Towards a Duality Result in Coalgebraic Modal Logic

Bart Jacobs

Department of Computer Science,

University of Nijmegen

Email: bart@cs.kun.nl

URL: http://www.cs.kun.nl/~bart

Contents

I. Introduction:

- Ultrafilters
- Standard duality in modal logic
- Rößiger's modal logic and completeness
- Temporal logic for coalgebras: $\Box$ via predicate lifting

II. Bounded stacks

- Specification
- Operators, Observers, Axioms
- Coalgebra on ultrafilters
- Adjunction (& duality)

III. Conclusions, references

Ultrafilters

Recall for a Boolean Algebra (BA) B, a filter of B is a subset $U \subseteq B$ which is

- closed under finite meets: $\top \in U$, and $x, y \in U \Rightarrow x \wedge y \in U$;
- upwardly closed: if $y \geq x$ for $x \in U$, then $y \in U$.

An ultrafilter is a filter which additionally satisfies:

- it contains either x or $\neg x$, but not both, for an arbitrary $x \in B$.

As a consequence: $x \vee y \in U \Rightarrow x \in U$ or $y \in U$.

We write spec B for the set of ultrafilters of B.

Standard duality in modal logic 1

Translations

- Given a transition system or Kripke structure $X \to \mathcal{P}(X)$, there is an associated nexttime operator $\Box$ on $\mathcal{P}(X)$ given by:

  $$\Box(\alpha) = \{x \in X \mid \forall y \in X.\, x \to y \implies y \in \alpha\}$$

  It makes $(\mathcal{P}(X), \Box)$ a Boolean Algebra with Operators (BAO).

- Conversely, given a BAO $(B, \Box)$, define a transition system on the set spec B of ultrafilters by:

  $$U \to V \stackrel{\text{def}}{\iff} \Box^{-1}(U) \subseteq V$$

These operations are both functorial.

Comparisons

- Starting from a BAO $(B, \Box)$, the canonical map

$$\eta_B \colon B \longrightarrow \mathcal{P}(\text{spec}\, B) \qquad \text{given by} \qquad b \longmapsto \{U \mid b \in U\}$$

  is a "map of BAO's".

- Starting with a transition system $X \to \mathcal{P}(X)$, there is also a canonical map

$$\varepsilon_X \colon X \longrightarrow \text{spec}\, \mathcal{P}(X) \qquad \text{given by} \qquad x \longmapsto \{\alpha \in \mathcal{P}(X) \mid x \in \alpha\}$$

  But it is folklore that ε is in general not a "map of coalgebras" (aka "bounded" morphism).

One gets a duality (both η and ε are isomorphisms) by restricting to so-called descriptive Kripke structures.

Rößiger's modal logic and completeness

Let T be a polynomial functor (on **Sets**) built up from the identity, constants, products, coproduct, exponents (with a constant) *and* powerset.

Rößiger introduces a modal language for T, with pathwise modal operators, and several axioms. Formulas are interpreted as subsets of the state space of a T-coalgebra. He proves:

- Two states are bisimilar if and only if they satisfy the same formulas;
- Completeness of the modal logic, via a terminal coalgebra constructed on maximally consistent sets of formulas.

Algebraically, these maximally consistent sets correspond to ultrafilters.

For a polynomial functor T as before, and a coalgebra $c\colon X \longrightarrow T(X)$, there is a single nexttime operator $\Box$ on $\mathcal{P}(X)$ given by

$$\Box(\alpha) = \{x \in X \mid c(x) \in \mathrm{Pred}(T)(\alpha)\}$$

where $\mathrm{Pred}(T)(\alpha) \subseteq T(X)$ is the so-called predicate lifting of α, defined by induction on T. This makes $(\mathcal{P}(X), \Box)$ into a "Galois algebra".

This generalizes the earlier definition of $\Box$ for transition systems, where $T = \mathcal{P}$.

Intuitively, $\Box(\alpha)$ holds of a state x if all possible successor states of x (along all paths) satisfy α. It is useful for expressing safety properties, but too crude for specification.

Aim:

Combine these approaches: refine the temporal logic of coalgebras into pathwise modal operators on predicates, so as to be able to reconstruct coalgebras from their operators. This would give an algebraic version of Rößiger's constructions.

Presentation:

An exemplaric case study: bounded stacks.

