Axiomatizing modal fixpoint logics

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

SYSMICS, 8 september 2016

(largely joint work with Enqvist, Seifan, Santocanale, Schröder, ...)

Overview

- Introduction
- Obstacles
- A general result
- A general framework
- Frame conditions
- Conclusions

Example

- Add master modality $\langle * \rangle$ to the language ML of modal logic
- $\langle * \rangle p := \bigvee_{n \in \omega} \Diamond^n p$
  $s \Vdash \langle * \rangle p$ iff there is a finite path from s to some p-state

Example

- Add master modality $\langle * \rangle$ to the language ML of modal logic
- $\langle * \rangle p := \bigvee_{n \in \omega} \Diamond^n p$
  $s \Vdash \langle * \rangle p$ iff there is a finite path from s to some p-state
- $\langle * \rangle p \leftrightarrow p \vee \Diamond \langle * \rangle p$

Example

- Add master modality $\langle * \rangle$ to the language ML of modal logic
- $\langle * \rangle p := \bigvee_{n \in \omega} \Diamond^n p$
  $s \Vdash \langle * \rangle p$ iff there is a finite path from s to some p-state
- $\langle * \rangle p \leftrightarrow p \vee \Diamond \langle * \rangle p$
- **Fact** $\langle * \rangle p$ is the least fixpoint of the 'equation' $x \leftrightarrow p \vee \Diamond x$

Example

- Add master modality $\langle * \rangle$ to the language ML of modal logic
- $\langle * \rangle p := \bigvee_{n \in \omega} \Diamond^n p$
  $s \Vdash \langle * \rangle p$ iff there is a finite path from s to some p-state
- $\langle * \rangle p \leftrightarrow p \vee \Diamond \langle * \rangle p$
- **Fact** $\langle * \rangle p$ is the least fixpoint of the 'equation' $x \leftrightarrow p \vee \Diamond x$
- Notation: $\langle * \rangle p \equiv \mu x . p \vee \Diamond x$.

Example

- Add master modality $\langle * \rangle$ to the language ML of modal logic
- $\langle * \rangle p := \bigvee_{n \in \omega} \Diamond^n p$
  $s \Vdash \langle * \rangle p$ iff there is a finite path from s to some p-state
- $\langle * \rangle p \leftrightarrow p \vee \Diamond \langle * \rangle p$
- **Fact** $\langle * \rangle p$ is the least fixpoint of the 'equation' $x \leftrightarrow p \vee \Diamond x$
- Notation: $\langle * \rangle p \equiv \mu x. p \vee \Diamond x$.
- Variant (PDL): $\langle \alpha^* \rangle \varphi := \mu x. \varphi \vee \langle \alpha \rangle x$

More examples

- $U\varphi\psi \equiv \varphi \vee (\psi \wedge \bigcirc U\varphi\psi)$

More examples

- $U\varphi\psi \equiv \varphi \vee (\psi \wedge \bigcirc U\varphi\psi)$
  $U\varphi\psi := \mu x.\varphi \vee (\psi \wedge \bigcirc x)$

More examples

- $U\varphi\psi \equiv \varphi \vee (\psi \wedge \bigcirc U\varphi\psi)$
  $U\varphi\psi := \mu x.\varphi \vee (\psi \wedge \bigcirc x)$
- $C\varphi := \varphi \wedge \bigwedge_i K_i\varphi \wedge \bigwedge_i K_i C(\bigwedge_i K_i\varphi) \wedge \dots$

More examples

- $U\varphi\psi \equiv \varphi \vee (\psi \wedge \bigcirc U\varphi\psi)$
  $U\varphi\psi := \mu x.\varphi \vee (\psi \wedge \bigcirc x)$
- $C\varphi := \varphi \wedge \bigwedge_i K_i\varphi \wedge \bigwedge_i K_i C(\bigwedge_i K_i\varphi) \wedge \ldots$
  $C\varphi \equiv \varphi \wedge \bigwedge_i K_i C\varphi$

More examples

- $U\varphi\psi \equiv \varphi \vee (\psi \wedge \bigcirc U\varphi\psi)$
  $U\varphi\psi := \mu x.\varphi \vee (\psi \wedge \bigcirc x)$
- $C\varphi := \varphi \wedge \bigwedge_i K_i\varphi \wedge \bigwedge_i K_i C(\bigwedge_i K_i\varphi) \wedge \dots$
  $C\varphi \equiv \varphi \wedge \bigwedge_i K_i C\varphi$
  $C\varphi := \nu x.\varphi \wedge \bigwedge_i K_i x$

Modal Fixpoint Logics

- Modal fixpoint languages extend basic modal logic with either
  - new fixpoint connectives such as $\langle * \rangle$, U, C, ...

Modal Fixpoint Logics

- Modal fixpoint languages extend basic modal logic with either
  - new fixpoint connectives such as $\langle * \rangle$, U, C, $\ldots \rightsquigarrow$ LTL, CTL, PDL
  - explicit fixpoint operators μx, νx

Modal Fixpoint Logics

- Modal fixpoint languages extend basic modal logic with either
  - new fixpoint connectives such as $\langle * \rangle$, U, C, $\ldots \rightsquigarrow$ LTL, CTL, PDL
  - explicit fixpoint operators μx, $\nu x \rightsquigarrow \mu$ML

Modal Fixpoint Logics

- Modal fixpoint languages extend basic modal logic with either
  - new fixpoint connectives such as $\langle * \rangle$, U, C, $\ldots \rightsquigarrow$ LTL, CTL, PDL
  - explicit fixpoint operators μx, $\nu x \rightsquigarrow \mu$ML
- Motivation 1: increase expressive power
  - e.g. enable specification of ongoing behaviour

Modal Fixpoint Logics

- Modal fixpoint languages extend basic modal logic with either
  - new fixpoint connectives such as $\langle * \rangle$, U, C, $\ldots \rightsquigarrow$ LTL, CTL, PDL
  - explicit fixpoint operators μx, $\nu x \rightsquigarrow \mu$ML
- Motivation 1: increase expressive power
  - e.g. enable specification of ongoing behaviour
- Motivation 2: generally nice computational properties

Modal Fixpoint Logics

- Modal fixpoint languages extend basic modal logic with either
  - new fixpoint connectives such as $\langle * \rangle$, U, C, $\ldots \rightsquigarrow$ LTL, CTL, PDL
  - explicit fixpoint operators μx, $\nu x \rightsquigarrow \mu$ML
- Motivation 1: increase expressive power
  - e.g. enable specification of ongoing behaviour
- Motivation 2: generally nice computational properties
- Combined: many applications in process theory, epistemic logic, ...

Modal Fixpoint Logics

- Modal fixpoint languages extend basic modal logic with either
  - new fixpoint connectives such as $\langle * \rangle$, U, C, $\ldots \rightsquigarrow$ LTL, CTL, PDL
  - explicit fixpoint operators μx, $\nu x \rightsquigarrow \mu$ML
- Motivation 1: increase expressive power
  - e.g. enable specification of ongoing behaviour
- Motivation 2: generally nice computational properties
- Combined: many applications in process theory, epistemic logic, ...
- Interesting mathematical theory:
  - interesting mix of algebraic|coalgebraic features
  - connections with theory of automata on infinite objects
  - game-theoretical semantics
  - interesting meta-logic

General Program

Understand modal fixpoint logics by studying the interaction between

- combinatorial
- algebraic and
- coalgebraic

aspects

Here: consider axiomatization problem

Axiomatization of fixpoints

Least fixpoint $\mu p.\varphi$ should be axiomatized by

Axiomatization of fixpoints

Least fixpoint $\mu p.\varphi$ should be axiomatized by

- a least (pre-)fixpoint axiom:

$$\varphi(\mu p.\varphi) \vdash \mu p.\varphi$$

- Park's induction rule

$$\frac{\varphi(\psi) \vdash \varphi}{\mu p.\varphi \vdash \psi}$$

(Here $\alpha \vdash_K \beta$ abbreviates $\vdash_K \alpha \to \beta$)

Axiomatization results for modal fixpoint logics

- LTL: Gabbay et alii (1980)
- PDL: Kozen & Parikh (1981)
- μML (aconjunctive fragment): Kozen (1983)
- CTL: Emerson & Halpern (1985)
- μML: Walukiewicz (1993/2000)
- CTL*: Reynolds (2000)
- LTL/CTL uniformly: Lange & Stirling (2001)
- common knowledge logics: various
- ...

