Fixpoint logics and automata: a coalgebraic approach

Yde Venema
ILLC-UvA
http://staff.science.uva.nl/~yde

Coimbra, FICS 6
12 September 2009

Logic & Automata

- Research area interfacing Logic and Theoretical Computer Science:

Logic & Automata

- Research area interfacing Logic and Theoretical Computer Science:
- applications in program specification and verification

Logic & Automata

- Research area interfacing Logic and Theoretical Computer Science:
- applications in program specification and verification
  - model checking of properties of processes

Logic & Automata

- Research area interfacing Logic and Theoretical Computer Science:
- applications in program specification and verification
  - model checking of properties of processes
- theory with long tradition:

Logic & Automata

- Research area interfacing Logic and Theoretical Computer Science:
- applications in program specification and verification
  - model checking of properties of processes
- theory with long tradition:
  - Büchi
  - Rabin
  - . . .

Key Connection

Key Connection

- automata are finite devices for classifying (potentially) infinite objects

Key Connection

- automata are finite devices for classifying (potentially) infinite objects
- automata are handy, non-linear generalizations of formulas

Key Connection

- automata are finite devices for classifying (potentially) infinite objects
- automata are handy, non-linear generalizations of formulas

  (& conversely: effectively transform automaton into equivalent formula)

Key Connection

- automata are finite devices for classifying (potentially) infinite objects
- automata are handy, non-linear generalizations of formulas

  (& conversely: effectively transform automaton into equivalent formula)
- apply automata theory to prove results about logic

Classification of Automata

Classification of Automata

- target objects: streams/trees/graphs/transition systems/. . .

Classification of Automata

- target objects: streams/trees/graphs/transition systems/. . .
- degree of interaction: determinism/nondeterminism/alternation

Classification of Automata

- target objects: streams/trees/graphs/transition systems/. . .
- degree of interaction: determinism/nondeterminism/alternation

  (perhaps better: nature of one-step interaction)

Classification of Automata

- target objects: streams/trees/graphs/transition systems/. . .
- degree of interaction: determinism/nondeterminism/alternation

  (perhaps better: nature of one-step interaction)
- acceptance condition: Büchi/Muller/parity/. . .

Acceptance Conditions

Acceptance Conditions

- A run of $\mathbb{A}$ on an object $\mathbb{S}$ (a match of the acceptance game) induces an infinite sequence $\rho = a_0 a_1 a_2 \ldots$

Acceptance Conditions

- A run of $\mathbb{A}$ on an object $\mathbb{S}$ (a match of the acceptance game) induces an infinite sequence $\rho = a_0 a_1 a_2 \ldots$
- Acceptance given by some set $Acc \subseteq A^\omega$, . . .

Acceptance Conditions

- A run of $\mathbb{A}$ on an object $\mathbb{S}$ (a match of the acceptance game) induces an infinite sequence $\rho = a_0 a_1 a_2 \ldots$
- Acceptance given by some set $Acc \subseteq A^{\omega}$, . . .
- . . . often expressed in terms of the set

$$Inf(\rho) := \{a \in A \mid a \text{ occurs infinitely often in } \rho\}.$$

Acceptance Conditions

- A run of $\mathbb{A}$ on an object $\mathbb{S}$ (a match of the acceptance game) induces an infinite sequence $\rho = a_0 a_1 a_2 \ldots$
- Acceptance given by some set $Acc \subseteq A^\omega$, . . .
- . . . often expressed in terms of the set

$$Inf(\rho) := \{a \in A \mid a \text{ occurs infinitely often in } \rho\}.$$

- Examples:

Name	Type	Acceptance condition
Büchi	$F \subseteq A$	$F \cap Inf(\rho) \neq \varnothing$
Muller	$\mathcal{M} \subseteq \wp(A)$	$Inf(\rho) \in \mathcal{M}$
parity	$\Omega : A \to \omega$	$\max\{\Omega(a) \mid a \in Inf(\rho)\}$ is even

Sample Results

Sample Results

Theorem (McNaughton, Safra)
Every NDet stream automaton has a Det (Muller/parity) equivalent.

Sample Results

Theorem (McNaughton, Safra)
Every NDet stream automaton has a Det (Muller/parity) equivalent.

. . . no generalization to automata on trees/transition systems. . .

Sample Results

Theorem (McNaughton, Safra)
Every NDet stream automaton has a Det (Muller/parity) equivalent.

. . . no generalization to automata on trees/transition systems. . .

Theorem (Muller & Schupp)
Every Alt tree automaton has a NDet Muller equivalent.

Sample Results

Theorem (McNaughton, Safra)
Every NDet stream automaton has a Det (Muller/parity) equivalent.

. . . no generalization to automata on trees/transition systems. . .

Theorem (Muller & Schupp)
Every Alt tree automaton has a NDet Muller equivalent.

. . . generalizations to automata on transition systems/. . .

Sample Results

Theorem (McNaughton, Safra)
Every NDet stream automaton has a Det (Muller/parity) equivalent.

. . . no generalization to automata on trees/transition systems. . .

Theorem (Muller & Schupp)
Every Alt tree automaton has a NDet Muller equivalent.

. . . generalizations to automata on transition systems/. . .

Theorem (Janin & Walukiewicz)
Every recognizable class of transition systems contains a small TS.

Sample Results

Theorem (McNaughton, Safra)
Every NDet stream automaton has a Det (Muller/parity) equivalent.

. . . no generalization to automata on trees/transition systems. . .

Theorem (Muller & Schupp)
Every Alt tree automaton has a NDet Muller equivalent.

. . . generalizations to automata on transition systems/. . .

Theorem (Janin & Walukiewicz)
Every recognizable class of transition systems contains a small TS.

. . . generalizes results on stream & tree automata.

Aim of talk

Aim of talk

Question

Which general principles underlie these results?

Aim of talk

Question

Which general principles underlie these results?

Basic observation

infinite words, trees, LTSs, . . . are coalgebras

Aim of talk

Question

Which general principles underlie these results?

Basic observation

infinite words, trees, LTSs, . . . are coalgebras

Main message

Key results in automata theory are part of Universal Coalgebra

Overview

- Logic & Automata
- Coalgebra
- Coalgebra Automata
- Towards Universal Automata Theory
- Final remarks

Overview

- Logic & Automata
- Coalgebra
- Coalgebra Automata
- Towards Universal Automata Theory
- Final remarks

Coalgebra

- Universal Coalgebra (Rutten, 2000) is
  a general mathematical theory for evolving systems

Coalgebra

- Universal Coalgebra (Rutten, 2000) is
  a general mathematical theory for evolving systems
- It provides a natural framework for notions like
  - behavior

Coalgebra

- Universal Coalgebra (Rutten, 2000) is
  a general mathematical theory for evolving systems
- It provides a natural framework for notions like
  - behavior
  - bisimulation/behavioral equivalence

Coalgebra

- Universal Coalgebra (Rutten, 2000) is
  a general mathematical theory for evolving systems
- It provides a natural framework for notions like
  - behavior
  - bisimulation/behavioral equivalence
  - invariants

Coalgebra

- Universal Coalgebra (Rutten, 2000) is
  a general mathematical theory for evolving systems
- It provides a natural framework for notions like
  - behavior
  - bisimulation/behavioral equivalence
  - invariants
- A coalgebra is a pair $\mathbb{S} = \langle S, \sigma : S \to \mathsf{F}S \rangle$, where
  - S is the set of states of $\mathbb{S}$,
  - σ is the transition map of the coalgebra,
  - F is the type of the coalgebra.

