Denotations for parity automata

S. Salvati INRIA, I. Walukiewicz CNRS
Université de Bordeaux

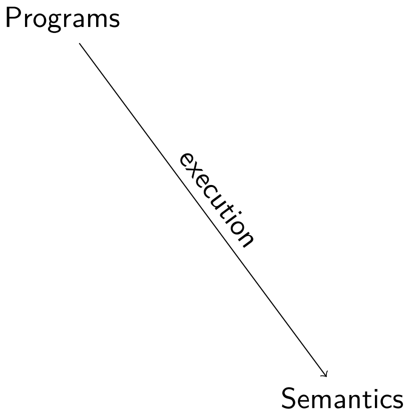
Shonan Meeting: Higher-Order Model Checking

Verification and Models

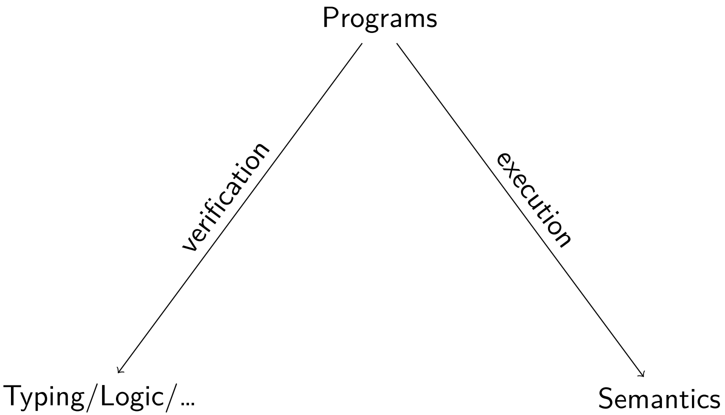
Schematology

Programs

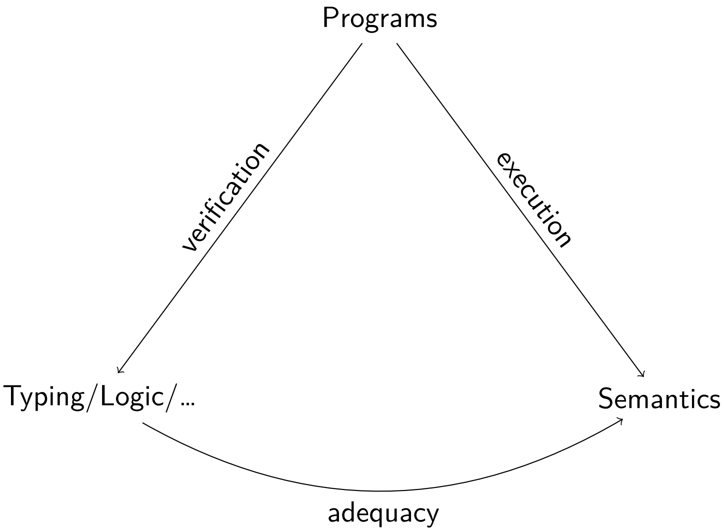
Schematology



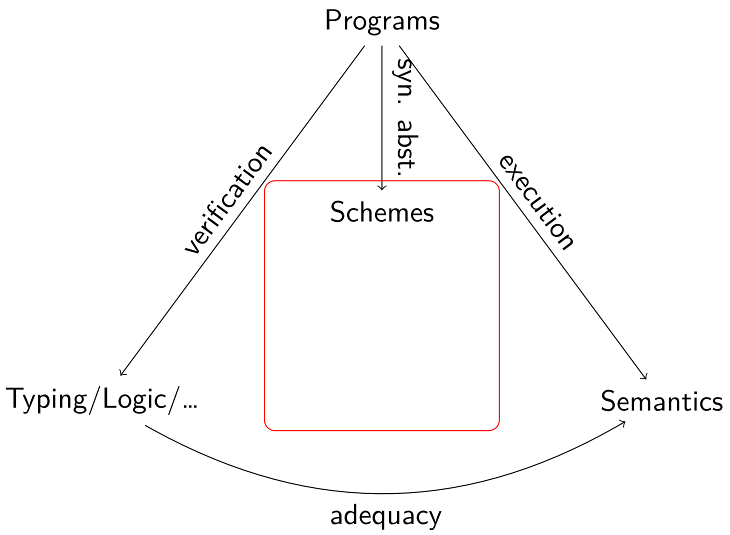
Schematology



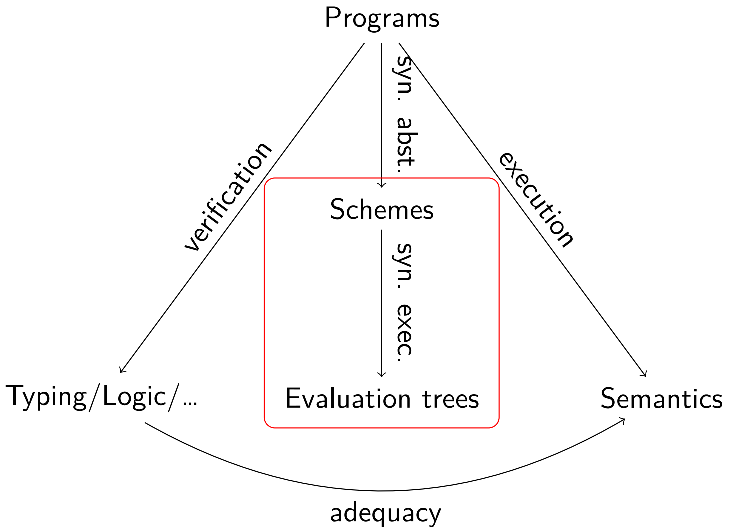
Schematology



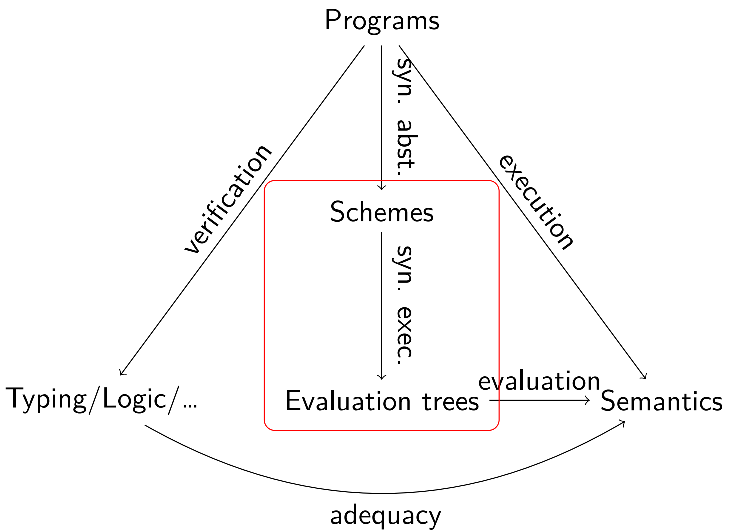
Schematology



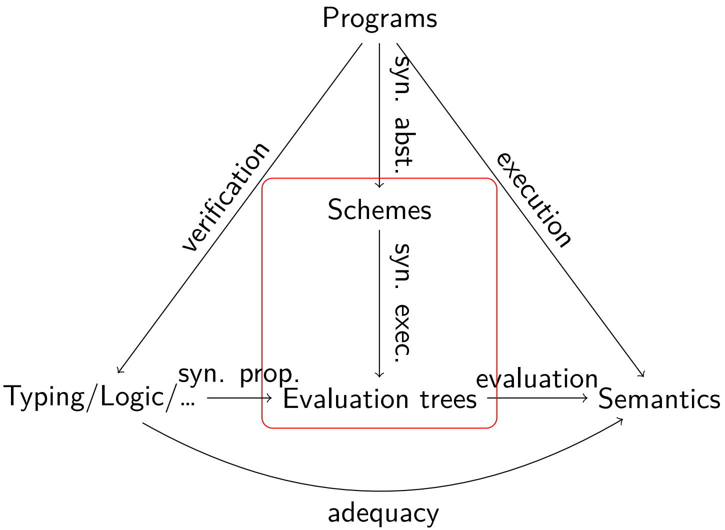
Schematology



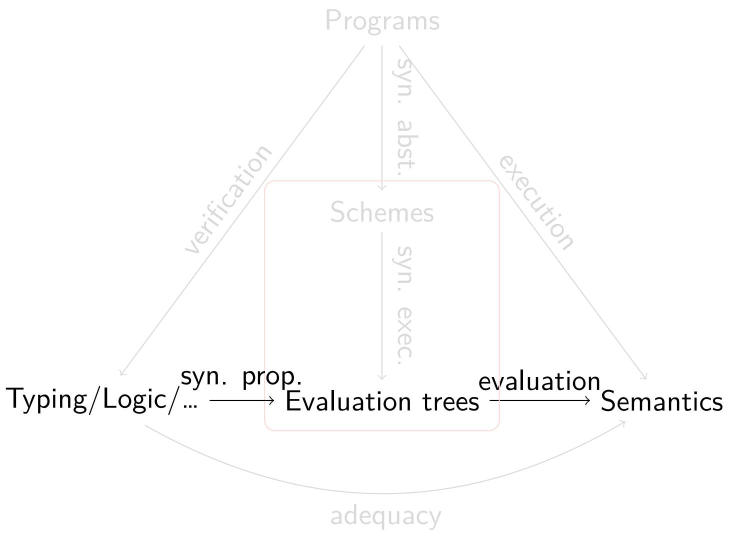
Schematology



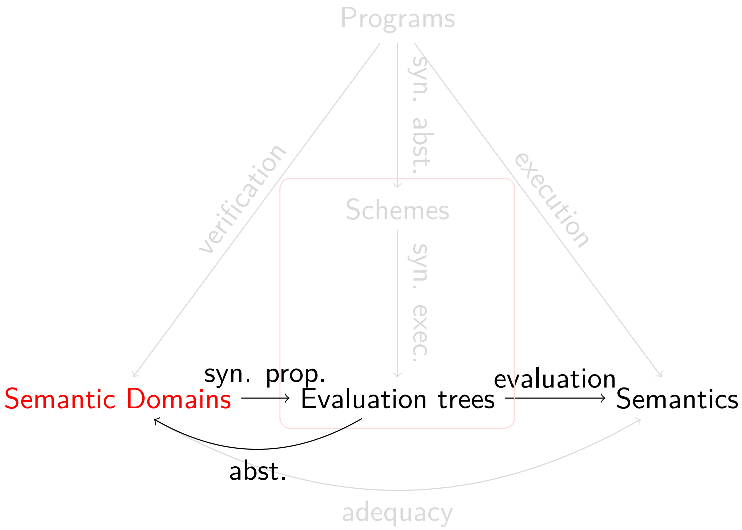
Schematology



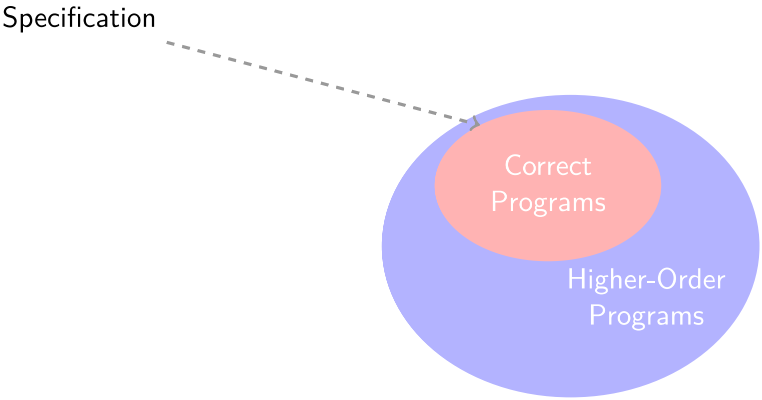
Schematology



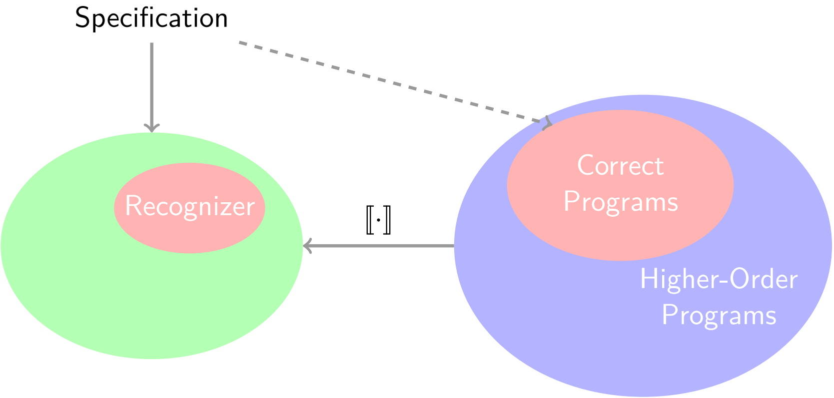
Schematology



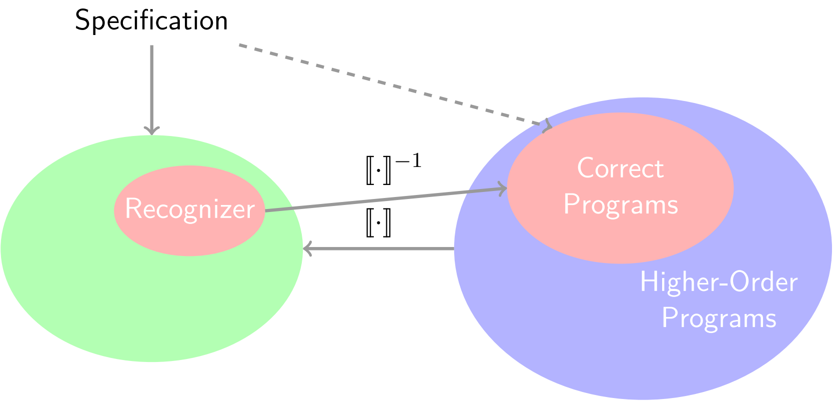
Programs and recognizability



Programs and recognizability



Programs and recognizability



Motivations

- Relating finite state methods with denotational methods

Motivations

- Relating finite state methods with denotational methods
- Reveal the invariants behind behavioral properties

Motivations

- Relating finite state methods with denotational methods
- Reveal the invariants behind behavioral properties
- Obtain decidability results by finiteness properties

Motivations

- Relating finite state methods with denotational methods
- Reveal the invariants behind behavioral properties
- Obtain decidability results by finiteness properties
- Compositional and Higher-Order by construction

Types: 0 is a type and $(A \to B)$ is a type if A and_ are types.

