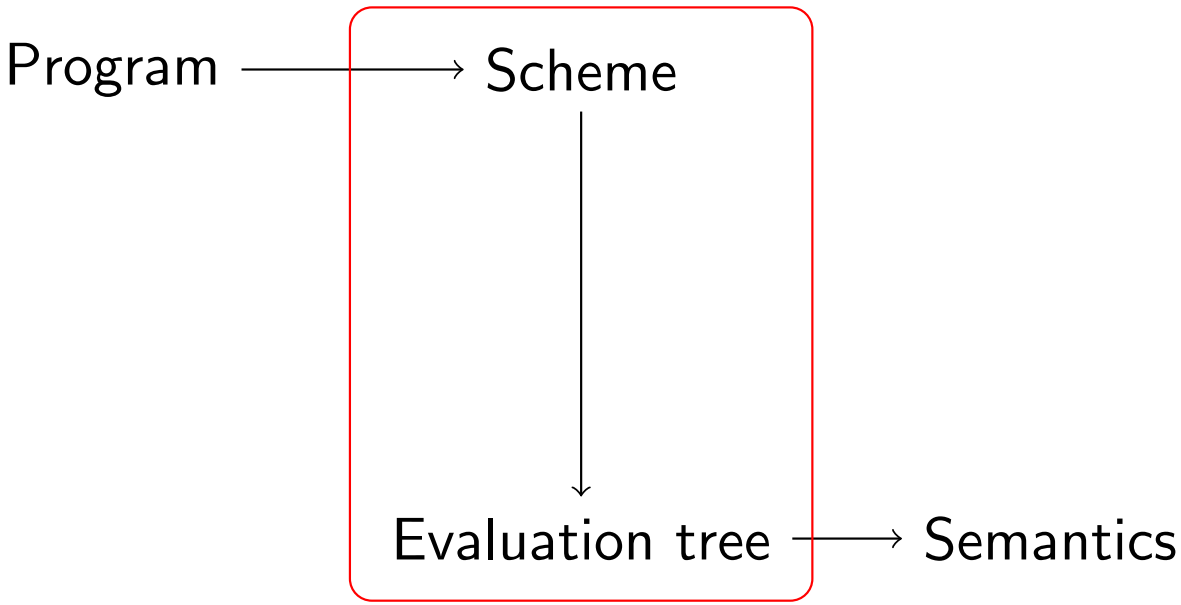
Model checking of higher-order programs and denotational semantics of λ-calculus.

Sylvain Salvati and Igor Waluckiewicz
INRIA, LaBRI, Université de Bordeaux

FREC Ile de Ré

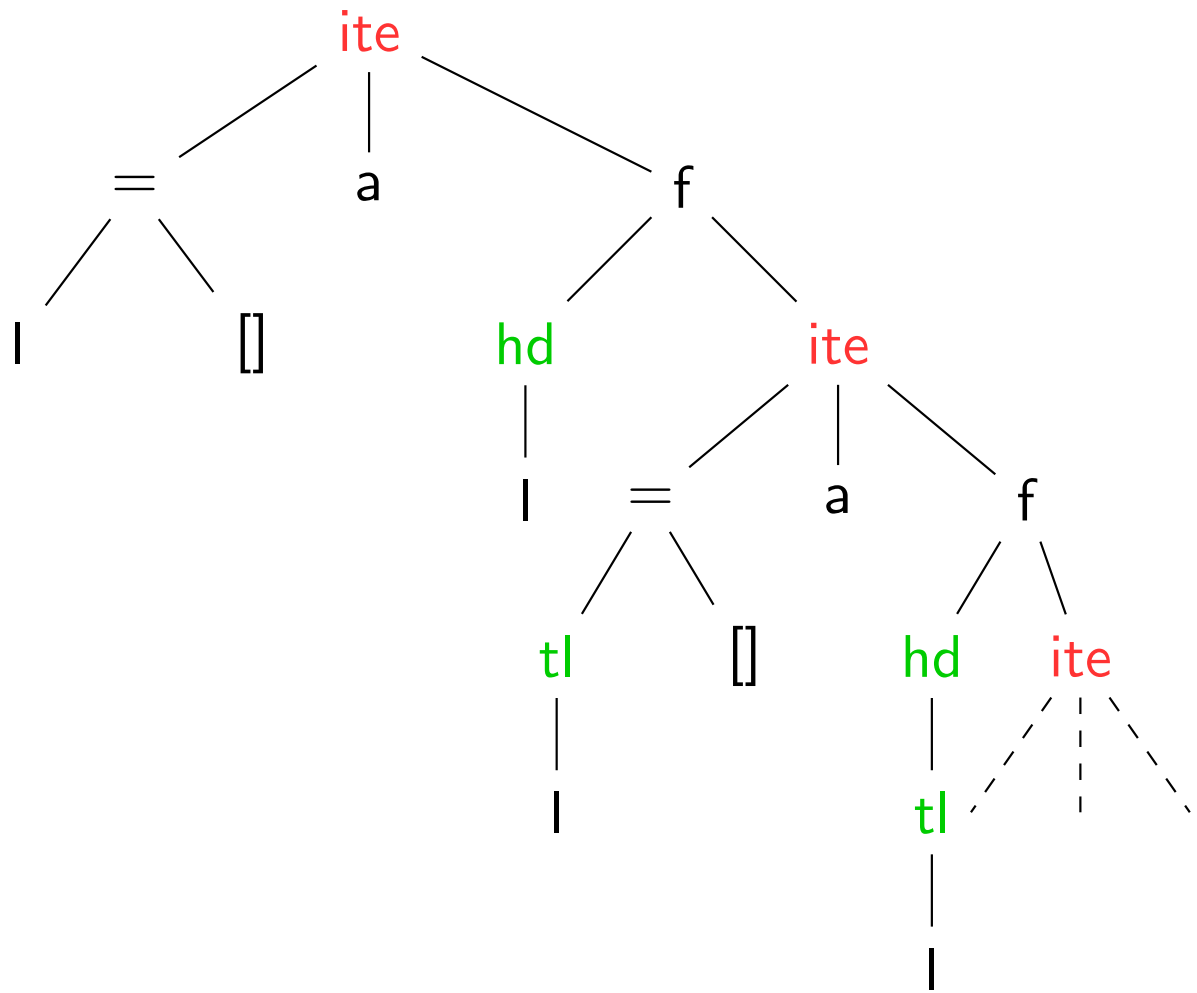
Program schemes model checking



Model checking

verifying finite state properties of evaluation trees

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



Higher-order schemes are λY-terms:

Types

$$\mathcal{T}: \quad \alpha, \beta, \gamma ::= 0 \mid (\alpha \to \beta)$$

λY-calculus

$$\Lambda Y: \quad M^\alpha, N^\beta ::= x^\alpha \mid c^\alpha \mid (\lambda x^\alpha . M^\beta)^{\alpha\to\beta} \mid (M^{\alpha\to\beta} N^\alpha)^\beta \mid (Y M^{\alpha\to\alpha})^\alpha$$

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

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

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

Higher-order schemes are λY-terms:

Types

$$\mathcal{T}: \quad \alpha, \beta, \gamma ::= 0 \mid (\alpha \to \beta)$$

λY-calculus

$$\Lambda Y: \quad M^\alpha, N^\beta ::= x^\alpha \mid c^\alpha \mid (\lambda x^\alpha.M^\beta)^{\alpha\to\beta} \mid (M^{\alpha\to\beta} N^\alpha)^\beta \mid (YM^{\alpha\to\alpha})^\alpha$$

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

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

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

Böhm tree for ΛY

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

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

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

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

otherwise:

$$BT(M) = \Omega$$

Böhm tree for ΛY

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

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

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

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

otherwise:

$$BT(M) = \Omega$$

Böhm tree for ΛY

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

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

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

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

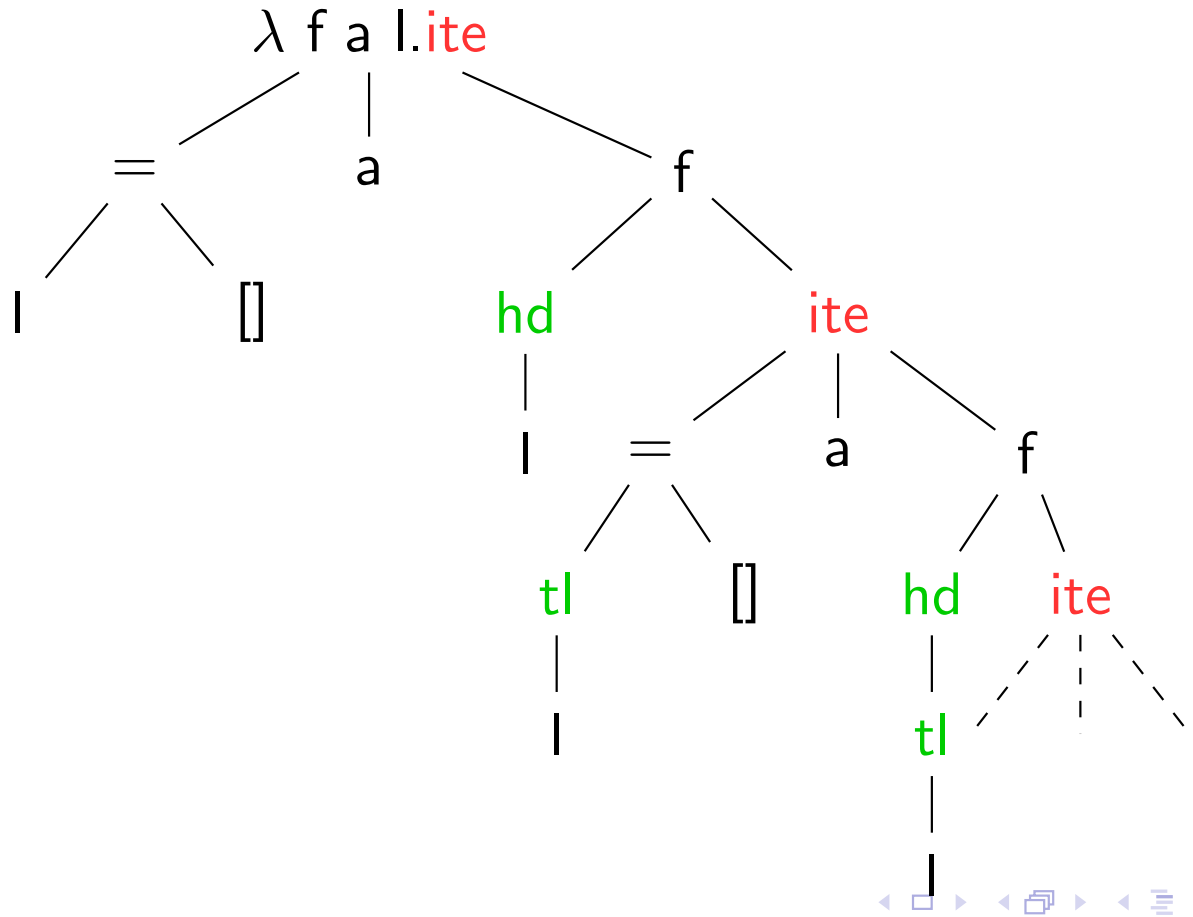
otherwise:

$$BT(M) = \Omega$$

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

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

$BT(M)$ is:



Exploring the limits of *effective* denotational semantics

Advantages of denotational semantics:

- Characterizes invariants modulo computation
- Plays the role of monoids/algebra in usual formal language

Exploring the limits of *effective* denotational semantics

Effective denotational semantics:

- Interpretation domains that can be effectively constructed at every types
- The interpretation of terms are all computable
- In practice, we use only finite domains of interpretations
- We want to understand the kinds of properties we can express on results produced by ΛY in this context because:
  - It gives simple decidability results
  - It allows to understand in a deeper way the nature of those properties and the kinds of algorithms they require
  - It may yield original domains of interpretation for ΛY

Exploring the limits of *effective* denotational semantics

Effective denotational semantics:

- Interpretation domains that can be effectively constructed at every types
- The interpretation of terms are all computable
- In practice, we use only finite domains of interpretations
- We want to understand the kinds of properties we can express on results produced by ΛY in this context because:
  - It gives simple decidability results
  - It allows to understand in a deeper way the nature of those properties and the kinds of algorithms they require
  - It may yield original domains of interpretation for ΛY

Exploring the limits of *effective* denotational semantics

Effective denotational semantics:

- Interpretation domains that can be effectively constructed at every types
- The interpretation of terms are all computable
- In practice, we use only finite domains of interpretations
- We want to understand the kinds of properties we can express on results produced by ΛY in this context because:
  - It gives simple decidability results
  - It allows to understand in a deeper way the nature of those properties and the kinds of algorithms they require
  - It may yield original domains of interpretation for ΛY

Exploring the limits of *effective* denotational semantics

Effective denotational semantics:

- Interpretation domains that can be effectively constructed at every types
- The interpretation of terms are all computable
- In practice, we use only finite domains of interpretations
- We want to understand the kinds of properties we can express on results produced by ΛY in this context because:
  - It gives simple decidability results
  - It allows to understand in a deeper way the nature of those properties and the kinds of algorithms they require
  - It may yield original domains of interpretation for ΛY

Exploring the limits of *effective* denotational semantics

Effective denotational semantics:

- Interpretation domains that can be effectively constructed at every types
- The interpretation of terms are all computable
- In practice, we use only finite domains of interpretations
- We want to understand the kinds of properties we can express on results produced by ΛY in this context because:
  - It gives simple decidability results
  - It allows to understand in a deeper way the nature of those properties and the kinds of algorithms they require
  - It may yield original domains of interpretation for ΛY

Exploring the limits of *effective* denotational semantics

Effective denotational semantics:

- Interpretation domains that can be effectively constructed at every types
- The interpretation of terms are all computable
- In practice, we use only finite domains of interpretations
- We want to understand the kinds of properties we can express on results produced by ΛY in this context because:
  - It gives simple decidability results
  - It allows to understand in a deeper way the nature of those properties and the kinds of algorithms they require
  - It may yield original domains of interpretation for ΛY

Exploring the limits of *effective* denotational semantics

Effective denotational semantics:

- Interpretation domains that can be effectively constructed at every types
- The interpretation of terms are all computable
- In practice, we use only finite domains of interpretations
- We want to understand the kinds of properties we can express on results produced by ΛY in this context because:
  - It gives simple decidability results
  - It allows to understand in a deeper way the nature of those properties and the kinds of algorithms they require
  - It may yield original domains of interpretation for ΛY

Exploring the limits of *effective* denotational semantics

Effective denotational semantics:

- Interpretation domains that can be effectively constructed at every types
- The interpretation of terms are all computable
- In practice, we use only finite domains of interpretations
- We want to understand the kinds of properties we can express on results produced by ΛY in this context because:
  - It gives simple decidability results
  - It allows to understand in a deeper way the nature of those properties and the kinds of algorithms they require
  - It may yield original domains of interpretation for ΛY

Models of ΛY

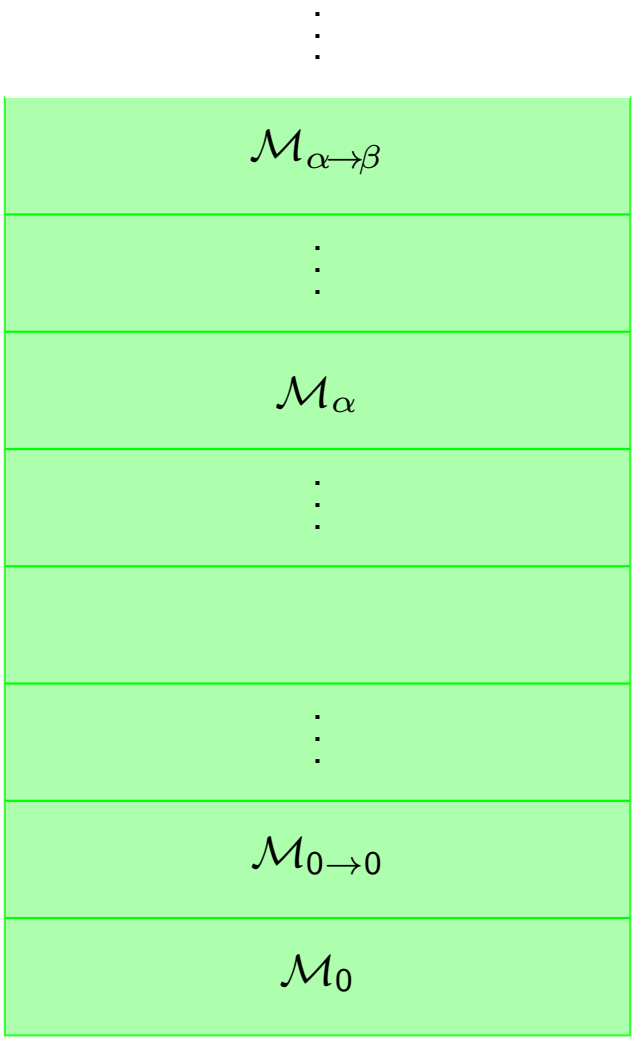
A model $\mathcal{M}$ is $((\mathcal{M}_\alpha)_{\alpha\in\mathcal{T}}, \bullet, \rho)$ where

- for every $f \in \mathcal{M}_{\alpha\to\beta}$ and $g \in \mathcal{M}_\alpha$, $f \bullet g \in \mathcal{M}_\beta$,

Models of ΛY

A model $\mathcal{M}$ is $((\mathcal{M}_\alpha)_{\alpha\in\mathcal{T}}, \bullet, \rho)$ where

