Expressivity of many-valued coalgebraic logics

Marta Bílková
(joint work with Matěj Dostál)

ICS ASCR in Prague

11 December 2015

Expressivity and its limits (what do we know)

Classical modal logics:

- The (finitary) $\Box, \Diamond$ modal logic over image finite Kripke frames is expressive for bisimilarity. (Hennessy-Milner property)
- From Kripke frames to arbitrary coalgebras (in Set): we have a generic definition of behavioral equivalence, (sometimes captured by bisimilarity),
  and there is a way how to create expressive languages for behavioral equivalence for a large class of coalgebra functors.
- Limits of expressivity: depends on the coalgebra functor, the kind of modalities we allow, and their arities.

Many-valued modal logics:

- The $\Box, \Diamond$ modal logic over image finite crisp Kripke frames with $\mathscr{V}$-valued valuations is not always expressive for bisimilarity (especially if we want to avoid constants for elements of $\mathscr{V}$).
- There is a full (algebraic) characterization in terms of those MTL chains $\mathscr{V}$ for which it is expressive. Also the propositional logic matters.
- Beyond this (crisp, MTL chains, $\Box, \Diamond$ based) not much has been known.

What about coalgebraic generalisations of many-valued modal logics?

Many valued Kripke semantics

We fix a residuated lattice $\mathscr{V}$ of truth values. A $\mathscr{V}$-valued Kripke model is a tuple $(W, R, \Vdash)$, where

$$R : W \times W \longrightarrow \mathscr{V}$$

is a $\mathscr{V}$-valued accessibility relation, and

$$\Vdash : \mathscr{L} \times W \longrightarrow \mathscr{V}$$

is a $\mathscr{V}$-valued valuation.

We call a model *crisp* whenever $R : W \times W \longrightarrow 2$.

Many valued Kripke semantics

Crucial is already the choice of the *propositional language*: it is usually the many-valued language of the variety we are interested in, and we take $\mathscr{V}$ from. But do we include *constants* for values in $\mathscr{V}$?

The modal part is interpreted as follows:

- $x \Vdash \Box a \;=\; \bigwedge_y (xRy \longrightarrow y \Vdash a)$
- $x \Vdash \Diamond a \;=\; \bigvee_y (xRy \;\&\; y \Vdash a)$

where $\bigwedge$, $\longrightarrow$, $\&$ and $\bigvee$ are computed in $\mathscr{V}$. For this, we need $\mathscr{V}$ to be a *complete* residuated lattice.

Coalgebras in Set

For an endofunctor

$$T : \mathsf{Set} \longrightarrow \mathsf{Set}$$

A coalgebra for T with a set of states X is a map

$$c : X \longrightarrow TX$$

T describes *one step* behavior. For this talk, T will mostly be a finitary and weak pullback preserving functor.

Example: If T is the powerset functor P, we obtain Kripke frames. If T is the finitary powerset functor P_ω, we obtain image finite Kripke frames.

Coalgebras in Set - examples

- coalgebras for the functor $\mathrm{Id} \times \mathrm{Id}$ are binary trees
- coalgebras for the functor $A \times \mathrm{Id}$ are streams over alphabet A
- coalgebras for the functor Id^A are labelled transition systems
- coalgebras for the functor $2 \times \mathrm{Id}^A$ are deterministic automata with input alphabet A
- coalgebras for the functor $2 \times (P\mathrm{Id})^A$ are non-deterministic automata
- coalgebras for the multiset functor are directed weighted graphs
- coalgebras for the $\mathscr{V}$-valued powerset functor $[\mathrm{Id}, \mathscr{V}]$ are many-valued Kripke frames
- ...

Morphisms of coalgebras

A map $f : X \longrightarrow Y$ is a morphism of coalgebras iff

$$\begin{tikzcd} X \arrow[r, "f"] \arrow[d, "c"'] & Y \arrow[d, "d"] \\ TX \arrow[r, "Tf"'] & TY \end{tikzcd}$$

Example: If T is the powerset functor P, we obtain precisely bounded morphisms.

