Translating Logics for Coalgebras

Dirk Pattinson

Institut für Informatik, LMU München

Abstract. This paper shows, that three different types of logics for coalgebras are institutions. The logics differ regarding the presentation of their syntax. In the first framework, abstract behavioural logic, one has a syntax-free representation of behavioural properties. We then turn to coalgebraic logic, the syntax of which is given as an initial algebra. The last framework, which we consider, is coalgebraic modal logic, the syntax of which is concretely given.

1 Introduction

This paper tries to contribute to the question, whether different types of logics, interpreted over coalgebras, carry the structure of an institution. Institutions, originally introduced by Goguen and Burstall, capture interplay between the transformation of systems and corresponding translations of logics. An institution therefore consists of two parts: A class of systems, and a class of logics, which can be used to describe properties of the systems under considerations. Both are linked by (semantical) transformation of systems and corresponding (syntactical) translation of the logics. If the systems, together with their logics, form an institution, we have the possibility to derive properties of a transformed system from properties of the original system, which makes the concept of institutions valuable in the stepwise process of building systems.

The class of systems we are dealing with in this paper, are coalgebras for an endofunctor on the category of sets. Coalgebras provide a uniform view on a large class of state-based systems, (see [20] for examples). In order to reason about coalgebraically modelled systems, modal logic has proven an appropriate tool ([10,15,8,19,7]).

Both the class of systems (coalgebras) and the corresponding class of (modal) logics are well understood – as long as we do not migrate between different types of systems (that is, between coalgebras for different functors) and leave the logics fixed. It is the purpose of this paper to add transformations between models and translation between the logics to the picture.

After recalling some basic terminology, we first address the question, whether *co*institutions are the appropriate framework in which one should consider logics for *co*algebras and their translations. We can rightfully say, that this is just a matter of taste: Every institution over a category Sig of signatures corresponds to a coinstitution over $\mathrm{Sig}^{\mathrm{op}}$, and vice versa (Proposition 1). In this light, we choose to work with coinstitution, which we feel are easier to work with in the context of coalgebras, mainly because we do not need to work with the dual category of signature morphisms.

Before dealing with translations on the logical side, we first study the transformation of models on semantical side. We argue that – working with coalgebras for endofunctors – natural transformations between functors provide us with a natural notion of signature morphism. This notion of signature morphism is then used to treat three different types of logics for coalgebras: abstract behavioural logic (the presentation of which is syntax free), coalgebraic logic (the syntax of which is abstract), and coalgebraic modal logic, where the syntax is concretely given.

We show, how to define translations between the logics for each of the three different types. It turns out that the institution property of abstract behavioural logic and coalgebraic modal logic is relatively easy to establish; in the case of coalgebraic logic, one needs a small extension of the syntax.

2 Preliminaries and Notation

In the whole paper, T denotes an endofunctor on the category Set of sets and functions.

2.1 Coalgebras

The definition of coalgebras (and their morphisms) dualises that of algebras for endofunctors:

Definition 1 (Coalgebras, Morphisms). *A* T-coalgebra *is a pair* (C, γ) *where* C *is a set and* $\gamma : C \to TC$ *is a function. A* morphism *between two* T*-coalgebras* (C, γ) *and* (D, δ) *is a function* $f : C \to D$*, which satisfies* $Tf \circ \gamma = \delta \circ f$*.*

Coalgebras, together with their morphisms, form a category, which we denote by $\mathsf{CoAlg}(T)$*.*

We think of coalgebras for an endofunctor as a general framework for state based systems, and we think of T as a *signature* for the T-coalgebras. Instantiating the framework with specific endofunctors (different signatures), we obtain different types of systems:

Example 1. (i) Suppose $T_1X = L \times X$ for a set L of labels. Then every state $c \in C$ of a T-coalgebra (C, γ) can be seen as producing an infinite trace of labels $l \in L$:

$$c = c_0 \xrightarrow{l_1} c_1 \xrightarrow{l_2} c_2 \longrightarrow \cdots$$

where $(l_k, c_k) = \gamma(c_{k-1})$ for $k > 0$.

(ii) For $T_2X = (O \times X)^I$, the T_2-coalgebras are Mealy Automata: Given $(C, \gamma) \in \mathsf{CoAlg}(T_2)$, a state $c \in C$ and an input $i \in I$, the transition function γ provides us with a new state $\pi_2 \circ \gamma(c)(i)$ and an output $o = \pi_1 \circ \gamma(c)(i) \in O$.

(iii) Suppose $TX = \mathcal{P}(X)^L$, where $\mathcal{P}$ is the covariant powerset functor. Then T-coalgebras are in 1-1 correspondence with labelled transition systems: Given $(C, \gamma) \in \mathsf{CoAlg}(T)$, put $c \xrightarrow{l} c'$ iff $c' \in \gamma(c)(l)$.

One of the appealing features of the general theory of coalgebras is, that T-coalgebras come with a meaningful built-in notion of behavioural equivalence:

Definition 2. *Suppose* (C, γ) *and* $(D, \delta) \in \mathsf{CoAlg}(T)$*. Then a pair of states* $(c, d) \in C \times D$ *is* behaviourally equivalent*, if there is* $(E, \epsilon) \in \mathsf{CoAlg}(T)$ *and a pair of morphisms* $f : (C, \gamma) \to (E, \epsilon)$ *and* $g : (D, \delta) \to (E, \epsilon)$ *such that* $f(c) = g(d)$*.*

This definition goes back to [11]; Rutten [20] has studied bisimulation, as defined by Aczel and Mendler [1] as fundamental notion of equivalence. Both notions agree if the signature functor preserves weak pullbacks; for functors not having this property, behavioural equivalence seems to be the more fundamental notion of equivalence (see [11] for discussion). In the examples, behavioural equivalence can be expressed as follows:

Example 2. (i) Let $T_1X = L \times X$ and suppose (C, γ) and $(D, \delta) \in \mathsf{CoAlg}(T_1)$ Then $(c, d) \in C \times D$ are behaviourally equivalent, if they produce the same trace of labels $l \in L$.

(ii) In the case $T_2X(O \times X)^I$, every state $c \in C$ of a T_2-coalgebra (C, γ) defines a function $f_c : I^\omega \to O^\omega$ (given $i = (i_n)_{n\in\omega}$, let $c = c_0$ and $(o_n, c_{n+1}) = \gamma(c_n)(i_n)$. Put $f_c(i) = (o_n)_{n\in\omega}$). We obtain that two states are behaviourally equivalent, if the associated functions are equal.

(iii) For $T_3X = \mathcal{P}(X)^L$ and (C, γ), $(D, \delta) \in \mathsf{CoAlg}(T_3)$, behavioural equivalence coincides with bisimulation, as used by Park [16] and Milner [14].

The significance of behavioural equivalence is that it identifies precisely those states, which cannot be distinguished from the outside. The logics, which we consider later, will all be invariant under behavioural equivalence.