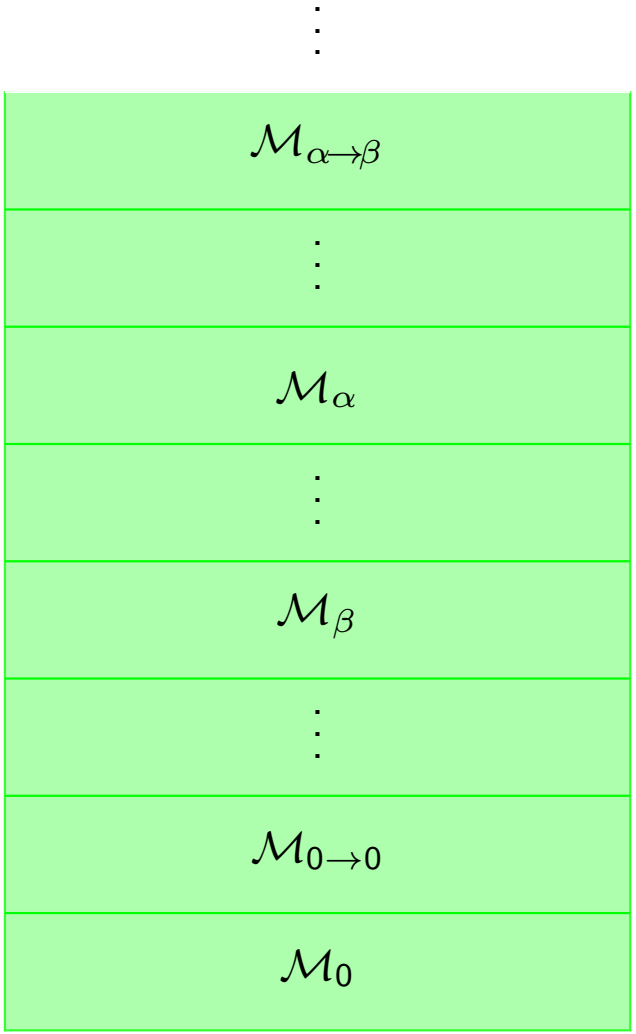
- for every $f \in \mathcal{M}_{\alpha\to\beta}$ and $g \in \mathcal{M}_\alpha$, $f \bullet g \in \mathcal{M}_\beta$,
- for every $f, f' \in \mathcal{M}_{\alpha\to\beta}$, for every $g \in \mathcal{M}_\alpha$ $f \bullet g = f' \bullet g$ implies $f = f'$



Models of ΛY

A model $\mathcal{M}$ is $((\mathcal{M}_\alpha)_{\alpha\in\mathcal{T}}, \bullet, \rho)$ where

- for every $f \in \mathcal{M}_{\alpha\to\beta}$ and $g \in \mathcal{M}_\alpha$, $f \bullet g \in \mathcal{M}_\beta$,
- for every $f, f' \in \mathcal{M}_{\alpha\to\beta}$, for every $g \in \mathcal{M}_\alpha$ $f \bullet g = f' \bullet g$ implies $f = f'$

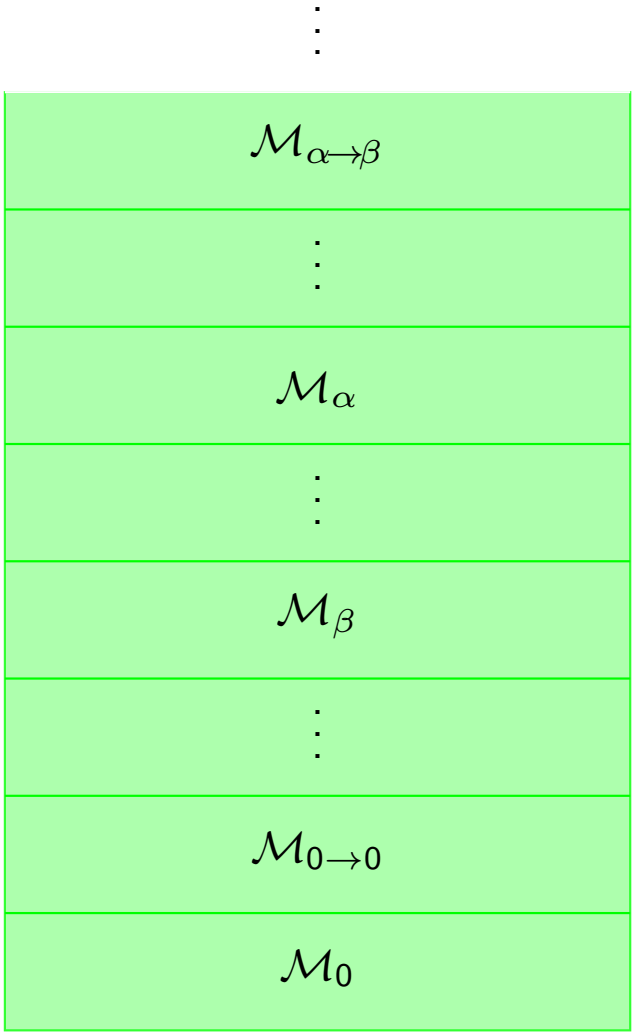


Models of ΛY

A model $\mathcal{M}$ is $((\mathcal{M}_\alpha)_{\alpha\in\mathcal{T}}, \bullet, \rho)$ where

- for every $f \in \mathcal{M}_{\alpha\to\beta}$ and $g \in \mathcal{M}_\alpha$, $f \bullet g \in \mathcal{M}_\beta$,
- for every $f, f' \in \mathcal{M}_{\alpha\to\beta}$, for every $g \in \mathcal{M}_\alpha$ $f \bullet g = f' \bullet g$ implies $f = f'$

Axioms of Interpretation



Models of ΛY

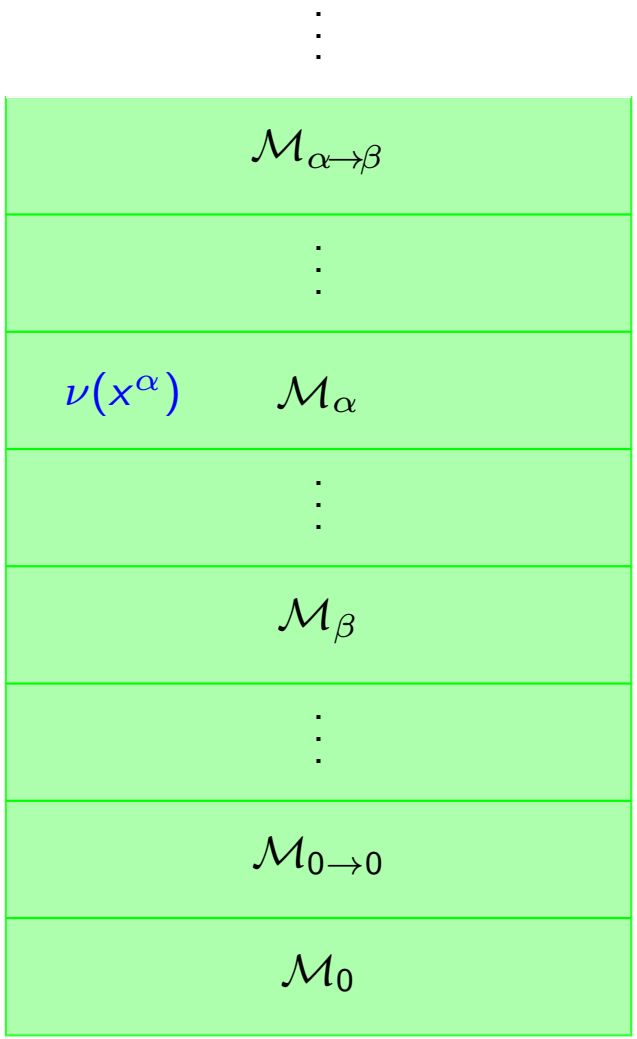
A model $\mathcal{M}$ is $((\mathcal{M}_\alpha)_{\alpha\in\mathcal{T}}, \bullet, \rho)$ where

- for every $f \in \mathcal{M}_{\alpha\to\beta}$ and $g \in \mathcal{M}_\alpha$, $f \bullet g \in \mathcal{M}_\beta$,
- for every $f, f' \in \mathcal{M}_{\alpha\to\beta}$, for every $g \in \mathcal{M}_\alpha$ $f \bullet g = f' \bullet g$ implies $f = f'$

Axioms of Interpretation

Given $\nu : \mathrm{Var} \to \mathcal{M}$.