Axiomatization results for modal fixpoint logics

- LTL: Gabbay et alii (1980)
- PDL: Kozen & Parikh (1981)
- μML (aconjunctive fragment): Kozen (1983)
- CTL: Emerson & Halpern (1985)
- μML: Walukiewicz (1993/2000)
- CTL*: Reynolds (2000)
- LTL/CTL uniformly: Lange & Stirling (2001)
- common knowledge logics: various
- ...

So what is the problem?

Axiomatization problem

Questions (2015)

- How to generalise these results to restricted frame classes?
- How to generalise results to similar logics, eg, the monotone μ-calculus?
- Does completeness transfer to fragments of μML? (Ex: game logic)
- What about proof theory?

Axiomatization problem

Questions (2015)

- How to generalise these results to restricted frame classes?
- How to generalise results to similar logics, eg, the monotone μ-calculus?
- Does completeness transfer to fragments of μML? (Ex: game logic)
- What about proof theory?

Compared to basic modal logic

- there are no sweeping general results such as Sahlqvist's theorem

Axiomatization problem

Questions (2015)

- How to generalise these results to restricted frame classes?
- How to generalise results to similar logics, eg, the monotone μ-calculus?
- Does completeness transfer to fragments of μML? (Ex: game logic)
- What about proof theory?

Compared to basic modal logic

- there are no sweeping general results such as Sahlqvist's theorem
- there is no no comprehensive completeness theory (duality, canonicity, filtration, . . .)

Overview

- Introduction
- Obstacles
- A general result
- A general framework
- Frame conditions
- Conclusions

Overview

- Introduction
- Obstacles
- A general result
- A general framework
- Frame conditions
- Conclusions

Obstacle 1: computational danger zone

Example

Obstacle 1: computational danger zone

Example

- Language: $\Diamond_R, \Diamond_U$

Obstacle 1: computational danger zone

Example

- Language: $\Diamond_R, \Diamond_U$
- Intended Semantics: $\mathbb{N} \times \mathbb{N}$
  - $(m, n)R(m', n')$ iff $m' = m + 1$ and $n' = n$
  - $(m, n)U(m', n')$ iff $m' = m$ and $n' = n + 1$

Obstacle 1: computational danger zone

Example

- Language: $\Diamond_R, \Diamond_U$
- Intended Semantics: $\mathbb{N} \times \mathbb{N}$
  - $(m,n)R(m',n')$ iff $m' = m+1$ and $n' = n$
  - $(m,n)U(m',n')$ iff $m' = m$ and $n' = n+1$
- Logic $\mathbf{K}G := \mathbf{K} +$
  - functionality: $\Diamond_R p \leftrightarrow \Box_R p$ and $\Diamond_U p \leftrightarrow \Box_U p$
  - confluence: $\Diamond_R \Box_U p \to \Box_U \Diamond_R p$

Obstacle 1: computational danger zone

Example

- Language: $\Diamond_R, \Diamond_U$
- Intended Semantics: $\mathbb{N} \times \mathbb{N}$
  - $(m, n)R(m', n')$ iff $m' = m + 1$ and $n' = n$
  - $(m, n)U(m', n')$ iff $m' = m$ and $n' = n + 1$
- Logic $\mathbf{K}G := \mathbf{K} +$
  - functionality: $\Diamond_R p \leftrightarrow \Box_R p$ and $\Diamond_U p \leftrightarrow \Box_U p$
  - confluence: $\Diamond_R \Box_U p \to \Box_U \Diamond_R p$
- $\mathbf{K}G$ is sound and complete with respect to its Kripke frames

Obstacle 1: computational danger zone

Example

- Language: $\Diamond_R, \Diamond_U$
- Intended Semantics: $\mathbb{N} \times \mathbb{N}$
  - $(m, n)R(m', n')$ iff $m' = m + 1$ and $n' = n$
  - $(m, n)U(m', n')$ iff $m' = m$ and $n' = n + 1$
- Logic $\mathbf{K}G := \mathbf{K} +$
  - functionality: $\Diamond_R p \leftrightarrow \Box_R p$ and $\Diamond_U p \leftrightarrow \Box_U p$
  - confluence: $\Diamond_R \Box_U p \to \Box_U \Diamond_R p$
- $\mathbf{K}G$ is sound and complete with respect to its Kripke frames
- Add master modality, $\langle * \rangle p := \mu x.p \vee \Diamond_R x \vee \Diamond_U x$

Obstacle 1: computational danger zone

Example

- Language: $\Diamond_R, \Diamond_U$
- Intended Semantics: $\mathbb{N} \times \mathbb{N}$
  - $(m,n)R(m',n')$ iff $m' = m+1$ and $n' = n$
  - $(m,n)U(m',n')$ iff $m' = m$ and $n' = n+1$
- Logic $\mathbf{K}G := \mathbf{K} +$
  - functionality: $\Diamond_R p \leftrightarrow \Box_R p$ and $\Diamond_U p \leftrightarrow \Box_U p$
  - confluence: $\Diamond_R \Box_U p \to \Box_U \Diamond_R p$
- $\mathbf{K}G$ is sound and complete with respect to its Kripke frames
- Add master modality, $\langle * \rangle p := \mu x.p \vee \Diamond_R x \vee \Diamond_U x$
- $\mu\mathbf{K}G$ is sound but incomplete with respect to its Kripke frames
  - **Proof:**

Obstacle 1: computational danger zone

Example

- Language: $\Diamond_R, \Diamond_U$
- Intended Semantics: $\mathbb{N} \times \mathbb{N}$
  - $(m, n)R(m', n')$ iff $m' = m + 1$ and $n' = n$
  - $(m, n)U(m', n')$ iff $m' = m$ and $n' = n + 1$
- Logic $\mathbf{K}G := \mathbf{K} +$
  - functionality: $\Diamond_R p \leftrightarrow \Box_R p$ and $\Diamond_U p \leftrightarrow \Box_U p$
  - confluence: $\Diamond_R \Box_U p \rightarrow \Box_U \Diamond_R p$
- $\mathbf{K}G$ is sound and complete with respect to its Kripke frames
- Add master modality, $\langle * \rangle p := \mu x.p \vee \Diamond_R x \vee \Diamond_U x$
- $\mu\mathbf{K}G$ is sound but incomplete with respect to its Kripke frames
  - **Proof:** Use recurrent tiling problem to show that

Obstacle 1: computational danger zone

Example

- Language: $\Diamond_R, \Diamond_U$
- Intended Semantics: $\mathbb{N} \times \mathbb{N}$
  - $(m, n)R(m', n')$ iff $m' = m + 1$ and $n' = n$
  - $(m, n)U(m', n')$ iff $m' = m$ and $n' = n + 1$
- Logic $\mathbf{K}G := \mathbf{K} +$
  - functionality: $\Diamond_R p \leftrightarrow \Box_R p$ and $\Diamond_U p \leftrightarrow \Box_U p$
  - confluence: $\Diamond_R \Box_U p \to \Box_U \Diamond_R p$
- $\mathbf{K}G$ is sound and complete with respect to its Kripke frames
- Add master modality, $\langle * \rangle p := \mu x.p \vee \Diamond_R x \vee \Diamond_U x$
- $\mu\mathbf{K}G$ is sound but incomplete with respect to its Kripke frames
  - **Proof:** Use recurrent tiling problem to show that
  - the $\Diamond_R, \Diamond_U, \langle * \rangle$-logic of $Fr(\mathbf{K}G)$ is not recursively enumerable

