An institutional approach to positive coalgebraic logic

Adriana Balan\textsuperscript{1} Alexander Kurz\textsuperscript{2}
Jiří Velebil\textsuperscript{3}

\textsuperscript{1}University Politehnica of Bucharest, Romania
\textsuperscript{2}University of Leicester, UK
\textsuperscript{3}Czech Technical University in Prague, Czech Republic

22nd International Workshop on Algebraic Development Techniques
September 2014 – Sinaia, Romania
Background

Modal logic is about Kripke frames

$$\langle X, X \xrightarrow{r} X \rangle$$

These are coalgebras for the powerset functor

$$X \to \mathcal{P}X, \ x \mapsto \{x' \in X \mid r(x', x)\}$$

More generally, replace $P$ by any functor $T : \text{Set} \to \text{Set}$

$T$-coalgebras capture LTS, (non)deterministic automata, Mealy machines, probabilistic/stochastic transition systems, . . .

Reasoning about $T$-coalgebras: coalgebraic (modal) logic $(L, \delta)$

Logic $L : \text{BA} \to \text{BA}$ functor

Semantics $\delta : LP \to PT^{\text{op}}$ natural transformation
Outline

Institution $\textbf{Ins}$ of Set-based coalgebraic logic
[Kurz-Hennicker02, Pattinson03, Cîrstea06, ...]

$\textbf{Ins}$ restricts to an institution $\textbf{Ins}_{wpb}$ having
– signatures: Set-functors which preserve weak pullbacks
– morphisms between signatures: weakly cartesian natural transformations
Positive coalgebraic logic

**Logic**  Axiomatization of the positive fragment of modal logic [Dunn95]

Dunn’s result naturally **generalize** from modal logic to coalgebraic logic [B-Kurz-Velebil13]

**Coalgebra**  Looking at simulations instead of bisimulations? Posets provide the environment for that

**Category Theory**  Posets link universal coalgebra and domain theory

Technical issue: to ensure the monotonicity of modal operators, need to **work in an ordered setting** (Poset-enriched category theory)
ABC of Poset-enriched category theory

**Poset-category:** hom-sets are ordered and composition preserves this order

**Poset-functor (locally monotone):** functor preserving the order on the hom-(po)sets

**Poset-natural transformation:** natural transformation
What a Poset-enriched institution might be?

\[ \textbf{Ins} = (\text{Sign}, \text{Mod}, \text{Sen}, \models) \]

- **Sign** Poset-category

- **Mod**: \( \text{Sign}^{op} \to \text{Poset-Cat} \) locally monotone functor

- **Sen**: \( \text{Sign} \to \text{Poset} \) locally monotone functor

- For each signature \( T \), a relation \( |\text{Mod}(T)| \models \overset{\models}{\to} \text{Sen}(T) \) such that

\[
M \models \text{Sen}(\sigma)(\varphi) \iff \text{Mod}(\sigma)(M) \models \varphi
\]

(for each \( \sigma : T \to \hat{T}, M \in \text{Mod}(\hat{T}), \varphi \in \text{Sen}(T) \))

A. Balan, A. Kurz, J. Velebil
An institutional approach to positive coalgebraic logic
WADT2014
What a Poset-enriched institution might be?

\textbf{Ins} = (\text{Sign, Mod, Sen, } \models)

- **Sign** category

- **Mod**: \text{Sign}^{\text{op}} \to \text{Poset-Cat} \to \text{Cat} \quad \text{functor}

- **Sen**: \text{Sign} \to \text{Poset} \to \text{Set} \quad \text{functor}

- For each signature \( T \), a relation \( \models \text{Mod}(T) \models \to \text{Sen}(T) \) such that

\[ M \models \text{Sen}(\sigma)(\varphi) \iff \text{Mod}(\sigma)(M) \models \varphi \]

(for each \( \sigma : T \to \hat{T}, M \in \text{Mod}(\hat{T}), \varphi \in \text{Sen}(T) \))
Outline

Institution $\text{Ins}$ of Set-based coalgebraic logic
[Kurz-Hennicker02, Pattinson03, Cîrstea06, ...]

$\text{Ins}$ restricts to an institution $\text{Ins}_{wpb}$ having
– signatures: Set-functors which preserve weak pullbacks
– morphisms between signatures: weakly cartesian natural transformations
Outline

Institution \textbf{Ins} of Set-based coalgebraic logic
[Kurz-Hennicker02, Pattinson03, Cîrstea06, \ldots ]

\textbf{Ins} restricts to an institution \textbf{Ins}_{wpb} having
– signatures: Set-functors which \textbf{preserve weak pullbacks}
– morphisms between signatures: \textbf{weakly cartesian} natural transformations

Institution \textbf{Ins}' of \textbf{Poset-based} coalgebraic logic, using the contravariant
adjunction Poset – DLat

\textbf{Ins}' restricts to an institution \textbf{Ins}_{ex-sq} having
– signatures: locally monotone functors which \textbf{preserve exact squares}
– morphisms between signatures: \textbf{weakly exact} natural transformations
Main result

Theorem

There is a (liberal) morphism of institutions between:

- The institution of Set-functors which preserve weak pullbacks and their strongly finitary coalgebraic logic $\text{Ins}_{\text{wpb}}$

- The institution of Poset-functors which preserve exact squares and their strongly finitary coalgebraic logic $\text{Ins}'_{\text{ex-sq}}$

sending a signature to its posetification, and assigning to each logic its positive fragment.
Two institutions of (positive) coalgebraic logic

Signatures

**Category of signatures**

\[ \text{Sign} = [\text{Set}, \text{Set}]^{\text{op}} \]

- Signature: functor \( T : \text{Set} \to \text{Set} \)
- Morphism \( T \to \hat{T} \) of signatures: natural transformation \( \sigma : \hat{T} \to T \)

(notice the change of direction!)