Tree signature $\Sigma = \{a, b, \dots\}$ all constants of type $0 \to 0 \to 0$ or of type 0.

λY-calculus

$$\Lambda Y : \qquad M^A, N^B \ ::= \ x^A \mid c^A \mid (\lambda x^A . M^B)^{A \to B} \mid (M^{A \to B} N^A)^B \mid (Y M^{A \to A})^A$$

$$(\beta) \qquad (\lambda x.M)N = M[N/x]$$

$$(\eta) \qquad \lambda x.Mx = M \text{ when } x \notin fv(M)$$

$$(\delta) \qquad YM = M(YM)$$

Böhm tree for ΛY

Böhm trees are a sort of infinite normal form for ΛY-terms

Böhm tree for ΛY

Böhm trees are a sort of infinite normal form for ΛY-terms

If M reduces to $\lambda x_1 \ldots x_n . h M_1 \ldots M_n$:

$$BT(M) = \lambda x_1 \ldots x_n . h$$

$$BT(M_1) \cdots\cdots\cdots\cdots BT(M_n)$$

Böhm tree for ΛY

Böhm trees are a sort of infinite normal form for ΛY-terms

If M reduces to $\lambda x_1 \dots x_n.hM_1 \dots M_n$:

$$BT(M) = \lambda x_1 \dots x_n.h$$

$$BT(M_1) \cdots\cdots\cdots\cdots BT(M_n)$$

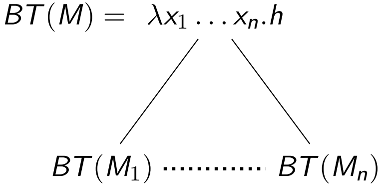
otherwise:

$$BT(M) = \Omega$$

Böhm tree for ΛY

Böhm trees are a sort of infinite normal form for ΛY-terms

If M reduces to $\lambda x_1 \dots x_n . h M_1 \dots M_n$:



otherwise:

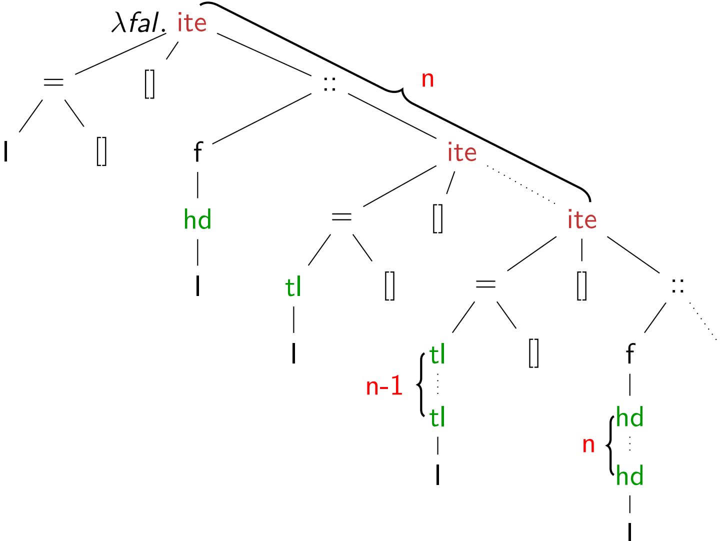
$$BT(M) = \Omega$$

When M is closed and of type 0, $BT(M)$ is an infinite tree: a higher-order tree.

Higher-order control flow

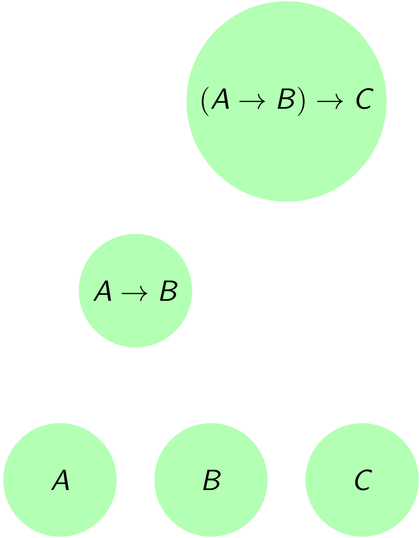
fold f a l = if l=[] then a else f (hd l) (fold f a (tl l))

$$M = Y\lambda \text{fold f a l.ite } (= \text{l } []) \text{ a } (\text{f } (\text{hd l}) (\text{fold f a } (\text{tl l})))$$