Obstacle 2: compactness failure

- Example: $\langle * \rangle p := \bigvee_{n \in \omega} \Diamond^n p$
  - $\{\langle * \rangle p\} \cup \{\Box^n \neg p \mid n \in \omega\}$ is finitely satisfiable but not satisfiable

Obstacle 2: compactness failure

- Example: $\langle * \rangle p := \bigvee_{n \in \omega} \Diamond^n p$
  - $\{\langle * \rangle p\} \cup \{\Box^n \neg p \mid n \in \omega\}$ is finitely satisfiable but not satisfiable
- Fixpoint logics have no nice Stone-based duality

Obstacle 3: fixpoint alternation

- tableaux: fixpoint unfolding
  - ν-fixpoints may be unfolded infinitely often
  - μ-fixpoints may only be unfolded finitely often

Obstacle 3: fixpoint alternation

- tableaux: fixpoint unfolding
  - ν-fixpoints may be unfolded infinitely often
  - μ-fixpoints may only be unfolded finitely often
- with every branch of tableau associate a trace graph

Obstacle 3: fixpoint alternation

- tableaux: fixpoint unfolding
  - ν-fixpoints may be unfolded infinitely often
  - μ-fixpoints may only be unfolded finitely often
- with every branch of tableau associate a trace graph
- obstacle 3a: conjunctions cause trace proliferation

Obstacle 3: fixpoint alternation

- tableaux: fixpoint unfolding
  - ν-fixpoints may be unfolded infinitely often
  - μ-fixpoints may only be unfolded finitely often
- with every branch of tableau associate a trace graph
- obstacle 3a: conjunctions cause trace proliferation
- obstacle 3b: fixpoint alternations cause intricate combinatorics

What to do?

What to do?

- consider simple frame conditions only (if at all)

What to do?

- consider simple frame conditions only (if at all)
- restrict language to fixpoints of simple formulas (avoid alternation)

What to do?

- consider simple frame conditions only (if at all)
- restrict language to fixpoints of simple formulas (avoid alternation)
- allow alternation, but develop suitable combinatorical framework

Overview

- Introduction
- Obstacles
- A general result
- A general framework
- Frame conditions
- Conclusions

Flat Modal Fixpoint Logics: Syntax

- Fix a basic modal formula $\gamma(x, \vec{p})$, positive in x

Flat Modal Fixpoint Logics: Syntax

- Fix a basic modal formula $\gamma(x, \vec{p})$, positive in x
- Add a fixpoint connective $\sharp_\gamma$ to the language of ML
  (arity of $\sharp_\gamma$ depends on γ but notation hides this)

Flat Modal Fixpoint Logics: Syntax

- Fix a basic modal formula $\gamma(x, \vec{p})$, positive in x
- Add a fixpoint connective $\sharp_\gamma$ to the language of ML
  (arity of $\sharp_\gamma$ depends on γ but notation hides this)
- Example: $Upq := \mu x.p \vee (q \wedge \bigcirc x)$,
  now: $Upq := \sharp_\gamma(p, q)$ with $\gamma = p \vee (q \wedge \bigcirc x)$
- Intended reading: $\sharp_\gamma(\vec{\varphi}) \equiv \mu x.\gamma(x, \vec{\varphi})$ for any $\vec{\varphi} = (\varphi_1, \ldots, \varphi_n)$.

Flat Modal Fixpoint Logics: Syntax

- Fix a basic modal formula $\gamma(x, \vec{p})$, positive in x
- Add a fixpoint connective $\sharp_\gamma$ to the language of ML
  (arity of $\sharp_\gamma$ depends on γ but notation hides this)
- Example: $Upq := \mu x.p \vee (q \wedge \bigcirc x)$,
  now: $Upq := \sharp_\gamma(p, q)$ with $\gamma = p \vee (q \wedge \bigcirc x)$
- Intended reading: $\sharp_\gamma(\vec{\varphi}) \equiv \mu x.\gamma(x, \vec{\varphi})$ for any $\vec{\varphi} = (\varphi_1, \ldots, \varphi_n)$.
- Obtain language ML_γ:

$$\varphi \ ::= \ p \mid \neg p \mid \bot \mid \top \mid \varphi_1 \vee \varphi_2 \mid \varphi_1 \wedge \varphi_2 \mid \Diamond_i \varphi \mid \Box_i \varphi \mid \sharp_\gamma(\vec{\varphi})$$

Flat Modal Fixpoint Logics: Syntax

- Fix a basic modal formula $\gamma(x, \vec{p})$, positive in x
- Add a fixpoint connective $\sharp_\gamma$ to the language of ML
  (arity of $\sharp_\gamma$ depends on γ but notation hides this)
- Example: $Upq := \mu x.p \vee (q \wedge \bigcirc x)$,
  now: $Upq := \sharp_\gamma(p, q)$ with $\gamma = p \vee (q \wedge \bigcirc x)$
- Intended reading: $\sharp_\gamma(\vec{\varphi}) \equiv \mu x.\gamma(x, \vec{\varphi})$ for any $\vec{\varphi} = (\varphi_1, \ldots, \varphi_n)$.
- Obtain language ML_γ:

$$\varphi \ ::= \ p \mid \neg p \mid \bot \mid \top \mid \varphi_1 \vee \varphi_2 \mid \varphi_1 \wedge \varphi_2 \mid \Diamond_i \varphi \mid \Box_i \varphi \mid \sharp_\gamma(\vec{\varphi})$$

- Examples: CTL, LTL, (PDL), ...

Flat Modal Fixpoint Logics: Kripke Semantics

- Kripke frame $\mathbb{S} = \langle S, R \rangle$ with $R \subseteq S \times S$.
- Complex algebra: $\mathbb{S}^+ := \langle \wp(S), \varnothing, S, \sim_S, \cup, \cap, \langle R \rangle \rangle$,
  $\langle R \rangle : \wp(S) \to \wp(S)$ given by
  $\langle R \rangle(X) := \{ s \in S \mid Rst \text{ for some } t \in X \}$

Flat Modal Fixpoint Logics: Kripke Semantics

- Kripke frame $\mathbb{S} = \langle S, R \rangle$ with $R \subseteq S \times S$.
- Complex algebra: $\mathbb{S}^+ := \langle \wp(S), \varnothing, S, \sim_S, \cup, \cap, \langle R \rangle \rangle$,
  $\langle R \rangle : \wp(S) \to \wp(S)$ given by
  $\langle R \rangle (X) := \{ s \in S \mid Rst \text{ for some } t \in X \}$
- Every modal formula $\varphi(p_1, \ldots, p_n)$ corresponds to a term function

$$\varphi^{\mathbb{S}} : \wp(S)^n \to \wp(S).$$

- γ positive in x, hence $\gamma^{\mathbb{S}}$ order preserving in x.