2.2 Institutions

Institutions [21,6] have been successfully used to describe the interplay between translation of logics and transformations of models along morphisms of signatures:

Definition 3. *Suppose* Sig *is a category (of signatures). An* institution *is a triple* (Mod, Sen, Sig) *where*

- Mod : Sig $\to$ $\mathsf{Cat}^{\mathrm{op}}$ *associates categories of models to signatures*
- Sen : Sig $\to$ Set *associates a set of sentences (formulas) to every signature, and*
- $\models$ *is a family* $(\models_S)$ *of relations* $\models_S \subseteq \mathrm{Mod}(S) \times \mathrm{Sen}(S)$*, indexed by the signatures* $S \in \mathrm{Sig}$

such that the satisfaction condition

$$\mathrm{Mod}(\sigma)(M) \models \phi \quad \iff M \models \mathrm{Sen}(\sigma)(\phi)$$

is satisfies for all $\sigma : S \to S'$*,* $M \in \mathrm{Mod}(S')$ *and* $\phi \in \mathrm{Sen}(S)$*.*

It is the purpose of the paper to establish the satisfaction condition for different types of logics, interpreted over coalgebras. For other examples of institutions, the reader is referred to the original paper by Goguen and Burstall [6]. Dually, we have

Definition 4. *A* coinstitution *over a category* Sig *of signatures consists of*

- *A functor* Mod : Sig $\to$ Cat
- *A functor* Sen : Sig $\to$ $\mathsf{Set}^{\mathrm{op}}$
- *A family* $\models_S$ *of relations* $\models_S \subseteq \mathrm{Mod}(S) \times \mathrm{Sen}(S)$*, indexed by the objects* S *of* Sig*,*

such that the dual of the satisfaction condition

$$\mathrm{Mod}(\sigma)(M) \models \phi \quad \iff M \models \mathrm{Sen}(\sigma)(\phi)$$

is satisfied for all $\sigma : S \to S'$*,* $M \in \mathrm{Mod}(S)$ *and* $\phi \in \mathrm{Sen}(S')$*.*

Note that, in a coinstitution, the translation Mod is covariant, whereas the translation Sen of sentences is contravariant. However, when dualising institutions, we do not obtain a new concept:

Proposition 1. *Suppose* Sig *is a category. Then there is a 1-1 correspondence between institutions over* Sig *and coinstitutions over* $\mathrm{Sig}^{\mathrm{op}}$*.*

Proof. Suppose (Mod, Set, Sig) is an institution over Sig. Then ($\mathrm{Mod}^{\mathrm{op}}, \mathrm{Sen}^{\mathrm{op}}, \models$) is a coinstitution over $\mathrm{Sig}^{\mathrm{op}}$; clearly this construction can be reversed.

In the light of this proposition, the concept of coinstitution is strictly speaking unnecessary. However, for the purposes of the present paper, we prefer to work with coinstitutions. This allows us to take a subcategory $\mathbb{S} \subseteq [T, T]$ of the category of endofunctors (instead of $\mathbb{S}^{\mathrm{op}}$) as a category of signatures.

3 Translation of Models

One of the goals of this paper is to show that three different conceptions of modal logic for coalgebras give rise to an institution. All three logics will be interpreted over coalgebras for endofunctors on sets. Since we think of the underlying endofunctor T as a signature for the corresponding T-coalgebras, signature morphisms need to mediate between endofunctors on Set. The obvious notion for signature morphisms are therefore *natural transformations* (see [13]). Thus, our category Sig of signatures will have endofunctors as objects and natural transformations as morphisms, that is, we take $\mathrm{Sig} \subseteq [\mathsf{Set}, \mathsf{Set}]$ as a (possibly non-full) subcategory of the functor category $[\mathsf{Set}, \mathsf{Set}]$. As far as signatures are concerned, this setup is common to all three types of logics, which we show to carry the structure of an institution. This section describes the model theoretic part, that is, the Mod functor, which translates models along signature morphisms. The translation between models described here is the same for all three conceptions of logics for coalgebras, which are later shown to carry the structure of an institution.

Before we start to study the translation of models (coalgebras) along signature morphisms, we first try to convince the reader that natural transformations are a indeed a natural choice for signature morphisms.

The key observation is the following:

Lemma 1. *Suppose $T, S : \mathsf{Set} \to \mathsf{Set}$ and $\sigma : S \to T$ is a natural transformation. Then $\sigma^\dagger : \mathsf{CoAlg}(S) \to \mathsf{CoAlg}(T)$, defined by $\sigma^\dagger(C, \gamma) = (C, \sigma(C) \circ \gamma)$, is functorial.*

Of course, this observation is not specific to the category of sets. We illustrate the use of natural transformations using the running examples introduced above.

Example 3. We consider natural transformations between the signatures discussed in Example 2.

(i) Every T_1-coalgebra (C, γ) can be viewed as Mealy automaton (that is, as T_2-coalgebra) if we simply ignore the input: put $\gamma'(c)(i) = \gamma(c)$ to obtain a transition function $\gamma' : C \to (C \times L)^I = T_2(C)$. On the level of natural transformations between the corresponding signature functors, this translation is accomplished by $\sigma : T_1 \to T_2$, with $\sigma(X) : L \times X \to (L \times X)^I$ defined by $\sigma(x)(i) = x$.

(ii) We can also view every Mealy automaton as a labelled transition system. Given a set I of inputs and O of outputs of the Mealy automaton, we put $L = O \times I$. Given $(C, \gamma) \in \mathsf{CoAlg}(T_2)$, we obtain a labelled transition system (i.e. a T_3-coalgebra) by letting $\gamma'(c) = \{(i, o, c') \mid \gamma(c)(i) = (o, c')\}$. Using natural transformations, the situation is as follows: Consider $\sigma : T_2 \to T_3$, given by $\sigma(X) : (O \times X)^I \to \mathcal{P}(X)^{O \times I}$ where $\sigma_1(X)(f)(o, i) = \{x \in X \mid f(i) = (o, x)\}$. We obtain $(C, \gamma') = \sigma^\dagger(C, \gamma)$.

Note that we can also treat coalgebras for endofunctors, which depend on an additional parameter in our framework:

Example 4. Suppose $T : \mathbb{C} \times \mathsf{Set} \to \mathsf{Set}$, where $\mathbb{C}$ is an arbitrary category of parameters. In order to emphasise the fact that we think of the first component as parameter, we write $T_A(X)$ for $T(A, X)$. Given a morphism $f : A \to B \in \mathbb{C}$, we obtain a natural transformation $\sigma(X) = T(f, \mathrm{id}_X)$. Identifying $C \in \mathbb{C}$ with the endofunctor T_C, we can thus treat $\mathbb{C}$ as a category of signatures for coalgebras.