- $[\![x^\alpha]\!]^{\mathcal{M}}_\nu = \nu(x^\alpha)$
- $[\![c^\alpha]\!]^{\mathcal{M}}_\nu = \rho(c^\alpha)$
- $[\![\lambda x^\alpha.M]\!]^{\mathcal{M}}_\nu \bullet a = [\![M]\!]^{\mathcal{M}}_{\nu[x:=a]}$
- $[\![M^{\alpha\to\beta} N^\alpha]\!]^{\mathcal{M}}_\nu = [\![M]\!]^{\mathcal{M}}_\nu \bullet [\![N]\!]^{\mathcal{M}}_\nu$
- $[\![Y]\!]^{\mathcal{M}}_\nu \bullet a = a \bullet ([\![Y]\!]^{\mathcal{M}}_\nu \bullet a)$



Models of ΛY

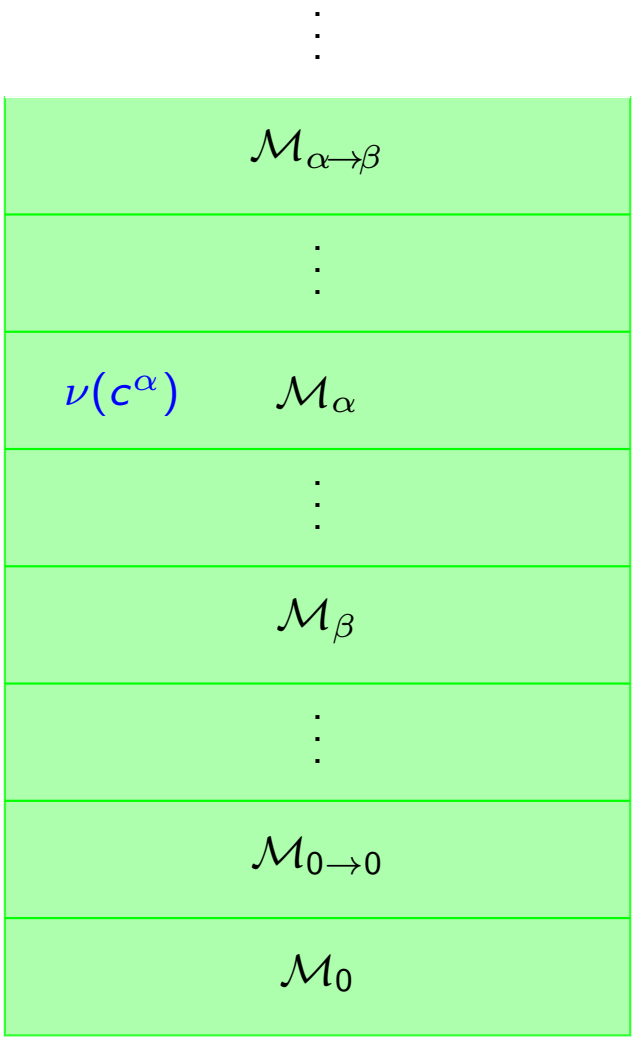
A model $\mathcal{M}$ is $((\mathcal{M}_\alpha)_{\alpha\in\mathcal{T}}, \bullet, \rho)$ where

- for every $f \in \mathcal{M}_{\alpha\to\beta}$ and $g \in \mathcal{M}_\alpha$, $f \bullet g \in \mathcal{M}_\beta$,
- for every $f, f' \in \mathcal{M}_{\alpha\to\beta}$, for every $g \in \mathcal{M}_\alpha$ $f \bullet g = f' \bullet g$ implies $f = f'$

Axioms of Interpretation

Given $\nu : \mathrm{Var} \to \mathcal{M}$.

- $[\![x^\alpha]\!]^{\mathcal{M}}_\nu = \nu(x^\alpha)$
- $[\![c^\alpha]\!]^{\mathcal{M}}_\nu = \rho(c^\alpha)$
- $[\![\lambda x^\alpha.M]\!]^{\mathcal{M}}_\nu \bullet a = [\![M]\!]^{\mathcal{M}}_{\nu[x:=a]}$
- $[\![M^{\alpha\to\beta} N^\alpha]\!]^{\mathcal{M}}_\nu = [\![M]\!]^{\mathcal{M}}_\nu \bullet [\![N]\!]^{\mathcal{M}}_\nu$
- $[\![Y]\!]^{\mathcal{M}}_\nu \bullet a = a \bullet ([\![Y]\!]^{\mathcal{M}}_\nu \bullet a)$



Models of ΛY

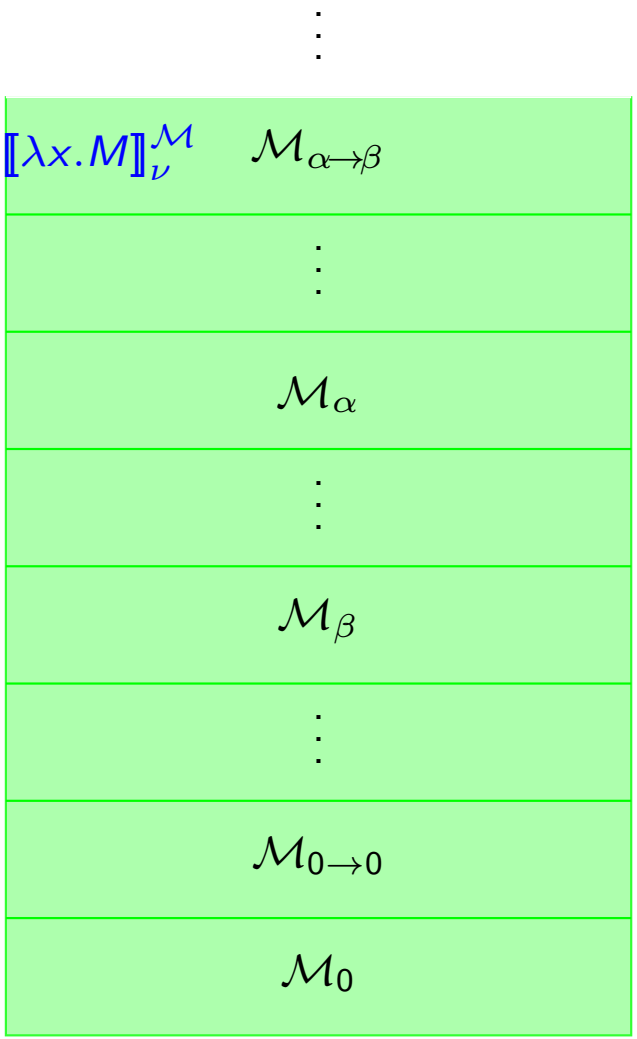
A model $\mathcal{M}$ is $((\mathcal{M}_\alpha)_{\alpha\in\mathcal{T}}, \bullet, \rho)$ where

- for every $f \in \mathcal{M}_{\alpha\to\beta}$ and $g \in \mathcal{M}_\alpha$, $f \bullet g \in \mathcal{M}_\beta$,
- for every $f, f' \in \mathcal{M}_{\alpha\to\beta}$, for every $g \in \mathcal{M}_\alpha$ $f \bullet g = f' \bullet g$ implies $f = f'$

Axioms of Interpretation

Given $\nu : \mathrm{Var} \to \mathcal{M}$.

- $[\![x^\alpha]\!]^{\mathcal{M}}_\nu = \nu(x^\alpha)$
- $[\![c^\alpha]\!]^{\mathcal{M}}_\nu = \rho(c^\alpha)$
- $[\![\lambda x^\alpha.M]\!]^{\mathcal{M}}_\nu \bullet a = [\![M]\!]^{\mathcal{M}}_{\nu[x:=a]}$
- $[\![M^{\alpha\to\beta} N^\alpha]\!]^{\mathcal{M}}_\nu = [\![M]\!]^{\mathcal{M}}_\nu \bullet [\![N]\!]^{\mathcal{M}}_\nu$
- $[\![Y]\!]^{\mathcal{M}}_\nu \bullet a = a \bullet ([\![Y]\!]^{\mathcal{M}}_\nu \bullet a)$



Models of ΛY

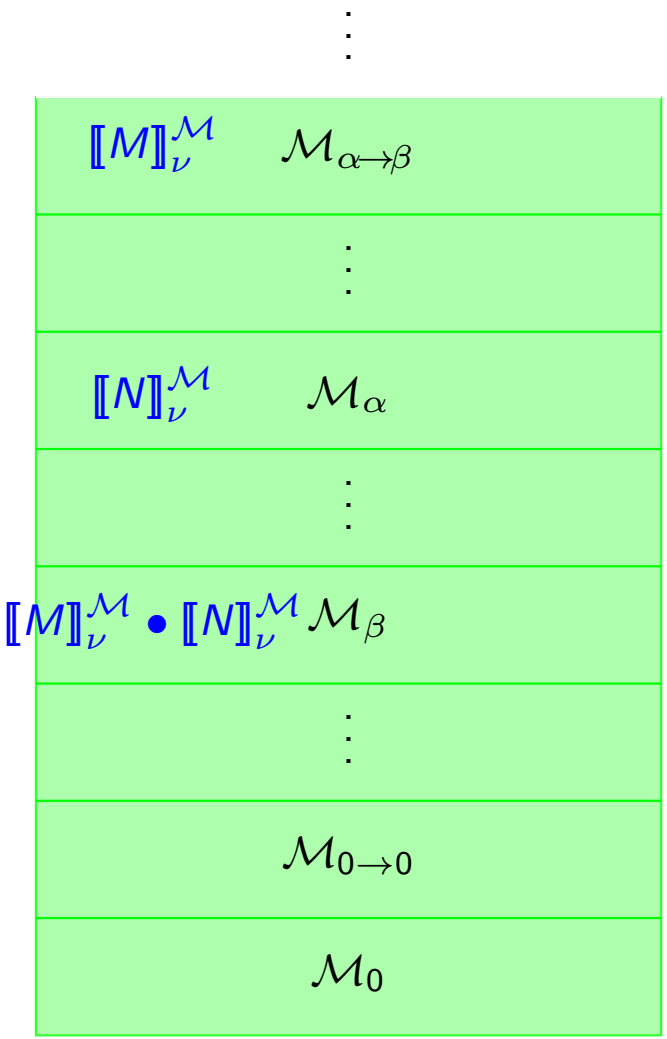
A model $\mathcal{M}$ is $((\mathcal{M}_\alpha)_{\alpha\in\mathcal{T}}, \bullet, \rho)$ where

- for every $f \in \mathcal{M}_{\alpha\to\beta}$ and $g \in \mathcal{M}_\alpha$, $f \bullet g \in \mathcal{M}_\beta$,
- for every $f, f' \in \mathcal{M}_{\alpha\to\beta}$, for every $g \in \mathcal{M}_\alpha$ $f \bullet g = f' \bullet g$ implies $f = f'$

Axioms of Interpretation

Given $\nu : \mathrm{Var} \to \mathcal{M}$.