**Poset-category of signatures**

\[ \text{Sign}' = [\text{Poset}, \text{Poset}]^{\text{op}} \]

- Signature: locally monotone functor \( T' : \text{Poset} \to \text{Poset} \)
- Morphism \( T' \to \hat{T}' \) of signatures: monotone natural transformation \( \sigma : \hat{T}' \to T' \)
Two institutions of (positive) coalgebraic logic

Signatures

- **Category of signatures**
  \[
  \text{Sign} = \text{[Set, Set]}^{\text{op}}
  \]

  - Signature: functor
    \[
    T : \text{Set} \to \text{Set}
    \]
  - Morphism \( T \to \hat{T} \) of signatures:
    natural transformation \( \sigma : \hat{T} \to T \)
    (notice the change of direction!)

- **Poset-category of signatures**
  \[
  \text{Sign'} = \text{[Poset, Poset]}^{\text{op}}
  \]

  - Signature: **locally monotone** functor
    \[
    T' : \text{Poset} \to \text{Poset}
    \]
  - Morphism \( T' \to \hat{T}' \) of signatures:
    **monotone** natural transformation
    \[
    \sigma : \hat{T}' \to T'
    \]
Two institutions of (positive) coalgebraic logic

A functor between signatures

\[ \text{Sign} = [\text{Set}, \text{Set}]^{\text{op}} \xrightarrow{\Phi} \text{Sign}' = [\text{Poset}, \text{Poset}]^{\text{op}} \]

1. For \( T : \text{Set} \to \text{Set} \), define
   \[ \Phi(T) := \text{Lan}_D(DT) : \text{Poset} \to \text{Poset} \]

2. For \( \sigma : \hat{T} \to T \), \( \Phi(\sigma) \) is the unique monotone natural transformation such that

\[
\begin{array}{ccc}
D\hat{T} & \xrightarrow{D\sigma} & DT \\
\downarrow \Phi(\hat{T})D & & \downarrow \Phi(T)D \\
\Phi(\hat{T})D & \xrightarrow{\Phi(\sigma)} & \Phi(T)D
\end{array}
\]
Two institutions of (positive) coalgebraic logic

A functor between signatures

\[
\text{Discrete Poset-functor}
\]

\[
\begin{array}{c}
\text{Sign} = [\text{Set, Set}]^{\text{op}} \\
\Phi \\
\text{Sign}' = [\text{Poset, Poset}]^{\text{op}}
\end{array}
\]

1. For \( T : \text{Set} \to \text{Set} \), define

\[
\Phi(T) := \text{Lan}_D(DT) : \text{Poset} \to \text{Poset}
\]

\( \text{Posetification} \)

2. For \( \sigma : \hat{T} \to T \), \( \Phi(\sigma) \) is the unique monotone natural transformation such that

\[
\begin{array}{c}
D \hat{T} \xrightarrow{D\sigma} DT \\
\downarrow \Phi(\hat{T})D \xrightarrow{\Phi(\sigma)} \Phi(T)D
\end{array}
\]
Two institutions of (positive) coalgebraic logic

Models

Reduct functor

\[ \text{Mod} : \text{[Set, Set]} \to \text{Cat} \]

Reduct Poset-functor

\[ \text{Mod}' : \text{[Set, Set]} \to \text{Poset} - \text{Cat} \]

1. Models: coalgebras

\[ T \xrightarrow{\hat{T}} \text{Coalg}(T) \]

2. Morphisms between models: coalgebra morphisms

\[ \hat{T} \xrightarrow{\sigma} T \]

\[ \text{Mod}(\hat{T}) = \text{Coalg}(\hat{T}) \]

\[ \text{Mod}(T) = \text{Coalg}(T) \]

\[ X \xrightarrow{\hat{c}} \hat{T}X \]

\[ X \xrightarrow{\hat{c}} \hat{T}X \xrightarrow{\sigma} TX \]
Two institutions of (positive) coalgebraic logic

Reduct functor

\[ \text{Mod} : \text{[Set, Set]} \rightarrow \text{Cat} \]

Reduct Poset-functor

\[ \text{Mod}' : \text{[Set, Set]} \rightarrow \text{Poset} \rightarrow \text{Cat} \]

1. Models: coalgebras

\[ T \rightarrow \text{Coalg}(T) \]

2. Morphisms between models: coalgebra morphisms

\[ \hat{T} \rightarrow \text{Mod}(\hat{T}) = \text{Coalg}(\hat{T}) \]

\[ \sigma \]

\[ \hat{T} \rightarrow \text{Mod}(T) = \text{Coalg}(T) \]

\[ X \xrightarrow{\hat{T}} \hat{T}X \]

\[ X \xrightarrow{\hat{T}} \hat{T}X \xrightarrow{\sigma} TX \]

A. Balan, A. Kurz, J. Velebil

An institutional approach to positive coalgebraic logic

WADT2014
Two institutions of (positive) coalgebraic logic

A transformation of models

There is a monotone-natural transformation

\[ \beta : \text{Mod} \rightarrow \text{Mod}' \circ \Phi \]

whose components \( \beta_T : \text{Coalg}(T) \rightarrow \text{Coalg}(\Phi(T)) \) are

\[ X \xrightarrow{c} TX \xleftarrow{T} DX \xrightarrow{Dc} DTX \cong \Phi(T)DX \]

Notice that each component \( \beta_T \) has a left adjoint!

A. Balan, A. Kurz, J. Velebil

An institutional approach to positive coalgebraic logic

WADT2014
The institution of Set-coalgebraic logic

Sentences

Context: standard contravariant adjunction of propositional logic

\[ \text{Set}^{\text{op}} \xleftrightarrow{S, U} \text{BA} \]

- \(P\) maps a set to the BA of its subsets
- \(S\) maps a BA to the set of its ultrafilters
The institution of Set-coalgebraic logic

Sentences

Context: standard contravariant adjunction of propositional logic

\[ T^{\text{op}} \circlearrowleft \overset{S}{\xrightarrow{\perp}} \text{BA} \circlearrowright \]

Signature \( T : \text{Set} \to \text{Set} \)

\( T \)-models: \( T \)-coalgebras

- \( P \) maps a set to the BA of its subsets
- \( S \) maps a BA to the set of its ultrafilters

Coalgebraic logic, abstractly

Syntax: functor \( L : \text{BA} \to \text{BA} \)

Semantics: natural transformation \( \delta : LP \to PT^{\text{op}} \)