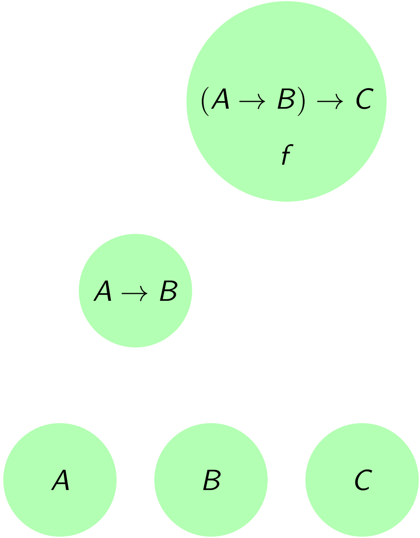
Finite models

$[\![\mathsf{M}, \nu]\!]$



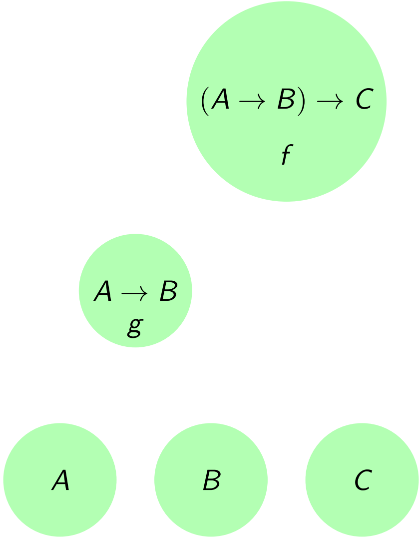
Finite models

$[\![M, \nu]\!]$



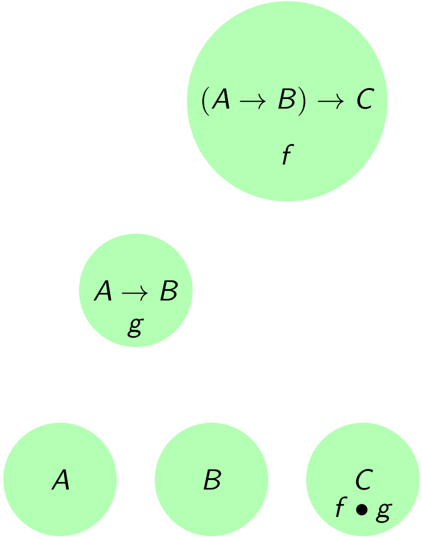
Finite models

$[\![\mathsf{M}, \nu]\!]$



Finite models

$[\![\mathsf{M}, \nu]\!]$

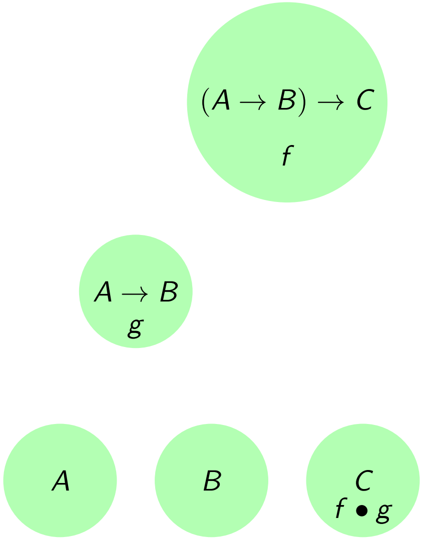


Finite models

$[\![\mathsf{M}, \nu]\!]$

Axioms

$[\![MN, \nu]\!] = [\![M, \nu]\!] \bullet [\![N, \nu]\!]$

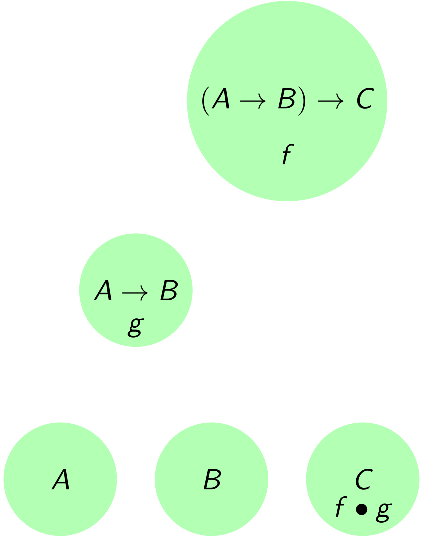


Finite models

$[\![\mathsf{M},\nu]\!]$

Axioms

$[\![MN,\nu]\!] = [\![M,\nu]\!] \bullet [\![N,\nu]\!]$
$[\![\lambda x.M,\nu]\!] \bullet f = [\![M,\nu[f/x]]\!]$

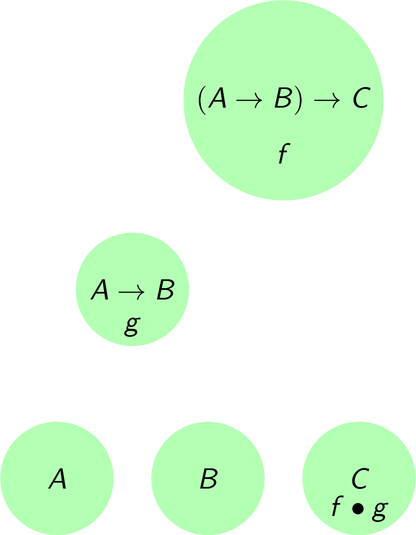


Finite models

$[\![\mathsf{M}, \nu]\!]$

Axioms

$[\![MN, \nu]\!] = [\![M, \nu]\!] \bullet [\![N, \nu]\!]$
$[\![\lambda x.M, \nu]\!] \bullet f = [\![M, \nu[f/x]]\!]$
$[\![Y, \nu]\!] \bullet f = f \bullet ([\![Y, \nu]\!] \bullet f)$



Finite models

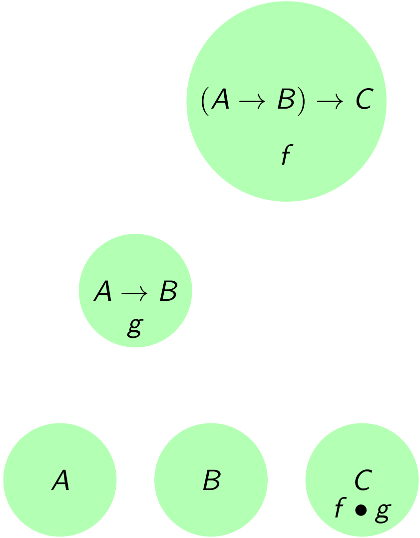
$[\![\text{M},\nu]\!]$

Axioms

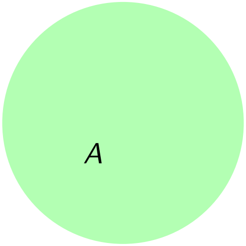
$[\![MN,\nu]\!] = [\![M,\nu]\!] \bullet [\![N,\nu]\!]$
$[\![\lambda x.M,\nu]\!] \bullet f = [\![M,\nu[f/x]]\!]$
$[\![Y,\nu]\!] \bullet f = f \bullet ([\![Y,\nu]\!] \bullet f)$

Lemma (Correctness)

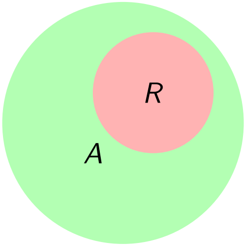
If $M =_{\beta\delta} N$, then for every ν, $[\![M,\nu]\!] = [\![N,\nu]\!]$.



Recognizability in the simply typed λ-calculus



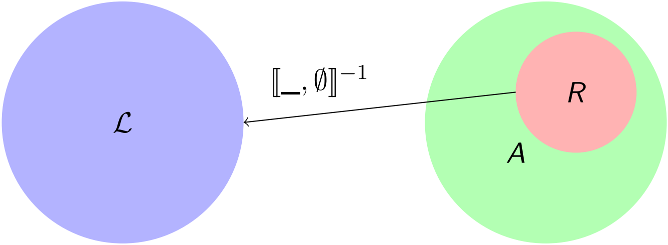
Recognizability in the simply typed λ-calculus



Recognizability in the simply typed λ-calculus

$\mathcal{L}$ is recognizable iff:

$$\mathcal{L} = \{M \mid [\![M, \emptyset]\!] \in R\}$$



Basic properties

Recognizable languages of λ-terms are:

- conservative extensions of recognizable languages of strings and trees,

Basic properties

Recognizable languages of λ-terms are:

- conservative extensions of recognizable languages of strings and trees,
- closed under boolean operations,

Basic properties

Recognizable languages of λ-terms are:

- conservative extensions of recognizable languages of strings and trees,
- closed under boolean operations,
- closed under inverse higher-order homomorphism,

Basic properties

Recognizable languages of λ-terms are:

- conservative extensions of recognizable languages of strings and trees,
- closed under boolean operations,
- closed under inverse higher-order homomorphism,
- not closed under relabeling.

Basic properties

Recognizable languages of λ-terms are:

- conservative extensions of recognizable languages of strings and trees,
- closed under boolean operations,
- closed under inverse higher-order homomorphism,
- not closed under relabeling.

- Singleton languages are recognizable [Statman 82]

Basic properties

Recognizable languages of λ-terms are:

- conservative extensions of recognizable languages of strings and trees,
- closed under boolean operations,
- closed under inverse higher-order homomorphism,
- not closed under relabeling.

- Singleton languages are recognizable [Statman 82]
- Emptiness is undecidable [Loader 01]

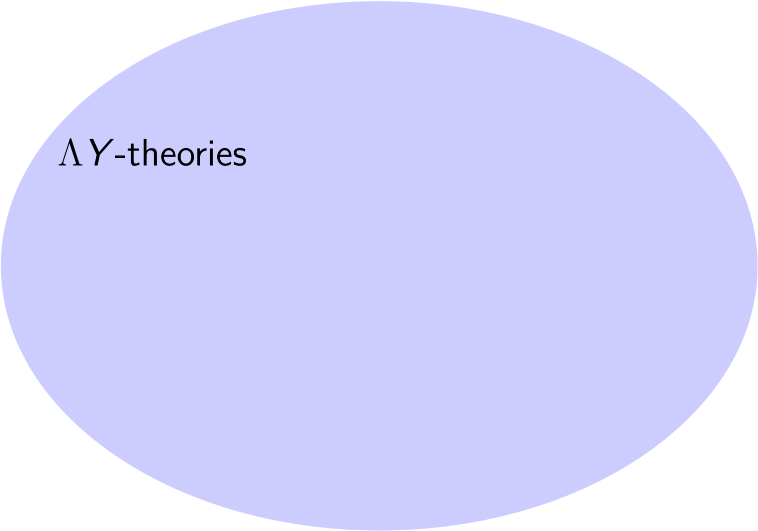
Basic properties

Recognizable languages of λ-terms are:

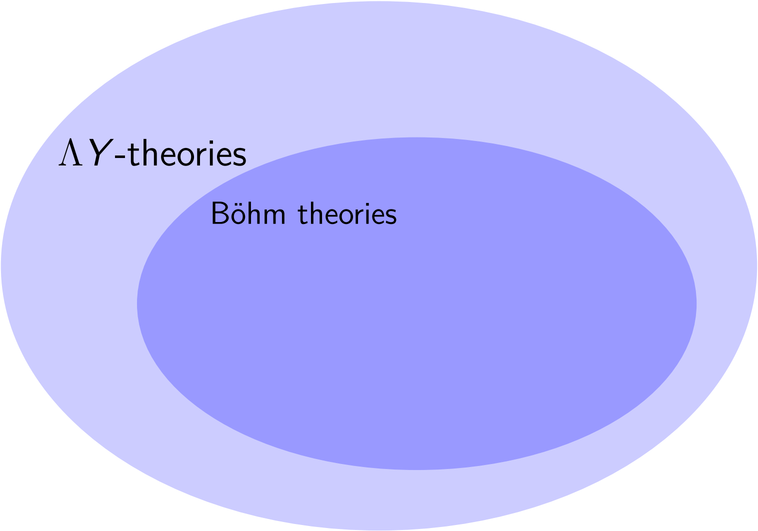
- conservative extensions of recognizable languages of strings and trees,
- closed under boolean operations,
- closed under inverse higher-order homomorphism,
- not closed under relabeling.