- $[\![x^\alpha]\!]^{\mathcal{M}}_\nu = \nu(x^\alpha)$
- $[\![c^\alpha]\!]^{\mathcal{M}}_\nu = \rho(c^\alpha)$
- $[\![\lambda x^\alpha.M]\!]^{\mathcal{M}}_\nu \bullet a = [\![M]\!]^{\mathcal{M}}_{\nu[x:=a]}$
- $[\![M^{\alpha\to\beta} N^\alpha]\!]^{\mathcal{M}}_\nu = [\![M]\!]^{\mathcal{M}}_\nu \bullet [\![N]\!]^{\mathcal{M}}_\nu$
- $[\![Y]\!]^{\mathcal{M}}_\nu \bullet a = a \bullet ([\![Y]\!]^{\mathcal{M}}_\nu \bullet a)$



Models of ΛY

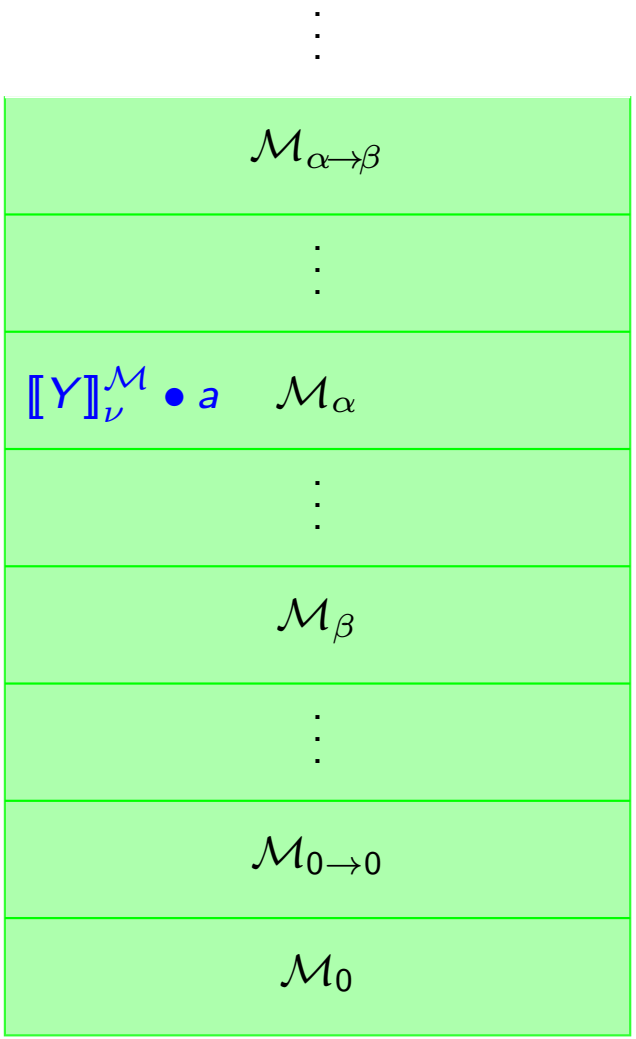
A model $\mathcal{M}$ is $((\mathcal{M}_\alpha)_{\alpha\in\mathcal{T}}, \bullet, \rho)$ where

- for every $f \in \mathcal{M}_{\alpha\to\beta}$ and $g \in \mathcal{M}_\alpha$, $f \bullet g \in \mathcal{M}_\beta$,
- for every $f, f' \in \mathcal{M}_{\alpha\to\beta}$, for every $g \in \mathcal{M}_\alpha$ $f \bullet g = f' \bullet g$ implies $f = f'$

Axioms of Interpretation

Given $\nu : \mathrm{Var} \to \mathcal{M}$.

- $[\![x^\alpha]\!]^{\mathcal{M}}_\nu = \nu(x^\alpha)$
- $[\![c^\alpha]\!]^{\mathcal{M}}_\nu = \rho(c^\alpha)$
- $[\![\lambda x^\alpha.M]\!]^{\mathcal{M}}_\nu \bullet a = [\![M]\!]^{\mathcal{M}}_{\nu[x:=a]}$
- $[\![M^{\alpha\to\beta} N^\alpha]\!]^{\mathcal{M}}_\nu = [\![M]\!]^{\mathcal{M}}_\nu \bullet [\![N]\!]^{\mathcal{M}}_\nu$
- $[\![Y]\!]^{\mathcal{M}}_\nu \bullet a = a \bullet ([\![Y]\!]^{\mathcal{M}}_\nu \bullet a)$



Models and recognition

Given a model $\mathcal{M}$, and $A \subseteq \mathcal{M}_\alpha$, A recognizes the language $\{M \mid [\![M]\!]^{\mathcal{M}} \in A\}$.

Known properties of models

Theorem (Henkin)

The following are equivalent:

- $M =_{\beta\delta\eta} N$
- *for all model* $\mathcal{M}$ *and every valuation* ν, $[\![M]\!]^{\mathcal{M}}_{\nu} = [\![N]\!]^{\mathcal{M}}_{\nu}$

Theorem (Statman 03)

Whether $M =_{\beta\delta\eta} N$ *is undecidable.*

Models are in general not effective.

The Monotone Model of λY

The monotone model over a finite lattice $(\mathcal{P}(X), \subseteq)$ is

$$\mathcal{D}_X = (\{(\mathcal{D}_\alpha, \sqsubseteq_\alpha)\}_{A \in \mathcal{T}}, \rho) \quad \rho : \mathsf{Cst} \to \mathcal{D}_X$$

where

- $\mathcal{D}_o = \mathcal{P}(X)$ and $f \sqsubseteq_o g$ iff $f \subseteq g$,

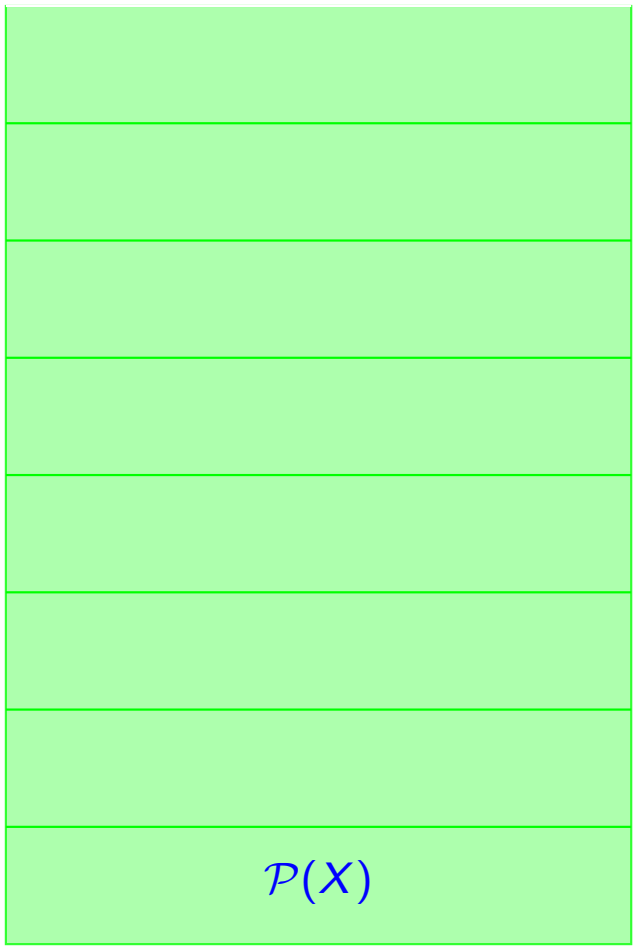
The Monotone Model of λY

The monotone model over a finite lattice $(\mathcal{P}(X), \subseteq)$ is

$$\mathcal{D}_X = (\{(\mathcal{D}_\alpha, \sqsubseteq_\alpha)\}_{A \in \mathcal{T}}, \rho) \quad \rho : \mathsf{Cst} \to \mathcal{D}_X$$