- \( \text{Alg}(L) \) is a variety
- \( L \) preserves sifted colimits
- \( L \) has a presentation by operations and equations
- \( L \) is determined by its restriction to the f. g. free BAs

A. Balan, A. Kurz, J. Velebil
An institutional approach to positive coalgebraic logic
WADT2014
The institution of Set-coalgebraic logic

Sentences

– Recall: predicate liftings of arity $n$ are natural transformations

$$\text{Set}(\_, 2^n) \rightarrow \text{Set}(\ T\_, 2)$$

– Equivalently, elements of $\text{Set}(\ T(2^n), 2) \cong UPT^{\text{op}} SFn$

(here $F \dashv U : \text{BA} \rightarrow \text{Set}$ is the monadic adjunction between the free BA functor and the forgetful one)

– Define $LFn ::= PT^{\text{op}} SFn$ on free finitely generated BA and extend continuously to all BA ($L = \text{Lan}_J(PT^{\text{op}} SJ)$, with $J : \text{BA}_f \rightarrow \text{BA}$ the inclusion functor)

– The semantics $\delta : LP \rightarrow PT$ is the transpose of the canonical morphism $L \rightarrow PT^{\text{op}} S$
The institution of Set-coalgebraic logic

Sentences

– The natural transformation $\delta$ provides one-step semantics
– To pass to the global semantics, have to iterate the one-step logic constructor $L$ and form the initial $L$-algebra $LI \xrightarrow{\text{in}} I$

The functor $\text{Sen} : \text{Sign} = [\text{Set}, \text{Set}]^\text{op} \to \text{Set}$

1. The set of $T$-sentences

$$T \xrightarrow{\text{Sen}} \text{Sen}(T) = UI$$

2. Translation of sentences

\[ \begin{array}{cccc}
\hat{T} & \xrightarrow{\delta} & PT^\text{op} & \xrightarrow{\text{in}} I \\
\sigma & \Rightarrow & \lambda P & \Rightarrow P\sigma \\
T & \xrightarrow{\delta} & \hat{LP} & \xrightarrow{\text{in}} \hat{I} \\
\end{array} \]

\[ \begin{array}{cccc}
\hat{I} & \xrightarrow{\lambda} & \hat{I} & \xrightarrow{\text{in}} I \\
\hat{P} & \xrightarrow{\delta} & \hat{LP} & \xrightarrow{\text{in}} \hat{I} \\
\end{array} \]

A. Balan, A. Kurz, J. Velebil
An institutional approach to positive coalgebraic logic
WADT2014
The institution of Set-coalgebraic logic

Satisfaction relation

- $T$-model (coalgebra) $X \xrightarrow{c} TX$
- $L$-algebra of subsets $LPX \xrightarrow{\delta} PT^{op}X \xrightarrow{Pc} PX$
- Unique $L$-algebra morphism $I \xrightarrow{\lbrack - \rbrack_{(X,c)}} PX$, $\varphi \mapsto \lbrack \varphi \rbrack_{(X,c)}$
- Satisfaction relation

$x \models_{(X,c)} \varphi \iff x \in \lbrack \varphi \rbrack_{(X,c)}$ 
$(X, c) \models \varphi \iff x \models_{(X,c)} \varphi$, $\forall x \in X$

Theorem

*The construction $\text{Ins} = (\text{Sign}, \text{Mod}, \text{Sen}, \models)$ is an institution.*
### Positive coalgebraic logic

\[
T^{\text{op}} \underset{\mathbf{Set}^{\text{op}}}{\xleftarrow{\downarrow}} \mathbf{BA} \underset{\downarrow}{\xrightarrow{\delta}} L
\]

### Coalgebraic logic

**Syntax:**
- functor $L : \mathbf{BA} \to \mathbf{BA}$

**Semantics:**
- natural transformation $\delta : L \ P \to P \ T^{\text{op}}$
  - $\text{Alg}(L)$ is a variety
  - $L$ has a presentation by operations and equations
  - $L$ preserves sifted colimits
  - $L$ is determined by its restriction to free f. g. BAs

---

A. Balan, A. Kurz, J. Velebil

An institutional approach to positive coalgebraic logic

WADT2014
Positive coalgebraic logic

\[
\begin{align*}
\text{Set}^{\text{op}} & \xrightarrow{\bot} \text{BA} \\
\text{Poset}^{\text{op}} & \xrightarrow{\bot} \text{DLat} \\
\end{align*}
\]

\(T^{\text{op}} \circ \text{Poset}^{\text{op}} \xrightarrow{\bot} \text{DLat} \circ L'\)

\(P'\) maps a poset to the DLat of its upsets.

\(S'\) associates to any DLat the poset of prime filters.

Poset-Coalgebraic logic

Syntax: locally monotone functor \(L' : \text{DLat} \to \text{DLat}\)

Semantics: monotone natural transformation \(\delta' : L'P' \to P'T'^{\text{op}}\)

- \(\text{Alg}(L)\) is an ordered variety
- \(L\) preserves Poset-sifted colimits
- \(L\) has a presentation by monotone operations and equations
- \(L\) is determined by its restriction to free f. g. DLs on discrete posets

A. Balan, A. Kurz, J. Velebil

An institutional approach to positive coalgebraic logic

WADT2014
Positive coalgebraic logic

– Predicate liftings of arity \( p \) are monotone natural transformations

\[
\text{Poset}(\neg, [p, 2]) \to \text{Poset}(T' \neg, 2), \quad (p \text{ finite poset})
\]

– That is, elements of the poset \( \text{Poset}(T'([p, 2]), 2) \cong U' P' T'^{\text{op}} S' F' p \)

(here \([X, Y]\) is the poset of monotone maps \( X \to Y \), and \( F' \dashv U' : \text{DLat} \to \text{Poset} \) is the Poset-monadic adjunction between the free DL functor and the forgetful one)

– Define \( L' F' Dn ::= P' T'^{\text{op}} S' F' Dn \) on free finitely generated DL on discrete generators and extend continuously to all DL

– The semantics \( \delta : L' P' \to P' T' \) is the transpose of the canonical morphism \( L' \to P' T'^{\text{op}} S' \)