4 Abstract Behavioural Logic

This shows, that abstract behavioural logic can be endowed with the structure of an institution. Abstract behavioural logic was studied in [10,11], where the term "logic" is understood in a very general sense:

Definition 5. *A* logic for coalgebras *is a set $\mathcal{L}$ (the* language *of the logic), together with a family $\models$ of relations, indexed by the T-coalgebras, such that $\models_{(C,\gamma)} \subseteq C \times \mathcal{L}$.*

We call a logic behavioural*,*

$$d \models_{(D,\delta)} \phi \iff c \models_{(C,\gamma)} \phi$$

for all formulas $\phi \in \mathcal{L}$ and all behaviourally equivalent states $(c, d) \in C \times D$. As usual, $(C, \gamma) \models \phi$ iff $\forall c \in C.c \models_{(C,\gamma)} \phi$, and $[\![\phi]\!]_{(C,\gamma)} = \{c \in C \mid c \models_{(C,\gamma)} \phi$.

The starting point of the investigations conducted in [10,11] is the representation of formulas of a behavioural logic as subsets of the final T-coalgebra (assuming it exists). This representation can be formulated as follows:

Proposition 2. *Suppose $(Z, \zeta) \in \mathsf{CoAlg}(T)$ is final and $\mathcal{L}$ is a behavioural logic for T-coalgebras. Then*

$$[\![\phi]\!]_{(C,\gamma)} = !^{-1}([\![\phi]\!]_{(Z,\zeta)})$$

for all $(C, \gamma) \in \mathsf{CoAlg}(T)$ and all $\phi \in \mathcal{L}$, where $! : C \to Z$ is the morphism given by finality.

Proof. Immediate from the definition of behavioural equivalence.

Thus, every formula ϕ of a behavioural logic can be semantically represented as a subset of the final T-coalgebra. Thus, if (Z, ζ) is final in $\mathsf{CoAlg}(T)$, we can view $\mathcal{P}(Z)$ as behavioural logic with $c \models_{(C,\gamma)} \phi$ if $!(c) \in \phi$, where $\phi \in \mathcal{P}(Z)$ and $! : C \to Z$ is the final morphism:

Definition 6. *Suppose (Z, ζ) is final in $\mathsf{CoAlg}(T)$. The* abstract behavioural logic $\mathcal{A}_T = \mathcal{P}(Z)$ *has subsets of the final T-coalgebra as formulas. Satisfaction is given by $c \models_{(C,\gamma)} \phi$ if $!(c) \in \phi$.*

It is immediately obvious from the definition of behavioural logic, that abstract behavioural logic is indeed behavioural. We now add signature morphisms to the picture. So suppose $\sigma : S \to T$ is a natural transformation. If (Z_S, ζ_S) is final in $\mathsf{CoAlg}(S)$, then $\sigma^\dagger(Z_S, \zeta_S) \in \mathsf{CoAlg}(T)$, thus, assuming (Z_T, ζ_T) is final in $\mathsf{CoAlg}(T)$, we have a unique morphism $! : Z_S \to Z_T$, the inverse image of which induces a translation $\mathcal{A}_T \to \mathcal{A}_S$ between the abstract logics associated to T and S.

Proposition 3. *Suppose $\sigma : S \to T$, and S, T allow for final coalgebras (Z_S, ζ_S) and (Z_T, ζ_T), respectively. Then*

$$(C, \gamma) \models \sigma^*(\phi) \quad \iff \quad \sigma^\dagger(C, \gamma) \models \phi$$

for all $(C, \gamma) \in \mathsf{CoAlg}(S)$ and all $\phi \in \mathcal{A}_T$, where $\sigma^* = !^{-1}$ for the unique morphism $! : \sigma^\dagger(Z_S, \zeta_S) \to (Z_T, \zeta_T)$, given by finality.

Proof. Suppose $(C, \gamma) \in \mathsf{CoAlg}(S)$ and consider the diagram

$$\begin{array}{ccccc}
C & \xrightarrow{u} & Z_S & \xrightarrow{v} & Z_T \\
\downarrow{\scriptstyle \gamma} & & \downarrow{\scriptstyle \zeta_S} & & \\
SC & \xrightarrow{Su} & SZ_S & & \downarrow{\scriptstyle \zeta_T} \\
\downarrow{\scriptstyle \sigma(C)} & & \downarrow{\scriptstyle \sigma(Z_S)} & & \\
TC & \xrightarrow{Tu} & TZ_S & \xrightarrow{Tv} & TZ_T
\end{array}$$

where u is the morphism given by finality of (Z_S, ζ_S) and v is the morphism given by finality of (Z_T, ζ_T). Suppose $\phi \in \mathcal{A}_T$ and $c \in C$. Then $c \models_{(C,\gamma)} \sigma^*(\phi)$ iff $u(c) \in v^{-1}(\phi)$ iff $u \circ v(c) \in \phi$ iff $c \models_{\sigma^\dagger(C,\gamma)} \phi$, since $v \circ u : \sigma^\dagger(C, \gamma) \to (Z_T, \zeta_T)$ is equal to the unique morphism given by finality of (Z, ζ_T).

If we take some care in setting up our category of signatures Sig as to ensure that every endofunctor $T \in \mathrm{Sig}$ admits a final coalgebra (otherwise abstract behavioural logic isn't meaningful), we obtain:

Theorem 1. *Suppose $\mathrm{Sig} \subseteq [\mathsf{Set}, \mathsf{Set}]$ is a subcategory such that every $T \in \mathrm{Sig}$ admits a final T-coalgebra. Let $\mathrm{Sen}(T) = \mathcal{A}(T)$ and $\mathrm{Sen}(\sigma) = \sigma^\dagger$. Then $(\mathrm{Sig}, \mathrm{Mod}, \mathrm{Sen}, \models)$, with $\models$ as in Definition 6, is a coinstitution.*

This theorem shows, that behavioural logics are an institution, if we replace the concrete syntax by a semantical abstraction. We now turn to coalgebraic logic, the language of which is given inductively as initial algebra.

5 Coalgebraic Logic

Coalgebraic Logic, due to Moss [15], is a modal logic, interpreted over coalgebras. The main feature of coalgebraic logic is the insight, that – on the level of T-coalgebras for an arbitrary endofunctor T – modal operators can be expressed using functor application. It turns out that coalgebraic logic, as originally defined by Moss [15] is *not* an institution: one cannot translate formulas along non-injective signature morphisms. However, adapting the definition slightly, we obtain a logic, which is an institution and into which coalgebraic logic can be conservatively embedded. In the original paper, the language of coalgebraic logic comprises a (in general proper) class of formulas, and is constructed by extending the endofunctor T to classes (assuming that T is standard and set-based). Here, we give an alternative (but equivalent) presentation of coalgebraic logic, which dispenses with the use of classes at the expense of assuming the existence of an inaccessible cardinal. Instead of assuming T to be standard and set-based, we assume that T is κ-accessible, for some inaccessible cardinal κ. In a nutshell, the accessibility condition assures that the image of T on a set is already determined by the image of T on sets of cardinality less than κ; this is a technical requirement wich ensures the existence of initial algebras, which constitute a part of the syntax of coalgebraic logic. We make this choice simply because we think that accessibility of an endofunctor is – for most readers – a more familiar concept than being standard and set based.