Coalgebra

- Universal Coalgebra (Rutten, 2000) is
  a general mathematical theory for evolving systems
- It provides a natural framework for notions like
  - behavior
  - bisimulation/behavioral equivalence
  - invariants
- A coalgebra is a pair $\mathbb{S} = \langle S, \sigma : S \to \mathsf{F}S \rangle$, where
  - S is the set of states of $\mathbb{S}$,
  - σ is the transition map of the coalgebra,
  - F is the type of the coalgebra.
- Sufficiently general to model concepts like:
  input, output, non-determinism, interaction, probability, . . .

Coalgebra

- Universal Coalgebra (Rutten, 2000) is
  a general mathematical theory for evolving systems
- It provides a natural framework for notions like
  - behavior
  - bisimulation/behavioral equivalence
  - invariants
- A coalgebra is a pair $\mathbb{S} = \langle S, \sigma : S \to \mathsf{F}S \rangle$, where
  - S is the set of states of $\mathbb{S}$,
  - σ is the transition map of the coalgebra,
  - F is the type of the coalgebra.
- Sufficiently general to model concepts like:
  input, output, non-determinism, interaction, probability, . . .
- A pointed F-coalgebra is a pair $(\mathbb{S}, s_0)$, with s_0 a state in $\mathbb{S}$.

Examples

- flows (eg streams/infinite words): $\mathsf{F}S = C \times S$
- bi-flows (eg infinite trees): $\mathsf{F}S = C \times S \times S$
- Kripke frames: $\mathsf{F}S = \wp(S)$
- Kripke models: $\mathsf{F}S = \wp(\mathsf{Prop}) \times \wp(S)$

Examples

- flows (eg streams/infinite words): $\mathsf{F}S = C \times S$
- bi-flows (eg infinite trees): $\mathsf{F}S = C \times S \times S$
- Kripke frames: $\mathsf{F}S = \wp(S)$
- Kripke models: $\mathsf{F}S = \wp(\mathsf{Prop}) \times \wp(S)$
- finite words: $\mathsf{F}S = (C \times S) \uplus \{\downarrow\}$

Examples

- flows (eg streams/infinite words): $\mathsf{F}S = C \times S$
- bi-flows (eg infinite trees): $\mathsf{F}S = C \times S \times S$
- Kripke frames: $\mathsf{F}S = \wp(S)$
- Kripke models: $\mathsf{F}S = \wp(\mathsf{Prop}) \times \wp(S)$
- finite words: $\mathsf{F}S = (C \times S) \uplus \{\downarrow\}$
- finite trees: $\mathsf{F}S = (C \times S \times S) \uplus \{\downarrow\}$

More Examples

- deterministic automata: $\mathsf{F}S = \{0,1\} \times S^C$

More Examples

- deterministic automata: $\mathsf{F}S = \{0,1\} \times S^C$
- labeled transition systems: $\mathsf{F}S = \wp(\mathsf{Prop}) \times (\wp S)^A$

More Examples

- deterministic automata: $\mathsf{F}S = \{0,1\} \times S^C$
- labeled transition systems: $\mathsf{F}S = \wp(\mathsf{Prop}) \times (\wp S)^A$
- (non-wellfounded) sets: $\mathsf{F}S = \wp S$

More Examples

- deterministic automata: $\mathsf{F}S = \{0,1\} \times S^C$
- labeled transition systems: $\mathsf{F}S = \wp(\mathsf{Prop}) \times (\wp S)^A$
- (non-wellfounded) sets: $\mathsf{F}S = \wp S$
- topologies: $\mathsf{F}S = \wp\wp(S)$

Coalgebras over other base categories than **Set**:

More Examples

- deterministic automata: $\mathsf{F}S = \{0,1\} \times S^C$
- labeled transition systems: $\mathsf{F}S = \wp(\mathsf{Prop}) \times (\wp S)^A$
- (non-wellfounded) sets: $\mathsf{F}S = \wp S$
- topologies: $\mathsf{F}S = \wp\wp(S)$

Coalgebras over other base categories than **Set**:

- Harsanyi type space (game theory),

More Examples

- deterministic automata: $\mathsf{F}S = \{0,1\} \times S^C$
- labeled transition systems: $\mathsf{F}S = \wp(\mathsf{Prop}) \times (\wp S)^A$
- (non-wellfounded) sets: $\mathsf{F}S = \wp S$
- topologies: $\mathsf{F}S = \wp\wp(S)$

Coalgebras over other base categories than **Set**:

- Harsanyi type space (game theory),
- descriptive general frames (modal logic)

More Examples

- deterministic automata: $\mathsf{F}S = \{0,1\} \times S^C$
- labeled transition systems: $\mathsf{F}S = \wp(\mathsf{Prop}) \times (\wp S)^A$
- (non-wellfounded) sets: $\mathsf{F}S = \wp S$
- topologies: $\mathsf{F}S = \wp\wp(S)$

Coalgebras over other base categories than **Set**:

- Harsanyi type space (game theory),
- descriptive general frames (modal logic)
- stochastic transition systems

More Examples

- deterministic automata: $\mathsf{F}S = \{0,1\} \times S^C$
- labeled transition systems: $\mathsf{F}S = \wp(\mathsf{Prop}) \times (\wp S)^A$
- (non-wellfounded) sets: $\mathsf{F}S = \wp S$
- topologies: $\mathsf{F}S = \wp\wp(S)$

Coalgebras over other base categories than **Set**:

- Harsanyi type space (game theory),
- descriptive general frames (modal logic)
- stochastic transition systems
- . . .

Coalgebras and their Morphisms

Definition: Let F be a set functor.

Coalgebras and their Morphisms

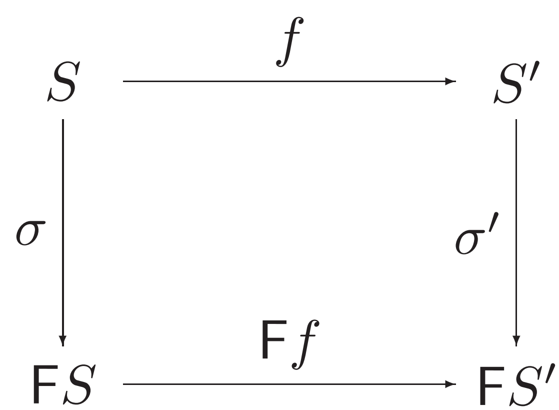
Definition: Let F be a set functor.

- An F-coalgebra is a pair $\mathbb{S} = \langle S, \sigma : S \to \mathsf{F}S \rangle$.

Coalgebras and their Morphisms

Definition: Let F be a set functor.

- An F-coalgebra is a pair $\mathbb{S} = \langle S, \sigma : S \to \mathsf{F}S \rangle$.
- A coalgebra homomorphism between two coalgebras $\mathbb{S}$ and $\mathbb{S}'$ is a map $f : S \to S'$ such that $\sigma' \circ f = \mathsf{F}f \circ \sigma$:



Algebra and Coalgebra

Relation between algebra & coalgebra characterized by both similarities and dualities

-

Algebra and Coalgebra

Relation between algebra & coalgebra characterized by both similarities and dualities

- construction vs observation
  $\mathbb{A} = \langle A, \alpha : A \leftarrow \mathsf{F}A \rangle$
  $\mathbb{C} = \langle C, \gamma : C \rightarrow \mathsf{F}C \rangle$

Algebra and Coalgebra

Relation between algebra & coalgebra characterized by both similarities and dualities

- construction vs observation
  $\mathbb{A} = \langle A, \alpha : A \leftarrow \mathsf{F}A \rangle$
  $\mathbb{C} = \langle C, \gamma : C \rightarrow \mathsf{F}C \rangle$
- congruence vs bisimulation

Algebra and Coalgebra

Relation between algebra & coalgebra characterized by both similarities and dualities

- construction vs observation
  $\mathbb{A} = \langle A, \alpha : A \leftarrow \mathsf{F}A \rangle$
  $\mathbb{C} = \langle C, \gamma : C \rightarrow \mathsf{F}C \rangle$
- congruence vs bisimulation
- induction vs coinduction

Algebra and Coalgebra

Relation between algebra & coalgebra characterized by both similarities and dualities

- construction vs observation
  $\mathbb{A} = \langle A, \alpha : A \leftarrow \mathsf{F}A \rangle$
  $\mathbb{C} = \langle C, \gamma : C \rightarrow \mathsf{F}C \rangle$
- congruence vs bisimulation
- induction vs coinduction
- ...