Flat Modal Fixpoint Logics: Kripke Semantics

- Kripke frame $\mathbb{S} = \langle S, R \rangle$ with $R \subseteq S \times S$.
- Complex algebra: $\mathbb{S}^+ := \langle \wp(S), \varnothing, S, \sim_S, \cup, \cap, \langle R \rangle \rangle$,
  $\langle R \rangle : \wp(S) \to \wp(S)$ given by
  $\langle R \rangle(X) := \{s \in S \mid Rst \text{ for some } t \in X\}$
- Every modal formula $\varphi(p_1, \ldots, p_n)$ corresponds to a term function

$$\varphi^{\mathbb{S}} : \wp(S)^n \to \wp(S).$$

- γ positive in x, hence $\gamma^{\mathbb{S}}$ order preserving in x.
- By Knaster-Tarski we may define $\sharp^{\mathbb{S}} : \wp(S)^n \to \wp(S)$ by

$$\sharp^{\mathbb{S}}(\vec{B}) := LFP.\gamma^{\mathbb{S}}(-, \vec{B}).$$

Flat Modal Fixpoint Logics: Kripke Semantics

- Kripke frame $\mathbb{S} = \langle S, R \rangle$ with $R \subseteq S \times S$.
- Complex algebra: $\mathbb{S}^+ := \langle \wp(S), \varnothing, S, \sim_S, \cup, \cap, \langle R \rangle \rangle$,
  $\langle R \rangle : \wp(S) \to \wp(S)$ given by
  $\langle R \rangle(X) := \{ s \in S \mid Rst \text{ for some } t \in X \}$
- Every modal formula $\varphi(p_1, \ldots, p_n)$ corresponds to a term function

$$\varphi^{\mathbb{S}} : \wp(S)^n \to \wp(S).$$

- γ positive in x, hence $\gamma^{\mathbb{S}}$ order preserving in x.
- By Knaster-Tarski we may define $\sharp^{\mathbb{S}} : \wp(S)^n \to \wp(S)$ by

$$\sharp^{\mathbb{S}}(\vec{B}) := LFP.\gamma^{\mathbb{S}}(-, \vec{B}).$$

- Kripke $\sharp$-algebra $\mathbb{S}^\sharp := \langle \wp(S), \varnothing, S, \sim_S, \cup, \cap, \langle R \rangle, \sharp^{\mathbb{S}} \rangle$.

Candidate Axiomatization

$\mathbf{K}_\gamma$:= **K** extended with

- prefixpoint axiom:
$$\gamma(\sharp(\vec{\varphi}), \vec{\varphi}) \vdash \sharp(\vec{\varphi})$$
- Park's induction rule:
$$\text{from } \gamma(\psi, \vec{\varphi}) \vdash \psi \text{ infer } \sharp_\gamma(\vec{\varphi}) \vdash \psi.$$

Flat Modal Fixpoint Logics: Algebraic completeness proof

Flat Modal Fixpoint Logics: Algebraic completeness proof

- Modal $\sharp$-algebra: $A = \langle A, \bot, \top, \neg, \wedge, \vee, \Diamond, \sharp \rangle$ with $\sharp : A^n \to A$ satisfying

$$\sharp(\vec{b}) = LFP.\gamma^A_{\vec{b}},$$

where $\gamma^A_{\vec{b}} : A \to A$ is given by $\gamma^A_{\vec{b}}(a) := \gamma^A(a, \vec{b})$.

Flat Modal Fixpoint Logics: Algebraic completeness proof

- Modal $\sharp$-algebra: $A = \langle A, \bot, \top, \neg, \wedge, \vee, \Diamond, \sharp \rangle$ with $\sharp : A^n \to A$ satisfying
$$\sharp(\vec{b}) = LFP.\gamma^A_{\vec{b}},$$
where $\gamma^A_{\vec{b}} : A \to A$ is given by $\gamma^A_{\vec{b}}(a) := \gamma^A(a, \vec{b})$.
- Axiomatically: modal $\sharp$-algebras satisfy
  - $\gamma(\sharp(\vec{y}), \vec{y}) \leq \sharp(\vec{y})$
  - if $\gamma(x, \vec{y}) \leq x$ then $\sharp(\vec{y}) \leq x$.
- Completeness for flat fixpoint logics: $\mathsf{Equ}(\mathsf{MA}_\sharp) \stackrel{?}{=} \mathsf{Equ}(\mathsf{KA}_\sharp)$

Flat Modal Fixpoint Logics: Algebraic completeness proof

- Modal $\sharp$-algebra: $A = \langle A, \bot, \top, \neg, \wedge, \vee, \Diamond, \sharp\rangle$ with $\sharp : A^n \to A$ satisfying
$$\sharp(\vec{b}) = LFP.\gamma^A_{\vec{b}},$$
where $\gamma^A_{\vec{b}} : A \to A$ is given by $\gamma^A_{\vec{b}}(a) := \gamma^A(a, \vec{b})$.
- Axiomatically: modal $\sharp$-algebras satisfy
  - $\gamma(\sharp(\vec{y}), \vec{y}) \leq \sharp(\vec{y})$
  - if $\gamma(x, \vec{y}) \leq x$ then $\sharp(\vec{y}) \leq x$.
- Completeness for flat fixpoint logics: $\text{Equ}(\text{MA}_\sharp) \stackrel{?}{=} \text{Equ}(\text{KA}_\sharp)$
- Two key concepts:

Flat Modal Fixpoint Logics: Algebraic completeness proof

- Modal $\sharp$-algebra: $A = \langle A, \bot, \top, \neg, \wedge, \vee, \Diamond, \sharp \rangle$ with $\sharp : A^n \to A$ satisfying
$$\sharp(\vec{b}) = LFP.\gamma^A_{\vec{b}},$$
where $\gamma^A_{\vec{b}} : A \to A$ is given by $\gamma^A_{\vec{b}}(a) := \gamma^A(a, \vec{b})$.
- Axiomatically: modal $\sharp$-algebras satisfy
  - $\gamma(\sharp(\vec{y}), \vec{y}) \leq \sharp(\vec{y})$
  - if $\gamma(x, \vec{y}) \leq x$ then $\sharp(\vec{y}) \leq x$.
- Completeness for flat fixpoint logics: $\mathsf{Equ}(\mathsf{MA}_\sharp) \stackrel{?}{=} \mathsf{Equ}(\mathsf{KA}_\sharp)$
- Two key concepts:
  - constructiveness
  - $\mathcal{O}$-adjointness

Constructiveness

- An $\mathrm{MA}_\sharp$-algebra $\mathbb{A}$ is constructive if

$$\sharp(\vec{b}) = \bigvee_{n \in \omega} \gamma_{\vec{b}}^{n}(\bot).$$

Constructiveness

- An $\mathrm{MA}_\sharp$-algebra $\mathbb{A}$ is constructive if

$$\sharp(\vec{b}) = \bigvee_{n \in \omega} \gamma_{\vec{b}}^{n}(\bot).$$

Note: we do not require $\mathbb{A}$ to be complete!

Constructiveness

- An $\mathrm{MA}_\sharp$-algebra $\mathbb{A}$ is constructive if

$$\sharp(\vec{b}) = \bigvee_{n \in \omega} \gamma_{\vec{b}}^{n}(\bot).$$

Note: we do not require $\mathbb{A}$ to be complete!

Theorem (Santocanale & Venema)
Let A be a countable, residuated, modal $\sharp$-algebra.
If A is constructive, then A can be embedded in a Kripke $\sharp$-algebra.

Constructiveness

- An $\mathrm{MA}_\sharp$-algebra $\mathbb{A}$ is constructive if

$$\sharp(\vec{b}) = \bigvee_{n\in\omega} \gamma^n_{\vec{b}}(\bot).$$

  Note: we do not require $\mathbb{A}$ to be complete!

Theorem (Santocanale & Venema)
Let A be a countable, residuated, modal $\sharp$-algebra.
If A is constructive, then A can be embedded in a Kripke $\sharp$-algebra.

Proof
Via a step-by-step construction/generalized Lindenbaum Lemma.
Alternatively, use Rasiowa-Sikorski Lemma.

$\mathcal{O}$-adjoints

Let $f : (P, \leq) \to (Q, \leq)$ be an order-preserving map.

$\mathcal{O}$-adjoints

Let $f : (P, \leq) \to (Q, \leq)$ be an order-preserving map.

- f is a (left) adjoint or residuated if it has a residual $g : Q \to P$ with

$$fp \leq q \iff p \leq gq.$$

$\mathcal{O}$-adjoints

Let $f : (P, \leq) \to (Q, \leq)$ be an order-preserving map.