The second condition we have to require is, that T extends to an endofunctor $\hat{T}$ on the category Rel of sets and relations (we often write $A \nrightarrow B$ for a relation $R \subseteq A \times B$). This extension is given by $\hat{T}X = TX$ for sets X and $\hat{T}R = T\pi_2 \circ (T\pi_1)^{-1}$, for a relation $R : A \nrightarrow B$ with associated projections $\pi_1 : R \to A$ and $\pi_2 : R \to B$ (this is as in [15]). It is well known (the original reference is [4]), that functoriality of $\hat{T}$ is equivalent to T preserving weak pullbacks. We now introduce syntax and semantics of coalgebraic logic, where we assume throughout the section, that T is κ-accessible for some inaccessible κ and preserves weak pullbacks and denote the bounded powerset functor by $\mathcal{P}_\kappa$, that is, $\mathcal{P}_\kappa(X) = \{\mathfrak{x} \subseteq X \mid \mathsf{card}(\mathfrak{x}) < \kappa\}$.

Definition 7. *Let $L_T = \mathcal{P}_\kappa + \mathcal{P}_\kappa \circ T$. The* syntax *of coalgebraic logic is the carrier $\mathcal{L}_T$ of the initial L_T-algebra $(\mathcal{L}_T, \iota_T)$.*

If $(C, \gamma) \in \mathsf{CoAlg}(T)$, put $d_C : \mathcal{P}_\kappa \mathcal{P}(C) \to \mathcal{P}(C), d_C(\mathfrak{x}) = \cap \mathfrak{x}$ and $e_C : \mathcal{P}_\kappa T\mathcal{P}(C) \to \mathcal{P}(C)$, $e_C(x) = \{c \in C \mid \exists y \in x.(\gamma(c), y) \in \hat{T}(\epsilon_C)\}$, where $\epsilon_C \subseteq C \times \mathcal{P}(C)$ is the membership relation.

The semantics *$[\![\cdot]\!]_{(C,\gamma)} : \mathcal{L}_T \to \mathcal{P}(C)$ of $\mathcal{L}_T$ with respect to (C, γ) is the unique function with $[d_C, e_C] \circ L_T([\![\cdot]\!]_{(C,\gamma)}) = [\![\cdot]\!]_{(C,\gamma)} \circ \iota_T$. If $c \in [\![\phi]\!]_{(C,\gamma)}$, we also write $c \models_{(C,\gamma)} \phi$; we drop the subscript (C, γ) whenever there is no danger of confusion; also $(C, \gamma) \models \phi$ iff $c \models_{(C,\gamma)} \phi$ for all $c \in C$.*

Note $\mathcal{L}$ contains $\mathtt{tt} = \bigwedge \emptyset$ and is closed under conjunctions of size $< \kappa$.

In the above definition, the auxiliary familiy of functions d_C is used to interpret conjunctions, and e takes care of the modalities. Note that the initial L_T-algebra $(\mathcal{L}_T, \iota_T)$ always exists since L_T is κ-accessible, see [2]. If $\mathrm{in}_1 : \mathcal{P}_\kappa \mathcal{L}_T \to \mathcal{P}_\kappa \mathcal{L}_T + \mathcal{P}_\kappa T \mathcal{L}_T$ and $\mathrm{in}_2 : T\mathcal{L}_T \to \mathcal{P}_\kappa \mathcal{L}_T + T\mathcal{L}_T$ denote the coproduct injections, we write $\bigwedge_T = \iota_T \circ \mathrm{in}_1$ and $\nabla_T = \iota \circ \mathrm{in}_2$. The language of coalgebraic logic can thus be described as the least set such that

$$\Phi \subseteq \mathcal{L}_T, \mathsf{card}(\Phi) < \kappa \implies \bigwedge_T \Phi \in \mathcal{L}_T$$

$$\phi \subseteq T\mathcal{L}_T \implies \nabla_T \phi \in \mathcal{L}_T$$

This presentation also highlights the (only) difference compared to Moss' original definition, where one does not take *subsets* of $T\mathcal{L}_T$ in the second clause, but *elements* of $T\mathcal{L}_T$.

If $(C, \gamma) \in \mathsf{CoAlg}(T)$, we then obtain

$$c \models \bigwedge_T \Phi \text{ iff } c \models \phi \text{ for all } \phi \in \Phi$$

$$c \models \nabla_T \Phi \text{ iff} (\gamma(c), \phi) \in \hat{T}(\models) \text{for some } \phi \in \Phi$$

for subsets $\Phi \subseteq \mathcal{L}_T$ of cardinality less than κ and $\phi \in T\mathcal{L}_T$.

We give a brief example of the nature of coalgebraic logic; for an in-depth discussion and more example see Moss' original article [15].

Example 5. Let $TX = L \times X$, where L is a set of labels; we drop the subscript "T" on $\mathcal{L}$ and ∇. As already mentioned, $\mathtt{tt} \in \mathcal{L}$ and obviously $[\![\mathtt{tt}]\!] = C$ for all $(C, \gamma) \in \mathsf{CoAlg}(T)$. If $l \in L$, we have $\{(l, \mathtt{tt})\} \subseteq T\mathcal{L}$, hence $\nabla\{(l, \mathtt{tt})\} \in \mathcal{L}$. Unravelling the definitions, one obtains $c \models \nabla\{(l, \mathtt{tt})\}$ if $\pi_1 \circ \gamma(c) = l$. In the same manner, one has $\nabla\{(m, \nabla\{(l, \mathtt{tt})\})\} \in \mathcal{L}$ for $m \in L$

with $c \models \nabla\{(m, \nabla\{(l, \mathtt{tt})\})\}$ iff the stream associated to c (cf. Example 1) begins with m and is followed by l.

Note that – if we restrict ourselves to singleton sets (as in the original paper [15]) as arguments of ∇, we cannot express the fact that a stream starts with l_0 *or* l_1 logically. This is the reason why coalgebraic logic, in its original formulation, fails to be a coinstitution: We cannot translate formulas along a signature morphism, which identifies two labels l_0 and l_1.

The generalisation of the original definition of coalgebraic logic does not allow us to distinguish bisimilar states. In other words, we have:

Proposition 4. *$\mathcal{L}_T$ is behavioural.*

Proof. It suffices to show that $f(c) \models_{(D,\delta)} \phi$ iff $c \models_{(C,\gamma)} \phi$ whenever $\phi \in \mathcal{L}_T$, $f : (C, \gamma) \to (D, \delta) \in \mathsf{CoAlg}(T)$ and $c \in C$. This follows from the fact that $f^{-1} : \mathcal{P}(D) \to \mathcal{P}(C)$ is a morphism of the L_T-algebras $(\mathcal{P}(D), [d_D, e_D])$ and $(\mathcal{P}(C), [d_C, e_C])$, where d_D, e_D, d_C and e_C are as in Definition 7.