- Singleton languages are recognizable [Statman 82]
- Emptiness is undecidable [Loader 01]
- Membership is non-elementary

Theory of Böhm trees

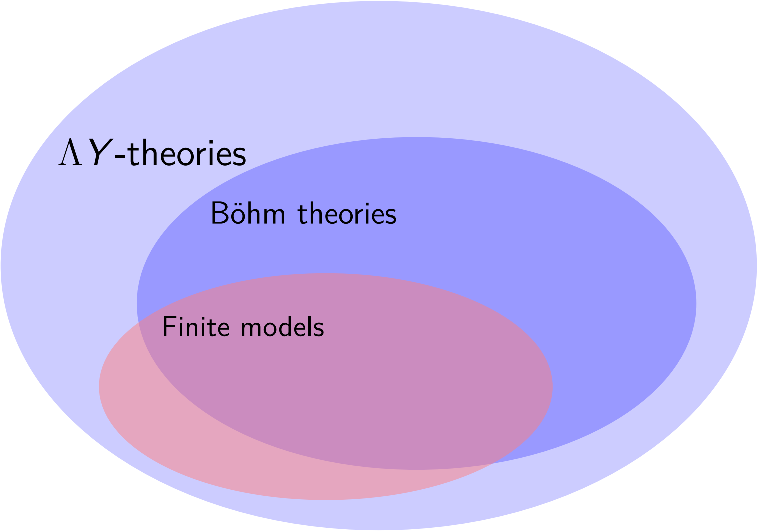


Theory of Böhm trees



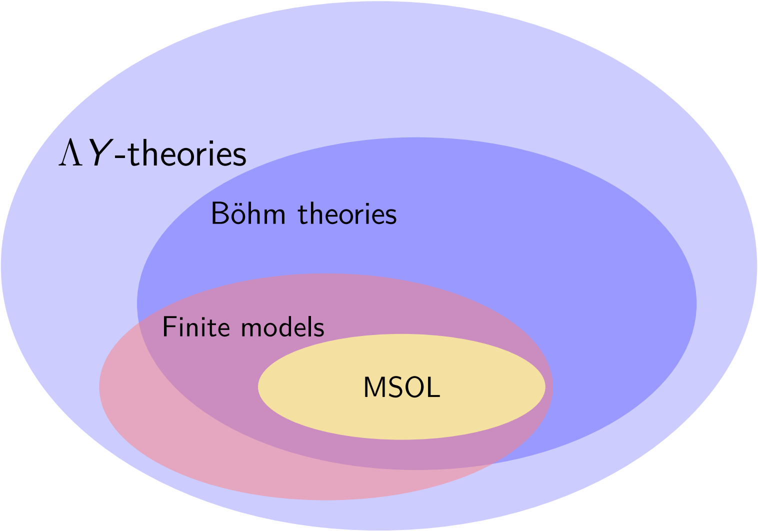
Böhm theories: $BT(M) = BT(N)$ implies for all ν, $[\![M, \nu]\!] = [\![N, \nu]\!]$

Theory of Böhm trees



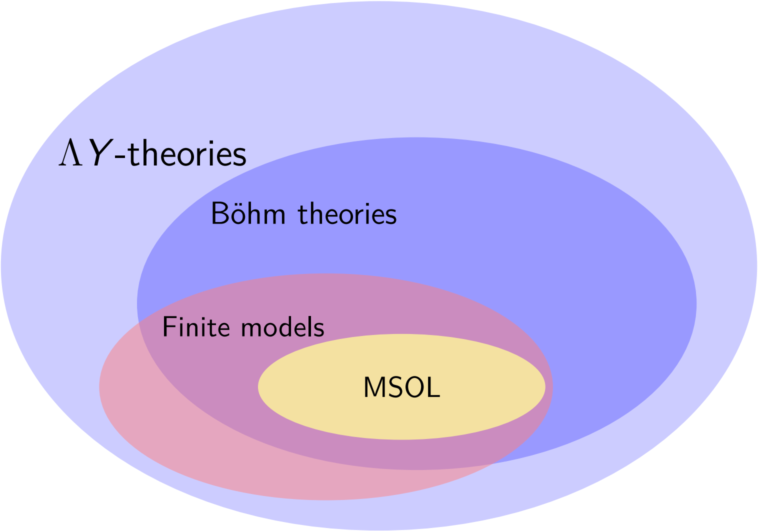
Böhm theories: $BT(M) = BT(N)$ implies for all ν, $[\![M, \nu]\!] = [\![N, \nu]\!]$

Theory of Böhm trees



Böhm theories: $BT(M) = BT(N)$ implies for all ν, $[\![M, \nu]\!] = [\![N, \nu]\!]$

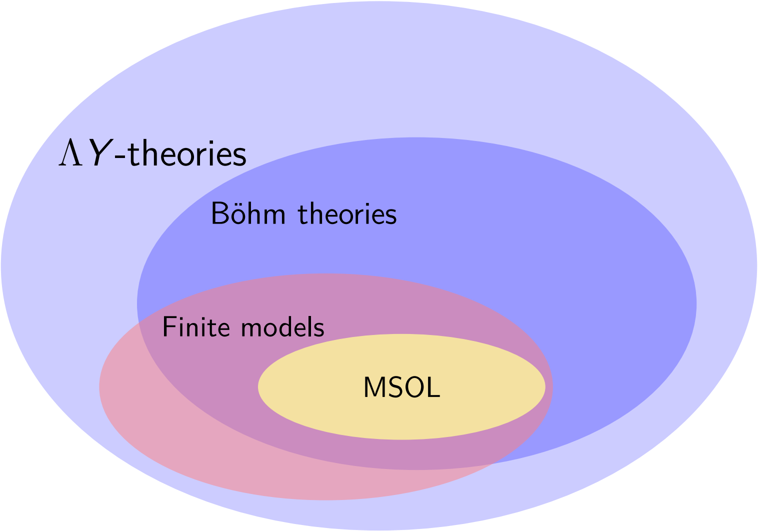
Theory of Böhm trees



Böhm theories: $BT(M) = BT(N)$ implies for all ν, $[\![M, \nu]\!] = [\![N, \nu]\!]$

- expressiveness of finite Böhm models? (See Pawel's talk)

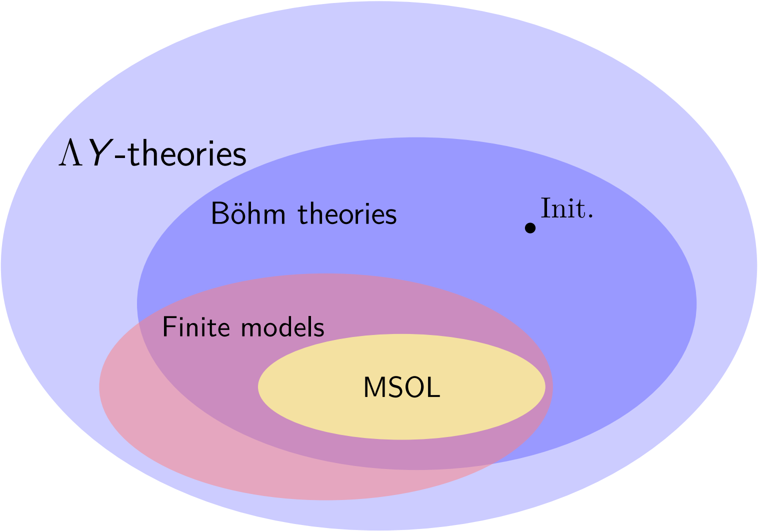
Theory of Böhm trees



Böhm theories: $BT(M) = BT(N)$ implies for all ν, $[\![M, \nu]\!] = [\![N, \nu]\!]$

- expressiveness of finite Böhm models? (See Pawel's talk)
- axiomatization of finite Böhm models?

Theory of Böhm trees



Böhm theories: $BT(M) = BT(N)$ implies for all ν, $[\![M,\nu]\!] = [\![N,\nu]\!]$

- expressiveness of finite Böhm models? (See Pawel's talk)
- axiomatization of finite Böhm models?

Monotone (Scott) models

$((\mathcal{M}_A, \leq_A)_A, [\![\_, \_]\!])$

- $(M_0, \leq_0)$ is a complete lattice,

Monotone (Scott) models

$((\mathcal{M}_A, \leq_A)_A, [\![\_, \_]\!])$

- $(M_0, \leq_0)$ is a complete lattice,
- $(M_{A\to B}, \leq_{A\to B})$ is the complete lattice of monotone functions f from $(M_A, \leq_A)$ to $(M_B, \leq_B)$, i.e. $a \leq_A b$ implies $f(a) \leq_B f(b)$, ordered pointwise.

Monotone (Scott) models

$((\mathcal{M}_A, \leq_A)_A, [\![\_, \_]\!])$

- $(M_0, \leq_0)$ is a complete lattice,
- $(M_{A\to B}, \leq_{A\to B})$ is the complete lattice of monotone functions f from $(M_A, \leq_A)$ to $(M_B, \leq_B)$, i.e. $a \leq_A b$ implies $f(a) \leq_B f(b)$, ordered pointwise.
- $[\![Y, \nu]\!](f) = \bigwedge\{f^n(\top) \mid n \in \mathbb{N}\}$.

Monotone (Scott) models

$((\mathcal{M}_A, \leq_A)_A, [\![\_, \_]\!])$

- $(M_0, \leq_0)$ is a complete lattice,
- $(M_{A\to B}, \leq_{A\to B})$ is the complete lattice of monotone functions f from $(M_A, \leq_A)$ to $(M_B, \leq_B)$, i.e. $a \leq_A b$ implies $f(a) \leq_B f(b)$, ordered pointwise.
- $[\![Y, \nu]\!](f) = \bigwedge\{f^n(\top) \mid n \in \mathbb{N}\}$.
- given $f \in M_A$, $g \in M_B$, $(f \mapsto g)(h) = \begin{cases} g \text{ when } g \leq h \\ \perp \text{ otherwise} \end{cases}$

Digretion: a extensional non-Böhm model

Take $(M_0, \leq) = (\mathcal{P}(\{q_0, q_1\}), \subseteq)$, we let $\mathbb{M}$ be the model so that $(M_A, \leq_A)$ is generated as in the monotone model. We then let:

- $S \downarrow_0 = S \cap \{q_0\}$ for $S \subseteq Q$,
- $f \downarrow_0 (g) = f(g) \downarrow_0$ for f of type $A \to B$ and g of type A.