- f is a (left) adjoint or residuated if it has a residual $g : Q \to P$ with

$$fp \leq q \iff p \leq gq.$$

- f is a (left) $\mathcal{O}$-adjoint if it has an $\mathcal{O}$-residual $G_f : Q \to \wp_\omega(P)$ with

$$fp \leq q \iff p \leq y \text{ for some } y \in G_f q.$$

$\mathcal{O}$-adjoints

Let $f : (P, \leq) \to (Q, \leq)$ be an order-preserving map.

- f is a (left) adjoint or residuated if it has a residual $g : Q \to P$ with

$$fp \leq q \iff p \leq gq.$$

- f is a (left) $\mathcal{O}$-adjoint if it has an $\mathcal{O}$-residual $G_f : Q \to \wp_\omega(P)$ with

$$fp \leq q \iff p \leq y \text{ for some } y \in G_f q.$$

Proposition (Santocanale 2005)

- f is a left adjoint iff f is a join-preserving $\mathcal{O}$-adjoint

$\mathcal{O}$-adjoints

Let $f : (P, \leq) \to (Q, \leq)$ be an order-preserving map.

- f is a (left) adjoint or residuated if it has a residual $g : Q \to P$ with

$$fp \leq q \iff p \leq gq.$$

- f is a (left) $\mathcal{O}$-adjoint if it has an $\mathcal{O}$-residual $G_f : Q \to \wp_\omega(P)$ with

$$fp \leq q \iff p \leq y \text{ for some } y \in G_f q.$$

Proposition (Santocanale 2005)

- f is a left adjoint iff f is a join-preserving $\mathcal{O}$-adjoint
- $\mathcal{O}$-adjoints are Scott continuous

$\mathcal{O}$-adjoints

Let $f : (P, \leq) \to (Q, \leq)$ be an order-preserving map.

- f is a (left) adjoint or residuated if it has a residual $g : Q \to P$ with

$$fp \leq q \iff p \leq gq.$$

- f is a (left) $\mathcal{O}$-adjoint if it has an $\mathcal{O}$-residual $G_f : Q \to \wp_\omega(P)$ with

$$fp \leq q \iff p \leq y \text{ for some } y \in G_f q.$$

Proposition (Santocanale 2005)

- f is a left adjoint iff f is a join-preserving $\mathcal{O}$-adjoint
- $\mathcal{O}$-adjoints are Scott continuous
- $\wedge$ is continuous but not an $\mathcal{O}$-adjoint.

Finitary $\mathcal{O}$-adjoints

Let $f : A^n \to A$ be an $\mathcal{O}$-adjoint with $\mathcal{O}$-residual G.

Finitary $\mathcal{O}$-adjoints

Let $f : A^n \to A$ be an $\mathcal{O}$-adjoint with $\mathcal{O}$-residual G.

- Inductively define $G^n : A \to \wp(A)$

$$\begin{aligned} G^0(a) &:= \{a\} \\ G^{n+1}(a) &:= G[G^n(a)] \end{aligned}$$

Finitary $\mathcal{O}$-adjoints

Let $f : A^n \to A$ be an $\mathcal{O}$-adjoint with $\mathcal{O}$-residual G.

- Inductively define $G^n : A \to \wp(A)$

$$\begin{aligned} G^0(a) &:= \{a\} \\ G^{n+1}(a) &:= G[G^n(a)] \end{aligned}$$

- Call f finitary if $G^\omega(a) := \bigcup_{n\in\omega} G^n(a)$ is finite.

Finitary $\mathcal{O}$-adjoints

Let $f : A^n \to A$ be an $\mathcal{O}$-adjoint with $\mathcal{O}$-residual G.

- Inductively define $G^n : A \to \wp(A)$

$$\begin{array}{rcl} G^0(a) & := & \{a\} \\ G^{n+1}(a) & := & G[G^n(a)] \end{array}$$

- Call f finitary if $G^\omega(a) := \bigcup_{n\in\omega} G^n(a)$ is finite.

Theorem (Santocanale 2005)
If $f : A \to A$ is a finitary $\mathcal{O}$-adjoint, then $LFP.f$, if existing, is constructive.

Adjoints on free algebras

Adjoints on free algebras

- Free modal ($\sharp$-)algebras have many $\mathcal{O}$-adjoints!

Adjoints on free algebras

- Free modal ($\sharp$-)algebras have many $\mathcal{O}$-adjoints!
  - cf. free distributive lattice are Heyting algebras,

Adjoints on free algebras

- Free modal ($\sharp$-)algebras have many $\mathcal{O}$-adjoints!
  - cf. free distributive lattice are Heyting algebras,
  - Whitman's rule for free lattices, ...

Adjoints on free algebras

- Free modal ($\sharp$-)algebras have many $\mathcal{O}$-adjoints!
  - cf. free distributive lattice are Heyting algebras,
  - Whitman's rule for free lattices, ...
- Call a modal formula γ untied in x if it belongs to

$$\gamma \ ::= \ x \ | \ \top \ | \ \gamma \vee \gamma \ | \ \psi \wedge \gamma \ | \ \nabla\{\gamma_1, \dots, \gamma_n\}$$

  where ψ does not contain x

Adjoints on free algebras

- Free modal ($\sharp$-)algebras have many $\mathcal{O}$-adjoints!
  - cf. free distributive lattice are Heyting algebras,
  - Whitman's rule for free lattices, ...
- Call a modal formula γ untied in x if it belongs to

$$\gamma \ ::= \ x \ | \ \top \ | \ \gamma \vee \gamma \ | \ \psi \wedge \gamma \ | \ \nabla\{\gamma_1, \ldots, \gamma_n\}$$

  where ψ does not contain x
  - Examples: $\Diamond x$, $\Box x$, $\Diamond x \wedge \Diamond\Diamond x \wedge \Box p$, $\Diamond x \wedge \Diamond\Box x \wedge \Box(\Diamond x \vee \Diamond\Box x)$, ...

Adjoints on free algebras

- Free modal ($\sharp$-)algebras have many $\mathcal{O}$-adjoints!
  - cf. free distributive lattice are Heyting algebras,
  - Whitman's rule for free lattices, ...
- Call a modal formula γ untied in x if it belongs to

$$\gamma \ ::= \ x \ \mid \ \top \ \mid \ \gamma \vee \gamma \ \mid \ \psi \wedge \gamma \ \mid \ \nabla\{\gamma_1, \ldots, \gamma_n\}$$

  where ψ does not contain x
  - Examples: $\Diamond x$, $\Box x$, $\Diamond x \wedge \Diamond\Diamond x \wedge \Box p$, $\Diamond x \wedge \Diamond\Box x \wedge \Box(\Diamond x \vee \Diamond\Box x)$, ...
  - Counterexamples: $\Diamond(x \wedge \Diamond x)$, $\Diamond x \wedge \Box\Diamond x$

Adjoints on free algebras

- Free modal ($\sharp$-)algebras have many $\mathcal{O}$-adjoints!
  - cf. free distributive lattice are Heyting algebras,
  - Whitman's rule for free lattices, ...
- Call a modal formula γ untied in x if it belongs to

$$\gamma \ ::= \ x \mid \top \mid \gamma \vee \gamma \mid \psi \wedge \gamma \mid \nabla\{\gamma_1, \ldots, \gamma_n\}$$

  where ψ does not contain x
  - Examples: $\Diamond x$, $\Box x$, $\Diamond x \wedge \Diamond\Diamond x \wedge \Box p$, $\Diamond x \wedge \Diamond\Box x \wedge \Box(\Diamond x \vee \Diamond\Box x)$, ...
  - Counterexamples: $\Diamond(x \wedge \Diamond x)$, $\Diamond x \wedge \Box\Diamond x$

Theorem (Santocanale & YV 2010)
Untied formulas are finitary $\mathcal{O}$-adjoints.

A general result

A general result

Theorem (Santocanale & YV 2010)
Let γ be untied wrt x. Then $\mathbf{K}_\gamma$ is sound and complete wrt its Kripke semantics.