Behavioral equivalence

Coalgebras $c : X \longrightarrow TX$ and $d : Y \longrightarrow TY$ are behavioraly equivalent iff there exists a coalgebra $z : Z \longrightarrow TZ$ such that

$$\begin{tikzcd}
X \arrow[r, "f"] \arrow[d, "c"'] & Z \arrow[d, "z"] & Y \arrow[l, "g"'] \arrow[d, "d"] \\
TX \arrow[r, "Tf"'] & TZ & TY \arrow[l, "Tg"]
\end{tikzcd}$$

Remark: if the final T-coalgebra exists, z might be taken to be the final coalgebra.

Bisimulations

A relation $B \subseteq X \times Y$ is a bisimulation between $c : X \longrightarrow TX$ and $d : Y \longrightarrow TY$ iff there is a coalgebra structure z on B which makes the projections into coalgebra morphisms:

$$
\begin{tikzcd}
X \arrow[d, "c"'] & B \arrow[l, "p_0"'] \arrow[r, "p_1"] \arrow[d, "z"', red] & Y \arrow[d, "d"] \\
TX & TB \arrow[l, "Tp_0"] \arrow[r, "Tp_1"'] & TY
\end{tikzcd}
$$

Equivalently (if T preserves weak pullbacks), using *relation lifting*, B is a bisimulation if

$$B(x, y) \text{ implies } \overline{T}(B)(c(x), d(y)).$$

Example: If T is the powerset functor P, we obtain precisely the standard there-and-back definition of a bisimulation.

Bisimulations

A relation $B \subseteq X \times Y$ is a bisimulation between $c : X \longrightarrow TX$ and $d : Y \longrightarrow TY$ iff there is a coalgebra structure z on B which makes the projections into coalgebra morphisms:

$$\begin{tikzcd} X \arrow[d, "c"'] & B \arrow[l, "p_0"'] \arrow[r, "p_1"] \arrow[d, "z"] & Y \arrow[d, "d"] \\ TX & TB \arrow[l, "Tp_0"] \arrow[r, "Tp_1"'] & TY \end{tikzcd}$$

Equivalently (if T preserves weak pullbacks), using *relation lifting*, B is a bisimulation if

$$B(x, y) \text{ implies } \overline{T}(B)(c(x), d(y)).$$

Remark: If T preserves weak pullbacks, *behavioral equivalence* and *bisimilarity* coincide.

Bisimulations

Example: A *bisimulation* between many-valued crisp Kripke models (P coalgebras with many-valued valuation) is a relation $B \subseteq X \times Y$ satisfying: xBy implies

1. $x \Vdash_c p = y \Vdash_d p$ for all atoms
2. $\forall x' \in c(x)\ \exists y'(y' \in d(y)$ and $x'By')$
3. $\forall y' \in d(y)\ \exists x'(x' \in c(x)$ and $x'By')$

Bisimilarity implies modal equivalence (for $\Box, \Diamond$). The converse is not true in general, not even for image-finite models. The propositional part matters.

Bisimulations

Example: A *bisimulation* between many-valued Kripke models ($P^{\mathscr{V}}$ coalgebras with many-valued valuation) is a relation $B \subseteq X \times Y$ satisfying: xBy implies

1. $x \Vdash_c p = y \Vdash_d p$ for all atoms
2. $c(x)(x') \leq \bigvee_{y' : x' B y'} d(y)(y')$
3. $d(y)(y') \leq \bigvee_{x' : x' B y'} c(x)(x')$

Bisimilarity implies modal equivalence (for $\Box, \Diamond$). The converse is not true in general, not even for image-finite models. The propositional part matters.

Expressivity for crisp many-valued frames

Theorem

For crisp frames, and $\mathscr{V}$ a complete MTL *chain, the following are equivalent:*

1. *the $\Box, \Diamond$ logic of image-finite frames satisfies the Hennessy-Milner theorem*
2. *$\mathscr{V}$ has distinguishing formula property*

For $\mathscr{V}$ being a complete BL chain with finite universe or $[0, 1]$, this yields expressivity iff $\mathscr{V}$ is a MV chain or the ordinal sum of two (hoop reducts of) MV chains.

G. Metcalfe and M. Martí. *A Hennessy-Milner property for many-valued modal logics.* In Advances in Modal Logic, volume 10, pages 407–420. 2014.

Two ways of ...