Digretion: a extensional non-Böhm model

Take $(M_0, \leq) = (\mathcal{P}(\{q_0, q_1\}), \subseteq)$, we let $\mathbb{M}$ be the model so that $(M_A, \leq_A)$ is generated as in the monotone model. We then let:

- $S \downarrow_0 = S \cap \{q_0\}$ for $S \subseteq Q$,
- $f \downarrow_0 (g) = f(g) \downarrow_0$ for f of type $A \to B$ and g of type A.

We then define:

$$fix(f) = \nu x. f(x \downarrow_0)))$$

Digretion: a extensional non-Böhm model

Take $(M_0, \leq) = (\mathcal{P}(\{q_0, q_1\}), \subseteq)$, we let $\mathbb{M}$ be the model so that $(M_A, \leq_A)$ is generated as in the monotone model. We then let:

- $S \downarrow_0 = S \cap \{q_0\}$ for $S \subseteq Q$,
- $f \downarrow_0 (g) = f(g) \downarrow_0$ for f of type $A \to B$ and g of type A.

We then define:

$$\mathit{fix}(f) = \nu x. f(x \downarrow_0)))$$

- for $f_1 = q_1 \mapsto q_0 \vee q_0 \mapsto q_1$, we have $\mathit{fix}(f_1) = \emptyset$,
- and for $f_2 = q_0 \mapsto q_0 \vee q_1 \mapsto q_1$, we have $\mathit{fix}(f_2) = \{q_0\}$

Digretion: a extensional non-Böhm model

Take $(M_0, \leq) = (\mathcal{P}(\{q_0, q_1\}), \subseteq)$, we let $\mathbb{M}$ be the model so that $(M_A, \leq_A)$ is generated as in the monotone model. We then let:

- $S \downarrow_0 = S \cap \{q_0\}$ for $S \subseteq Q$,
- $f \downarrow_0 (g) = f(g) \downarrow_0$ for f of type $A \to B$ and g of type A.

We then define:

$$fix(f) = \nu x.f(x \downarrow_0)))$$

- for $f_1 = q_1 \mapsto q_0 \vee q_0 \mapsto q_1$, we have $fix(f_1) = \emptyset$,
- and for $f_2 = q_0 \mapsto q_0 \vee q_1 \mapsto q_1$, we have $fix(f_2) = \{q_0\}$

But $f_2 = f_1 \circ f_1$.
So if a is a constant so that $[\![a]\!] = f_1$, interpreting Y as fix gives $[\![Y(\lambda x.ax)]\!] = \emptyset$ and $[\![Y(\lambda x.a(ax))]\!] = \{q_0\}$.

Ω-blind trivial properties

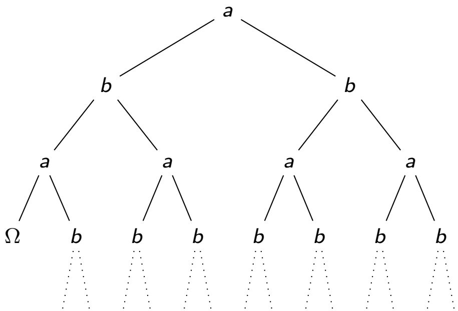
Theorem (S. Waluckiewicz 13)

Scott models recognize boolean combinations of Ω-blind trivial properties.

Ω-blind trivial properties

Theorem (S. Waluckiewicz 13)

Scott models recognize boolean combinations of Ω-blind trivial properties.

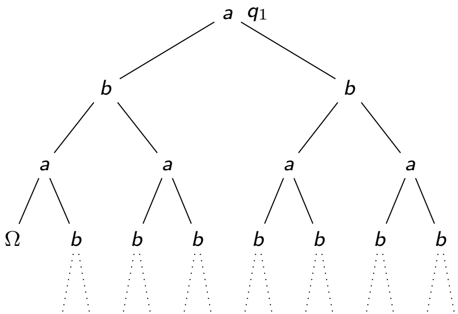


$$\delta(a, q_1) = (q_2, q_2) \qquad \delta(b, q_2) = (q_1, q_1)$$

Ω-blind trivial properties

Theorem (S. Waluckiewicz 13)

Scott models recognize boolean combinations of Ω-blind trivial properties.

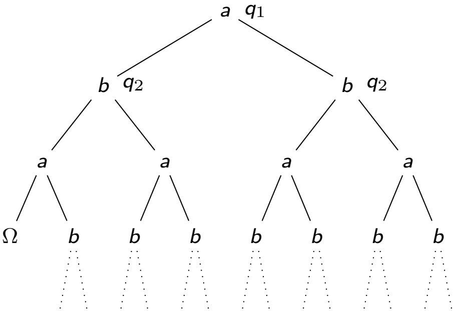


$$\delta(a, q_1) = (q_2, q_2) \qquad \delta(b, q_2) = (q_1, q_1)$$

Ω-blind trivial properties

Theorem (S. Waluckiewicz 13)

Scott models recognize boolean combinations of Ω-blind trivial properties.

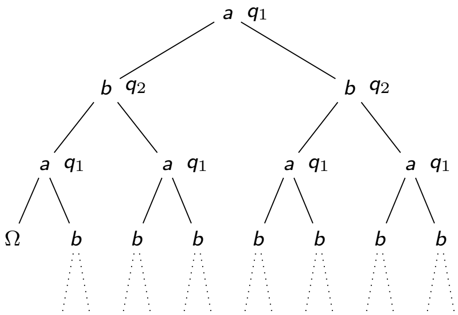


$$\delta(a, q_1) = (q_2, q_2) \qquad \delta(b, q_2) = (q_1, q_1)$$

Ω-blind trivial properties

Theorem (S. Waluckiewicz 13)

Scott models recognize boolean combinations of Ω-blind trivial properties.

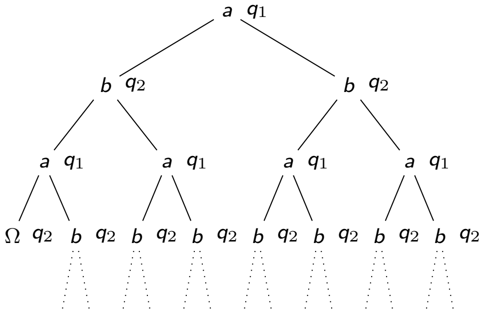


$$\delta(a, q_1) = (q_2, q_2) \qquad \delta(b, q_2) = (q_1, q_1)$$

Ω-blind trivial properties

Theorem (S. Waluckiewicz 13)

Scott models recognize boolean combinations of Ω-blind trivial properties.

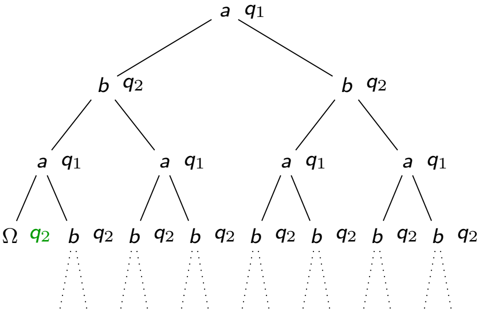


$$\delta(a, q_1) = (q_2, q_2) \qquad \delta(b, q_2) = (q_1, q_1)$$

Ω-blind trivial properties

Theorem (S. Waluckiewicz 13)

Scott models recognize boolean combinations of Ω-blind trivial properties.

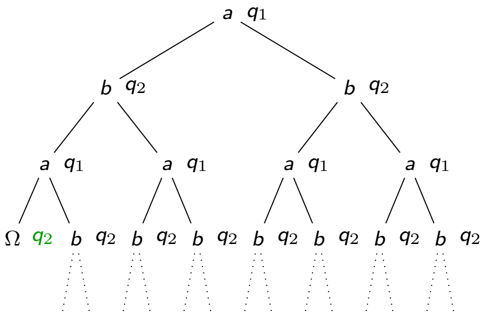


$$\delta(a, q_1) = (q_2, q_2) \qquad \delta(b, q_2) = (q_1, q_1)$$

Ω-blind trivial properties

Theorem (S. Waluckiewicz 13)

Scott models recognize boolean combinations of Ω-blind trivial properties.



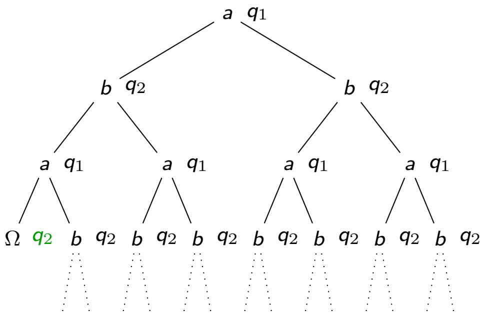
$$\delta(a, q_1) = (q_2, q_2) \qquad \delta(b, q_2) = (q_1, q_1)$$

The recognizing model is defined: $(\mathcal{M}_0, \leq_0) = (\mathcal{P}(\{q_1, q_2\}, \subseteq))$

Ω-blind trivial properties

Theorem (S. Waluckiewicz 13)

Scott models recognize boolean combinations of Ω-blind trivial properties.

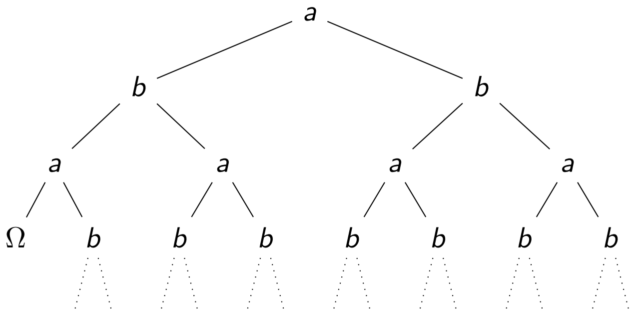


$$\delta(a, q_1) = (q_2, q_2) \qquad \delta(b, q_2) = (q_1, q_1)$$

The recognizing model is defined: $(\mathcal{M}_0, \leq_0) = (\mathcal{P}(\{q_1, q_2\}, \subseteq))$
In [S. Walukiewicz 13], it is showed how to build insightful models.