To see that f^{-1} is a morphism of algebras, it suffices to show that

$$\begin{array}{ccc} \mathcal{P}_\kappa T\mathcal{P}(D) & \xrightarrow{\mathcal{P}_\kappa T(f^{-1})} & \mathcal{P}_\kappa T\mathcal{P}(C) \\ {\scriptstyle e_D}\downarrow & & \downarrow{\scriptstyle e_C} \\ \mathcal{P}(D) & \xrightarrow[f^{-1}]{} & \mathcal{P}(D) \end{array}$$

commutes. For $c \in C$ and $x \in \mathcal{P}_\kappa T\mathcal{P}D$, we have

$$\begin{aligned} & c \in e_C \circ \mathcal{P}_\kappa T(f^{-1})(x) \\ \text{iff } & \exists y \in x.(Tf \circ \gamma(c), y) \in \hat{T}(\in_D) \\ \text{iff } & \exists y \in x.(\delta \circ f(c), y) \in \hat{T}(\in_D) \\ \text{iff } & c \in f^{-1} \circ e_D(x), \end{aligned}$$

which shows the claim.

We now turn to show that coalgebraic logic forms an institution. Here, a little care is needed when setting up the category of signatures and the category of models: Recall that we have required T to be κ-accessible for some inaccessible κ. To show the satisfaction condition (and to define the appropriate translations), we need to restrict the cardinality of the models to $< \kappa$ and require that T restricts to the full subcategory of sets, which are of cardinality less than κ. Working with classes, this would be unnecessary – we would have to require the dual condition that T can be continuously extended to classes.

Definition 8. *A κ-accessible endofunctor is* below κ *if $|TX| < \kappa$ whenever $|X| < \kappa$.*

Most κ-accessible functors are indeed below κ. The prime example of a κ-accessible functor, which is not below κ is the constant functor with value κ. The following lemma gives a characterisation of functors below κ, which just depends on the value of the functor at 1.

Lemma 2. *Suppose T is κ-accessible. Then T is below κ if $|T1| < \kappa$.*

Proof. If $|X| < \kappa$, then the diagram $(\{x\} \hookrightarrow X \mid x \in X)$ is κ-filtered. The claim follows from κ being inaccessible and from the construction of κ-filtered colimits (see [3]).

In order establish the satisfaction condition, we additionally have to require that the natural transformation σ is compatible with the extensions $\hat{S}$ and $\hat{T}$ to relations. That is, we require that $G(\sigma) : \hat{S} \to \hat{T}$ is natural, where $G(\sigma)(X) = G(\sigma(X)) : \hat{S}X \mathrel{\mapstochar\rightarrow} \hat{T}X$ is defined as the graph of $\sigma(X)$, for X a set. In this case, we call σ *relational*.

Many natural transformations can be shown to be relational using the following criterion:

Lemma 3. *A natural transformation $\sigma : S \to T$ is relational, if every naturality square,*

$$\begin{array}{ccc} SA & \xrightarrow{\sigma(A)} & TA \\ {\scriptstyle Sf}\downarrow & & \downarrow{\scriptstyle Tf} \\ SB & \xrightarrow[\sigma(B)]{} & TB \end{array}$$

where $f : A \to B$, is a weak pullback.

Proof. Suppose A, B are sets and $R : A \mathrel{\mapstochar\rightarrow} B$ is a relation; we need to show that

$$\begin{array}{ccc} SA & \xrightarrow{G(\sigma(A))} & TA \\ {\scriptstyle \hat{S}(R)}\downarrow & & \downarrow{\scriptstyle \hat{T}(R)} \\ SB & \xrightarrow[G(\sigma(B))]{} & TB \end{array}$$

commutes in Rel.

First suppose that $(x, y) \in G(\sigma(B)) \circ \hat{S}(R)$. Thus there is some $x_1 \in \hat{S}(R)$ with $S\pi_1(x_1) = x$ and $\sigma(B) \circ S\pi_2(x_1) = y$. Put $y_1 = \sigma(R)(x_1)$. Then $T\pi_1(y_1) = \sigma(A)(x)$ and $T\pi_2(y_1) = y$, hence $(x, y) \in \hat{T}(R) \circ G(\sigma(A))$.

Now let $(x, y) \in \hat{T}(R) \circ G(\sigma(A))$. As above, there is $y_1 \in TR$ with $T\pi_1(y_1) = \sigma(A)(x)$ and $T\pi_2(y_1) = y$. Since

$$\begin{array}{ccc} SR & \xrightarrow{\sigma(R)} & TR \\ {\scriptstyle S\pi_1}\downarrow & & \downarrow{\scriptstyle T\pi_1} \\ SA & \xrightarrow[\sigma(A)]{} & TA \end{array}$$

is a weak pullback, there is $x_1 \in SR$ with $\sigma(R)(x_1) = y_1$ and $S\pi_1(x_1) = x$. Using naturality of σ, we obtain $S\pi_1(x_1) = x$ and $\sigma(B) \circ S\pi_2(x_1) = y$, so $(x, y) \in \sigma(B) \circ SR$.

Using the fact that products, coproducts, the powerset functor, identity functor and constant functors preserve weak pullbacks, we have the following criterion, which can be applied to a large class of signatures, obtained via parameterised functors (cf. Example 4).

Corollary 1. *Suppose $T : \mathsf{Set} \times \mathsf{Set} \to \mathsf{Set}$ is built using products, coproducts, the powerset functor, identity functor and constant functors only. Then, given $f : A \to B$, the natural transformation $T(f, \mathrm{id}) : T_A \to T_B$ is relational.*

Given a (not necessary relational) transformation $\sigma : S \to T$, we can define a translation $\sigma^* : \mathcal{L}_T \to \mathcal{L}_S$ as follows: Since $\mathcal{L}_T$ supports the structure ι_T of an initial T-algebra, every L_T-algebra structure $t : L_T\mathcal{L}_S \to \mathcal{L}_S$ defines a unique mapping $\sigma^* : \mathcal{L}_T \to \mathcal{L}_S$ with $\sigma^* \circ \iota_T = t \circ L_T(\sigma^*)$. So we have to find an appropriate L_T-algebra structure t on $\mathcal{L}_S$. We let $t = [t_1, t_2]$ where $t_1 : \mathcal{P}_\kappa\mathcal{L}_S \to \mathcal{L}_S$ is intersection and $t_2 : \mathcal{P}_\kappa \circ T\mathcal{L}_S \to \mathcal{L}_S$ is given as $t_2 = \nabla_T \circ \sigma(\mathcal{L}_S)^{-1}$. Note that $\sigma(\mathcal{L}_S)^{-1}$ maps $\mathcal{P}_\kappa(T\mathcal{L}_S) \to \mathcal{P}_\kappa(S\mathcal{L}_S)$.