Coalgebraic Logic

Coalgebraic Logic

- Need coalgebraic languages & logics to describe and reason about behaviour.

Coalgebraic Logic

- Need coalgebraic languages & logics to describe and reason about behaviour.
- Coalgebras are a natural generalization of Kripke structures

Coalgebraic Logic

- Need coalgebraic languages & logics to describe and reason about behaviour.
- Coalgebras are a natural generalization of Kripke structures

  $\Rightarrow$ (try to) use suitable extensions/generalizations of modal languages

Coalgebraic Logic

- Need coalgebraic languages & logics to describe and reason about behaviour.
- Coalgebras are a natural generalization of Kripke structures

  $\Rightarrow$ (try to) use suitable extensions/generalizations of modal languages

- Kurz:

$$\frac{\text{Modal Logic}^*}{\text{Coalgebra}} = \frac{\text{Equational Logic}}{\text{Algebra}}$$

Coalgebraic Logic

- Need coalgebraic languages & logics to describe and reason about behaviour.
- Coalgebras are a natural generalization of Kripke structures

  $\Rightarrow$ (try to) use suitable extensions/generalizations of modal languages
- Kurz:

$$\frac{\text{Modal Logic}^*}{\text{Coalgebra}} = \frac{\text{Equational Logic}}{\text{Algebra}}$$

* with fixpoint operators

Overview

- Logic & Automata
- Coalgebra
- Coalgebra Automata
- Towards Universal Automata Theory
- Final remarks

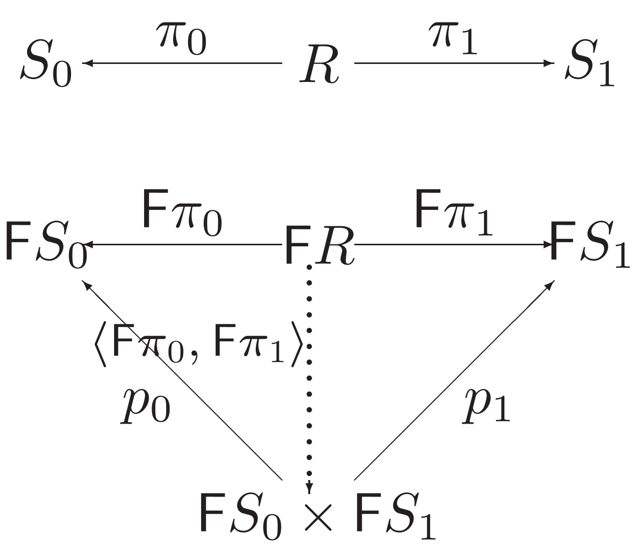
Relation Lifting

Definition: Given relation $R \subseteq S_0 \times S_1$, define $\overline{\mathsf{F}}R \subseteq \mathsf{F}S_0 \times \mathsf{F}S_1$:

Relation Lifting

Definition: Given relation $R \subseteq S_0 \times S_1$, define $\overline{\mathsf{F}}R \subseteq \mathsf{F}S_0 \times \mathsf{F}S_1$:

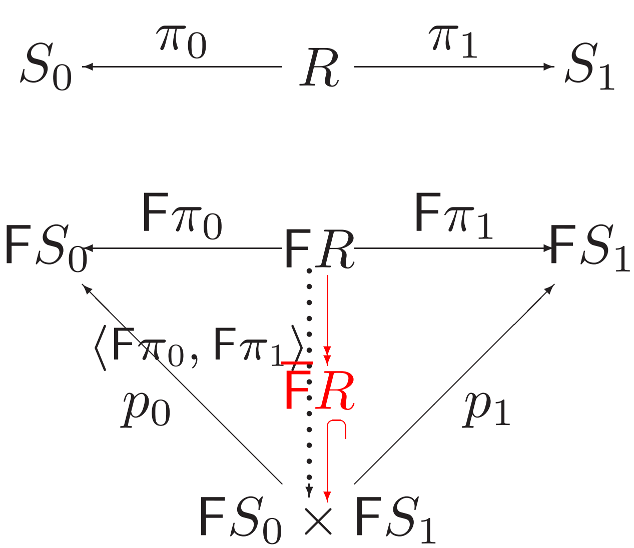
Consider the following diagram:



Relation Lifting

Definition: Given relation $R \subseteq S_0 \times S_1$, define $\overline{\mathsf{F}}R \subseteq \mathsf{F}S_0 \times \mathsf{F}S_1$:

Consider the following diagram:



$\overline{\mathsf{F}}R := \{(\mathsf{F}\pi_0\rho, \mathsf{F}\pi_1\rho) \mid \rho \in \mathsf{F}R\}$.

Relation Lifting, continued

Example:

Relation Lifting, continued

Example:

- C-labelled binary trees:

$$(c, s_l, s_r)\overline{\mathsf{B}}R(c', s'_l, s'_r) \text{ iff } c = c', s_l R s'_l \text{ and } s_r R s'_r.$$

Relation Lifting, continued

Example:

- C-labelled binary trees:

$$(c, s_l, s_r)\overline{\mathsf{B}}R(c', s'_l, s'_r) \text{ iff } c = c', s_l R s'_l \text{ and } s_r R s'_r.$$

- Kripke frames:

$$\begin{array}{lll} X\overline{\mathsf{P}}RX' & \text{iff} & \forall x \in X.\exists x' \in X'.xRx' \\ & \text{and} & \forall x \in X.\exists x' \in X'.xRx'. \end{array}$$

Relation Lifting, continued

Example:

- C-labelled binary trees:

$$(c, s_l, s_r)\overline{\mathsf{B}}R(c', s_l', s_r') \text{ iff } c = c', s_l R s_l' \text{ and } s_r R s_r'.$$

- Kripke frames:

$$\begin{array}{lll} X\overline{\mathsf{P}}RX' & \text{iff} & \forall x \in X.\exists x' \in X'.xRx' \\ & \text{and} & \forall x \in X.\exists x' \in X'.xRx'. \end{array}$$

- Kripke models:

$$(\pi, X)\overline{\mathsf{K}}R(\pi', X') \text{ iff } \pi = \pi' \text{ and } X\overline{\mathsf{P}}RX'$$

Bisimulations via Relation Lifting

Definition: With $\mathbb{S} = (S, \sigma : S \to \mathsf{F}S)$ and $\mathbb{S}' = (S', \sigma' : S \to \mathsf{F}S)$ two F-coalgebras,

Bisimulations via Relation Lifting

Definition: With $\mathbb{S} = (S, \sigma : S \to \mathsf{F}S)$ and $\mathbb{S}' = (S', \sigma' : S \to \mathsf{F}S)$ two F-coalgebras,

- $Z \subseteq S \times S'$ is a bisimulation if $(\sigma(s), \sigma(s')) \in \overline{F}(Z)$ for all $(s, s') \in Z$

Bisimulations via Relation Lifting

Definition: With $\mathbb{S} = (S, \sigma : S \to \mathsf{F}S)$ and $\mathbb{S}' = (S', \sigma' : S \to \mathsf{F}S)$ two F-coalgebras,

- $Z \subseteq S \times S'$ is a bisimulation if $(\sigma(s), \sigma(s')) \in \overline{F}(Z)$ for all $(s, s') \in Z$
- s and s' are bisimilar, notation: $s \leftrightarroweq s'$, if sZs' for some bisimulation Z.