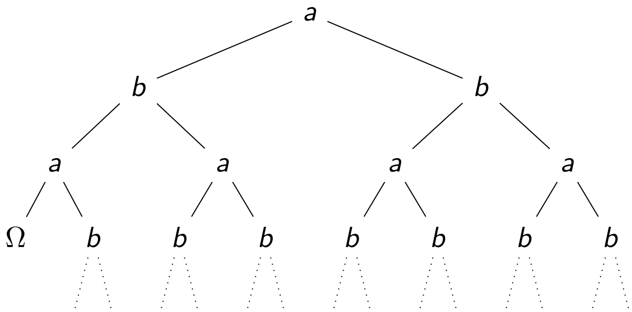
First step towards Parity Conditions: weak MSOL

weak MSOL



$$
\begin{aligned}
\delta(a, q_1) =& \{\{(q_0, q_0)\}\} \\
\delta(b, q_2) =& \{\{(q_1, q_1), (q_2, q_2)\}\} \\
\delta(a, q_0) = \delta(b, q_0)& \{\{(q_0, q_0)\}\} \\
\delta(a, q_2) =& \{\{(q_2, q_2)\}\} \\
\delta(b, q_1) =& \{\{(q_1, q_1)\}\}
\end{aligned}
$$

weak MSOL



$$\delta(a, q_1) = \{\{(q_0, q_0)\}\}$$

$$\delta(b, q_2) = \{\{(q_1, q_1), (q_2, q_2)\}\}$$

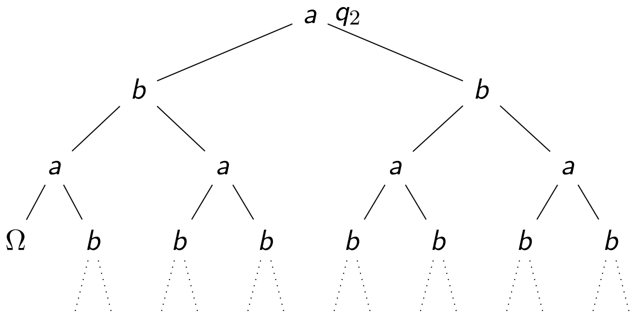
$$\delta(a, q_0) = \delta(b, q_0)\{\{(q_0, q_0)\}\}$$

$$\delta(a, q_2) = \{\{(q_2, q_2)\}\}$$

$$\delta(b, q_1) = \{\{(q_1, q_1)\}\}$$

$$\mathtt{rk}(q_i) = i$$

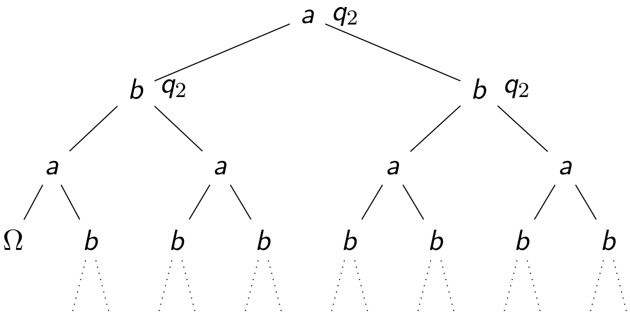
weak MSOL



$$\delta(a, q_1) = \{\{(q_0, q_0)\}\}$$
$$\delta(b, q_2) = \{\{(q_1, q_1), (q_2, q_2)\}\}$$
$$\delta(a, q_0) = \delta(b, q_0)\{\{(q_0, q_0)\}\}$$
$$\delta(a, q_2) = \{\{(q_2, q_2)\}\}$$
$$\delta(b, q_1) = \{\{(q_1, q_1)\}\}$$

$$\mathtt{rk}(q_i) = i$$

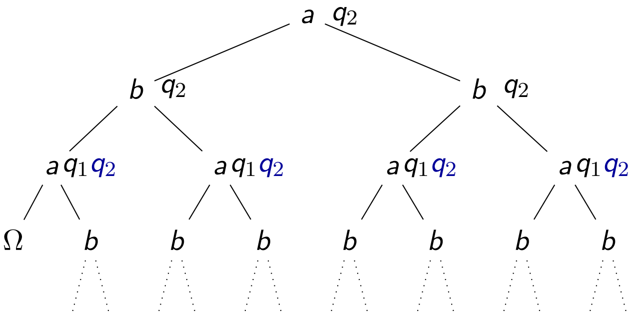
weak MSOL



$$\delta(a, q_1) = \{\{(q_0, q_0)\}\}$$
$$\delta(b, q_2) = \{\{(q_1, q_1), (q_2, q_2)\}\}$$
$$\delta(a, q_0) = \delta(b, q_0)\{\{(q_0, q_0)\}\}$$
$$\delta(a, q_2) = \{\{(q_2, q_2)\}\}$$
$$\delta(b, q_1) = \{\{(q_1, q_1)\}\}$$

$$\mathtt{rk}(q_i) = i$$

weak MSOL



$$\delta(a, q_1) = \{\{(q_0, q_0)\}\}$$

$$\delta(b, q_2) = \{\{(q_1, q_1), (q_2, q_2)\}\}$$

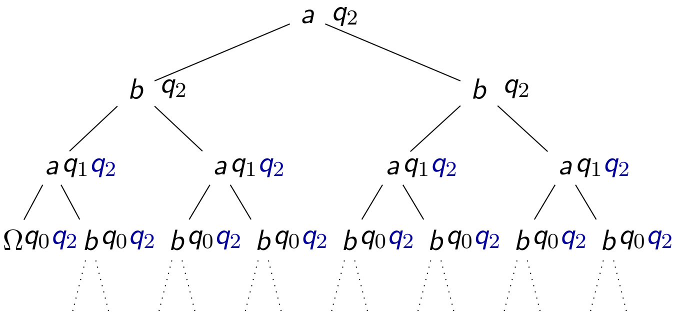
$$\delta(a, q_0) = \delta(b, q_0)\{\{(q_0, q_0)\}\}$$

$$\delta(a, q_2) = \{\{(q_2, q_2)\}\}$$

$$\delta(b, q_1) = \{\{(q_1, q_1)\}\}$$

$$\mathtt{rk}(q_i) = i$$

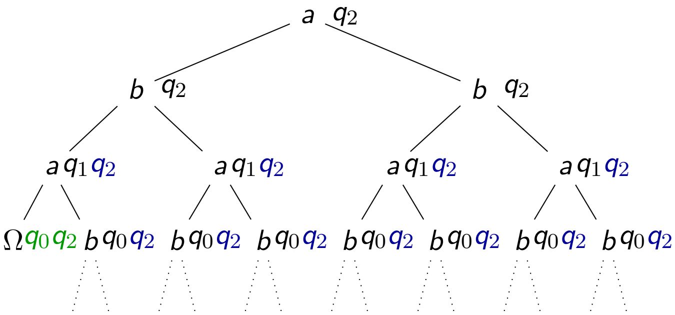
weak MSOL



$$\delta(a, q_1) = \{\{(q_0, q_0)\}\}$$
$$\delta(b, q_2) = \{\{(q_1, q_1), (q_2, q_2)\}\}$$
$$\delta(a, q_0) = \delta(b, q_0)\{\{(q_0, q_0)\}\}$$
$$\delta(a, q_2) = \{\{(q_2, q_2)\}\}$$
$$\delta(b, q_1) = \{\{(q_1, q_1)\}\}$$

$$\mathtt{rk}(q_i) = i$$

weak MSOL



$$\delta(a, q_1) = \{\{(q_0, q_0)\}\}$$

$$\delta(b, q_2) = \{\{(q_1, q_1), (q_2, q_2)\}\}$$

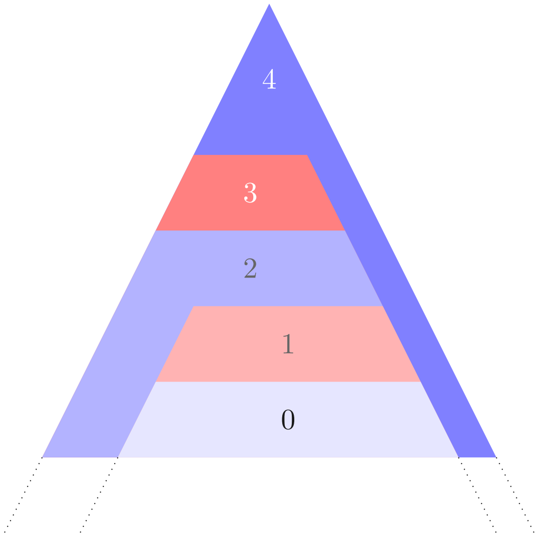
$$\delta(a, q_0) = \delta(b, q_0)\{\{(q_0, q_0)\}\}$$

$$\delta(a, q_2) = \{\{(q_2, q_2)\}\}$$

$$\delta(b, q_1) = \{\{(q_1, q_1)\}\}$$

$$\mathtt{rk}(q_i) = i$$

Structure of weak Parity automata accepting runs



Layered monotone models

The layered monotone model over the finite lattices $\mathcal{L}_0 = (Q_0, \leq_0)$, ..., $\mathcal{L}_k = (Q_k, \leq_k)$:

$$\mathcal{D} = (\{(\mathcal{D}_A, \sqsubseteq_A)\}_{A \in \mathcal{T}\sqcup}, \rho) \quad \rho : \mathsf{Cst} \to \mathcal{D}$$

where

- $\mathcal{D}_0 = \mathcal{L}_0 \times \cdots \times \mathcal{L}_k$ and $f \sqsubseteq_0 g$ is the product order,
- $e = (a_1, \ldots, a_k)$, $e_{|i} = (a_1, \ldots, a_i)$,

Layered monotone models

The layered monotone model over the finite lattices $\mathcal{L}_0 = (Q_0, \leq_0)$, ..., $\mathcal{L}_k = (Q_k, \leq_k)$:

$$\mathcal{D} = (\{(\mathcal{D}_A, \sqsubseteq_A)\}_{A \in \mathcal{T}\sqcup}, \rho) \quad \rho : \mathsf{Cst} \to \mathcal{D}$$

where

- $\mathcal{D}_0 = \mathcal{L}_0 \times \cdots \times \mathcal{L}_k$ and $f \sqsubseteq_0 g$ is the product order,
- $e = (a_1, \ldots, a_k)$, $e_{|i} = (a_1, \ldots, a_i)$,

Layered monotone models

The layered monotone model over the finite lattices $\mathcal{L}_0 = (Q_0, \leq_0)$, ..., $\mathcal{L}_k = (Q_k, \leq_k)$:

$$\mathcal{D} = (\{(\mathcal{D}_A, \sqsubseteq_A)\}_{A \in \mathcal{T}\sqcup}, \rho) \quad \rho : \mathsf{Cst} \to \mathcal{D}$$

where

- $\mathcal{D}_0 = \mathcal{L}_0 \times \cdots \times \mathcal{L}_k$ and $f \sqsubseteq_0 g$ is the product order,
- $e = (a_1, \ldots, a_k)$, $e_{|i} = (a_1, \ldots, a_i)$,
- $\mathcal{D}_{B \to C} = [\mathcal{D}_B \to_l \mathcal{D}_C] = \{f \in [\mathcal{D}_B \to_m \mathcal{D}_C] \mid \forall g, g' \in \mathcal{D}_B, \forall i \leq k, g_{|i} = g'_{|i} \Rightarrow (f(g))_{|i} = (f(g'))_{|i}\}$
  $\sqsubseteq_{B \to C}$= pointwise ordering.