– Logic \( (L', \delta') \) is expressive [Kapulkin-Kurz-Velebil12], for finitary \( T' \) which preserves embeddings
Positive coalgebraic logic

– Predicate liftings of arity $p$ are monotone natural transformations

$$\text{Poset}(-, [p, 2]) \to \text{Poset}(T', -, 2), \quad (p \text{ finite poset})$$

– That is, elements of the poset $\text{Poset}(T'([p, 2]), 2) \cong U' P' T'^{\text{op}} S' F' p$

(Here $[X, Y]$ is the poset of monotone maps $X \to Y$, and $F' \dashv U' : \text{DLat} \to \text{Poset}$ is the Poset-monadic adjunction between the free DL functor and the forgetful one)

– Define $L' F'Dn ::= P' T'^{\text{op}} S' F'Dn$ on free finitely generated DL on discrete generators and extend continuously to all DL

– The semantics $\delta : L' P' \to P' T'$ is the transpose of the canonical morphism $L' \to P' T'^{\text{op}} S'$

– Logic $(L', \delta')$ is expressive [Kapulkin-Kurz-Velebil12], for finitary $T'$ which preserves embeddings
The institution of Poset-coalgebraic logic

Sentences and satisfaction relation

- $\text{Sen}' : \text{Sign}' \to \text{Poset}$ \textbf{locally monotone} functor

  $$T' : \text{Poset} \to \text{Poset} \quad \text{iff} \quad \text{Sen}(T') := U'I' \text{ poset of sentences}$$

  (where $L'I' \to I'$ is the initial $L'$-algebra)

- $T'$-coalgebra $X \xrightarrow{c} T'X \quad \text{iff} \quad \llbracket \cdot \rrbracket (X, c) : I' \to P'X$

  (a formula is sent to the \textbf{upperset} of states satisfying it)

- Satisfaction relation

  $$x \models_{(X, c)} \varphi \iff x \in \llbracket \varphi \rrbracket (X, c), \quad (X, c) \models \varphi \iff \forall x \in X, \ x \models_{(X, c)} \varphi$$

  \textbf{Monotone wrt simulations}

\textbf{About simulations}

A. Balan, A. Kurz, J. Velebil

\textbf{An institutional approach to positive coalgebraic logic}

WADT2014
The institution of Poset-based coalgebraic logic

Theorem

*The construction* \( \text{Ins}' = (\text{Sign}', \text{Mod}', \text{Sen}', \models) \) *is an institution, where:*

1. \( \text{Sign}' = [\text{Poset}, \text{Poset}]^{op} \)
2. \( \text{Mod}' : \text{Sign}'^{op} \rightarrow \text{Poset-Cat} \rightarrow \text{Cat}, \quad \text{Mod}(T) = \text{Coalg}(T) \)
3. \( \text{Sen}' : \text{Sign}' \rightarrow \text{Poset} \rightarrow \text{Set}, \quad \text{Sen}'(T) = U'I' \)
4. *The satisfaction relation* \( \models \subseteq |\text{Mod}'(T')| \times \text{Sen}(T') \) *is defined as earlier*
The positive fragment of coalgebraic logic

Theorem (B-Kurz-Velebil13)

Let \( T : \text{Set} \to \text{Set} \) such that:

- \( T \) preserves weak pullbacks
- \( T' = \text{Lan}_D(DT) \) is the posetification of \( T \)
- (\( L, \delta \)) and (\( L', \delta' \)) are the (strongly finitary) logics of \( T \) and \( T' \)

Then \( L' \) is the positive fragment of \( L \). More precisely, there is an isomorphism

\[
\begin{array}{ccc}
\text{DLat} & \xrightarrow{L'} & \text{DLat} \\
\downarrow{\cong} & & \downarrow{\cong} \\
\text{BA} & \xrightarrow{L} & \text{BA}
\end{array}
\]

compatible with semantics \( \delta : LP \to PT^{\text{op}} \) and \( \delta' : L'P' \to P'T'^{\text{op}} \)
Two institutions of (positive) coalgebraic logic

Relating the sentences

\[
\begin{array}{c}
\text{DLat} \xrightarrow{L'} \text{DLat} \\
W \searrow \downarrow \swarrow \nearrow W \\
\text{BA} \xrightarrow{L} \text{BA}
\end{array}
\]

Apply the isomorphism above to construct a \text{monotone} natural transformation between sentences

\[
\alpha : \text{Sen}' \circ \Phi \rightarrow \text{Sen}
\]

restricted to signature functors \(T\) which \text{preserve weak pullbacks} and weakly cartesian natural transformations

\[
\begin{array}{c}
L'W'I' \xrightarrow{\text{in}'} I' \\
\downarrow \searrow \searrow \nearrow W'I' \downarrow U'I'
\end{array}
\]

\[
\begin{array}{c}
L'W'I \xrightarrow{\text{in}} WLI \xrightarrow{\text{Win}} WI \\
\searrow \downarrow \downarrow \nearrow U'W'I \xrightarrow{\text{Win}} UI
\end{array}
\]
Main result

Theorem

There is a (liberal) morphism of institutions between:

- The institution of Set-functors which preserve weak pullbacks and their strongly finitary coalgebraic logic $\text{Ins}_{\text{wpb}}$
- The institution of Poset-functors which preserve exact squares and their strongly finitary coalgebraic logic $\text{Ins}'_{\text{ex-sq}}$

sending a signature to its posetification, and assigning to each logic its positive fragment.
Some references

B-Kurz11  *Finitary functors: from Set to Preord and Poset*, CALCO2011


Pattinson03  *Translating logics for coalgebras*, WADT2002
Thank you!
Thank you!
Thank you!
Examples

1. \( T = \mathcal{P} \) (finite) powerset functor

   Logic \( \mathcal{L}A \) is the BA generated by \( \Box a \), for \( a \in A \), \( \text{wrt} \ \Box \) preserving finite meets

   Semantics \( \delta_X : \mathcal{L}P \mathcal{X} \to \mathcal{P} \mathcal{P}^{\text{op}} \mathcal{X}, \ \Box a \mapsto \{ b \in \mathcal{P} \mathcal{X} \mid b \subseteq a \} \)

2. \( T = \mathcal{N} \) the neighbourhood functor.