Bisimulations via Relation Lifting

Definition: With $\mathbb{S} = (S, \sigma : S \to \mathsf{F}S)$ and $\mathbb{S}' = (S', \sigma' : S \to \mathsf{F}S)$ two F-coalgebras,

- $Z \subseteq S \times S'$ is a bisimulation if $(\sigma(s), \sigma(s')) \in \overline{F}(Z)$ for all $(s, s') \in Z$
- s and s' are bisimilar, notation: $s \leftrightarroweq s'$, if sZs' for some bisimulation Z.

This generalizes known examples!

Bisimulations via Relation Lifting

Definition: With $\mathbb{S} = (S, \sigma : S \to \mathsf{F}S)$ and $\mathbb{S}' = (S', \sigma' : S \to \mathsf{F}S)$ two F-coalgebras,

- $Z \subseteq S \times S'$ is a bisimulation if $(\sigma(s), \sigma(s')) \in \overline{F}(Z)$ for all $(s, s') \in Z$
- s and s' are bisimilar, notation: $s \leftrightarroweq s'$, if sZs' for some bisimulation Z.

This generalizes known examples!

Proposition: $\{B \mid B : \mathbb{S}_0 \leftrightarroweq \mathbb{S}_1\}$ is a complete lattice, with unions as joins, and $\leftrightarroweq$ as top.

Bisimulations via Relation Lifting

Definition: With $\mathbb{S} = (S, \sigma : S \to \mathsf{F}S)$ and $\mathbb{S}' = (S', \sigma' : S \to \mathsf{F}S)$ two F-coalgebras,

- $Z \subseteq S \times S'$ is a bisimulation if $(\sigma(s), \sigma(s')) \in \overline{F}(Z)$ for all $(s, s') \in Z$
- s and s' are bisimilar, notation: $s \leftrightarroweq s'$, if sZs' for some bisimulation Z.

This generalizes known examples!

Proposition: $\{B \mid B : \mathbb{S}_0 \leftrightarroweq \mathbb{S}_1\}$ is a complete lattice, with unions as joins, and $\leftrightarroweq$ as top.

Proof: Bisimulations are postfixpoints of the monotone operation

$$R \mapsto \{(s_0, s_1) \mid (\sigma_0 s_0, \sigma_1 s_1) \in \overline{\mathsf{F}}R\}$$

The relation $\leftrightarroweq$ is its greatest fixpoint.

Bisimilarity Game

Bisimilarity game $\mathsf{B}(\mathbb{A}, \mathbb{S})$ for F-coalgebras $\mathbb{A} = \langle A, \alpha \rangle$ and $\mathbb{S} = \langle S, \sigma \rangle$:

Position	Player	Admissible moves
$(a, s) \in A \times S$	$\exists$	$\{Z \mid (\alpha(a), \sigma(s)) \in \overline{\mathsf{F}}(Z)\}$
$Z \subseteq A \times S$	$\forall$	$Z = \{(b, t) \mid (b, t) \in Z\}$

Bisimilarity Game

Bisimilarity game $\mathsf{B}(\mathbb{A}, \mathbb{S})$ for F-coalgebras $\mathbb{A} = \langle A, \alpha \rangle$ and $\mathbb{S} = \langle S, \sigma \rangle$:

Position	Player	Admissible moves
$(a, s) \in A \times S$	$\exists$	$\{Z \mid (\alpha(a), \sigma(s)) \in \overline{\mathsf{F}}(Z)\}$
$Z \subseteq A \times S$	$\forall$	$Z = \{(b, t) \mid (b, t) \in Z\}$

Winning conditions:

Bisimilarity Game

Bisimilarity game $\mathsf{B}(\mathbb{A},\mathbb{S})$ for F-coalgebras $\mathbb{A} = \langle A, \alpha \rangle$ and $\mathbb{S} = \langle S, \sigma \rangle$:

Position	Player	Admissible moves
$(a,s) \in A \times S$	$\exists$	$\{Z \mid (\alpha(a), \sigma(s)) \in \overline{\mathsf{F}}(Z)\}$
$Z \subseteq A \times S$	$\forall$	$Z = \{(b,t) \mid (b,t) \in Z\}$

Winning conditions:

- finite matches are lost by the player who gets stuck,

Bisimilarity Game

Bisimilarity game $\mathsf{B}(\mathbb{A}, \mathbb{S})$ for F-coalgebras $\mathbb{A} = \langle A, \alpha \rangle$ and $\mathbb{S} = \langle S, \sigma \rangle$:

Position	Player	Admissible moves
$(a, s) \in A \times S$	$\exists$	$\{Z \mid (\alpha(a), \sigma(s)) \in \overline{\mathsf{F}}(Z)\}$
$Z \subseteq A \times S$	$\forall$	$Z = \{(b, t) \mid (b, t) \in Z\}$

Winning conditions:

- finite matches are lost by the player who gets stuck,
- infinite matches are won by $\exists$.

Bisimilarity Game

Bisimilarity game $\mathsf{B}(\mathbb{A}, \mathbb{S})$ for F-coalgebras $\mathbb{A} = \langle A, \alpha \rangle$ and $\mathbb{S} = \langle S, \sigma \rangle$:

Position	Player	Admissible moves
$(a, s) \in A \times S$	$\exists$	$\{Z \mid (\alpha(a), \sigma(s)) \in \overline{\mathsf{F}}(Z)\}$
$Z \subseteq A \times S$	$\forall$	$Z = \{(b, t) \mid (b, t) \in Z\}$

Winning conditions:

- finite matches are lost by the player who gets stuck,
- infinite matches are won by $\exists$.

Theorem: For all a, s: $(a, s) \in Win_{\exists}(\mathsf{B})$ iff $\mathbb{A}, a \leftrightarrows \mathbb{S}, s$.

Coalgebra Automata

Definition Let F be a set functor.

A (nondeterministic) F-automaton is a quadruple $\mathbb{A} = \langle A, a_I, \Delta, \Omega \rangle$ with

Coalgebra Automata

Definition Let F be a set functor.

A (nondeterministic) F-automaton is a quadruple $\mathbb{A} = \langle A, a_I, \Delta, \Omega \rangle$ with

- A a finite set of states

Coalgebra Automata

Definition Let F be a set functor.

A (nondeterministic) F-automaton is a quadruple $\mathbb{A} = \langle A, a_I, \Delta, \Omega \rangle$ with

- A a finite set of states
- a_I is the starting state

Coalgebra Automata

Definition Let F be a set functor.

A (nondeterministic) F-automaton is a quadruple $\mathbb{A} = \langle A, a_I, \Delta, \Omega \rangle$ with

- A a finite set of states
- a_I is the starting state
- $\Delta : A \to \mathsf{PF}A$ is the transition function

Coalgebra Automata

Definition Let F be a set functor.

A (nondeterministic) F-automaton is a quadruple $\mathbb{A} = \langle A, a_I, \Delta, \Omega \rangle$ with

- A a finite set of states
- a_I is the starting state
- $\Delta : A \to \mathsf{PF}A$ is the transition function

(variation: alternating automata have $\Delta : A \to \mathsf{PPF}A$)

Coalgebra Automata

Definition Let F be a set functor.