Proposition 5. *Suppose* $\sigma : S \to T$ *is relational and* $(C, \gamma) \in \mathsf{CoAlg}(S)$. *Then*

$$(C, \gamma) \models \sigma^*(\phi) \iff \sigma^\dagger(C, \gamma) \models \phi$$

for all $\phi \in \mathcal{L}_T$, *provided* $|C| < \kappa$.

Proof. Let $d = d_C$ and $e = e_C$ be as in Definition 7 and suppose $d^\dagger(\mathfrak{x}) = \cap\mathfrak{x}$ and $e^\dagger(\mathfrak{x}) = \{c \in C \mid \exists y \in x.(\sigma(C) \circ \gamma(c), y) \in \hat{T}(\in_C)\}$. Then, by definition of $[\![\cdot]\!]_T$, we have $[\![\cdot]\!]_T \circ [\bigwedge_T, \nabla_T] = [d^\dagger, e^\dagger] \circ \mathcal{P}_\kappa[\![\cdot]\!]_T + \mathcal{P}_\kappa T[\![\cdot]\!]_T$. Consider the following diagram:

$$\begin{array}{ccccc}
L_T\mathcal{L}_T & \xrightarrow{L_T\sigma^*} & L_T\mathcal{L}_S & \xrightarrow{L_T[\![\cdot]\!]_S} & L_T\mathcal{P}(C) \\
\Big\downarrow & & \Big\downarrow{\scriptstyle \mathrm{id}+\sigma(\mathcal{L}_S)^{-1}} & & \Big\downarrow{\scriptstyle \mathrm{id}+\sigma(\mathcal{P}C)^{-1}} \\
{\scriptstyle [\bigwedge_T, \nabla_T]} & & L_S\mathcal{L}_S & \xrightarrow{L_S[\![\cdot]\!]_S} & L_S\mathcal{P}(C) \\
\Big\downarrow & & \Big\downarrow{\scriptstyle [\bigwedge_S, \nabla_S]} & & \Big\downarrow{\scriptstyle [d,e]} \\
\mathcal{L}_T & \xrightarrow{\sigma^*} & \mathcal{L}_S & \xrightarrow{[\![\cdot]\!]_S} & \mathcal{P}(C)
\end{array}$$

The left hand square commutes by definition of σ^* and the lower right hand square by definition of $[\![\cdot]\!]_S$. We show that

(i) $[d, e] \circ (\mathrm{id} + \sigma(\mathcal{P}C)^{-1}) = [d^\dagger, e^\dagger]$
(ii) The top right corner commutes.

Both claims then entail the satisfaction condition as stated.

Ad 1: Since σ is relational, we have $\hat{T}(\in_C) \circ G(\sigma(C)) = G(\sigma(\mathcal{P}C)) \circ \hat{S}(\in_C)$. Now let $c \in C$ and $x \in \mathcal{P}_\kappa T\mathcal{P}(C)$. We have

$$\begin{aligned}
&c \in e^\dagger(x) \\
\text{iff } &\exists y \in x.(\gamma(c), y) \in G(\sigma(\mathcal{P}C)) \circ S(\in_C) \\
\text{iff } &\exists z \in \sigma(\mathcal{P}C)^{-1}(x).(\gamma(c), z) \in \hat{S}(\in_C) \\
\text{iff } &c \in e \circ \sigma(\mathcal{P}C)^{-1}(x).
\end{aligned}$$

Ad 2: If R is a relation, we denote the opposite relation by R^{op}. Then, for a function f, we have $G(Tf)^{\mathrm{op}} = \hat{T}((Gf)^{\mathrm{op}})$, and similarly for S. Also note that $\hat{T}(G[\![\cdot]\!]_s^{\mathrm{op}}) \circ G(\sigma(\mathcal{P}C)) = G(\sigma(\mathcal{L}_S) \circ \hat{S}(G[\![\cdot]\!]_S^{\mathrm{op}})$ since σ is relational. Having said that, we obtain for $\phi \in \mathcal{P}_\kappa T\mathcal{L}_S$ and $c \in S\mathcal{P}C$:

$$\begin{aligned}
&c \in \mathcal{P}_\kappa S[\![\cdot]\!]_S \circ \sigma(\mathcal{L}_S)^{-1}(\phi) \\
\text{iff } &\exists \psi \in \phi.(c, \psi) \in G(\sigma(\mathcal{L}_S)) \circ \hat{S}(G[\![\cdot]\!]_S^{\mathrm{op}}) \\
\text{iff } &\exists \psi \in \phi.(c, \psi) \in \hat{T}(G[\![\cdot]\!]_S^{\mathrm{op}}) \circ G(\sigma(\mathcal{P}C)) \\
\text{iff } &c \in \sigma(\mathcal{P}C)^{-1} \circ \mathcal{P}_\kappa T[\![\cdot]\!]_S(\phi),
\end{aligned}$$

that is, the satisfaction condition holds.

Taking some care when choosing signatures and models, coalgebraic logic is a coinstitution ($\mathrm{Mod}^\kappa(T)$ is the full subcategory of T-coalgebras with carrier $< \kappa$):

Theorem 2. *Suppose* $\mathrm{Sig} \subseteq [\mathsf{Set}, \mathsf{Set}]$ *is a subcategory such that*

- *Each* $T \in \mathrm{Sig}$ *is below* κ
- *Each* $\sigma : S \to T \in \mathrm{Sig}$ *is relational.*

Then $(\mathrm{Sig}, \mathrm{Mod}^\kappa, \mathrm{Sen}, \models)$*, with* $\mathrm{Sen}(T) = \mathcal{L}_T$ *and* $\mathrm{Sen}(\sigma) = \sigma^*$*, is a coinstitution.*

6 Coalgebraic Modal Logic

We have seen in the previous sections, that abstract behavioural logic and coalgebraic logic are coinstitutions. The formulation of abstract modal logic is completely syntax-free; the language of coalgebraic modal logic is abstract in that it is given as initial algebra. We now investigate coalgebraic modal logic, the language of which is concretely given as propositional logic, enriched with modal operators. Coalgebraic modal logic is based on the observation, that predicate liftings, which we now introduce, generalise modal operators from Kripke models to coalgebras for arbitrary signature functors.

Predicate liftings were first considered by Jacobs and Hermida [9] in the context of coinduction principles and later by Rößiger [18] and Jacobs [8] in the context of modal logic. There, as well as in the related paper [18], predicate liftings appear as syntactically defined entities, and naturality is a derived property. The notion of predicate lifting used in the present exposition is more general, and takes naturality as the defining property.

Definition 9. *A* predicate lifting *for* T *is a natural transformation* $\lambda : 2 \to 2 \circ T$*, where* $2 : \mathsf{Set} \to \mathsf{Set}^{\mathrm{op}}$ *denotes the contravariant powerset functor.*

The next example shows, that predicate liftings do not only capture modal operators, but can also be used to interpret atomic propositions.