   Logic \( \mathcal{L}A \) is the BA generated by \( \Box a \), for \( a \in A \), no equations

   Semantics \( \delta_X : \mathcal{L}P \mathcal{X} \to \mathcal{P} \mathcal{N}^{\text{op}} \mathcal{X}, \ \Box a \mapsto \{ s \in \mathcal{N} \mathcal{X} \mid a \in s \} \)
More examples...

1. \( T = \mathcal{M} \) the (finite) multisets functor

   **Logic** \( LA \) is the BA generated by \( \diamond_n a \), for \( a \in A \), wrt \( \diamond_n \) preserving finite joins

   **Semantics** \( \delta_X : LPX \to P\mathcal{M}^{\text{op}}X \),
   \[
   \diamond_n a \mapsto \{ \varphi \in \mathcal{M}X \mid \text{card } \varphi(x) \geq n \}, \text{ for } n \in \mathbb{N}
   \]

2. \( T = \mathcal{D} \) (finite) probability functor

   **Logic** \( LA \) is the BA generated by \( \diamond_q a \), for \( a \in A \), wrt \( \diamond_q \) preserving finite joins

   **Semantics** \( \delta_X : LPX \to P\mathcal{D}^{\text{op}}X \), \( \diamond_q a \mapsto \{ d \in \mathcal{D}X \mid \sum_{x \in a} d(x) \geq q \} \)
   for \( q \in \mathbb{Q} \cap [0, 1] \)
Posetifications - or how to extend functors from sets to posets

**Functor** \( T : \text{Set} \rightarrow \text{Set} \)

**Extension** Locally monotone functor

\( T' : \text{Poset} \rightarrow \text{Poset} \)

**Posetification** Extension with universal property \( T' = \text{Lan}_D(DT) \)

Poset-left Kan extension

\[
\begin{array}{ccc}
\text{Set} & \xrightarrow{D} & \text{Poset} \\
\downarrow T & & \downarrow T' \\
\text{Set} & \xrightarrow{D} & \text{Poset}
\end{array}
\]
Theorem (B-Kurz-Velebil13)

1. **Existence**

   *Posetification exists for any functor* $T : \text{Set} \rightarrow \text{Set}*

2. **A characterisation of left Kan extensions to posets**

   *For locally monotone* $T' : \text{Poset} \rightarrow \text{Poset}, \ TFAE*

   - $T'$ is $\text{Lan}_D(DT)$ for some $T : \text{Set} \rightarrow \text{Set}$
   - $T'$ preserves discrete posets and coinserters of simplicial resolutions

3. **Taking posetifications is functorial**

   $$[\text{Set, Set}] \longrightarrow [\text{Poset, Poset}], \ T \mapsto \text{Lan}_D(DT)$$

(proof technique: use a ”simplicial representation” of posets)

A. Balan, A. Kurz, J. Velebil

*An institutional approach to positive coalgebraic logic*
Examples

Kripke functors

\[ T ::= \text{Id} \mid T_{X_0} \mid T_0 + T_1 \mid T_0 \times T_1 \mid T^A \]

Posetifications are as expected:

- Posetification of \( \text{Id}_{\text{Set}} \) is \( \text{Id}_{\text{Poset}} \)
- Posetification of the constant functor at set \( X_0 \) is the constant functor at discrete poset \((X_0, =)\)
- Posetification of (co)product functor is again the (co)product, this time in Poset
- Posetification of exponential functor \( TX = X^A \) is again exponential in Poset
Examples (continued)

Motivating example: \( T = \mathcal{P} \), the (finite) power-set functor

Posetification is the (finitely generated) convex power-set functor, with the Egli-Milner order.
Motivating example: $T = \mathcal{P}$, the (finite) power-set functor

Posetification is the (finitely generated) convex power-set functor, with the Egli-Milner order.

Distribution functor $\mathcal{D}X = \{ d : X \rightarrow [0, 1] \mid \sum_{x \in X} d(x) = 1 \}$

Coalgebras: Markov chains

Posetification: $\mathcal{D}'(X, \leq)$ is $\mathcal{D}X$, with order given by

$$d \leq d' \iff \exists \omega \in \mathcal{D}(X \times X) \cdot \begin{cases} \omega(x, y) > 0 \Rightarrow x \leq y \\ \sum_{y \in X} \omega(x, y) = d(x) \\ \sum_{x \in X} \omega(x, y) = d'(y) \end{cases}$$

Multiset functor $\mathcal{M}X = \{ \phi : X \rightarrow \mathbb{N} \mid \text{supp}(\phi) < \infty \}$

Coalgebras: multigraphs

Posetification: still to compute...
Simplicial representation of posets

\[ X \text{ poset} \implies \text{diagram of (discrete po)sets:} \quad X_1 \xrightarrow{d_0} X_0 \xrightarrow{d_1} \]

- \( X_0 \) is the underlying set of \( X \)
- \( X_1 \) is the set of comparable pairs \( X_1 = \{(x, x') \in X \mid x \leq x'\} \)
- \( d_0, d_1 : X_1 \rightarrow X_0 \) the usual projections

The co inserter of the diagram is \( X \) (coinserter = ordered analogue of a coequalizer)

**The left Kan extension (posetification) of any** \( T : \text{Set} \rightarrow \text{Set} \)

Put \( T'X := \text{coins}(Td_0, Td_1) \), for a poset \( X \)

The assignment \( X \mapsto T'X \) is locally monotone, coincides with \( T \) on discrete posets and can be exhibited as left Kan extension of \( DT \) along \( D \)
Example

- $\hat{T} : \text{Set} \rightarrow \text{Set}, \quad \hat{T}X = 2 \times X^A$
  $\hat{T}$-coalgebras are deterministic automata with alphabet $A$ and binary outputs, deciding if a state is accepting or not.

- $T : \text{Set} \rightarrow \text{Set}, \quad TX = (\mathcal{P}X)^A$
  $T$-coalgebras are LTS, with label set $A$.