A general result

Theorem (Santocanale & YV 2010)
Let γ be untied wrt x. Then $\mathbf{K}_\gamma$ is sound and complete wrt its Kripke semantics.

Notes

A general result

Theorem (Santocanale & YV 2010)
Let γ be untied wrt x. Then $\mathbf{K}_\gamma$ is sound and complete wrt its Kripke semantics.

Notes

- Santocanale & YV have fully general result for extended axiom system.

A general result

Theorem (Santocanale & YV 2010)
Let γ be untied wrt x. Then $\mathbf{K}_\gamma$ is sound and complete wrt its Kripke semantics.

Notes

- Santocanale & YV have fully general result for extended axiom system.
- Schröder & YV have similar results for wider coalgebraic setting.

Overview

- Introduction
- Obstacles
- A general result
- A general framework
- Frame conditions
- Conclusions

The modal μ-calculus

- [+] natural extension of basic modal logic with fixpoint operators
- [+] expressive: LTL, CTL, PDL, CTL*, $\ldots \subseteq \mu$ML
- [+] good computational properties
- [+] nice meta-logical theory

The modal μ-calculus

- [+] natural extension of basic modal logic with fixpoint operators
- [+] expressive: LTL, CTL, PDL, CTL*, $\ldots \subseteq \mu$ML
- [+] good computational properties
- [+] nice meta-logical theory
- [−] hard to understand (nested) fixpoint operators

The modal μ-calculus

- [+] natural extension of basic modal logic with fixpoint operators
- [+] expressive: LTL, CTL, PDL, CTL*, $\ldots \subseteq \mu$ML
- [+] good computational properties
- [+] nice meta-logical theory
- [−] hard to understand (nested) fixpoint operators
- [−] theory of μML isolated from theory of ML

The modal μ-calculus

- $[+]$ natural extension of basic modal logic with fixpoint operators
- $[+]$ expressive: LTL, CTL, PDL, CTL*, $\ldots \subseteq \mu$ML
- $[+]$ good computational properties
- $[+]$ nice meta-logical theory
- $[-]$ hard to understand (nested) fixpoint operators
- $[-]$ theory of μML isolated from theory of ML
  - this applies in particular to the completeness result

The modal μ-calculus

- $[+]$ natural extension of basic modal logic with fixpoint operators
- $[+]$ expressive: LTL, CTL, PDL, CTL*, $\ldots \subseteq \mu$ML
- $[+]$ good computational properties
- $[+]$ nice meta-logical theory
- $[-]$ hard to understand (nested) fixpoint operators
- $[-]$ theory of μML isolated from theory of ML
  - this applies in particular to the completeness result

Most results on μML use automata ...

Logic & Automata

Logic & Automata

Automata in Logic

- long & rich history (Büchi, Rabin, ...)
- mathematically interesting theory
- many practical applications
- automata for μML:
  - Janin & Walukiewicz (1995): μ-automata (nondeterministic)
  - Wilke (2002): modal automata (alternating)

Modal automata

Fix a set X of proposition letters; PX is a set of colours

- A modal automaton is a triple $\mathbb{A} = (A, \Theta, Acc)$, where
  - A is a finite set of states
  - $\Theta : A \times \mathsf{P}X \to 1\mathrm{ML}(A)$ is the transition map
  - $Acc \subseteq A^\omega$ is the acceptance condition

Modal automata

Fix a set X of proposition letters; PX is a set of colours

- A modal automaton is a triple $\mathbb{A} = (A, \Theta, Acc)$, where
  - A is a finite set of states
  - $\Theta : A \times \mathsf{P}X \to 1\mathrm{ML}(A)$ is the transition map
  - $Acc \subseteq A^\omega$ is the acceptance condition
- An initialized automaton is a pair $(\mathbb{A}, a)$ with $a \in A$

Modal automata

Fix a set X of proposition letters; PX is a set of colours

- A modal automaton is a triple $\mathbb{A} = (A, \Theta, Acc)$, where
  - A is a finite set of states
  - $\Theta : A \times \mathrm{PX} \to 1\mathrm{ML}(A)$ is the transition map
  - $Acc \subseteq A^{\omega}$ is the acceptance condition
- An initialized automaton is a pair $(\mathbb{A}, a)$ with $a \in A$
- Parity automata: Acc is given by map $\Omega : A \to \omega$
  - Given $\rho \in A^{\omega}$, $Inf(\rho) := \{a \in A \mid a \text{ occurs infinitely often in } \pi_b\}$
  - $Acc_{\Omega} := \{\rho \in A^{\omega} \mid \max\{\Omega(a) \mid a \in Inf(\rho)\} \text{ is even }\}$

Modal automata

Fix a set X of proposition letters; PX is a set of colours

- A modal automaton is a triple $\mathbb{A} = (A, \Theta, Acc)$, where
  - A is a finite set of states
  - $\Theta : A \times \mathrm{PX} \to 1\mathrm{ML}(A)$ is the transition map
  - $Acc \subseteq A^\omega$ is the acceptance condition
- An initialized automaton is a pair $(\mathbb{A}, a)$ with $a \in A$
- Parity automata: Acc is given by map $\Omega : A \to \omega$
  - Given $\rho \in A^\omega$, $Inf(\rho) := \{a \in A \mid a \text{ occurs infinitely often in } \pi_b\}$
  - $Acc_\Omega := \{\rho \in A^\omega \mid \max\{\Omega(a) \mid a \in Inf(\rho)\} \text{ is even }\}$
- Our approach: automata are formulas

One-step logic 1ML

- Let A be a set of variables with $A \cap X = \varnothing$
- One-step formulas: $\Diamond(a \wedge b)$, $\Box a \wedge \Diamond b$, $\top$, $\Diamond\bot$,...
- A one-step model is a pair (U, m) with $m : U \to \mathsf{P}A$ a marking
  - write $U, m, u \Vdash^0 a$ if $a \in m(u)$

One-step logic 1ML

- Let A be a set of variables with $A \cap X = \varnothing$
- One-step formulas: $\Diamond(a \wedge b)$, $\Box a \wedge \Diamond b$, $\top$, $\Diamond\bot$,...
- A one-step model is a pair (U, m) with $m : U \to \mathsf{P}A$ a marking
  - write $U, m, u \Vdash^0 a$ if $a \in m(u)$
- One-step modal language $1\mathrm{ML}(\mathsf{X}, A)$ over A

$$\begin{array}{lll} \alpha & ::= & \Diamond\pi \mid \Box\pi \mid \bot \mid \top \mid \alpha \vee \alpha \mid \alpha \wedge \alpha \\ \pi & ::= & a \in A \mid \bot \mid \top \mid \pi \vee \pi \mid \pi \wedge \pi \end{array}$$

One-step logic 1ML

- Let A be a set of variables with $A \cap X = \varnothing$
- One-step formulas: $\Diamond(a \wedge b)$, $\Box a \wedge \Diamond b$, $\top$, $\Diamond\bot, \ldots$
- A one-step model is a pair (U, m) with $m : U \to \mathsf{P}A$ a marking
  - write $U, m, u \Vdash^0 a$ if $a \in m(u)$
- One-step modal language $1\mathrm{ML}(\mathrm{X}, A)$ over A

$$
\begin{array}{lcl}
\alpha & ::= & \Diamond\pi \mid \Box\pi \mid \bot \mid \top \mid \alpha \vee \alpha \mid \alpha \wedge \alpha \\
\pi & ::= & a \in A \mid \bot \mid \top \mid \pi \vee \pi \mid \pi \wedge \pi
\end{array}
$$

- One-step semantics interprets $1\mathrm{ML}(A)$ over one-step models, e.g.
  - $(U, m) \Vdash^1 \Box a$ iff $\forall u \in U. u \Vdash^0 a$
  - $(U, m) \Vdash^1 \Diamond(a \wedge b)$ iff $\exists u \in U. u \Vdash^0 a \wedge b$