where

- $\mathcal{D}_o = \mathcal{P}(X)$ and $f \sqsubseteq_o g$ iff $f \subseteq g$,



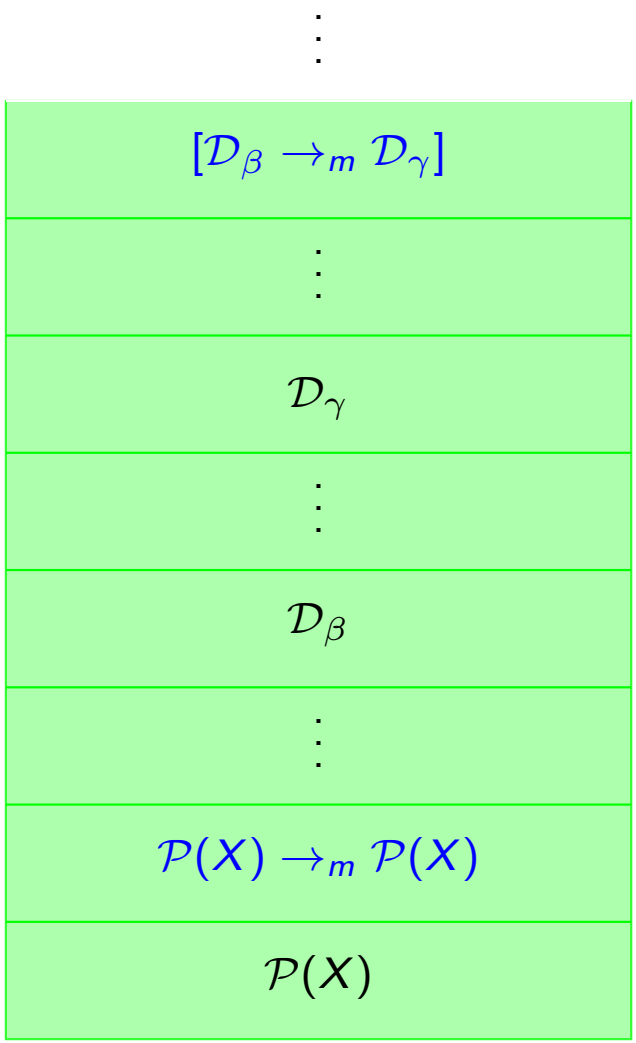
The Monotone Model of λY

The monotone model over a finite lattice $(\mathcal{P}(X), \subseteq)$ is

$$\mathcal{D}_X = (\{(\mathcal{D}_\alpha, \sqsubseteq_\alpha)\}_{A \in \mathcal{T}}, \rho) \quad \rho : \mathsf{Cst} \to \mathcal{D}_X$$

where

- $\mathcal{D}_o = \mathcal{P}(X)$ and $f \sqsubseteq_o g$ iff $f \subseteq g$,
- $\mathcal{D}_{\beta\to\gamma} = [\mathcal{D}_\beta \to_m \mathcal{D}_\gamma]$
  $\sqsubseteq_{\beta\to\gamma}$= pointwise ordering.



The Monotone Model of λY

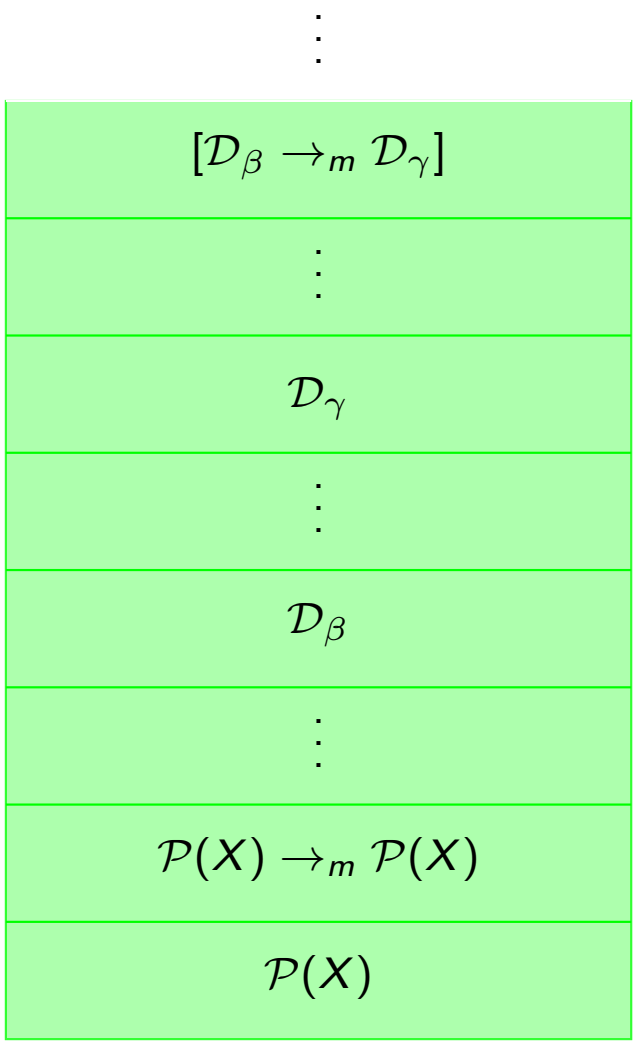
The monotone model over a finite lattice $(\mathcal{P}(X), \subseteq)$ is

$$\mathcal{D}_X = (\{(\mathcal{D}_\alpha, \sqsubseteq_\alpha)\}_{A \in \mathcal{T}}, \rho) \quad \rho : \mathsf{Cst} \to \mathcal{D}_X$$

where

- $\mathcal{D}_o = \mathcal{P}(X)$ and $f \sqsubseteq_o g$ iff $f \subseteq g$,
- $\mathcal{D}_{\beta\to\gamma} = [\mathcal{D}_\beta \to_m \mathcal{D}_\gamma]$
  $\sqsubseteq_{\beta\to\gamma}$= pointwise ordering.

$[\![Y^\alpha]\!]^{\mathcal{D}_X}_\nu(a) = \bigwedge_{n\in\mathbb{N}} a^n(\top_\alpha)$



Known properties of monotone models

Theorem (Statman 82)

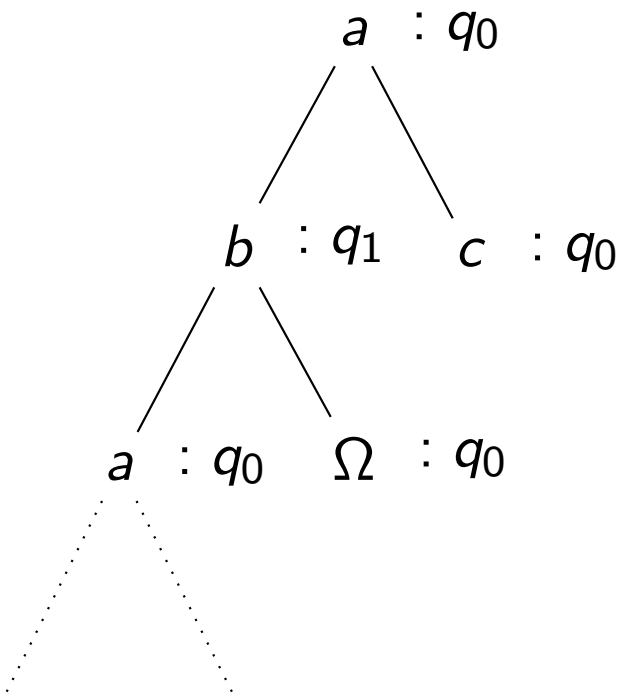
The following are equivalent:

- $BT(M) = BT(N)$,
- *for every monotone models* $\mathcal{M}$ *and every valuation* ν, $[\![M]\!]^{\mathcal{M}}_{\nu} = [\![N]\!]^{\mathcal{M}}_{\nu}$.

Theorem (Loader 00)

Given a monotone model $\mathcal{M}$ *and* f *in* $\mathcal{M}_\alpha$, *whether there is* M *so that* $[\![M]\!]^{\mathcal{M}} = f$ *is undecidable.*

Automata with Trivial Acceptance Condition (TAC) and Böhm trees



a, q_0	(q_1, q_0)
a, q_0	(q_0, q_0)
b, q_1	(q_0, q_1)

c	q_0
Ω	q_0, q_1

Ω-blind TAC: Ω is accepted by any state

Insightful TAC: Ω is accepted only by certain states

Ω-blind TAC and monotone models

Theorem

Given $\mathcal{D}_X$ and A, M is recognized by A iff $BT(M)$ is accepted by a boolean combination of automata with Ω-blind TAC.

Ω-blind TAC and monotone models

Theorem

Given $\mathcal{D}_X$ and A, M is recognized by A iff $BT(M)$ is accepted by a boolean combination of automata with Ω-blind TAC.