- Natural transformation $\sigma : \hat{T} \rightarrow T$, $\sigma_X : 2 \times X^A \rightarrow (\mathcal{P}X)^A$,
  $\sigma_X(i, f)(a) = \{f(a)\}$

  Then $\text{Mod}(\sigma) : \text{Coalg}(\hat{T}) \rightarrow \text{Coalg}(T)$ transforms a deterministic automata into a LTS forgetting whether the resulting state is accepting outputs.
Poset-functors and their coalgebras

\( T : \text{Poset} \rightarrow \text{Poset} \) **locally monotone functor**

\( T \)-coalgebras

- Partially ordered set of states \( X = (X, \leq) \)
- Monotone transition map \( X \xrightarrow{c} TX \)
- Monotone translation map \( X \xrightarrow{f} Y \)

Poset-category \( \text{Coalg}(T) \)
Poset-functors and their coalgebras

$T : \text{Poset} \to \text{Poset}$ locally monotone functor

$T$-coalgebras

- Partially ordered set of states $X = (X, \leq)$
- Monotone transition map $X \overset{c}{\to} TX$
- Monotone translation map $X \overset{f}{\to} Y$

Poset-category $\text{Coalg}(T)$

Examples

- ordered automata: $X \to X^A \times \{0, 1\}$, with $A$ the (discrete) input set
Poset-functors and their coalgebras

\( T : \text{Poset} \to \text{Poset} \) \textbf{locally monotone functor}

\( T \)-coalgebras

- Partially ordered set of states \( X = (X, \leq) \)
- Monotone transition map \( X \xrightarrow{c} TX \)
- Monotone translation map \( X \xrightarrow{f} Y \)

Poset-category \( \text{Coalg}(T) \)

Examples

- ordered automata: \( X \to X^A \times \mathbb{2} \), with \( A \) the (discrete) input set

\( X \xrightarrow{f} Y \)
\( c \downarrow \quad \downarrow \quad d \)
\( TX \xrightarrow{Tf} TY \)

\( x \leq x' \)
Poset-functors and their coalgebras

\[ T : \text{Poset} \rightarrow \text{Poset} \text{  locally monotone functor} \]

**\( T \)-coalgebras**

- Partially ordered set of states \( X = (X, \leq) \)
- Monotone transition map \( X \xrightarrow{c} T X \)
- Monotone translation map \( X \xrightarrow{f} Y \)

Poset-category \( \text{Coalg}(T) \)

**Examples**

- ordered automata: \( X \rightarrow X^A \times 2 \), with \( A \) the (discrete) input set
Poset-functors and their coalgebras

\( T : \text{Poset} \to \text{Poset} \) \textbf{locally monotone functor}

\( T \)-coalgebras

- Partially ordered set of states \( X = (X, \leq) \)
- Monotone transition map \( X \xrightarrow{c} TX \)
- Monotone translation map \( X \xrightarrow{f} Y \)

Poset-category \( \text{Coalg}(T) \)

Examples

- ordered automata: \( X \to X^A \times 2 \), with \( A \) the (discrete) input set
Poset-functors and their coalgebras

$T : \text{Poset} \to \text{Poset}$ \textit{locally monotone functor}

$T$-coalgebras

- Partially ordered set of states $X = (X, \leq)$
- Monotone transition map $X \xrightarrow{c} TX$
- Monotone translation map $X \xrightarrow{f} Y$

Poset-category $\text{Coalg}(T)$

Examples

- ordered automata: $X \to X^A \times \mathbb{2}$, with $A$ the (discrete) input set

\[
x \leq x' \quad \Downarrow \quad a \Downarrow \quad a \Downarrow \quad y \leq y'
\]
Poset-functors and their coalgebras

\( T : \text{Poset} \to \text{Poset} \) \textit{locally monotone functor}

\( T \)-coalgebras

- Partially ordered set of states \( X = (X, \leq) \)
- Monotone transition map \( X \xrightarrow{c} TX \)
- Monotone translation map \( X \xrightarrow{f} Y \)

Poset-category \( \text{Coalg}(T) \)

Examples

- ordered automata: \( X \to X^A \times \mathbb{2} \), with \( A \) the (discrete) input set

\[ x \leq x' \]
\[ a \]
\[ y \leq y' \]

- ordered Kripke frames: \( X \to \mathcal{P}_cX \), with \( \mathcal{P}_c \) the convex powerset functor ordered by

\[ U, V \in \mathcal{P}_cX, \ U \subseteq V \iff ( \forall x \in U \ \exists y \in V. \ x \leq y ) \land ( \forall y \in V \ \exists x \in U. \ x \leq y ) \]
Monotone relations

A monotone relation $X \rightharpoonup Y$ is a monotone map $r : Y^{\text{op}} \times X \to \mathcal{P}$

$$y' \leq y \land r(y, x) \land x \leq x' \implies r(y', x')$$

Each monotone map $f : X \to Y$ produce two (adjoint) monotone relations:

$$X \xrightarrow{f^\Diamond} Y \quad f^\Diamond(y, x) \iff y \leq f(x)$$

$$Y \xrightarrow{f^\Diamond} X \quad f^\Diamond(x, y) \iff f(x) \leq y$$

Each monotone relation $X \rightharpoonup Y$ can be represented as a cospan

$$Y \xrightarrow{r_0} \text{Coll}(r) \xleftarrow{r_1} X, \quad r = r_0^\Diamond \circ r_1^\Diamond$$

where the collage $\text{Coll}(r)$ is $Y + X$, with order

$$y \leq y' \quad x \leq x' \quad r(y, x)$$

A. Balan, A. Kurz, J. Velebil

An institutional approach to positive coalgebraic logic

WADT2014
Relation lifting

$T : \text{Poset} \to \text{Poset}$ locally monotone functor, $Y \overset{r}{\rightarrow} X$ monotone relation

$$\forall Y \overset{r_0}{\rightarrow} \text{Coll}(r) \; \overset{r_1}{\rightarrow} X$$

$$\forall TY \overset{Tr_0}{\rightarrow} T\text{Coll}(r) \; \overset{Tr_1}{\rightarrow} TX$$
Relation lifting