... designing an expressive language for T coalgebras over a propositional logic (e.g. classical, positive or many-valued), based on the idea that modal part should describe all (enough) behaviour patterns given by T:

- adding a single modality of arity T (idea of Larry Moss)
- adding all possible (or sufficiently many) n-ary modalities for all $n \in N$ (logic based on a Stone type duality, or eq. on predicate liftings)

Logic of all modalities based on a logical connection

- M.M. Bonsangue and A. Kurz. *Duality for logics of transition systems*. In Vladimiro Sassone, editor, Foundations of Software Science and Computational Structures, volume 3441 of Lecture Notes in Computer Science, pages 455–469. 2005.
- A. Kurz and R. Leal. *Modalities in the Stone age: A comparison of coalgebraic logics.* Theoretical Computer Science, 430:88–116, 2012.

Logic of all modalities based on a logical connection

We start with a Stone-type dual adjunction of the following form:

$$\begin{tikzcd} \mathsf{Set}^{op} \arrow[loop left, "T^{op}"] \arrow[rr, bend right, "Pred"'] & \arrow[phantom, "\perp"] & \mathsf{Alg} \arrow[ll, bend right, "Stone"'] \end{tikzcd}$$

Where Alg is a variety (e.g. BA, DL)

$$Pred = \mathsf{Set}(-, 2) \text{ and } Stone = \mathsf{Alg}(-, 2),$$

or Alg is a variety of residuated lattices and

$$Pred = \mathsf{Set}(-, \mathscr{V}) \text{ and } Stone = \mathsf{Alg}(-, \mathscr{V}),$$

Logic of all modalities based on a logical connection

We start with a Stone-type dual adjunction of the following form:

$$\begin{tikzcd} \mathrm{Set}^{op} \arrow[loop left, "T^{op}"] \arrow[rr, bend right, "Pred"'] \arrow[rr, phantom, "\perp"] & & \mathrm{Alg} \arrow[ll, bend right, "Stone"'] \arrow[loop right, "L"] \end{tikzcd}$$

The functor L describes how to add *one layer* of *modalities*

$$L = Pred.T^{op}.Stone$$

Description of L

It is enough to define L on finitely generated algebras from Alg (and extend to arbitrary A via colimits)

$$
\begin{tikzcd}
\mathsf{Set}^{op} \arrow[loop left, "T^{op}"] \arrow[rr, bend right, "Pred"'] \arrow[rr, phantom, "\perp"] & & \mathsf{Alg} \arrow[ll, bend right, "Stone"'] \arrow[loop right, "L"] \arrow[d, bend left, "U"] \arrow[d, phantom, "\dashv"] \\
& & \mathsf{Set} \arrow[u, bend left, "F"]
\end{tikzcd}
$$

$$LFn = Pred.T^{op}.Stone.Fn \simeq [T(\mathscr{V}^n), \mathscr{V}]$$

Semantics of modalities

An n-ary modality $\heartsuit$ is semantically a map $\heartsuit : T(\mathscr{V}^n) \longrightarrow \mathscr{V}$
On a coalgebra $c : X \longrightarrow TX$ with valuation $\|.\| : \mathscr{L} \longrightarrow Pred X$ it is interpreted as follows:

$$X \xrightarrow{c} TX \xrightarrow{T\|\bar{a}\|} T(\mathscr{V}^n) \xrightarrow{\heartsuit} \mathscr{V}$$

Example: boolean semantics of $\Box : P2 \longrightarrow 2$ is the map assigning 1 to $\emptyset$ and $\{1\}$, i.e. the meet.
On a coalgebra $c : X \longrightarrow PX$ it is interpreted as follows:

$$X \xrightarrow{c} PX \xrightarrow{P\|a\|} P(2) \xrightarrow{\Box} 2$$

for any x the result is 1 iff $c(x) = \emptyset$ or $\forall y \in c(x)\ y \Vdash a$.