A (nondeterministic) F-automaton is a quadruple $\mathbb{A} = \langle A, a_I, \Delta, \Omega \rangle$ with

- A a finite set of states
- a_I is the starting state
- $\Delta : A \to \mathsf{PF}A$ is the transition function

  (variation: alternating automata have $\Delta : A \to \mathsf{PPF}A$)
- $\Omega : A \to \omega$ is the parity acceptance condition

Acceptance Game

Acceptance game $\mathsf{B}(\mathbb{A}, \mathbb{S})$ of F-automaton $\mathbb{A} = \langle A, a_I \Delta, \Omega \rangle$ on F-coalgebra $\mathbb{S} = \langle S, \sigma \rangle$:

Position	Player	Admissible moves	Parity
$(a, s) \in A \times S$	$\exists$	$\{(\alpha, s) \in \mathsf{F}A \times S \mid \alpha \in \Delta(a)\}$	$\Omega(a)$
$(\alpha, s) \in \mathsf{F}A \times S$	$\exists$	$\{Z \mid (\alpha, \sigma(s)) \in \overline{\mathsf{F}}(Z)\}$	0
$Z \subseteq A \times S$	$\forall$	$Z = \{(b, t) \mid (b, t) \in Z\}$	0

Acceptance Game

Acceptance game $\mathcal{B}(\mathbb{A}, \mathbb{S})$ of F-automaton $\mathbb{A} = \langle A, a_I \Delta, \Omega \rangle$ on F-coalgebra $\mathbb{S} = \langle S, \sigma \rangle$:

Position	Player	Admissible moves	Parity
$(a, s) \in A \times S$	$\exists$	$\{(\alpha, s) \in \mathsf{F}A \times S \mid \alpha \in \Delta(a)\}$	$\Omega(a)$
$(\alpha, s) \in \mathsf{F}A \times S$	$\exists$	$\{Z \mid (\alpha, \sigma(s)) \in \overline{\mathsf{F}}(Z)\}$	0
$Z \subseteq A \times S$	$\forall$	$Z = \{(b, t) \mid (b, t) \in Z\}$	0

Winning conditions:

Acceptance Game

Acceptance game $\mathsf{B}(\mathbb{A}, \mathbb{S})$ of F-automaton $\mathbb{A} = \langle A, a_I \Delta, \Omega \rangle$ on F-coalgebra $\mathbb{S} = \langle S, \sigma \rangle$:

Position	Player	Admissible moves	Parity
$(a, s) \in A \times S$	$\exists$	$\{(\alpha, s) \in \mathsf{F}A \times S \mid \alpha \in \Delta(a)\}$	$\Omega(a)$
$(\alpha, s) \in \mathsf{F}A \times S$	$\exists$	$\{Z \mid (\alpha, \sigma(s)) \in \overline{\mathsf{F}}(Z)\}$	0
$Z \subseteq A \times S$	$\forall$	$Z = \{(b, t) \mid (b, t) \in Z\}$	0

Winning conditions:

- finite matches are lost by the player who gets stuck,

Acceptance Game

Acceptance game $\mathsf{B}(\mathbb{A}, \mathbb{S})$ of F-automaton $\mathbb{A} = \langle A, a_I \Delta, \Omega \rangle$ on F-coalgebra $\mathbb{S} = \langle S, \sigma \rangle$:

Position	Player	Admissible moves	Parity
$(a, s) \in A \times S$	$\exists$	$\{(\alpha, s) \in \mathsf{F}A \times S \mid \alpha \in \Delta(a)\}$	$\Omega(a)$
$(\alpha, s) \in \mathsf{F}A \times S$	$\exists$	$\{Z \mid (\alpha, \sigma(s)) \in \overline{\mathsf{F}}(Z)\}$	0
$Z \subseteq A \times S$	$\forall$	$Z = \{(b, t) \mid (b, t) \in Z\}$	0

Winning conditions:

- finite matches are lost by the player who gets stuck,
- infinite matches are won as specified by Ω'.

Acceptance Game

Acceptance game $\mathsf{B}(\mathbb{A}, \mathbb{S})$ of F-automaton $\mathbb{A} = \langle A, a_I \Delta, \Omega \rangle$ on F-coalgebra $\mathbb{S} = \langle S, \sigma \rangle$:

Position	Player	Admissible moves	Parity
$(a, s) \in A \times S$	$\exists$	$\{(\alpha, s) \in \mathsf{F}A \times S \mid \alpha \in \Delta(a)\}$	$\Omega(a)$
$(\alpha, s) \in \mathsf{F}A \times S$	$\exists$	$\{Z \mid (\alpha, \sigma(s)) \in \overline{\mathsf{F}}(Z)\}$	0
$Z \subseteq A \times S$	$\forall$	$Z = \{(b, t) \mid (b, t) \in Z\}$	0

Winning conditions:

- finite matches are lost by the player who gets stuck,
- infinite matches are won as specified by Ω'.

Definition: $\mathbb{A}$ accepts $(\mathbb{S}, s)$ if $(a_I, s) \in Win_\exists(\mathcal{B}(\mathbb{A}, \mathbb{S}))$.

Coalgebra Automata & ‘Standard’ Automata

Standard automata on words, trees, transition systems, etc, are all special instances of coalgebra automata

Coalgebra Automata & ‘Standard’ Automata

Standard automata on words, trees, transition systems, etc, are all special instances of coalgebra automata

Usual presentation of nondeterministic tree automata:

$$\Delta : A \times C \to \mathsf{P}(A \times A)$$

Coalgebra Automata & ‘Standard’ Automata

Standard automata on words, trees, transition systems, etc, are all special instances of coalgebra automata

Usual presentation of nondeterministic tree automata:

$$\Delta : A \times C \to \mathsf{P}(A \times A)$$

This is equivalent to the coalgebraic presentation where

$$\Delta : A \to \mathsf{P}(C \times A \times A) :$$

Coalgebra Automata & 'Standard' Automata

Standard automata on words, trees, transition systems, etc, are all special instances of coalgebra automata

Usual presentation of nondeterministic tree automata:

$$\Delta : A \times C \to \mathsf{P}(A \times A)$$

This is equivalent to the coalgebraic presentation where

$$\Delta : A \to \mathsf{P}(C \times A \times A) :$$

$$\begin{aligned}
(A \times C) \to \mathsf{P}(A \times A) &\cong (A \times C) \to ((A \times A) \to 2) \\
&\cong (A \times C \times A \times A) \to 2 \\
&\cong A \to ((C \times A \times A) \to 2) \\
&\cong A \to \mathsf{P}(C \times A \times A)
\end{aligned}$$

General Picture

General Picture

- Coalgebra automata are similar to the coalgebras themselves:

$$
\begin{array}{llll}
\text{F-coalgebra} & \mathbb{S} & = & \langle S, \sigma : S \to \mathsf{F}S \rangle \\
\text{F-automaton} & \mathbb{A} & = & \langle A, \Delta : A \to \mathsf{P}(\mathsf{F}A), Acc \rangle
\end{array}
$$

General Picture

- Coalgebra automata are similar to the coalgebras themselves:

F-coalgebra	$\mathbb{S}$	$=$	$\langle S, \sigma : S \to \mathsf{F}S \rangle$
F-automaton	$\mathbb{A}$	$=$	$\langle A, \Delta : A \to \mathsf{P}(\mathsf{F}A), Acc \rangle$

- Acceptance generalizes bisimilarity.

General Picture

- Coalgebra automata are similar to the coalgebras themselves:

  F-coalgebra $\mathbb{S} = \langle S, \sigma : S \to \mathsf{F}S \rangle$

  F-automaton $\mathbb{A} = \langle A, \Delta : A \to \mathsf{P}(\mathsf{F}A), Acc \rangle$

- Acceptance generalizes bisimilarity.
- Separate the combinatorics (Acc) from the dynamics (Δ).

Results in Universal Coalgebra

(The following are joint results with Clemens Kupke)

Theorem: Let F be an arbitrary set functor preserving weak pullbacks.

Results in Universal Coalgebra

(The following are joint results with Clemens Kupke)

Theorem: Let F be an arbitrary set functor preserving weak pullbacks.