Layered monotone models

The layered monotone model over the finite lattices $\mathcal{L}_0 = (Q_0, \leq_0)$, ..., $\mathcal{L}_k = (Q_k, \leq_k)$:

$$\mathcal{D} = (\{(\mathcal{D}_A, \sqsubseteq_A)\}_{A \in \mathcal{T}\sqcup}, \rho) \quad \rho : \mathsf{Cst} \rightarrow \mathcal{D}$$

where

- $\mathcal{D}_0 = \mathcal{L}_0 \times \cdots \times \mathcal{L}_k$ and $f \sqsubseteq_0 g$ is the product order,
- $e = (a_1, \ldots, a_k)$, $e_{|i} = (a_1, \ldots, a_i)$,
- $\mathcal{D}_{B \rightarrow C} = [\mathcal{D}_B \rightarrow_l \mathcal{D}_C] = \{f \in [\mathcal{D}_B \rightarrow_m \mathcal{D}_C] \mid \forall g, g' \in \mathcal{D}_B, \forall i \leq k, g_{|i} = g'_{|i} \Rightarrow (f(g))_{|i} = (f(g'))_{|i}\}$
  $\sqsubseteq_{B \rightarrow C}$= pointwise ordering.
- $\mathcal{D}_{B \rightarrow C|i} = [\mathcal{D}_{B|i} \rightarrow_l \mathcal{D}_{C|i}]$

Layered monotone models

The layered monotone model over the finite lattices $\mathcal{L}_0 = (Q_0, \leq_0)$, ..., $\mathcal{L}_k = (Q_k, \leq_k)$:

$$\mathcal{D} = (\{(\mathcal{D}_A, \sqsubseteq_A)\}_{A \in \mathcal{T}\sqcup}, \rho) \quad \rho : \mathsf{Cst} \to \mathcal{D}$$

where

- $\mathcal{D}_0 = \mathcal{L}_0 \times \cdots \times \mathcal{L}_k$ and $f \sqsubseteq_0 g$ is the product order,
- $e = (a_1, \ldots, a_k)$, $e_{|i} = (a_1, \ldots, a_i)$,
- $\mathcal{D}_{B \to C} = [\mathcal{D}_B \to_l \mathcal{D}_C] = \{f \in [\mathcal{D}_B \to_m \mathcal{D}_C] \mid \forall g, g' \in \mathcal{D}_B, \forall i \leq k, g_{|i} = g'_{|i} \Rightarrow (f(g))_{|i} = (f(g'))_{|i}\}$
  $\sqsubseteq_{B \to C}$= pointwise ordering.
- $\mathcal{D}_{B \to C|i} = [\mathcal{D}_{B|i} \to_l \mathcal{D}_{C|i}]$
- for $f \in \mathcal{D}_{B \to C}$, $f_{|i}(g_{|i}) = (f(g))_{|i}$,

Layered monotone models

The layered monotone model over the finite lattices $\mathcal{L}_0 = (Q_0, \leq_0)$, ..., $\mathcal{L}_k = (Q_k, \leq_k)$:

$$\mathcal{D} = (\{(\mathcal{D}_A, \sqsubseteq_A)\}_{A \in \mathcal{T} \sqcup}, \rho) \quad \rho : \mathsf{Cst} \to \mathcal{D}$$

where

- $\mathcal{D}_0 = \mathcal{L}_0 \times \cdots \times \mathcal{L}_k$ and $f \sqsubseteq_0 g$ is the product order,
- $e = (a_1, \ldots, a_k)$, $e_{|i} = (a_1, \ldots, a_i)$,
- $\mathcal{D}_{B \to C} = [\mathcal{D}_B \to_l \mathcal{D}_C] = \{f \in [\mathcal{D}_B \to_m \mathcal{D}_C] \mid \forall g, g' \in \mathcal{D}_B, \forall i \leq k, g_{|i} = g'_{|i} \Rightarrow (f(g))_{|i} = (f(g'))_{|i}\}$
  $\sqsubseteq_{B \to C}$= pointwise ordering.
- $\mathcal{D}_{B \to C|i} = [\mathcal{D}_{B|i} \to_l \mathcal{D}_{C|i}]$
- for $f \in \mathcal{D}_{B \to C}$, $f_{|i}(g_{|i}) = (f(g))_{|i}$,
- $\mathcal{D}_{B \to C, i} = \{f \in \mathcal{D}_{B \to B} \mid \forall g \in \mathcal{D}_B, \forall j \neq i, \pi_j(f(g)) = \bot_{\mathcal{D}_{C,j}}\}$.

Layered monotone models

The layered monotone model over the finite lattices $\mathcal{L}_0 = (Q_0, \leq_0)$, ..., $\mathcal{L}_k = (Q_k, \leq_k)$:

$$\mathcal{D} = (\{(\mathcal{D}_A, \sqsubseteq_A)\}_{A \in \mathcal{T}\sqcup}, \rho) \quad \rho : \mathsf{Cst} \to \mathcal{D}$$

where

- $\mathcal{D}_0 = \mathcal{L}_0 \times \cdots \times \mathcal{L}_k$ and $f \sqsubseteq_0 g$ is the product order,
- $e = (a_1, \ldots, a_k)$, $e_{|i} = (a_1, \ldots, a_i)$,
- $\mathcal{D}_{B \to C} = [\mathcal{D}_B \to_l \mathcal{D}_C] = \{f \in [\mathcal{D}_B \to_m \mathcal{D}_C] \mid \forall g, g' \in \mathcal{D}_B, \forall i \leq k, g_{|i} = g'_{|i} \Rightarrow (f(g))_{|i} = (f(g'))_{|i}\}$
  $\sqsubseteq_{B \to C}$= pointwise ordering.
- $\mathcal{D}_{B \to C|i} = [\mathcal{D}_{B|i} \to_l \mathcal{D}_{C|i}]$
- for $f \in \mathcal{D}_{B \to C}$, $f_{|i}(g_{|i}) = (f(g))_{|i}$,
- $\mathcal{D}_{B \to C,i} = \{f \in \mathcal{D}_{B \to B} \mid \forall g \in \mathcal{D}_B, \forall j \neq i, \pi_j(f(g)) = \bot_{\mathcal{D}_{C,j}}\}$.

Lemma

For all A, $\mathcal{D}_A$ is isomorphic to $\mathcal{D}_{A,0} \times \cdots \times \mathcal{D}_{A,k}$.

Towards a Semantics of Y: Galois Connections

For $f = (f_1, \ldots, f_i)$ in $\mathcal{D}_{A|i}$ we let:

- $f^{\uparrow} = (f_1, \ldots, f_i, \top_{A,i})$,
- $f^{\downarrow} = (f_1, \ldots, f_i, \bot_{A,i})$

For $f = (f_1, \ldots, f_i, f_{i+1})$ in $\mathcal{D}_{A|i+1}$ we let:

$$\overline{f} = (f_1, \ldots, f_i)$$

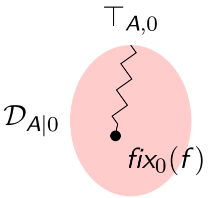
We have, for $f \in \mathcal{D}_{A|i}$ and $g \in \mathcal{D}_{A|i+1}$:

- $\overline{g} \leq f$ iff $g \leq f^{\uparrow}$,
- $f \leq \overline{g}$ iff $f^{\downarrow} \leq g$

Towards a Semantics of Y

We inductively define fix_i as an element of $\mathcal{D}_{A \to A | i}$:

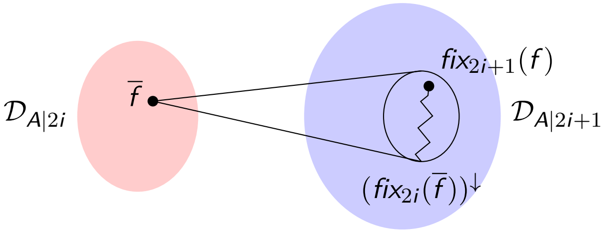
- $\mathit{fix}_0(f) = \prod\{f^n(\top_{A,0}) \mid n \in \mathbb{N}\}$



Towards a Semantics of Y

We inductively define fix_i as an element of $\mathcal{D}_{A \to A|i}$:

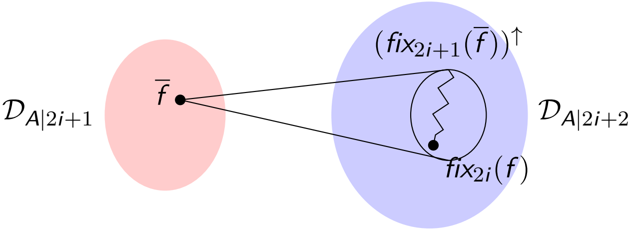
- $\mathit{fix}_0(f) = \bigsqcap\{f^n(\top_{A,0}) \mid n \in \mathbb{N}\}$
- $\mathit{fix}_{2i+1}(f) = \bigsqcup\{f^n((\mathit{fix}_{2i}(\overline{f}))^{\downarrow}) \mid n \in \mathbb{N}\}$



Towards a Semantics of Y

We inductively define fix_i as an element of $\mathcal{D}_{A \to A|i}$:

- $fix_0(f) = \bigsqcap\{f^n(\top_{A,0}) \mid n \in \mathbb{N}\}$
- $fix_{2i+1}(f) = \bigsqcup\{f^n((fix_{2i}(\overline{f}))^{\downarrow}) \mid n \in \mathbb{N}\}$
- $fix_{2i+2}(f) = \bigsqcap\{f^n((fix_{2i+1}(\overline{f}))^{\uparrow}) \mid n \in \mathbb{N}\}$



Layered monotone models and weak automata

Theorem (S. Walukiewicz 15)

Given $\mathcal{D}$ a layered monotone model and $A \subseteq \mathcal{D}_0$, M is recognized by A iff $BT(M)$ is accepted by a weak alternating parity automaton.