Example 6. Suppose $TX = \mathcal{P}(X) \times \mathcal{P}(A)$. Then every T-coalgebra (C, γ) defines a Kripke model $\mathbb{K}(C, \gamma) = (C, R, V)$ over the set A of atomic propositions: the accessibility relation is given by $(c, c') \in R$ iff $c' \in \pi_1 \circ \gamma(c)$ and for $a \in A$ we have $V(a) = \{c \in C \mid a \in \pi_2 \circ \gamma(c)\}$.

Now, for a set C, consider the operation $\lambda(C) : \mathcal{P}(C) \to \mathcal{P}(TC)$ given by $\lambda(C)(\mathfrak{c}) = \{(\mathfrak{c}', \mathfrak{a}) \in TC \mid \mathfrak{c}' \subseteq \mathfrak{c}\}$. It is easy to see that λ defines a predicate lifting for T. Now suppose $(C, \gamma) \in \mathsf{CoAlg}(T)$ and $\mathfrak{c} \subseteq C$, which we think of as the interpretation of a modal formula ϕ. Under the correspondence outlined above, we have $\gamma^{-1} \circ \lambda(C)(\mathfrak{c}) = \{c \in C \mid \forall c'.(c, c') \in R \implies c' \in \mathfrak{c}\}$, corresponding to the interpretation of the formula $\Box\phi$.

For the case of atomic propositions, consider the constant lifting, defined by $\alpha(C)(\mathfrak{c}) = \{(\mathfrak{c}', \mathfrak{a}) \in TC \mid a \in \mathfrak{a}\}$. Again, an easy calculation shows that α is a predicate lifting. Identifying T-coalgebras with Kripke models via the correspondence above, we obtain for $(C, \gamma) \in \mathsf{CoAlg}(T)$ and an arbitrary subset $\mathfrak{c} \subseteq C$ that $\gamma^{-1} \circ \alpha(\mathfrak{c}) = V(a)$, that is, the set of states which validate the proposition a.

This leads us to study propositional logic, enriched with predicate liftings, as a logic for coalgebras.

Definition 10. *Suppose* $T : \mathsf{Set} \to \mathsf{Set}$ *and* Λ *is a set of predicate liftings for* T*. The* language $\mathcal{L}(\Lambda_T)$ *of coalgebraic modal logic associated with* T *and* Λ *is the least set according to the grammar*

$$\phi ::= \mathrm{ff} \mid \phi \to \psi \mid [\lambda]\phi \qquad (\lambda \in \Lambda).$$

Given $(C,\gamma) \in \mathsf{CoAlg}(T)$*, the semantics* $[\![\phi]\!]_{(C,\gamma)} = [\![\phi]\!]$ *of formulas* $\phi \in \mathcal{L}(\Lambda)$ *is given by:*

$$[\![\mathrm{ff}]\!] = \emptyset$$
$$[\![\phi \to \psi]\!] = (C \setminus [\![\phi]\!]) \cup [\![\psi]\!]$$
$$[\![[\lambda]\phi]\!] = \gamma^{-1} \circ \lambda(C)([\![\phi]\!]).$$

As usual, we write $c \models_{(C,\gamma)} \phi$ *(and drop the subscript if there is no danger of confusion), if* $c \in [\![\phi]\!]_{(C,\gamma)}$*. As usual, we write* $(C,\gamma) \models \phi$ *if* $c \models_{(C,\gamma)} \phi$ *for all* $c \in C$*.*

An easy induction on the structure of formulas shows, that coalgebraic modal logic cannot distinguish between states, which are behaviourally equivalent.

Lemma 4. *Coalgebraic modal logic is behavioural.*

Proof. By induction on the structure of formulas, one shows that $[\![\phi]\!]_{(C,\gamma)} = f^{-1}([\![\phi]\!]_{(D,\delta)})$ for $\phi \in \mathcal{L}(\Lambda)$ and a morphism of coalgebras $f : (C,\gamma) \to (D,\delta)$. The claim follows from the definition of behavioural equivalence.

We now investigate the effect of signature morphisms on formulas. The key observation is the following:

Lemma 5. *Suppose* $\sigma : S \to T$ *is natural and* λ *is a predicate lifting for* T*. Then* $\sigma^{-1} \circ \lambda$ *is a predicate lifting for* S*.*

The proof is a straightforward calculation, and therefore omitted. That is, a signature morphism $\sigma : S \to T$ translates the modal operators associated with T to modal operators for S. This defines an inductive translation between languages for S to languages for T:

Definition 11. *Suppose* $\sigma : S \to T$ *and suppose that* Λ_T, Λ_S *are sets of predicate liftings for* T *and* S*, respectively. Let* $\sigma^{-1}(\Lambda) = \{\sigma^{-1} \circ \lambda \mid \lambda \in \Lambda\}$*. If* $\Lambda_S \subseteq \sigma^{-1}(\Lambda_T)$*, we define* $\sigma^* : \mathcal{L}(\Lambda_S) \to \mathcal{L}(\sigma^{-1}(\Lambda_T))$ *by*

$$\sigma^*(\mathrm{ff}) = \mathrm{ff}$$
$$\sigma^*(\phi \wedge \psi) = \sigma^*(\phi) \wedge \sigma^*(\phi)$$
$$\sigma^*([\lambda]\phi) = [\sigma^{-1} \circ \lambda]\phi$$

Using this translation, we have the following property, which immediately entails the satisfaction condition:

Lemma 6. *Suppose* $\sigma : S \to T$ *and* Λ_T, Λ_S *are sets of predicate liftings for* T *and* S*, respectively, with* $\sigma^{-1}(\lambda_T) \subseteq \Lambda_S$*. Then*

$$[\![\sigma^*(\phi)]\!]_{(C,\gamma)} = [\![\phi]\!]_{\sigma^\dagger(C,\gamma)}$$

for all $(C,\gamma) \in \mathsf{CoAlg}(S)$ *and all* $\phi \in \mathcal{L}(\Lambda_T)$*.*

Proof. We proceed by induction on the structure of formulas and do the only interesting case $\phi = [\lambda]\psi$; by induction hypothesis we may assume that $[\![\sigma^*\psi]\!]_{(C,\gamma)} = [\![\psi]\!]_{\sigma^\dagger(C,\gamma)}$. We obtain

$$\begin{aligned} [\![\sigma^*([\lambda]\phi)]\!]_{(C,\gamma)} &= \gamma^{-1} \circ \sigma^{-1} \circ \lambda(C)([\![\sigma^*(\psi)]\!]_{(C,\gamma)}) \\ &= (\gamma^\dagger)^{-1} \circ \lambda(C)([\![\psi]\!]_{\sigma^\dagger(C,\gamma)}) \\ &= [\![[\lambda]\psi]\!]_{\sigma^\dagger(C,\gamma)}, \end{aligned}$$

which finishes the proof.

Again, we have to pay some attention when setting up the category of signatures in order to obtain an institution.