- Finite Model Property If a $\langle A, a_I, \Delta : A \to \mathsf{P}(\mathsf{F}A), \Omega\rangle$ accepts some coalgebra, then it accepts some $\langle A', \alpha, a_I\rangle$ with $A' \subseteq A$ and $\alpha(a) \in \Delta(a)$.

Results in Universal Coalgebra

(The following are joint results with Clemens Kupke)

Theorem: Let F be an arbitrary set functor preserving weak pullbacks.

- Finite Model Property If a $\langle A, a_I, \Delta : A \to \mathsf{P}(\mathsf{F}A), \Omega\rangle$ accepts some coalgebra, then it accepts some $\langle A', \alpha, a_I\rangle$ with $A' \subseteq A$ and $\alpha(a) \in \Delta(a)$.
- Determinization There is a construction transforming an alternating F-automaton to an equivalent nondeterministic one.

Results in Universal Coalgebra

(The following are joint results with Clemens Kupke)

Theorem: Let F be an arbitrary set functor preserving weak pullbacks.

- Finite Model Property If a $\langle A, a_I, \Delta : A \to \mathsf{P}(\mathsf{F}A), \Omega\rangle$ accepts some coalgebra, then it accepts some $\langle A', \alpha, a_I\rangle$ with $A' \subseteq A$ and $\alpha(a) \in \Delta(a)$.
- Determinization There is a construction transforming an alternating F-automaton to an equivalent nondeterministic one.
- Closure Properties The recognizable languages (classes of F-coalgebras that are accepted by some F-automaton) are closed under union, intersection, and projection (modulo bisimulation).

Results in Universal Coalgebra

(The following are joint results with Clemens Kupke)

Theorem: Let F be an arbitrary set functor preserving weak pullbacks.

- Finite Model Property If a $\langle A, a_I, \Delta : A \to \mathsf{P}(\mathsf{F}A), \Omega \rangle$ accepts some coalgebra, then it accepts some $\langle A', \alpha, a_I \rangle$ with $A' \subseteq A$ and $\alpha(a) \in \Delta(a)$.
- Determinization There is a construction transforming an alternating F-automaton to an equivalent nondeterministic one.
- Closure Properties The recognizable languages (classes of F-coalgebras that are accepted by some F-automaton) are closed under union, intersection, and projection (modulo bisimulation).
- New! The recognizable languages are closed under complementation.

(This is a joint result with Christian Kissig).

Results in Fixpoint Logic

Results in Fixpoint Logic

The above results have various corollaries in fixpoint logics.

Overview

- Logic & Automata
- Coalgebra
- Coalgebra Automata
- Towards Universal Automata Theory
- Final remarks

Predicate Liftings

Definition: A predicate lifting for a set functor F is a natural transformation

$$\lambda : \breve{\mathsf{P}} \longrightarrow \breve{\mathsf{P}} \circ \mathsf{F}$$

where $\breve{\mathsf{P}}$ is the contravariant power set functor.

An n-ary predicate lifting for F is a natural transformation $\lambda : \breve{\mathsf{P}}^n \longrightarrow \breve{\mathsf{P}} \circ \mathsf{F}$.

Predicate Liftings

Definition: A predicate lifting for a set functor F is a natural transformation

$$\lambda : \breve{\mathsf{P}} \rightarrow \breve{\mathsf{P}} \circ \mathsf{F}$$

where $\breve{\mathsf{P}}$ is the contravariant power set functor.

An n-ary predicate lifting for F is a natural transformation $\lambda : \breve{\mathsf{P}}^n \rightarrow \breve{\mathsf{P}} \circ \mathsf{F}$.

Define a modal operator $\heartsuit_\lambda$:

$$\begin{array}{lll} \mathbb{S}, s \Vdash \heartsuit_\lambda \varphi & \text{iff} & \sigma(s) \in \lambda_S([\![\varphi]\!]). \\ \mathbb{S}, s \Vdash \heartsuit_\lambda (\varphi_1, \ldots, \varphi_n) & \text{iff} & \sigma(s) \in \lambda_S([\![\varphi_1, \ldots, [\![\varphi_n]\!]\!]). \end{array}$$

Predicate Liftings

Definition: A predicate lifting for a set functor F is a natural transformation

$$\lambda : \breve{\mathsf{P}} \to \breve{\mathsf{P}} \circ \mathsf{F}$$

where $\breve{\mathsf{P}}$ is the contravariant power set functor.

An n-ary predicate lifting for F is a natural transformation $\lambda : \breve{\mathsf{P}}^n \to \breve{\mathsf{P}} \circ \mathsf{F}$.

Define a modal operator $\heartsuit_\lambda$:

$$\begin{array}{lll} \mathbb{S}, s \Vdash \heartsuit_\lambda \varphi & \text{iff} & \sigma(s) \in \lambda_S([\![\varphi]\!]). \\ \mathbb{S}, s \Vdash \heartsuit_\lambda(\varphi_1, \ldots, \varphi_n) & \text{iff} & \sigma(s) \in \lambda_S([\![\varphi_1, \ldots, [\![\varphi_n]\!]\!]). \end{array}$$

Naturality of λ guarantees bisimulation invariance of the modalities $\heartsuit_\lambda$.

Example: Probabilistic Modal Logic

Let D_ω be the finitary distribution functor given by

$$\begin{aligned}
\mathsf{D}_\omega S &:= \left\{\mu : S \to [0,1] \mid \sum_{s\in S}\mu(s) = 1 \ \&\ |Supp(\mu)| < \omega\right\} \\
\mathsf{D}_\omega(f : S \to S') &:= \Big(\mu : S \to [0,1]\Big) \mapsto \lambda s'. \sum_{s\in f^{-1}(s')} \mu(s).
\end{aligned}$$

D_ω-coalgebras can be seen as probabilistic frames.

Example: Probabilistic Modal Logic

Let D_ω be the finitary distribution functor given by

$$\begin{array}{lcl} \mathsf{D}_\omega S & := & \left\{\mu : S \to [0,1] \mid \sum_{s \in S} \mu(s) = 1 \ \& \ |Supp(\mu)| < \omega\right\} \\ \mathsf{D}_\omega(f : S \to S') & := & \left(\mu : S \to [0,1]\right) \mapsto \lambda s'. \sum_{s \in f^{-1}(s')} \mu(s). \end{array}$$

D_ω-coalgebras can be seen as probabilistic frames.

For $\mathsf{F} = \mathsf{D}_\omega$ and $q \in \mathbb{Q}$, the lifting $\lambda^q : \breve{\mathsf{P}} \to \breve{\mathsf{P}}\mathsf{D}_\omega$ is given by

$$\lambda^q_S(X \in \mathsf{P}S) := \{\mu : S \to [0,1] \mid \textstyle\sum_{x \in X} \mu(x) \geq q\}$$

Example: Probabilistic Modal Logic

Let D_ω be the finitary distribution functor given by

$$\begin{array}{lcl} \mathsf{D}_\omega S & := & \left\{\mu : S \to [0,1] \mid \sum_{s \in S} \mu(s) = 1 \ \& \ |Supp(\mu)| < \omega \right\} \\ \mathsf{D}_\omega(f : S \to S') & := & \left(\mu : S \to [0,1]\right) \mapsto \lambda s'. \sum_{s \in f^{-1}(s')} \mu(s). \end{array}$$

D_ω-coalgebras can be seen as probabilistic frames.