Acceptance game

- Represent Kripke model as pair $\mathbb{S} = (S, \sigma)$ with $\sigma : S \to \mathsf{P}\mathsf{X} \times \mathsf{P}S$

Acceptance game $\mathcal{A}(\mathbb{A}, \mathbb{S})$ of $\mathbb{A} = \langle A, \Theta, Acc \rangle$ on $\mathbb{S} = \langle S, \sigma \rangle$:

Position	Player	Admissible moves
$(a, s) \in A \times S$	$\exists$	$\{m : \sigma_R(s) \to \mathsf{P}A \mid \sigma(s), m \Vdash^1 \Theta(a)\}$
$m : S \breve{\to} \mathsf{P}A$	$\forall$	$\{(b, t) \mid b \in m(t)\}$

Acceptance game

- Represent Kripke model as pair $\mathbb{S} = (S, \sigma)$ with $\sigma : S \to \mathsf{P}\mathsf{X} \times \mathsf{P}S$

Acceptance game $\mathcal{A}(\mathbb{A}, \mathbb{S})$ of $\mathbb{A} = \langle A, \Theta, Acc \rangle$ on $\mathbb{S} = \langle S, \sigma \rangle$:

Position	Player	Admissible moves
$(a, s) \in A \times S$	$\exists$	$\{m : \sigma_R(s) \to \mathsf{P}A \mid \sigma(s), m \Vdash^1 \Theta(a)\}$
$m : S \breve{\to} \mathsf{P}A$	$\forall$	$\{(b, t) \mid b \in m(t)\}$

Winning conditions:

- finite matches are lost by the player who gets stuck,
- infinite matches are won as specified by the acceptance condition:
  - match $\pi = (a_0, s_0)m_0(a_1, s_1)m_1 \ldots$ induces list $\pi_A := a_0 a_1 a_2 \ldots$
  - $\exists$ wins if $\pi_A \in Acc$

Acceptance game

- Represent Kripke model as pair $\mathbb{S} = (S, \sigma)$ with $\sigma : S \to \mathsf{P}\mathsf{X} \times \mathsf{P}S$

Acceptance game $\mathcal{A}(\mathbb{A}, \mathbb{S})$ of $\mathbb{A} = \langle A, \Theta, Acc \rangle$ on $\mathbb{S} = \langle S, \sigma \rangle$:

Position	Player	Admissible moves
$(a, s) \in A \times S$	$\exists$	$\{m : \sigma_R(s) \to \mathsf{P}A \mid \sigma(s), m \Vdash^1 \Theta(a)\}$
$m : S \stackrel{\smile}{\to} \mathsf{P}A$	$\forall$	$\{(b, t) \mid b \in m(t)\}$

Winning conditions:

- finite matches are lost by the player who gets stuck,
- infinite matches are won as specified by the acceptance condition:
  - match $\pi = (a_0, s_0) m_0 (a_1, s_1) m_1 \ldots$ induces list $\pi_A := a_0 a_1 a_2 \ldots$
  - $\exists$ wins if $\pi_A \in Acc$

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

Themes

Basis

- There are well-known translations: formulas $\leftrightarrow$ automata

Themes

Basis

- There are well-known translations: formulas $\leftrightarrow$ automata

Goal:

- Understand modal fixpoint logics via corresponding automata

Themes

Basis

- There are well-known translations: formulas $\leftrightarrow$ automata

Goal:

- Understand modal fixpoint logics via corresponding automata

Perspective:

- automata are generalized formulas with interesting inner structure
- automata separate the dynamics (Θ) from the combinatorics (Ω)

Themes

Basis

- There are well-known translations: formulas $\leftrightarrow$ automata

Goal:

- Understand modal fixpoint logics via corresponding automata

Perspective:

- automata are generalized formulas with interesting inner structure
- automata separate the dynamics (Θ) from the combinatorics (Ω)

Leading question:

- Which properties of modal parity automata are determined
  - already at one-step level

Themes

Basis

- There are well-known translations: formulas $\leftrightarrow$ automata

Goal:

- Understand modal fixpoint logics via corresponding automata

Perspective:

- automata are generalized formulas with interesting inner structure
- automata separate the dynamics (Θ) from the combinatorics (Ω)

Leading question:

- Which properties of modal parity automata are determined
  - already at one-step level
  - by the interaction of combinatorics and dynamics

Automata & ...

Theorem
There are maps $\mathbb{B}_{\_} : \mu\mathsf{ML} \to \mathsf{Aut}(\mathsf{ML}_1)$ and $\xi : \mathsf{Aut}(\mathsf{ML}_1) \to \mu\mathsf{ML}$ that
(1) preserve meaning: $\varphi \equiv \mathbb{B}_{\varphi}$ and $\mathbb{A} \equiv \xi(\mathbb{A})$

Automata & . . .

Theorem
There are maps $\mathbb{B}_{\_} : \mu\mathsf{ML} \to \mathsf{Aut}(\mathsf{ML}_1)$ and $\xi : \mathsf{Aut}(\mathsf{ML}_1) \to \mu\mathsf{ML}$ that
(1) preserve meaning: $\varphi \equiv \mathbb{B}_\varphi$ and $\mathbb{A} \equiv \xi(\mathbb{A})$
(2) interact nicely with Booleans, modalities, fixpoints, and substitution

Automata & . . .

Theorem

There are maps $\mathbb{B}_{\_} : \mu\mathsf{ML} \to \mathsf{Aut}(\mathsf{ML}_1)$ and $\xi : \mathsf{Aut}(\mathsf{ML}_1) \to \mu\mathsf{ML}$ that
(1) preserve meaning: $\varphi \equiv \mathbb{B}_\varphi$ and $\mathbb{A} \equiv \xi(\mathbb{A})$
(2) interact nicely with Booleans, modalities, fixpoints, and substitution
(3) satisfy $\varphi \equiv_{\mathsf{K}} \xi(\mathbb{B}_\varphi)$

Automata & ...

Theorem
There are maps $\mathbb{B}_{-} : \mu\mathsf{ML} \to \mathsf{Aut}(\mathsf{ML}_1)$ and $\xi : \mathsf{Aut}(\mathsf{ML}_1) \to \mu\mathsf{ML}$ that
(1) preserve meaning: $\varphi \equiv \mathbb{B}_{\varphi}$ and $\mathbb{A} \equiv \xi(\mathbb{A})$
(2) interact nicely with Booleans, modalities, fixpoints, and substitution
(3) satisfy $\varphi \equiv_{\mathsf{K}} \xi(\mathbb{B}_{\varphi})$

As a corollary, we may apply proof-theoretic concepts to automata

Completeness at one-step level

- Given $\alpha, \alpha' \in 1\mathrm{ML}$ define $\models^1 \alpha \leq \alpha'$ if for all (U, m):

$$(U, m) \Vdash^1 \alpha \text{ implies } (U, m) \Vdash^1 \alpha'.$$

Completeness at one-step level

- Given $\alpha, \alpha' \in$ 1ML define $\models^1 \alpha \leq \alpha'$ if for all (U, m):

$$(U, m) \Vdash^1 \alpha \text{ implies } (U, m) \Vdash^1 \alpha'.$$

- A one-step derivation system is a set **H** of one-step axioms and one-step rules operating on inequalities $\pi \leq \pi'$, $\alpha \leq \alpha'$.