Semantics of modalities

An n-ary modality $\heartsuit$ is semantically a map $\heartsuit : T(\mathscr{V}^n) \longrightarrow \mathscr{V}$
On a coalgebra $c : X \longrightarrow TX$ with valuation $\|.\| : \mathscr{L} \longrightarrow PredX$ it is interpreted as follows:

$$X \xrightarrow{c} TX \xrightarrow{T\|\bar{a}\|} T(\mathscr{V}^n) \xrightarrow{\heartsuit} \mathscr{V}$$

Example: Many-valued semantics of $\Box : P\mathscr{V} \longrightarrow \mathscr{V}$ is the meet.
On a coalgebra $c : X \longrightarrow PX$ it is interpreted as follows:

$$X \xrightarrow{c} PX \xrightarrow{P\|a\|} P(\mathscr{V}) \xrightarrow{\Box} \mathscr{V}$$

for any x the result is $c(x) \Vdash \Box a = \bigwedge_{y \in c(x)} y \Vdash a$.

Semantics of modalities

An n-ary modality $\heartsuit$ is semantically a map $\heartsuit : T(\mathscr{V}^n) \longrightarrow \mathscr{V}$
On a coalgebra $c : X \longrightarrow TX$ with valuation $\|.\| : \mathscr{L} \longrightarrow Pred X$ it is interpreted as follows:

$$X \xrightarrow{c} TX \xrightarrow{T\|\bar{a}\|} T(\mathscr{V}^n) \xrightarrow{\heartsuit} \mathscr{V}$$

Example: Many-valued semantics of $\Box : [\mathscr{V}, \mathscr{V}] \longrightarrow \mathscr{V}$ is:

$$\bigwedge_{v \in \mathscr{V}} (\sigma(v) \longrightarrow v)$$

On a coalgebra $c : X \longrightarrow PX$ it is interpreted as follows:

$$X \xrightarrow{c} [X, \mathscr{V}] \xrightarrow{[\|a\|, \mathscr{V}]} [\mathscr{V}, \mathscr{V}] \xrightarrow{\Box} \mathscr{V}$$

for any x the result is $c(x) \Vdash \Box a = \bigwedge_y (c(x)(y) \longrightarrow y \Vdash a)$.

Predicate liftings

- D. Pattinson. Expressive logics for coalgebras via terminal sequence induction. Notre Dame J. Formal Logic, (45):19–33, 2004.
- L. Schröder. *Expressivity of coalgebraic modal logic: The limits and beyond.* Theoretical Computer Science, 390:230–247, 2008.

Modalities = Predicate liftings

Modalities (n-ary):

$$\heartsuit : T\mathscr{V}^n \longrightarrow \mathscr{V}$$

Predicate liftings (n-ary, natural in X):

$$\hat{\heartsuit}_X : [X, \mathscr{V}^n] \longrightarrow [TX, \mathscr{V}]$$

Modalities and PL are in one-to-one correspondence:

$$\hat{\heartsuit}_X(Q) = \heartsuit(TQ) : TX \longrightarrow \mathscr{V}$$

and

$$\heartsuit = \hat{\heartsuit}_{\mathscr{V}^n}(\mathrm{id}_{\mathscr{V}^n})$$

Separating sets of PL

Predicate liftings (n-ary, natural in X):

$$\hat{\heartsuit}_X : [X, \mathscr{V}^n] \longrightarrow [TX, \mathscr{V}]$$

have their transpose:

$$\hat{\heartsuit}^\flat_X : TX \longrightarrow [[X, \mathscr{V}^n], \mathscr{V}]$$

Definition

A set Λ of modalities is called *separating* iff

$$(\hat{\heartsuit}^\flat_X : TX \longrightarrow [[X, \mathscr{V}^n], \mathscr{V}])_{\heartsuit \in \Lambda}$$

is *jointly injective* for all X.

Existence of separating sets

Theorem

A (finitary) functor T admits a separating set of predicate liftings iff the source

$$(Tf : TX \longrightarrow T(\mathscr{V}^n))_{f:X \longrightarrow \mathscr{V}^n}$$

is jointly injective for each X.

Example: $\mathscr{V} = 2$, $T = P_\omega P_\omega$ does not admit a separating set of unary predicate liftings: given a finite set X and any f,

$$(Tf)\{A \subseteq X \mid |A| \leq 2\} = (Tf)P_\omega X.$$

Existence of separating sets

Theorem

A (finitary) functor T admits a separating set of predicate liftings iff the source

$$(Tf : TX \longrightarrow T(\mathscr{V}^n))_{f:X\longrightarrow\mathscr{V}^n}$$

is jointly injective for each X.