Assume we wish to specify a bounded stack, which may contain at most N elements, from a type D. This means that not only pop can fail (when the stack is empty), but also push (when the stack is full). Therefore we use the following interface.

BEGIN BoundedStack$[D : \mathsf{TYPE}, N : \mathbb{N}]$

OPERATIONS

size: $\mathsf{Self} \longrightarrow \{0, 1, \ldots, N\}$

push: $\mathsf{Self} \times D \longrightarrow \mathsf{Self} + \mathsf{Self}$

pop: $\mathsf{Self} \longrightarrow \mathsf{Self} + (D \times \mathsf{Self})$

ASSERTIONS

• • •

END BoundedStack

The polynomial functor that is extracted from this specification is:

$$T(X) = \{0, 1, \ldots, N\} \times (X + X)^D \times (X + (D \times X))$$

A coalgebra $X \longrightarrow T(X)$ for this functor can be identified with a 3-tuple of functions

$$\begin{aligned} \mathsf{size} &\colon X \longrightarrow \{0, 1, \ldots, N\} \\ \mathsf{push} &\colon X \times D \longrightarrow X + X \\ \mathsf{pop} &\colon X \longrightarrow X + (D \times X) \end{aligned}$$

as in the previous specification.

Operators, Observers, Axioms 1

Assume a bounded stack coalgebra $\langle \mathsf{size}, \mathsf{push}, \mathsf{pop} \rangle \colon X \longrightarrow T(X)$.

The different paths to X in $T(X)$ lead to modal operators on $\mathcal{P}(X)$:

$$\begin{array}{rcl}
\mathsf{PushF}(d)(\alpha) & = & \{x \in X \mid \forall y \in X.\, \mathsf{push}(x,d) = \kappa_1(y) \Rightarrow \alpha(y)\} \\
\mathsf{PushS}(d)(\alpha) & = & \{x \in X \mid \forall y \in X.\, \mathsf{push}(x,d) = \kappa_2(y) \Rightarrow \alpha(y)\} \\
\mathsf{PopF}(\alpha) & = & \{x \in X \mid \forall y \in X.\, \mathsf{pop}(x) = \kappa_1(y) \Rightarrow \alpha(y)\} \\
\mathsf{PopS}(\alpha) & = & \{x \in X \mid \forall y \in X.\, \forall d \in D.\, \mathsf{pop}(x) = \kappa_2(d,y) \Rightarrow \alpha(y)\}
\end{array}$$

And the constants in $T(X)$ to the observers

$$\begin{array}{rcl}
\mathsf{Size}(n) & = & \{x \in X \mid \mathsf{size}(x) = n\} \\
\mathsf{Top}(d) & = & \{x \in X \mid \forall y \in X.\, \forall e \in D.\, \mathsf{top}(x) = \kappa_2(e,y) \Rightarrow e = d\}.
\end{array}$$

There is a systematic way to define these operators and observers, using induction on the structure of the functor.

Also, there is a systematic way to obtain the following axioms for them:

$$\mathsf{PushF}(d)(\bot) \mathbin{\dot\vee} \mathsf{PushS}(d)(\bot) \qquad\qquad \dot{\bigvee_{n\in\{0,1,\ldots,N\}}} \mathsf{Size}(n)$$

$$\mathsf{PopF}(\bot) \mathbin{\dot\vee} \mathsf{PopS}(\bot) \qquad\qquad \neg\mathsf{PopS}(\bot) \supset \dot{\bigvee_{d\in D}} \mathsf{Top}(d)$$

(where $\bot$ is the empty subset, and $\dot\vee$ is exclusive or).

In this way we obtain that $\mathcal{P}(X)$ is a BAO of type T.

(where T is still the Bounded Stack interface functor).

Operators, Observers, Axioms 3

In general, a BAO of type T consists of a BA B with operators

$$\mathsf{PushF}(d), \mathsf{PushS}(d), \mathsf{PopF}, \mathsf{PopS} \colon B \longrightarrow B$$

which preserve finite meets $\top, \wedge$ and non-empty joins $\vee$, together with observers

$$\mathsf{Size}(n), \mathsf{Top}(d) \in B$$

satisfying the axioms above.