Proof $\Leftarrow$

$\mathcal{A} = (Q, \delta)$

Take the monotone model $\mathcal{M}$ so that $\mathcal{M}_0 = \mathcal{P}(Q)$ and

- $[\![a]\!](Q_1, Q_2) = \{q \mid \exists (q_1, q_2).(q_1, q_2) \in (Q_1 \times Q_2) \cap \delta(a, q)\}$
- $[\![c]\!] = \delta(c)$

$q \in [\![M]\!]^{\mathcal{M}}$ iff $\mathcal{A}$ accepts $BT(M)$ from q.

Ω-blind TAC and monotone models

Theorem

Given $\mathcal{D}_X$ and A, M is recognized by A iff $BT(M)$ is accepted by a boolean combination of automata with Ω-blind TAC.

Proof $\Rightarrow$

Given a monotone model $\mathcal{M}$ we define $\mathcal{A}_{\mathcal{M}} = (\mathcal{M}_0, \delta)$ so that:

- $\delta(q, a) = \{(q_1, q_2) \mid q \leq [\![a]\!](q_1, q_2)\}$
- $\delta(c) = [\![c]\!]$.

$\mathcal{A}_{\mathcal{M}}$ accepts $BT(M)$ from q iff $[\![M]\!] \geq q$.

Beyond monotone models

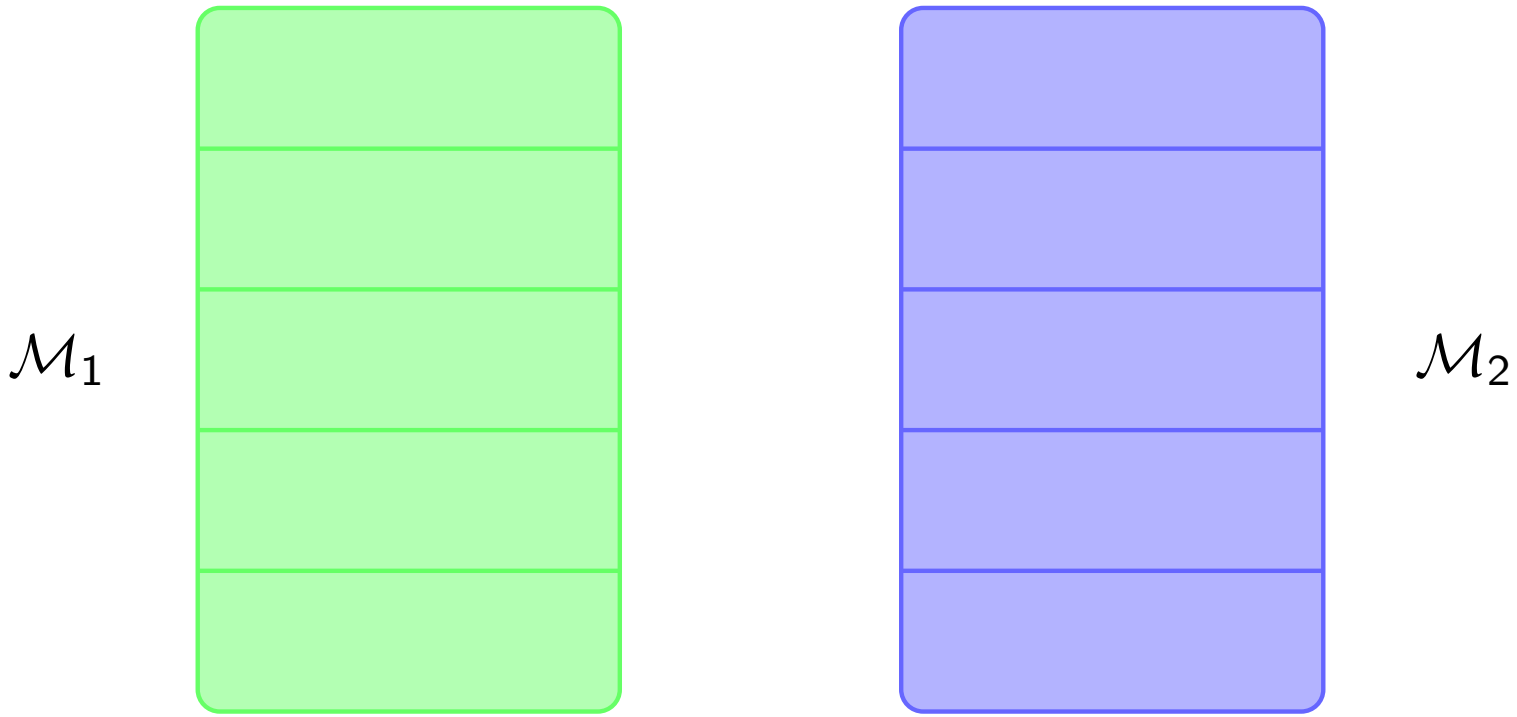
We are going to construct models for

- Insightful automata
- weak MSO

Important ingredients

- Logical relations
- Galois connections

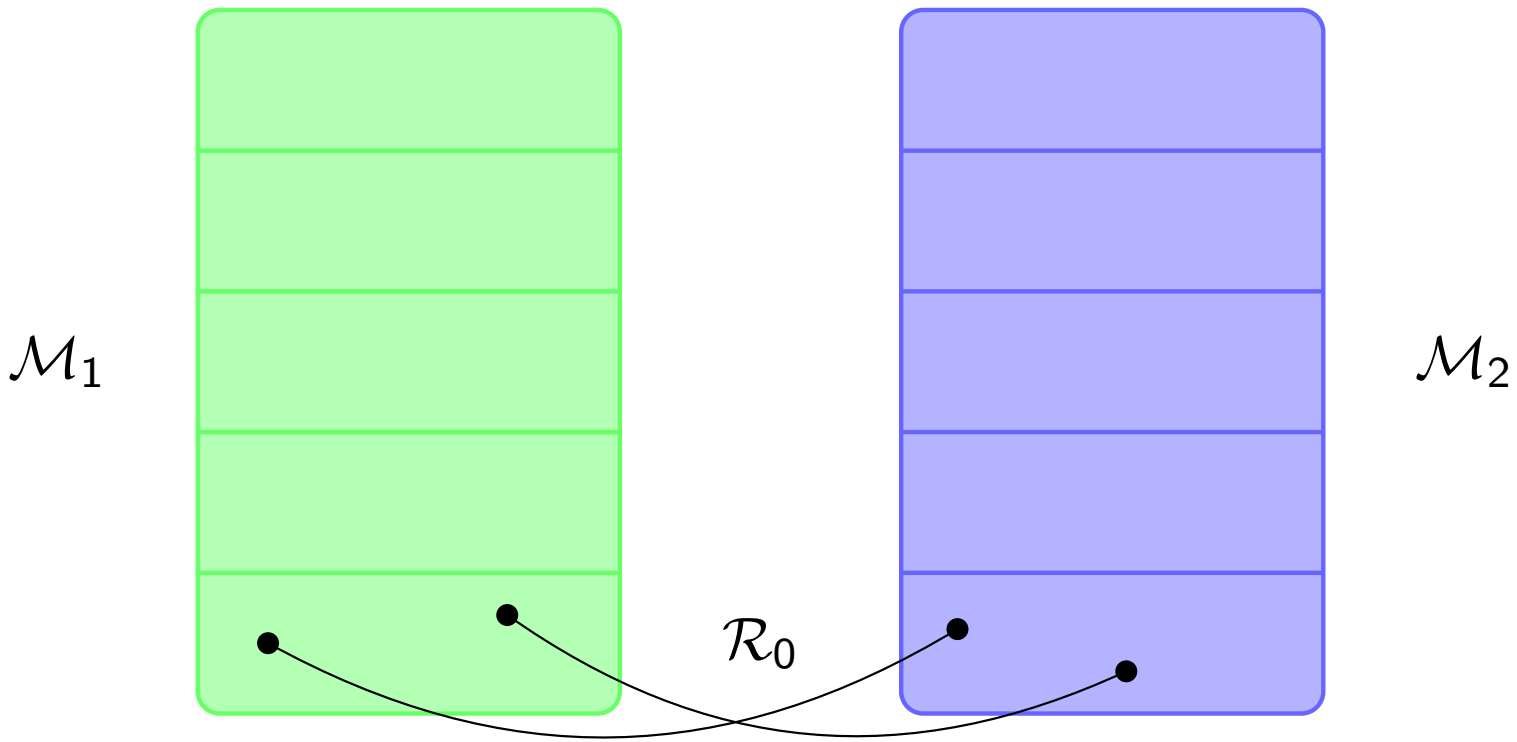
Logical relations



Lemma (Fundamental Lemma)