Corollary

Every finitary functor admits a separating set of predicate liftings.

Namely, the set of all liftings is separating.

Expressivity for $\mathscr{V} = 2$ via separation

Theorem

Let T be finitary, $\mathscr{V} = 2$, and Λ a separating set of PL. Then $\mathscr{L}(\Lambda)$ is expressive for behavioral equivalence.

For T preserving weak pullbacks this gives expressivity also for bisimilarity.

Question: Where exactly in the proof $\mathscr{V} = 2$ is important?

Separating sets of PL

Theorem

Assume T admits a separating set of PL, i.e. Λ_T is separating. Then for all $\Lambda \subseteq \Lambda_T$ TFAE:

- *Λ is separating,*
- *$\{\heartsuit(T\sigma) \mid \sigma : \mathscr{V}^n \longrightarrow \mathscr{V}^n, \heartsuit \in \Lambda\}$ is separating,*
- *$t \neq t'$ in $T\mathscr{V}^n$ implies $\exists \sigma : \mathscr{V}^n \longrightarrow \mathscr{V}^n, \heartsuit \in \Lambda$ such that*

$$\heartsuit(T\sigma)(t) \neq \heartsuit(T\sigma)(t').$$

Notice: for $\mathscr{V} = 2$ and $n = 1$, each $\sigma : \mathscr{V} \longrightarrow \mathscr{V}$ is definable in the boolean language (CPC is functionally complete).

Condition on $\mathscr{V}$

A set Λ of modalities is called *separating* iff

$$(\hat{\heartsuit}^{\flat}_{X} : TX \longrightarrow [[X, \mathscr{V}^{n}], \mathscr{V}])_{\heartsuit \in \Lambda}$$

is *jointly injective* for all X.

Meaning: for each $t \neq t'$ in TX, there is $\sigma : X \longrightarrow \mathscr{V}^{n}$ and $\heartsuit \in \Lambda$, such that

$$\heartsuit(T\sigma)(t) \neq \heartsuit(T\sigma)(t').$$

We need σ to be expressible as a tuple $(\|a_1\|_X, \ldots, \|a_n\|_X)$ of predicates given by formulas in the language of $\mathscr{V}$.

Condition on $\mathscr{V}$

We call $\sigma : \mathscr{V}^n \longrightarrow \mathscr{V}^n$ expressible, if there are n terms $\sigma_1, \ldots, \sigma_n$ in n variables in the language of $\mathscr{V}$, such that:

$$\sigma(v_1, \ldots, v_n) = (\sigma_1(v_1, \ldots, v_n), \ldots, \sigma_n(v_1, \ldots, v_n)).$$

The condition: $t \neq t'$ in $T\mathscr{V}^n$ implies there exists $\sigma : \mathscr{V}^n \longrightarrow \mathscr{V}^n$ expressible, and $\heartsuit \in \Lambda$ such that

$$\heartsuit(T\sigma)(t) \neq \heartsuit(T\sigma)(t').$$

Example: for $n = 1$, $T = P$, $\Lambda = \{\Box, \Diamond\}$ and $\mathscr{V}$ a complete MTL chain, the condition is equivalent to the one given by Metcalfe and Martí.

Expressivity

Theorem

Let T be finitary, Λ a separating set of PL, and $\mathscr{V}$ satisfies the condition (w.r.t. T and Λ). Then $\mathscr{L}(\Lambda)$ is expressive for behavioral equivalence.

For T preserving weak pullbacks this gives expressivity also for bisimilarity (which is easier to prove independently on the above).

The condition for $\mathscr{V}$ is sufficient and necessary.

Proof sketch for bisimilarity

We prove that the modal equivalence $\equiv_{\mathscr{L}(\Lambda)}$ between coalgebras c and d is a bisimulation, in particular that $x \equiv y$ implies $c(x)\ \overline{T}\!\equiv d(y)$.