Excursion

The modal operators and observers can be used for formulating the Bounded Stack assertions. For example:

$$\forall d \in D.\, \mathsf{Size}(N) \Longrightarrow \mathsf{PushS}(d)(\bot)$$
$$\mathsf{Size}(0) \Longrightarrow \mathsf{PopS}(\bot)$$
$$\forall d \in D.\, \forall \alpha \in \mathsf{Prop}.\, \mathsf{Size}(N) \Longrightarrow (\mathsf{PushF}(d)(\alpha) \Longleftrightarrow \alpha)$$
$$\forall d \in D.\, \forall n \in \{0, 1, \ldots, N-1\}.\, \mathsf{Size}(n) \Longrightarrow \mathsf{PushS}(d)(\mathsf{Size}(n+1))$$
$$\forall d \in D.\, \mathsf{PushS}(d)(\mathsf{Top}(d))$$
$$\forall n \in \{0, 1, \ldots, N-2\}.\, \mathsf{Size}(n+1) \Longrightarrow \mathsf{PopS}(\mathsf{Size}(n))$$

The equivalence $\Longleftrightarrow$ is used to describe bisimilarity.

We have seen how a coalgebra gives rise to a BAO. One can also go in the opposite direction.

Assume therefore a BAO B of type T. A coalgebra $\mathsf{spec}\,B \longrightarrow T(\mathsf{spec}\,B)$ can be defined:

$$\begin{aligned}\mathsf{size}(U) &= \text{the unique } m \text{ with } \mathsf{Size}(m) \in U \\ &\qquad\qquad \Big(\text{using } \dot{\bigvee_{n\in\{0,1,\ldots,N\}}} \mathsf{Size}(n) = \top \in U\Big)\end{aligned}$$

$$\mathsf{push}(U,d) = \begin{cases} \kappa_1(\mathsf{PushF}(d)^{-1}(U)) & \text{if } \neg\mathsf{PushF}(d)(\bot) \in U \\ \kappa_2(\mathsf{PushS}(d)^{-1}(U)) & \text{if } \neg\mathsf{PushS}(d)(\bot) \in U \end{cases}$$

$$\mathsf{pop}(U) = \begin{cases} \kappa_1(\mathsf{PopF}^{-1}(U)) & \text{if } \neg\mathsf{PopF}(\bot) \in U \\ \kappa_2(d, \mathsf{PopS}^{-1}(U)) & \text{if } \neg\mathsf{PopS}(\bot) \in U \text{ and } \mathsf{Top}(d) \in U. \end{cases}$$

Main Result

In this case the canonical maps $\eta_B \colon B \longrightarrow \mathcal{P}(\mathsf{spec}\, B)$ and $\varepsilon_X \colon X \longrightarrow \mathsf{spec}\, \mathcal{P}(X)$ are appropriate homomorphisms, and we get an adjunction of the form:

$$
\begin{tikzcd}
\mathbf{BAO}(T) \arrow[d, bend right, "\mathsf{spec}"'] \arrow[d, phantom, "\dashv"] \\
\mathbf{CoAlg}(T)^{\mathrm{op}} \arrow[u, bend right, "\mathcal{P}"']
\end{tikzcd}
$$

This is also the main result of the paper, but then for a general polynomial functor (without powerset). It requires a detailed analysis of paths for a functor, in order to describe the modal operators, observers and their properties.

(Each adjunction gives rise to a duality, by restriction to the subcategories of objects for which the (co)unit is an isomorphism.)

- For a polynomial functor T (without powerset $\mathcal{P}$) we now have modal operators, observers and axioms forming a modal language (much like Rößiger's) which is useful for specification. There is an adjunction between
  - "dynamical" models, namely coalgebras;
  - "algebraic" models, namely BAOs.
- Such a relation can be exploited to transfer results and techniques. Typically, to obtain Birkhoff's variety result in modal logic. This is done in recent work of Goldblatt, but only for functors which even don't have coproducts $+$.
- A further integretation is needed with the classical work on duality for Kripke structures, via proper handling of the powerset functor.

- Larry Moss, Coalgebraic Logic, *Annals of Pure & Applied Logic*, to appear.
- Martin Rößiger, From Modal Logic to Terminal Coalgebras, *Theoretical Computer Science*, to appear.
- Martin Rößiger, Coalgebras and Modal Logic, *This volume*.
- Bart Jacobs, The Temporal Logic of Coalgebras via Galois Algebras, Techn. Rep. CSI-R9906, Univ. Nijmegen.
- Robert Goldblatt, Duality for some Categories of Coalgebras, manuscript, 2000.