Theorem 3. *Suppose* $\mathrm{Sig} \subseteq [\mathsf{Set}, \mathsf{Set}]$ *is a subcategory, and*

- Λ_T *is a set of predicate liftings for all* $T \in \mathrm{Sig}$*, and*
- $\sigma^{-1}(\Lambda_T) \subseteq \Lambda_S$ *for all* $\sigma : S \to T \in \mathrm{Sig}$.

Then $(\mathrm{Sig}, \mathrm{Mod}, \mathrm{Sen}, \models)$ *is an institution, where* $\mathrm{Sen}(T) = \mathcal{L}(\Lambda_T)$ *for* $T \in \mathrm{Sig}$ *and* $\mathrm{Sen}(\sigma) = \sigma^*$ *for* $\sigma : S \to T \in \mathrm{Sig}$.

7 Conclusions and Related Work

We have addressed the question whether logics for coalgebras can be translated along signature morphisms, as to form an institution. The answer was "in general yes, but one has to take a little care when setting up the framework."

It is well known, that algebras form institutions with respect to different kinds of logics. Therefore, one might be lead to expect that *co*algebras and their logics congregate in some kind of *co*institution. This is true to the same extent as coalgebras and their logics form an institution, since there is a one-to-one correspondence between institutions over a category $\mathbb{S}$ and coinstitutions over $\mathbb{S}^{\mathrm{op}}$ (Proposition 1).

Hence, instead of showing that coalgebras and their logics form an institution, we can equivalently show that they are coinstitutions. We prefer the latter, since we feel more comfortable with a category $\mathbb{S} \subseteq [T, T]$ of signatures than with its opposite; but this is clearly just a matter of taste.

We then showed that the dual of the satisfaction condition holds for three different types of logics for coalgebras: Abstract Behavioural Logic, Coalgebraic Logic and Coalgebraic Modal Logic. The framework of abstract behavioural logic is based on the observation, that formulas of a behavioural logic can be represented as subsets of the final coalgebra, if the latter exists. This leads to a translation not of formulas, but of the associated representations, resulting in an institution (Theorem 1). For the second type of logic, Moss' coalgebraic logic, the syntax needed to be modified slightly to obtain an institution. We have showed that this modification does not increase the expressive power of the logic (Proposition 4) and gives rise to an institution (Theorem 2). The third framework which we have studied is coalgebraic modal logic, which – in contrast to the ones mentioned before – comes with a concrete syntax, given by a set of predicate liftings for the endofunctor under consideration. The key observation here is, that predicate liftings translate along signature morphisms (Lemma 5), thus giving rise to an inductively defined translation between logics for different signature functors. This translation is well-behaved, witnessed by the fact that coalgebraic modal logic also forms an institution (Theorem 3).

The question whether logics for coalgebras form institutions was also taken up in [5,17]. In [5], the satisfaction condition was established for an inductively defined class of functors, so-called "Kripke polynomial functors", on a category of sorted sets. In contrast, our approach is purely semantical and can be seen to subsume the one-sorted case, treated in [5]. In [17], the satisfaction condition was only established for the case of coalgebraic modal logic. A purely semantical study about the relationship between categories of coalgebras for parameterised endofunctors was already carried out in [12].

References

1. P. Aczel and N. Mendler. A Final Coalgebra Theorem. In D. H. Pitt et al, editor, *Category Theory and Computer Science*, volume 389 of *Lect. Notes in Comp. Sci.*, pages 357–365. Springer, 1989.
2. J. Adámek. Free algebras and automata realizations in the language of categories. *Comment. Math. Univ. Carolinae*, 15:589–602, 1974.
3. F. Borceux. *Handbook of Categorical Algebra*, volume 2. Cambridge University Press, 1994.
4. A. Carboni, G. Kelly, and R. Wood. A 2-categorical approach to change of base and geometric morphisms I. *Cahiers de Topologie et Géometrié Différentielle Catégoriques*, 32(1):47–95, 1991.
5. Corina Cirstea. Institutionalizing coalgebraic modal logic. In L. Moss, editor, *Coalgebraic Methods in Computer Science (CMCS 2002)*, volume 65 of *Electr. Notes in Theoret. Comp. Sci.* Elsevier Science Publishers, 2002.
6. J. Goguen and R. Burstall. Institutions: Abstract Model Theory for Specification and Programming. *Journal of the Association for Computing Machinery*, 39(1), 1992.
7. R. Goldblatt. A calculus of terms for coalgebras of polynomial functors. In Marina Lenisa Andrea Corradini and Ugo Montanari, editors, *Coalgebraic Methods in Computer Science (CMCS'2001)*, volume 44 of *Electr. Notes in Theoret. Comp. Sci.*, 2001.
8. B. Jacobs. Many-sorted coalgebraic modal logic: a model-theoretic study. *Theoret. Informatics and Applications*, 35(1):31–59, 2001.
9. B. Jacobs and C. Hermida. Structural Induction and Coinduction in a Fibrational Setting. *Information and Computation*, 145:107–152, 1998.
10. A. Kurz. A Co-Variety-Theorem for Modal Logic. In *Proceedings of Advances in Modal Logic 2, Uppsala, 1998.* Center for the Study of Language and Information, Stanford University, 2000.
11. A. Kurz. *Logics for Coalgebras and Applications to Computer Science.* PhD thesis, Universität München, April 2000.
12. A. Kurz and D. Pattinson. Coalgebras and Modal Logics for Parameterised Endofunctors. Technical report, CWI, 2000.
13. S. MacLane. *Categories for the Working Mathematician.* Springer, 1971.
14. R. Milner. *Communication and Concurrency.* International series in computer science. Prentice Hall, 1989.
15. L. Moss. Coalgebraic Logic. *Annals of Pure and Applied Logic*, 96:277–317, 1999.
16. D. Park. Concurrency and Automata on Infinite Sequences. In P. Deussen, editor, *5th GI Conference*, volume 104 of *Lect. Notes in Comp. Sci.* Springer, 1981.
17. D. Pattinson. *Expressivity Results in the Modal Logic of Coalgebras.* PhD thesis, Universität München, June 2001.
18. M. Rößiger. Coalgebras and Modal Logic. In H. Reichel, editor, *Coalgebraic Methods in Computer Science (CMCS'2000)*, volume 33 of *Electr. Notes in Theoret. Comp. Sci.*, 2000.
19. M. Rößiger. From Modal Logic to Terminal Coalgebras. *Theor. Comp. Sci.*, 260:209–228, 2001.
20. J. Rutten. Universal Coalgebra: A theory of systems. *Theor. Comp. Sci.*, 249(1):3 – 80, 2000.
21. Andrzej Tarlecki. Institutions: An Abstract Framework for Formal Specifications. In E. Astesiano, H.-J. Kreowski, and B. Krieg-Brückner, editors, *Algebraic Fondations of System Specification*, volume 2. Springer, 1999.