Layered monotone models and weak automata

Theorem (S. Walukiewicz 15)

Given $\mathcal{D}$ a layered monotone model and $A \subseteq \mathcal{D}_0$, M is recognized by A iff $BT(M)$ is accepted by a weak alternating parity automaton.

- The model *lives* inside the monotone model where we have removed meaningless functions.

Layered monotone models and weak automata

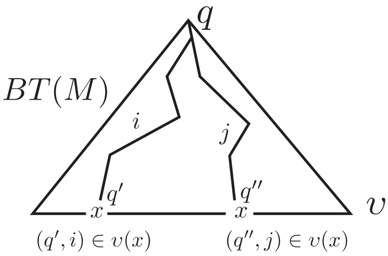
Theorem (S. Walukiewicz 15)

Given $\mathcal{D}$ a layered monotone model and $A \subseteq \mathcal{D}_0$, M is recognized by A iff $BT(M)$ is accepted by a weak alternating parity automaton.

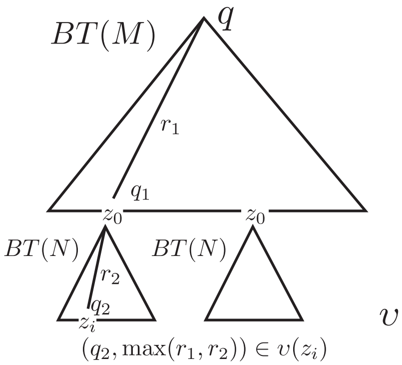
- The model *lives* inside the monotone model where we have removed meaningless functions.
- Dualities in the model $\longrightarrow$ means for reasonning for proving or refuting properties.

Models for MSOL

Color modalities (1)



Color modalities (2)



General principles

- Maintaining the information of the maximal color seen from the root of a Böhm tree to occurrences of variables.
- We use Scott domains *enriched* with this information.
- As for weak MSOL we remove *meaningless interpretations*.

Enriched Scott domains

Fix a parity automaton $\mathcal{A}$, $rk(q)$ is the color associated to q.

Enriched domain

$$\mathcal{R}_0 = \mathcal{P}(\{(q, r) : q \in Q \text{ and } rk(q) \leq r \leq m\})$$

$$h\downharpoonleft_r = \{(q, i) \in h : r \leq i\} \cup \{(q, j) : (q, r) \in h, rk(q) \leq j \leq r\}$$

Lemma

For $h \in \mathcal{R}_0$, $q \in Q$, and $r, r_1, r_2 \in [m]$:

- $(h\downharpoonleft_{r_1})\downharpoonleft_{r_2} = h\downharpoonleft_{\max(r_1,r_2)}$;
- $(q, rk(q)) \in h\downharpoonleft_r$ *iff* $(q, \max(r, rk(q))) \in h$

$$h{\Downarrow_q} = \{r \mid (q, r) \in h\}$$

Result domain

Fix a parity automaton $\mathcal{A}$, $rk(q)$ is the color associated to q.

Result domain

$$\mathcal{D}_0 = \mathcal{P}(Q)$$

$$f \cdot r = \{(q, r) : q \in \mathcal{R}_0 \text{ and } rk(q) \leq r\}$$

$$f{\Downarrow_q} = f \cap \{q\}$$

For h in $\mathcal{R}_0$, let

$$h^{\partial} = \{q : (q, rk(q)) \in h\}$$

Going higher-order

Enriched domain $\mathcal{R}_{A \to B}$ is the set of monotone functions from $\mathcal{R}_A$ to $\mathcal{R}_B$ so that:

$$\forall g \in \mathcal{R}_A.\ \forall q \in Q.\ (f(g))\Downarrow_q = (f(g\lfloor_{rk(q)}))\Downarrow_q$$

Where $f\Downarrow_q(g) = f(g)\Downarrow_q$, $f\lfloor_{rk(q)}(g) = f(g)\lfloor_{rk(q)}$ and $f^{\partial}(g) = f(g)^{\partial}$

Result domain

$\mathcal{D}_{A \to B}$ is the set of monotone functions from $\mathcal{R}_A$ to $\mathcal{D}_B$ so that:

$$\forall g \in \mathcal{R}_A.\ \forall q \in Q.\ (f(g))\Downarrow_q = (f(g\lfloor_{rk(q)}))\Downarrow_q$$

Where $f\Downarrow_q(g) = f(g)\Downarrow_q$ and $f \cdot r(g) = f(g) \cdot r$.

Interpretation of terms

$$\llbracket x, \nu \rrbracket = (\nu(x))^{\partial}$$
$$\llbracket a, \nu \rrbracket h_1 \dots h_k = \{q : \exists_{(q_1,\dots,q_k)\in(q,a)}\ q_i \in (h_i\lfloor_{rk(q)})^{\partial} \text{ for all } i\}$$
$$\llbracket \lambda x.M, \nu \rrbracket h = \llbracket M, \nu[h/x] \rrbracket$$
$$\llbracket MN, \nu \rrbracket = \llbracket M, \nu \rrbracket \langle\langle N, \nu \rangle\rangle \qquad \text{where } \langle\langle N, \nu \rangle\rangle = \bigvee_{r=0}^{m} \big(\llbracket N, \nu\lfloor_r \rrbracket \cdot r\big)$$
$$\text{and } \nu\lfloor_r(x) = \nu(x)\lfloor_r$$
$$\llbracket Y, \nu \rrbracket h = \mu f_m.\nu f_{m-1} \dots \mu f_1.\nu f_0.\ (h\lfloor_l)^{\partial}\big(\bigvee_{i=0}^{l} f_i \cdot i\big)$$

Interpretation of terms

$$
\begin{aligned}
&[\![x,\nu]\!] = (\nu(x))^{\partial}\\
&[\![a,\nu]\!]h_1\dots h_k = \{q : \exists_{(q_1,\dots,q_k)\in(q,a)}\ q_i \in (h_i\lfloor_{rk(q)})^{\partial} \text{ for all } i\}\\
&[\![\lambda x.M,\nu]\!]h = [\![M,\nu[h/x]]\!]\\
&[\![MN,\nu]\!] = [\![M,\nu]\!]\langle\!\langle N,\nu\rangle\!\rangle \qquad \text{where } \langle\!\langle N,\nu\rangle\!\rangle = \bigvee_{r=0}^{m}\big([\![N,\nu\lfloor_r]\!]\cdot r\big)\\
&\qquad\qquad\qquad\qquad\qquad\qquad \text{and } \nu\lfloor_r(x) = \nu(x)\lfloor_r\\
&[\![Y,\nu]\!]h = \mu f_m.\nu f_{m-1}\dots\mu f_1.\nu f_0.\ (h\lfloor_l)^{\partial}\big(\bigvee_{i=0}^{l} f_i\cdot i\big)
\end{aligned}
$$

Theorem (Soundness (S. Walukiewicz 15))

If $BT(M) = BT(N)$ then $[\![M,\nu]\!] = [\![N,\nu]\!]$.

Interpretation of terms

$$
\begin{aligned}
&[\![x,\nu]\!] = (\nu(x))^{\partial}\\
&[\![a,\nu]\!]h_1\dots h_k = \{q : \exists_{(q_1,\dots,q_k)\in(q,a)}\ q_i \in (h_i{\downharpoonleft}_{rk(q)})^{\partial} \text{ for all } i\}\\
&[\![\lambda x.M,\nu]\!]h = [\![M,\nu[h/x]]\!]\\
&[\![MN,\nu]\!] = [\![M,\nu]\!]\langle\!\langle N,\nu\rangle\!\rangle \qquad \text{where } \langle\!\langle N,\nu\rangle\!\rangle = \bigvee_{r=0}^{m}\big([\![N,\nu{\downharpoonleft}_r]\!]\cdot r\big)\\
&\qquad\qquad\qquad\qquad\qquad\quad\ \text{and } \nu{\downharpoonleft}_r(x) = \nu(x){\downharpoonleft}_r\\
&[\![Y,\nu]\!]h = \mu f_m.\nu f_{m-1}\dots\mu f_1.\nu f_0.\ (h{\downharpoonleft}_l)^{\partial}\big(\bigvee_{i=0}^{l} f_i\cdot i\big)
\end{aligned}
$$

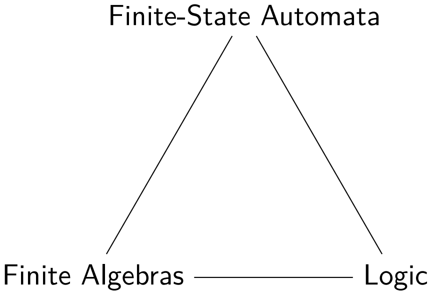
Theorem (Soundness (S. Walukiewicz 15))

If $BT(M) = BT(N)$ then $[\![M,\nu]\!] = [\![N,\nu]\!]$.

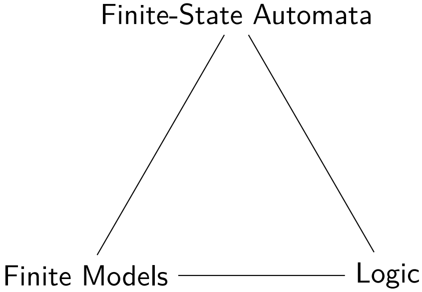
Theorem (Completeness (S. Walukiewicz 15))

For M closed and of type 0, $q \in [\![M]\!]$ iff $\mathcal{A}$ has an accepting run starting from q on $BT(M)$.

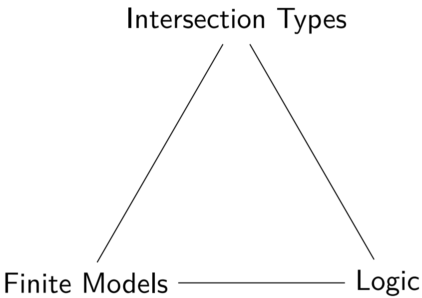
Conclusion



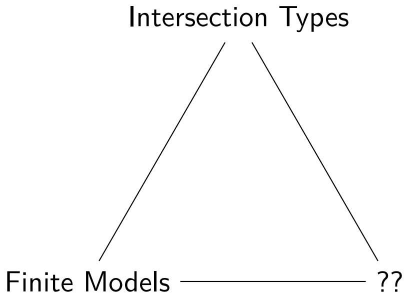
Conclusion



Conclusion



Conclusion



Announcement

Igor and I have a PhD fellowship starting this Autumn in Bordeaux.
We will welcome any good student willing to work on this topic.