$T : \text{Poset} \rightarrow \text{Poset}$ locally monotone functor, $Y \xrightarrow{r} X$ monotone relation

$$
\begin{array}{ccc}
Y & \xleftarrow{r} & X \\
\downarrow{r_0} & & \downarrow{r_1} \\
\text{Coll}(r) & & \text{Coll}(r) \\
\end{array}
$$

$$
\begin{array}{ccc}
TY & \xleftarrow{(Tr_0)} & TX \\
\downarrow{(Tr_1)} & & \\
T\text{Coll}(r) & & T\text{Coll}(r) \\
\end{array}
$$

$r = r_0 \circ r_1$
Relation lifting

$T : \text{Poset} \to \text{Poset}$ locally monotone functor, $Y \xrightarrow{r} X$ monotone relation

\[
\begin{array}{ccc}
Y & \xrightarrow{r} & X \\
\downarrow{r_0} & & \downarrow{r_1} \\
\text{Coll}(r) & & \\
\end{array} \quad \begin{array}{ccc}
TY & \xleftarrow{\text{Rel}_T(r)} & TX \\
\downarrow{(Tr_0)} & & \downarrow{(Tr_1)} \\
T\text{Coll}(r) & & \\
\end{array}
\]

$r = r_0 \circ r_1$

Relation lifting $\text{Rel}_T(r) = (Tr_0) \circ (Tr_1)$
Relation lifting

\( T : \text{Poset} \rightarrow \text{Poset} \) locally monotone functor, \( Y \xrightarrow{r} X \) monotone relation

\[
\begin{aligned}
\text{Coll}(r) & \xleftarrow{\cdot} \bullet \rightarrow \bullet \rightarrow \bullet \rightarrow \bullet \rightarrow \bullet \\
& \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \\
& \text{Coll}(r) & \text{Coll}(r) & \text{Coll}(r) & \text{Coll}(r) & \text{Coll}(r)
\end{aligned}
\]

\( r = r_0 \diamond \circ r_1 \diamond \)

Relation lifting \( \text{Rel}_T(r) = (Tr_0) \diamond \circ (Tr_1) \diamond \)

\( \text{Rel}_T(r)(v, u) \iff Tr_0(v) \leq Tr_1(u) \text{ in } T\text{Coll}(r) \)
Relation lifting

\( T : \text{Poset} \rightarrow \text{Poset} \) locally monotone functor, \( Y \xrightarrow{r} X \) monotone relation

\[
\begin{array}{ccc}
Y & \xleftarrow{r} & X \\
\downarrow{r_0} & & \downarrow{r_1} \\
\text{Coll}(r) & \xleftarrow{r} & \text{Coll}(r)
\end{array}
\quad
\begin{array}{ccc}
TY & \xleftarrow{\text{Rel}_T(r)} & TX \\
\downarrow{(Tr_0)} & & \downarrow{(Tr_1)} \\
T\text{Coll}(r) & \xleftarrow{(Tr_0)} & T\text{Coll}(r)
\end{array}
\]

\( r = r_0 \circ r_1 \)

Relation lifting \( \text{Rel}_T(r) = (Tr_0) \circ (Tr_1) \)

\( \text{Rel}_T(r)(v, u) \iff Tr_0(v) \leq Tr_1(u) \) in \( T\text{Coll}(r) \)

\( T \) preserves \text{exact squares} \implies \text{well-behaved relation lifting}
Exact squares

\[
\begin{array}{c}
E \xrightarrow{p_1} Y \\
\downarrow \downarrow \downarrow \downarrow \\
X \xrightarrow{f} Z
\end{array}
\quad \quad \quad
\begin{array}{c}
E \xrightarrow{p_0} Y \\
\downarrow \downarrow \downarrow \downarrow \\
X \xrightarrow{f} Z
\end{array}
\]

Exact square: \( p_0 \leq g \) and \( p_1 = g \)

(exact square = the ordered analogue of weak pullback)
Exact squares

Exact square: $E \xrightarrow{p_1} Y$ \quad and \quad $f(x) \leq g(y)$ \quad \exists w \in E. \ (x \leq \alpha(w) \land \beta(w) \leq y)$

(exact square = the ordered analogue of weak pullback)
Relation lifting - examples

1. \( T = \mathcal{P}_c \) the convex powerset \( X \rightarrow Y \)

\[
\text{Rel}_{\mathcal{P}_c}(r)(v, u) \iff (\forall y \in v \exists x \in u. r(y, x)) \land (\forall x \in u \exists y \in v. r(y, x))
\]

(here \( u \in \mathcal{P}_c(X), v \in \mathcal{P}_c(Y) \) are convex subsets)

2. \( T = (-)^A \times \mathcal{P} \)

\[
\text{Rel}_T(r)(v, u) \iff (\forall a \in A, r(x_a, y_a)) \land (i \leq j)
\]

where \( u = ((x_a)_{a \in A}, i) \in TX, v = ((y_a)_{a \in A}, j) \in TY \)
Simulations

Locally monotone functor $T' : \text{Poset} \to \text{Poset}$

Coalgebras $X \overset{c}{\rightarrow} T' X$, $Y \overset{d}{\rightarrow} T' Y$
Simulations

Locally monotone functor $T' : \text{Poset} \to \text{Poset}$

Coalgebras $X \xrightarrow{c} T'X$, $Y \xrightarrow{d} T'Y$

A monotone relation $X \xrightarrow{r} Y$ is called a simulation if

\[
\begin{array}{c}
x \rightsquigarrow c(x) \\
\downarrow r \\
y \rightsquigarrow d(y)
\end{array}
\]
Simulations

Locally monotone functor $T' : \text{Poset} \to \text{Poset}$

Coalgebras $X \xrightarrow{c} T'X$, $Y \xrightarrow{d} T'Y$

A monotone relation $X \xrightarrow{r} Y$ is called a simulation if

$$\xymatrix{ x \ar@{~>}[r] \ar[d]_r & c(x) \ar[d]_{\text{Rel}_T(r)} \\
 y \ar@{~>}[r] & d(y) }$$
Simulations