Completeness at one-step level

- Given $\alpha, \alpha' \in 1\mathrm{ML}$ define $\models^1 \alpha \leq \alpha'$ if for all (U, m):

$$(U, m) \Vdash^1 \alpha \text{ implies } (U, m) \Vdash^1 \alpha'.$$

- A one-step derivation system is a set **H** of one-step axioms and one-step rules operating on inequalities $\pi \leq \pi'$, $\alpha \leq \alpha'$.
- Example: the core of basic modal logic **K** consists of
  - monotonicity rule for $\Diamond$: $a \leq b \ / \ \Diamond a \leq \Diamond b$
  - normality ($\Diamond\bot \leq \bot$) and additivity ($\Diamond(a \vee b) \leq \Diamond a \vee \Diamond b$) axioms

Completeness at one-step level

- Given $\alpha, \alpha' \in$ 1ML define $\models^1 \alpha \leq \alpha'$ if for all (U, m):

$$(U, m) \Vdash^1 \alpha \text{ implies } (U, m) \Vdash^1 \alpha'.$$

- A one-step derivation system is a set **H** of one-step axioms and one-step rules operating on inequalities $\pi \leq \pi'$, $\alpha \leq \alpha'$.
- Example: the core of basic modal logic **K** consists of
  - monotonicity rule for $\Diamond$: $a \leq b \ / \ \Diamond a \leq \Diamond b$
  - normality ($\Diamond\bot \leq \bot$) and additivity ($\Diamond(a \vee b) \leq \Diamond a \vee \Diamond b$) axioms
- A derivation system **H** is one-step sound and complete if

$$\vdash_{\mathbf{H}} \alpha \leq \alpha' \text{ iff } \models^1 \alpha \leq \alpha'.$$

Completeness at one-step level

- Given $\alpha, \alpha' \in$ 1ML define $\models^1 \alpha \leq \alpha'$ if for all (U, m):

$$(U, m) \Vdash^1 \alpha \text{ implies } (U, m) \Vdash^1 \alpha'.$$

- A one-step derivation system is a set **H** of one-step axioms and one-step rules operating on inequalities $\pi \leq \pi'$, $\alpha \leq \alpha'$.
- Example: the core of basic modal logic **K** consists of
  - monotonicity rule for $\Diamond$: $a \leq b \ / \ \Diamond a \leq \Diamond b$
  - normality ($\Diamond\bot \leq \bot$) and additivity ($\Diamond(a \vee b) \leq \Diamond a \vee \Diamond b$) axioms
- A derivation system **H** is one-step sound and complete if

$$\vdash_{\mathbf{H}} \alpha \leq \alpha' \text{ iff } \models^1 \alpha \leq \alpha'.$$

- For more on this, check the literature on coalgebra (Cîrstea, Pattinson, Schröder,...)

General result

Theorem Assume that

- $\mathcal{L}$ is a one-step language with an adequate disjunctive base
- **H** is a one-step sound and complete axiomatization for $\mathcal{L}$

Then **H**μ is a sound and complete axiomatization for $\mu\mathcal{L}$.

General result

Theorem Assume that

- $\mathcal{L}$ is a one-step language with an adequate disjunctive base
- **H** is a one-step sound and complete axiomatization for $\mathcal{L}$

Then **H**μ is a sound and complete axiomatization for $\mu\mathcal{L}$.

Proof
'De- and re-constructing' Walukiewicz' proof – automata in leading role

General result

Theorem Assume that

- $\mathcal{L}$ is a one-step language with an adequate disjunctive base
- **H** is a one-step sound and complete axiomatization for $\mathcal{L}$

Then **H**μ is a sound and complete axiomatization for $\mu\mathcal{L}$.

Proof
'De- and re-constructing' Walukiewicz' proof – automata in leading role

Examples:

- linear time μ-calculus,

General result

Theorem Assume that

- $\mathcal{L}$ is a one-step language with an adequate disjunctive base
- **H** is a one-step sound and complete axiomatization for $\mathcal{L}$

Then **H**μ is a sound and complete axiomatization for $\mu\mathcal{L}$.

Proof
'De- and re-constructing' Walukiewicz' proof – automata in leading role

Examples:

- linear time μ-calculus, *k*-successor μ-calculus,

General result

Theorem Assume that

- $\mathcal{L}$ is a one-step language with an adequate disjunctive base
- **H** is a one-step sound and complete axiomatization for $\mathcal{L}$

Then **H**μ is a sound and complete axiomatization for $\mu\mathcal{L}$.

Proof
'De- and re-constructing' Walukiewicz' proof – automata in leading role

Examples:

- linear time μ-calculus, *k*-successor μ-calculus,
  standard modal μ-calculus,

General result

Theorem Assume that

- $\mathcal{L}$ is a one-step language with an adequate disjunctive base
- **H** is a one-step sound and complete axiomatization for $\mathcal{L}$

Then **H**μ is a sound and complete axiomatization for $\mu\mathcal{L}$.

Proof
'De- and re-constructing' Walukiewicz' proof – automata in leading role

Examples:

- linear time μ-calculus, *k*-successor μ-calculus,
  standard modal μ-calculus, graded μ-calculus,

General result

Theorem Assume that

- $\mathcal{L}$ is a one-step language with an adequate disjunctive base
- **H** is a one-step sound and complete axiomatization for $\mathcal{L}$

Then **H**μ is a sound and complete axiomatization for $\mu\mathcal{L}$.

Proof
'De- and re-constructing' Walukiewicz' proof – automata in leading role

Examples:

- linear time μ-calculus, k-successor μ-calculus, standard modal μ-calculus, graded μ-calculus, monotone modal μ-calculus,

General result

Theorem Assume that

- $\mathcal{L}$ is a one-step language with an adequate disjunctive base
- **H** is a one-step sound and complete axiomatization for $\mathcal{L}$

Then **H**μ is a sound and complete axiomatization for $\mu\mathcal{L}$.

Proof
'De- and re-constructing' Walukiewicz' proof – automata in leading role

Examples:

- linear time μ-calculus, k-successor μ-calculus,
  standard modal μ-calculus, graded μ-calculus,
  monotone modal μ-calculus, game μ-calculus, ...

Overview

- Introduction
- Obstacles
- A general result
- A general framework
- Frame conditions
- Conclusions

Frame conditions

Conjecture Let $\mathbf{L}$ be an extension of $\mathbf{K}_\Gamma$ or $\mathbf{K}\mu$
with an axiom set Φ such that each $\varphi \in \Phi$

- is canonical
- corresponds to a universal first-order frame condition.

Then $\mathbf{L}$ is sound and complete for the class of frames satisfying Φ.

Overview

- Introduction
- Obstacles
- A general result
- A general framework
- Frame conditions
- Conclusions

But first:

TACL

But first:

TACL

TOPOLOGY, ALGEBRA AND CATEGORIES IN LOGIC 2017

But first:

TOPOLOGY, ALGEBRA AND CATEGORIES IN LOGIC 2017

2017 June 20–24	:	TACL School
2017 June 26–30	:	TACL Conference

`www.cs.cas.cz/tacl2017`

Conclusions

Conclusions

- general completeness result for flat fixpoint logics

Conclusions

- general completeness result for flat fixpoint logics
- framework for proving completeness for μ-calculi

Conclusions

- general completeness result for flat fixpoint logics
- framework for proving completeness for μ-calculi
- perspective for bringing automata into proof theory

Future work

- prove conjecture!
- completeness for fragments of μML (game logic!)
  - many μML-fragments have interesting automata-theoretic counterparts!
- interpolation for fixpoint logics (PDL!)
- fixpoint logics on non-boolean basis
  - non-boolean automata?
- proof theory for modal automata
- further explore notion of $\mathcal{O}$-adjointness
- ...

References

- L. Santocanale & YV. Completeness for flat modal fixpoint logic APAL 2010
- L. Schröder & YV. Completeness for flat coalgebraic fixpoint logic submitted (short version appeared in CONCUR 2010)
- S. Enqvist, F. Seifan & YV. Completeness for coalgebraic fixpoint logic CSL 2016.
- S. Enqvist, F. Seifan & YV. Completeness for the modal μ-calculus: separating the combinatorics from the dynamics, ILLC Prepublications PP-2016-33.
- YV. Lecture notes on the modal μ-calculus. Manuscript, ILLC, 2012.

http://staff.science.uva.nl/~yde

THANK YOU!