$ such that $E \nvdash l = r$ if $C, (l, Z), (r, Z')$ (otherwise $E \vdash l = r$ if C would follow by **case-analysis**). We distinguish two sub-cases:

(a) $Z \neq Z'$.

$E \nvdash l = r$ if $C, (l, Z), (r, Z')$ gives $E \nvdash l' = r'$ if $C, (l, Z), (r, Z')$ for any $l', r' \in T_\Delta[\mathcal{C}']_h$ with $covar(l') \cap covar(r') = \emptyset$ (otherwise **contradiction** could be applied). Then, Lemma 5.3 gives $\varphi \in F_{E,h}$ such that C, (l, Z), (r, Z') hold in φ. That is, $\varphi \in F_{E,h}$ satisfies the conditions C, but $l_{F_E}(\varphi) \neq r_{F_E}(\varphi)$ (since $l_{F_E}(\varphi) \in \iota_Z(F_v)$ and $r_{F_E}(\varphi) \in \iota_{Z'}(F_{v'})$, with $Z \neq Z'$).

(b) $Z = Z'$.

Since $E \nvdash l = r$ if $C, (l, Z), (r, Z)$, it follows by Lemma 5.4 that there exists $\varphi \in F_{E,h}$ such that C holds in φ but $l_{F_E}(\varphi) \neq r_{F_E}(\varphi)$.

In both of the above sub-cases we can infer that $F_E \not\models_\Delta l = r$ if C, which contradicts the hypothesis (as $E \models_\Delta e$ implies $F_E \models_\Delta e$). Hence, $E \vdash e$.

This concludes the proof for the case when all the covariables in $\mathcal{C}$ are visible-sorted. The proof for the case when $\mathcal{C}$ also contains hidden-sorted covariables is similar, except that one considers, instead of a final Δ-coalgebra, a cofree Δ-coalgebra over the S-sorted set C given by: $C_h = \{0_h, 1_h\}$ for $h \in H$, and $C_v = D_v$ for $v \in V$. □

The crucial results in the above proof were Lemma 5.3 and Lemma 5.4. Lemma 5.3 states that whenever a set of coequations is inconsistent w.r.t a given sort and a set of conditions for this sort, a contradiction for the given conditions can be syntactically derived from the coequations. Lemma 5.4 states that if two coterms constrained to the same visible-sorted covariable can not be proved equal under certain conditions, then the final coalgebra of the specification contains a state which satisfies all the conditions, but distinguishes the two coterms.

Remark 5.6 In theory, the number of applications of the **case-analysis** rule needed to infer a given coequation may be arbitrarily large. However, in practice, case analysis on the result type of coterms *matching* the lhs/rhs of coequations in the specification (i.e. yielding the lhs/rhs of the coequation when a substitution is applied to them) proves sufficient in most cases.

6 Related Work

The approach presented here is not entirely a generalisation of the one in [1], the reason being the absence of any algebraic features, as a result of using a coalgebraic syntax. Most importantly, coequations can not be used to constrain state observations to particular data values. However, in our opinion, such constraints should only be imposed to observations of particular states (such as the ones yielded by certain constructors), and therefore should not be considered when coalgebraically specifying the state space.

In addition to destructors of form $\sigma : h \to h$ and $\alpha : h \to v$, parameterised destructors of form $\sigma : h\ v \to h$ and $\alpha : h\ v \to v'$ were also considered in [1]. Our approach can be easily extended to accommodate such operations, as well as more general operations of form $\delta : h\ v \to s_1 \ldots s_n$. In this generalised setting, coterms are defined inductively by:

- $Z \in T_\Delta[\mathcal{C}]_s$ for $Z \in \mathcal{C}_s$ and $s \in S$
- $[t_1, \ldots, t_n]\delta(d) \in T_\Delta[\mathcal{C}]_h$ for $\delta \in \Delta_{hv,s_1 \ldots s_n}$, $d \in D_v$ and $t_i \in T_\Delta[\mathcal{C}]_{s_i}$, $i = 1, \ldots, n$

The completeness result also generalises, the only additional requirement being that the sets D_v, with $v \in V$, are enumerable. However, we believe that operations such as the ones above should not be regarded as destructors, especially if their visible arguments may take infinitely many values. In particular, operations denoting a *change* of state (rather than a *property* of states) should not be considered at this stage. (In [1], viewing *parameterised methods* as destructors resulted in complications when attempting to define such operations by equations, as arbitrary algebraic terms were not allowed in equations.)

Finally, we briefly comment on the differences between our equational approach and the modal logic approach of [6]. A first difference stands in the way of capturing state observations – in [6], the result of state observations may be undefined, whereas here any observation yields a well-defined result (at the expense of additional information in coterms). Another significant difference stands in the fact that coequations *relate* different state observations, while the formulae of modal logic *define* the values of single state observations. Thus, our approach provides a framework for specifying constraints involving the *structure* of state-based systems (including the absence of certain components, or the sharing of subcomponents by certain system components), while modal logic provides a framework for constraining the values yielded by observations of particular states.

7 Conclusions and Future Work

Destructor signatures induced by polynomial endofunctors in the form of products of finite coproducts have been used to specify ways of observing system states, and a sound and complete equational calculus for reasoning about observational properties of states has been developed. The duality between the endofunctors considered here and the ones inducing (many-sorted) algebraic signatures has provided useful insights, yielding notions of coterm, covariable and coequation, dual to the algebraic notions of term, variable and equation.

Coequations are sufficiently expressive to capture constraints regarding the structure of system states. However, the specification of state-based, dynamical systems also involves constraints regarding the relationship between *constructing* system states and *observing* such states. For instance, one usually specifies object-oriented systems by defining the state constructors in terms of their effect on the particular observations that can be made about the result they yield. An approach that integrates algebraic and coalgebraic techniques in order to allow the specification of this relationship is therefore needed to fully specify state-based, dynamical systems. Such an approach should clearly distinguish between (algebraic) operations used to construct new states, and (coalgebraic) operations used to observe properties of existing states.

References

[1] A. Corradini. A complete calculus for equational deduction in coalgebraic specification. In F. Parisi-Presicce, editor, *Recent Trends in Algebraic Development Techniques*, volume 1376 of *LNCS*. Springer, 1998.

[2] B. Jacobs. Inheritance and cofree constructions. In P. Cointe, editor, *European Conference on Object-Oriented Programming*, volume 1098 of *LNCS*. Springer, 1996.

[3] B. Jacobs. Objects and classes, coalgebraically. In B. Freitag, C.B. Jones, C. Lengauer, and H.-J. Schek, editors, *Object Orientation with Parallelism and Persistence*. Kluwer Academic Publishers, 1996.

[4] L.S. Moss. Coalgebraic logic. To appear in the Annals of Pure and Applied Logic.

[5] H. Reichel. An approach to object semantics based on terminal coalgebras. *Mathematical Structures in Computer Science*, 5, 1995.

[6] M. Rößiger. From modal logic to terminal coalgebras. To appear in Theoretical Computer Science.

[7] J. Rutten. A calculus of transition systems (towards universal coalgebra). Technical Report CS-R9503, CWI, 1995.

[8] J. Rutten. Universal coalgebra: a theory of systems. Technical Report CS-R9652, CWI, 1996.