For $\mathsf{F} = \mathsf{D}_\omega$ and $q \in \mathbb{Q}$, the lifting $\lambda^q : \breve{\mathsf{P}} \to \breve{\mathsf{P}}\mathsf{D}_\omega$ is given by

$\lambda^q_S(X \in \mathsf{P}S) := \{\mu : S \to [0,1] \mid \sum_{x \in X} \mu(x) \geq q\}$

This gives rise to probabilistic modal logic:

$$\mathbb{S}, s \Vdash \heartsuit_q \varphi \iff \sigma_s([\![\varphi]\!]) \geq q.$$

Predicate Liftings: More Examples

Predicate Liftings: More Examples

- For $\mathsf{F} = \mathsf{P}$, the liftings $\lambda^{\Diamond}, \lambda^{\Box} : \breve{\mathsf{P}} \to \breve{\mathsf{P}}\mathsf{P}$ are given by

$$\begin{array}{lcl} \lambda^{\Diamond}_S(A) & := & \{X \subseteq S \mid X \cap A \neq \varnothing\} \\ \lambda^{\Box}_S(A) & := & \{X \subseteq S \mid X \subseteq A\} \end{array}$$

Predicate Liftings: More Examples

- For $\mathsf{F} = \mathsf{P}$, the liftings $\lambda^{\Diamond}, \lambda^{\Box} : \breve{\mathsf{P}} \to \breve{\mathsf{P}}\mathsf{P}$ are given by

$$\begin{array}{rcl} \lambda^{\Diamond}_S(A) & := & \{X \subseteq S \mid X \cap A \neq \varnothing\} \\ \lambda^{\Box}_S(A) & := & \{X \subseteq S \mid X \subseteq A\} \end{array}$$

The associated modalities satisfy $\heartsuit_{\lambda^{\Diamond}} \equiv \Diamond$ and $\heartsuit_{\lambda^{\Box}} \equiv \Box$.

Predicate Liftings: More Examples

- For $\mathsf{F} = \mathsf{P}$, the liftings $\lambda^{\Diamond}, \lambda^{\Box} : \breve{\mathsf{P}} \to \breve{\mathsf{P}}\mathsf{P}$ are given by

$$\lambda^{\Diamond}_S(A) := \{X \subseteq S \mid X \cap A \neq \varnothing\}$$

$$\lambda^{\Box}_S(A) := \{X \subseteq S \mid X \subseteq A\}$$

  The associated modalities satisfy $\heartsuit_{\lambda^{\Diamond}} \equiv \Diamond$ and $\heartsuit_{\lambda^{\Box}} \equiv \Box$.

- graded modal logic

Predicate Liftings: More Examples

- For $\mathsf{F} = \mathsf{P}$, the liftings $\lambda^{\Diamond}, \lambda^{\Box} : \breve{\mathsf{P}} \to \breve{\mathsf{P}}\mathsf{P}$ are given by

$$\begin{aligned} \lambda^{\Diamond}_S(A) &:= \{X \subseteq S \mid X \cap A \neq \varnothing\} \\ \lambda^{\Box}_S(A) &:= \{X \subseteq S \mid X \subseteq A\} \end{aligned}$$

  The associated modalities satisfy $\heartsuit_{\lambda^{\Diamond}} \equiv \Diamond$ and $\heartsuit_{\lambda^{\Box}} \equiv \Box$.

- graded modal logic

- coalition logic

Predicate Liftings: More Examples

- For $\mathsf{F} = \mathsf{P}$, the liftings $\lambda^{\Diamond}, \lambda^{\Box} : \breve{\mathsf{P}} \to \breve{\mathsf{P}}\mathsf{P}$ are given by

$$\begin{aligned} \lambda^{\Diamond}_S(A) &:= \{X \subseteq S \mid X \cap A \neq \varnothing\} \\ \lambda^{\Box}_S(A) &:= \{X \subseteq S \mid X \subseteq A\} \end{aligned}$$

  The associated modalities satisfy $\heartsuit_{\lambda^{\Diamond}} \equiv \Diamond$ and $\heartsuit_{\lambda^{\Box}} \equiv \Box$.

- graded modal logic
- coalition logic
- conditional logic
- . . .

Coalgebraic Modal (Fixpoint) Logic

ML_Λ For a family Λ of predicate liftings, use $\{\heartsuit_\lambda \mid \lambda \in \Lambda\}$ as modalities:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid \heartsuit_\lambda(\varphi_1, \ldots, \varphi_n).$$

Coalgebraic Modal (Fixpoint) Logic

ML_Λ For a family Λ of predicate liftings, use $\{\heartsuit_\lambda \mid \lambda \in \Lambda\}$ as modalities:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid \heartsuit_\lambda(\varphi_1, \ldots, \varphi_n).$$

Semantics:

$$\mathbb{S}, s \Vdash \heartsuit_\lambda(\varphi_0, \ldots, \varphi_n) \iff \sigma(s) \in \lambda([\![\varphi_1]\!], \ldots, [\![\varphi_n]\!]).$$

Coalgebraic Modal (Fixpoint) Logic

ML_Λ For a family Λ of predicate liftings, use $\{\heartsuit_\lambda \mid \lambda \in \Lambda\}$ as modalities:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid \heartsuit_\lambda(\varphi_1, \ldots, \varphi_n).$$

Semantics:

$$\mathbb{S}, s \Vdash \heartsuit_\lambda(\varphi_0, \ldots, \varphi_n) \iff \sigma(s) \in \lambda([\![\varphi_1]\!], \ldots, [\![\varphi_n]\!]).$$

A predicate lifting λ is monotone if $\forall i. X_i \subseteq Y_i$ implies $\lambda(\vec{X}) \subseteq \lambda(\vec{Y})$.

Coalgebraic Modal (Fixpoint) Logic

ML_Λ For a family Λ of predicate liftings, use $\{\heartsuit_\lambda \mid \lambda \in \Lambda\}$ as modalities:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid \heartsuit_\lambda(\varphi_1, \ldots, \varphi_n).$$

Semantics:

$$\mathbb{S}, s \Vdash \heartsuit_\lambda(\varphi_0, \ldots, \varphi_n) \iff \sigma(s) \in \lambda([\![\varphi_1]\!], \ldots, [\![\varphi_n]\!]).$$

A predicate lifting λ is monotone if $\forall i. X_i \subseteq Y_i$ implies $\lambda(\vec{X}) \subseteq \lambda(\vec{Y})$.

For a family Λ of such liftings, introduce fixpoint extension $\mu\mathrm{ML}_\Lambda$ of ML_Λ.

Theorem (Cîrstea, Kupke & Pattinson [CSL'09]))
If a family Λ of monotone liftings admits a 'reasonable' one-step tableau, the satisfiability problem for $\mu\mathrm{ML}_\Lambda$ is in EXPTIME.

One-step semantics

Definition: $\Lambda^{\heartsuit} A := \{\heartsuit_\lambda(a_1, \ldots, a_n) \mid \lambda \in \Lambda, a_i \in A\}$.

$\mathcal{L}_0 B$ is the set of lattice terms over B:

$$\varphi ::= b \in B \mid \bot \mid \top \mid \varphi \vee \varphi \mid \varphi \wedge \varphi$$

One-step semantics

Definition: $\Lambda^{\heartsuit} A := \{\heartsuit_\lambda(a_1, \ldots, a_n) \mid \lambda \in \Lambda, a_i \in A\}$.

$\mathcal{L}_0 B$ is the set of lattice terms over B:

$$\varphi ::= b \in B \mid \bot \mid \top \mid \varphi \vee \varphi \mid \varphi \wedge \varphi$$

Definition: An A-valuation over S is a map $v : A \to \mathcal{P}(S)$.

One-step semantics

Definition: $\Lambda^{\heartsuit} A := \{\heartsuit_\lambda(a_1, \ldots, a_n) \mid \lambda \in \Lambda, a_i \in A\}$.

$\mathcal{L}_0 B$ is the set of lattice terms over B:

$$\varphi ::= b \in B \mid \bot \mid \top \mid \varphi \vee \varphi \mid \varphi \wedge \varphi$$

Definition: An A-valuation over S is a map $v : A \to \mathcal{P}(S)$.