- Assume $\neg(c(x)\ \overline{T}\!\equiv d(y))$.
- Consider (bases of) $c(x)$ and $d(y)$ to be $x_1 \dots x_k$ and $y_1 \dots y_l$ resp.,
- for each $x_i \not\equiv y_j$ fix a distinguishing formula $a_{i,j}$. Thus we have up to kl formulas.
- Put $n = kl$ and consider the maps:

$$f = \overrightarrow{\|a\|_c} : \mathrm{b}(c(x)) \longrightarrow \mathscr{V}^n \ \text{ and } \ g = \overrightarrow{\|a\|_d} : \mathrm{b}(d(y)) \longrightarrow \mathscr{V}^n$$

Proof sketch for bisimilarity

- for each $x_i \not\equiv y_j$ fix a distinguishing formula $a_{i,j}$. Thus we have up to kl formulas.
- Put $n = kl$ and consider the maps:

$$f = \overrightarrow{\|a\|_c} : \mathrm{b}(c(x)) \longrightarrow \mathscr{V}^n \text{ and } g = \overrightarrow{\|a\|_d} : \mathrm{b}(d(y)) \longrightarrow \mathscr{V}^n$$

- Then (restriction of) $\equiv$ is the pullback of the two maps. Therefore $T\equiv$ is the pullback of Tf and Tg.
- Because $\neg(c(x)\ \overline{T}{\equiv}\ d(y))$, we know that $(Tf)c(x) \neq (Tg)(d(y))$ in $T\mathscr{V}^n$.

Proof sketch for bisimilarity

- Put $n = kl$ and consider the maps:

$$f = \overrightarrow{\|a\|_c} : \mathrm{b}(c(x)) \longrightarrow \mathscr{V}^n \quad \text{and} \quad g = \overrightarrow{\|a\|_d} : \mathrm{b}(d(y)) \longrightarrow \mathscr{V}^n$$

- Then (restriction of) $\equiv$ is the pullback of the two maps. Therefore $T\equiv$ is the pullback of Tf and Tg.
- Because $\neg(c(x)\ \overline{T}\equiv d(y))$, we know that $(Tf)c(x) \neq (Tg)(d(y))$ in $T\mathscr{V}^n$.
- By separation there is $\heartsuit \in \Lambda$ and σ *expressible* such that

$$\heartsuit(T\sigma)(Tf)(c(x)) \neq \heartsuit(T\sigma)(Tg)(d(y))$$

- The formula $\heartsuit(\sigma_1(\vec{a}), \ldots, \sigma_n(\vec{a}))$ distinguishes x and y.

Idea of proof for behavioral equivalence

$$\begin{tikzcd} \mathrm{Set}^{op} \arrow[loop left, "T^{op}"] \arrow[rr, bend right, "Pred"'] \arrow[rr, phantom, "\perp"] & & \mathrm{Alg} \arrow[ll, bend right, "Stone"'] \arrow[loop right, "L"] \end{tikzcd}$$

$$L = Pred.T^{op}.Stone \quad \text{and} \quad \mathscr{L} \quad \text{the initial} \quad L-\text{algebra}$$

Expressivity is expressed by the injectivity of

$$TSA \xrightarrow{\eta_{TSA}} SPTSA \xrightarrow{S\delta_{SA}} SLPSA \xrightarrow{SL\varepsilon_A} SLA$$

which follows from the assumptions above (separation), and which, for $A = \mathscr{L}$, yields the injectivity of

$$T(\mathscr{L}, \mathscr{V}) \longrightarrow (\mathscr{L}, \mathscr{V}),$$

therefore we can define a coalgebra on $(\mathscr{L}, \mathscr{V})$, witnessing the behavioral equivalence of states having same theories.

Moss' many-valued logic

- Uses a single ∇ modality, of arity T.
- Because the semantics uses lifting of $\Vdash$, we need lifting for many-valued relations. This works well if $\mathscr{V}$ is a complete Heyting algebra.
- We can prove expressivity if moreover $\mathscr{V}$ is a chain (or if $\top$ is join-prime), under a similar separation condition on $\mathscr{V}$.

M. Bílková and M. Dostál. *Many-valued relation lifting and Moss' coalgebraic logic.* In CALCO 2013, pages 66–79. 2013.

Further work

- Case study!
- Other notions of bisimulation? (We can vary the atomic harmony condition.)
- Expressivity w.r.t. behavioral equivalence via characteristic formulas.
- Push the results on Moss' many-valued logic further.