Locally monotone functor $T' : \text{Poset} \to \text{Poset}$

Coalgebras $X \xrightarrow{c} T'X$, $Y \xrightarrow{d} T'Y$

A monotone relation $X \xrightarrow{r} Y$ is called a simulation if

$$
\begin{align*}
x & \xrightarrow{\sim} c(x) \\
r & \mid \quad \text{Rel}_T(r) \\
y & \xrightarrow{\sim} d(y)
\end{align*}
$$

$$
\begin{align*}
X & \xrightarrow{c^\Diamond} TX \\
r & \mid \quad \leq \quad \text{Rel}_T(r) \\
Y & \xrightarrow{d^\Diamond} TY
\end{align*}
$$
Simulations

Locally monotone functor $T' : \text{Poset} \to \text{Poset}$

Coalgebras $X \xrightarrow{c} T'X$, $Y \xrightarrow{d} T'Y$

A monotone relation $X \xrightarrow{r} Y$ is called a simulation if

$$
\begin{align*}
  x & \sim c(x) \\
  y & \sim d(y)
\end{align*}
$$

A coalgebra morphism $X \xrightarrow{f} Y$ induces simulations $X \xrightarrow{f\Diamond} Y$ and $Y \xrightarrow{f\Diamond} X$

Simulations are closed under composition if $T'$ preserves exact squares

Satisfaction relation is monotone wrt simulation order on states:

$$
\begin{align*}
  r(y, x) \land (x \Vdash \varphi) \land \varphi \leq \psi \Longrightarrow (y \Vdash \psi)
\end{align*}
$$

for all simulations $X \xrightarrow{r} Y$, states $x \in X, y \in Y$ and formulae $\varphi, \psi \in I'$
Motivating example

**Signature** $T = P$ (finite) powerset functor

**Logic** $LA$ is the BA generated by $(\Box a)_{a \in A}$ such that

$\Box(a \land b) = \Box a \land \Box b$

**Semantics** $\delta_X : LPX \to PP^{op}X$, $\Box a \mapsto \{ b \in PX \mid b \subseteq a \}$

**Posetification** $T' = P_c$ (finitely generated) convex powerset functor

**Logic** $L'A$ is the DLat generated by $(\Box a, \Diamond a)_{a \in A}$b such that

$\Box(a \land b) = \Box a \land \Box b$, $\Diamond(a \lor b) = \Diamond a \lor \Diamond b$

$\Box a \land \Diamond b \leq \Diamond(a \land b), \quad \Box(a \lor b) \leq \Diamond a \lor \Box b$

**Semantics**

$\delta'_X : LP'X \to PP'^{op}X$, $\frac{}{\Box a \mapsto \{ b \in PX \mid b \subseteq a \}}$

$\Diamond a \mapsto \{ b \in PX \mid b \cap a \neq \emptyset \}$

**Translation** $L'W \cong WL$ induce the DL morphism $\alpha_P$

$\alpha_P(\Diamond \varphi) = \neg \Box \neg \alpha_P(\varphi)$, $\alpha_P(\Box \varphi) = \Box \alpha_P(\varphi)$
Motivating example

**Signature** \( T = \mathcal{P} \) (finite) powerset functor

**Logic** \( \mathcal{L}A \) is the BA generated by \( (\Box a)_{a \in A} \) such that
\[
\Box (a \land b) = \Box a \land \Box b
\]

**Semantics** \( \delta_X : \mathcal{L}P X \rightarrow \mathcal{P}X^{\text{op}} \), \( \Box a \mapsto \{ b \in \mathcal{P}X \mid b \subseteq a \} \)

**Posetification** \( T' = \mathcal{P}_c \) (finitely generated) convex powerset functor

**Logic** \( \mathcal{L}'A \) is the DLat generated by \( (\Box a, \Diamond a)_{a \in A} \) such that
\[
\Box (a \land b) = \Box a \land \Box b, \quad \Diamond (a \lor b) = \Diamond a \lor \Diamond b
\]
\[
\Box a \land \Diamond b \leq \Diamond (a \land b), \quad \Box (a \lor b) \leq \Diamond a \lor \Box b
\]

**Semantics**
\[
\delta'_X : \mathcal{L}'P' X \rightarrow \mathcal{P}'X^{\text{op}} \times \left\{ \begin{array}{l}
\Box a \mapsto \{ b \in \mathcal{P}X \mid b \subseteq a \}
\Diamond a \mapsto \{ b \in \mathcal{P}X \mid b \cap a \neq \emptyset \}
\end{array} \right.
\]

**Translation** \( \mathcal{L}'W \cong \mathcal{W}L \) induce the DL morphism \( \alpha_P \)
\[
\alpha_P(\Diamond \varphi) = \neg \Box \neg \alpha_P(\varphi) \quad \alpha_P(\Box \varphi) = \Box \alpha_P(\varphi)
\]
Motivating example

**Signature** $T = \mathcal{P}$ (finite) powerset functor

**Logic** $LA$ is the BA generated by $(\square a)_{a \in A}$ such that

$\square(a \land b) = \square a \land \square b$

**Semantics** $\delta_X : LPX \to PP^{op}X$, $\square a \mapsto \{ b \in PX \mid b \subseteq a \}$

**Posetification** $T' = \mathcal{P}_c$ (finitely generated) convex powerset functor

**Logic** $L'A$ is the DLat generated by $(\square a, \Diamond a)_{a \in A}$ such that

$\square(a \land b) = \square a \land \square b$, $\Diamond(a \lor b) = \Diamond a \lor \Diamond b$

$\square a \land \Diamond b \leq \Diamond(a \land b)$, $\square(a \lor b) \leq \Diamond a \lor \square b$

**Semantics**

$\delta'_X : L'P'X \to P'P'^{op}X$, $\left\{\begin{array}{l}
\square a \mapsto \{ b \in PX \mid b \subseteq a \} \\
\Diamond a \mapsto \{ b \in PX \mid b \cap a \neq \emptyset \}
\end{array}\right.$

**Translation** $L'W \cong WL$ induce the DL morphism $\alpha_P$

$\alpha_P(\Diamond \varphi) = \lnot \square \lnot \alpha_P(\varphi)$  $\alpha_P(\square \varphi) = \square \alpha_P(\varphi)$