Associate a notion of one-step satisfaction $\models^v \subseteq \mathsf{F}S \times \mathcal{L}_0 \Lambda^{\heartsuit} A$:

$$\begin{array}{lll} \mathsf{F}S, \tau \models^v \heartsuit_\lambda a & \text{iff} & \tau \in \lambda(v(a)) \\ \mathsf{F}S, \tau \models^v \varphi_0 \wedge \varphi_1 & \text{iff} & \mathsf{F}S, \tau \models^v \varphi_0 \text{ and } \mathsf{F}S, \tau \models^v \varphi_1 \\ \mathsf{F}S, \tau \models^v \varphi_0 \vee \varphi_1 & \text{iff} & \mathsf{F}S, \tau \models^v \varphi_0 \text{ or } \mathsf{F}S, \tau \models^v \varphi_1 \end{array}$$

One-step semantics

Definition: $\Lambda^{\heartsuit}A := \{\heartsuit_\lambda(a_1, \ldots, a_n) \mid \lambda \in \Lambda, a_i \in A\}$.

$\mathcal{L}_0 B$ is the set of lattice terms over B:

$$\varphi ::= b \in B \mid \bot \mid \top \mid \varphi \vee \varphi \mid \varphi \wedge \varphi$$

Definition: An A-valuation over S is a map $v : A \to \mathcal{P}(S)$.

Associate a notion of one-step satisfaction $\models^v \subseteq \mathsf{F}S \times \mathcal{L}_0\Lambda^{\heartsuit}A$:

$$\begin{array}{lll} \mathsf{F}S, \tau \models^v \heartsuit_\lambda a & \text{iff} & \tau \in \lambda(v(a)) \\ \mathsf{F}S, \tau \models^v \varphi_0 \wedge \varphi_1 & \text{iff} & \mathsf{F}S, \tau \models^v \varphi_0 \text{ and } \mathsf{F}S, \tau \models^v \varphi_1 \\ \mathsf{F}S, \tau \models^v \varphi_0 \vee \varphi_1 & \text{iff} & \mathsf{F}S, \tau \models^v \varphi_0 \text{ or } \mathsf{F}S, \tau \models^v \varphi_1 \end{array}$$

Note: With $v : A \to \mathcal{P}(S)$, associate a relation $Z_v := \{(a, s) \mid s \in v(a)\}$.

Λ-Automata

Definition: Let Λ be a family of predicate liftings for a set functor F.

A Λ-automaton is a quadruple $\mathbb{A} = \langle A, a_I, \delta, \Omega \rangle$ with

- A a finite set of states, of which a_I is the starting state
- $\delta : A \to \mathcal{L}_0 \Lambda^{\heartsuit} A$ is the transition function
- $\Omega : A \to \omega$ is the parity acceptance condition

Λ-Automata

Definition: Let Λ be a family of predicate liftings for a set functor F.

A Λ-automaton is a quadruple $\mathbb{A} = \langle A, a_I, \delta, \Omega \rangle$ with
- A a finite set of states, of which a_I is the starting state
- $\delta : A \to \mathcal{L}_0\Lambda^{\heartsuit}A$ is the transition function
- $\Omega : A \to \omega$ is the parity acceptance condition

Definition: Acceptance game of a Λ-automaton $\mathbb{A}$ and a T-coalgebra $\mathbb{S}$:

Position	Player	Admissible moves	Priority
$(a, s) \in A \times S$	$\exists$	$\{v : A \to \mathcal{P}(S) \mid TS, \sigma(s) \models^v \delta(a)\}$	$\Omega(a)$
$v : A \to \mathcal{P}(S)$	$\forall$	$Z_v = \{\{(a', s') \mid s' \in v(a')\}$	0

Bounded Model Property

(The following is joint work with Gaëlle Fontaine and Raul Leal)

Theorem: Let $\mathbb{A}$ be some Λ-automaton such that $L(\mathbb{A}) \neq \varnothing$.
Then $\mathbb{A}$ accepts some pointed coalgebra $(\mathbb{S}, s)$ of size exponential in $|A|$.

Bounded Model Property

(The following is joint work with Gaëlle Fontaine and Raul Leal)

Theorem: Let $\mathbb{A}$ be some Λ-automaton such that $L(\mathbb{A}) \neq \varnothing$.
Then $\mathbb{A}$ accepts some pointed coalgebra $(\mathbb{S}, s)$ of size exponential in $|A|$.

In fact, our proof provides a direct construction of $(\mathbb{S}, s)$ from $\mathbb{A}$.

Towards Universal Automata Theory

Question: Which Λ have an EXPTIME nonemptiness problem?

Towards Universal Automata Theory

Question: Which Λ have an EXPTIME nonemptiness problem?

Question: What is the expressive power of various languages of the form $\mu\mathrm{ML}_\Lambda$?

Towards Universal Automata Theory

Question: Which Λ have an EXPTIME nonemptiness problem?

Question: What is the expressive power of various languages of the form $\mu\mathrm{ML}_{\Lambda}$?

Question: What do concepts like nondeterminism mean in this setting?

Towards Universal Automata Theory

Question: Which Λ have an EXPTIME nonemptiness problem?

Question: What is the expressive power of various languages of the form μML_Λ?

Question: What do concepts like nondeterminism mean in this setting?

Question: Can we naturally extend coalgebraic automata theory to non-set-based coalgebras?

Towards Universal Automata Theory

Question: Which Λ have an EXPTIME nonemptiness problem?

Question: What is the expressive power of various languages of the form $\mu\mathrm{ML}_\Lambda$?

Question: What do concepts like nondeterminism mean in this setting?

Question: Can we naturally extend coalgebraic automata theory to non-set-based coalgebras?

. . .

Overview

- Logic & Automata
- Coalgebra
- Coalgebra Automata
- Towards Universal Automata Theory
- Final remarks

Further Coalgebraic Inspiration

Further Coalgebraic Inspiration

- Model theory of modal μ-calculus (joint work with Gaëlle Fontaine)

Further Coalgebraic Inspiration

- Model theory of modal μ-calculus (joint work with Gaëlle Fontaine)
- Axiomatics of modal fixpoint logics (joint work with Luigi Santocanale)

Conclusions

Conclusions

- Coalgebra provides a sensible abstraction level for automata theory.

Conclusions

- Coalgebra provides a sensible abstraction level for automata theory.
- Our understanding of modal fixpoint logics may benefit from a coalgebraic perspective.

Research directions

Research directions

- universal automata theory

Research directions

- universal automata theory
- completeness theory for coalgebraic fixpoint logics.

Research directions

- universal automata theory
- completeness theory for coalgebraic fixpoint logics.
- automata and fixpoints in algebra|coalgebra duality.

References

- G. Fontaine. Continuous fragment of the mu-calculus. Proceedings CSL, 2008.
- C. Kissig and Y. Venema. Complementation for coalgebra automata. In *Proceedings CALCO 2009*, pp 81-96, Springer LNCS 5728.
- C. Kupke and Y. Venema. Coalgebraic automata theory: basic results. *Logical Methods in Computer Science*, 2008.
- C. Cîrstea, C. Kupke and D. Pattinson. EXPTIME Tableaux for Coalgebraic mu-calculi. Proceedings CSL 2009.
- Y. Venema. Algebras and coalgebras. In P. Blackburn, J. van Benthem, and F. Wolter, editors, *Handbook of Modal Logic*, pp 331–426. Elsevier, 2006.
- Y. Venema. Automata and fixed point logic: a coalgebraic perspective. *Information and Computation*, 204:637–678, 2006.
- L. Santocanale and Y. Venema. Completeness for Flat Modal Fixpoint Logics, submitted. (Abstract available: Proceedings LPAR 2007, pp. 499-513, Springer LNCS 4790).