If $[\![Y]\!]^{\mathcal{M}_1} \mathcal{R}\ [\![Y]\!]^{\mathcal{M}_2}$, $[\![c]\!]^{\mathcal{M}_1}\ \mathcal{R}\ [\![c]\!]^{\mathcal{M}_2}$, ν_1 *and* ν_2 *so that for every* x, $\nu_1(x)\ \mathcal{R}\ \nu_2(x)$ *then for every* M: $[\![M]\!]^{\mathcal{M}_1}_{\nu_1}\ \mathcal{R}\ [\![M]\!]^{\mathcal{M}_2}_{\nu_2}$

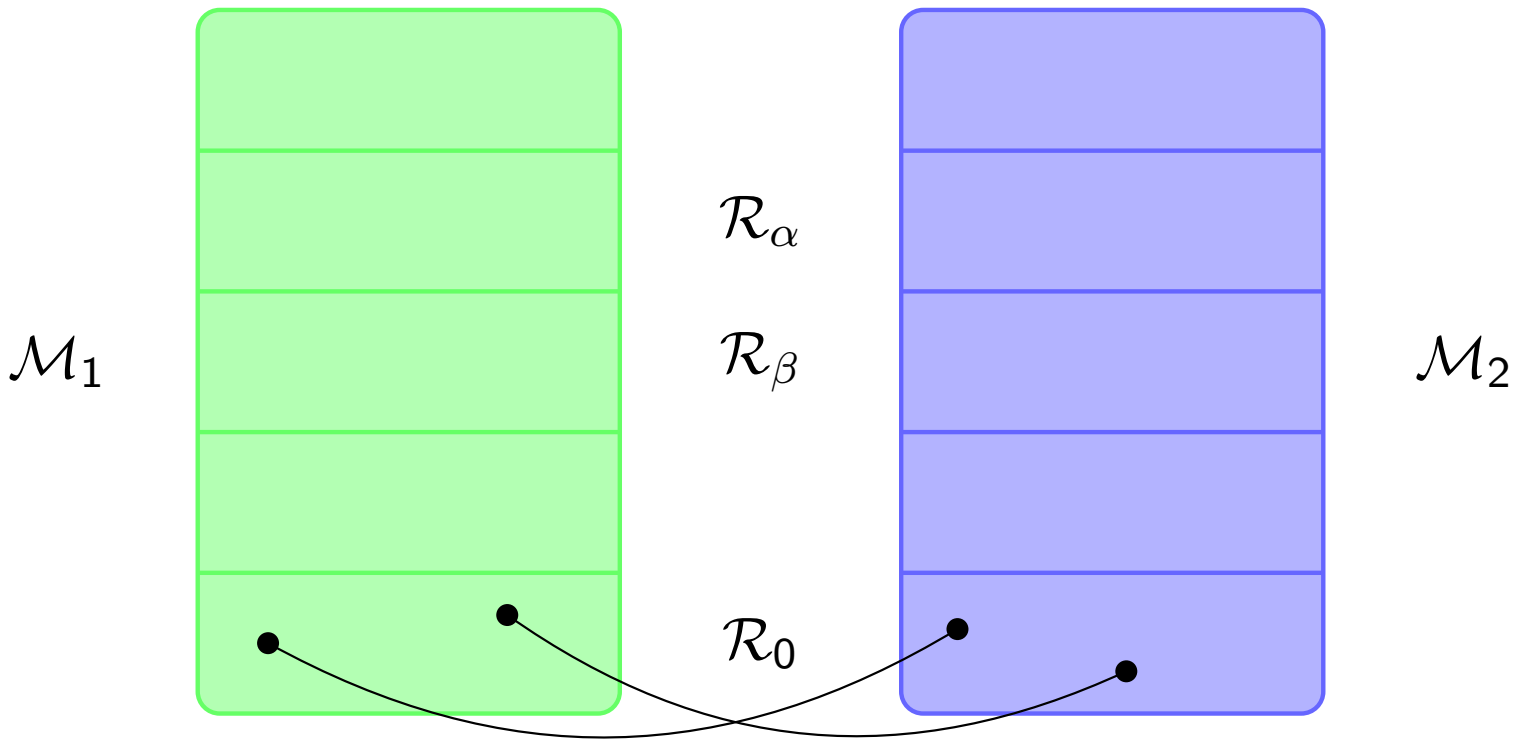
Logical relations



Lemma (Fundamental Lemma)

If $[\![Y]\!]^{\mathcal{M}_1} \mathcal{R}\ [\![Y]\!]^{\mathcal{M}_2}$, $[\![c]\!]^{\mathcal{M}_1}\ \mathcal{R}\ [\![c]\!]^{\mathcal{M}_2}$, ν_1 *and* ν_2 *so that for every* x, $\nu_1(x)\ \mathcal{R}\ \nu_2(x)$ *then for every* M: $[\![M]\!]^{\mathcal{M}_1}_{\nu_1}\ \mathcal{R}\ [\![M]\!]^{\mathcal{M}_2}_{\nu_2}$

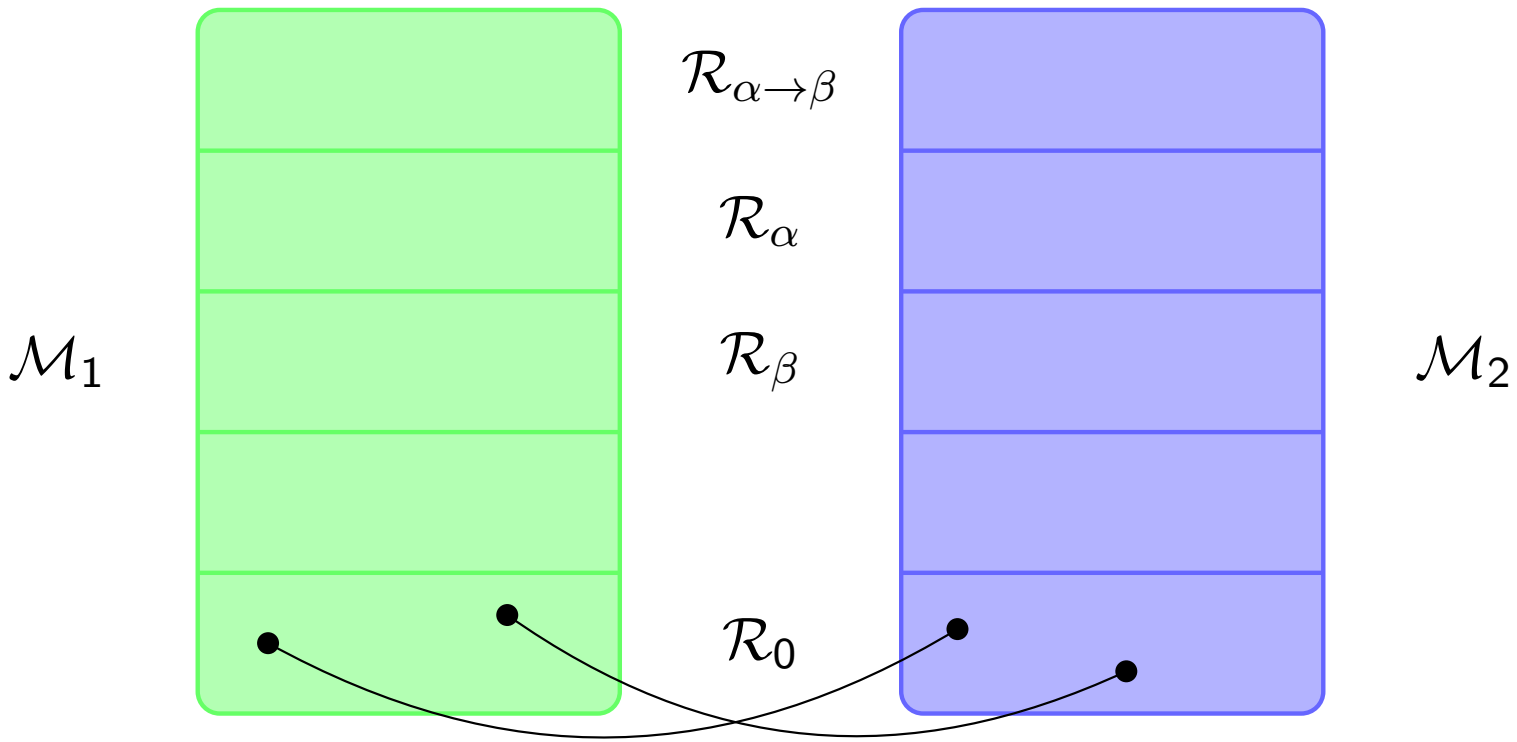
Logical relations



Lemma (Fundamental Lemma)

If $[\![Y]\!]^{\mathcal{M}_1} \mathcal{R} [\![Y]\!]^{\mathcal{M}_2}$, $[\![c]\!]^{\mathcal{M}_1} \mathcal{R} [\![c]\!]^{\mathcal{M}_2}$, ν_1 *and* ν_2 *so that for every* x, $\nu_1(x) \mathcal{R} \nu_2(x)$ *then for every* M: $[\![M]\!]^{\mathcal{M}_1}_{\nu_1} \mathcal{R} [\![M]\!]^{\mathcal{M}_2}_{\nu_2}$

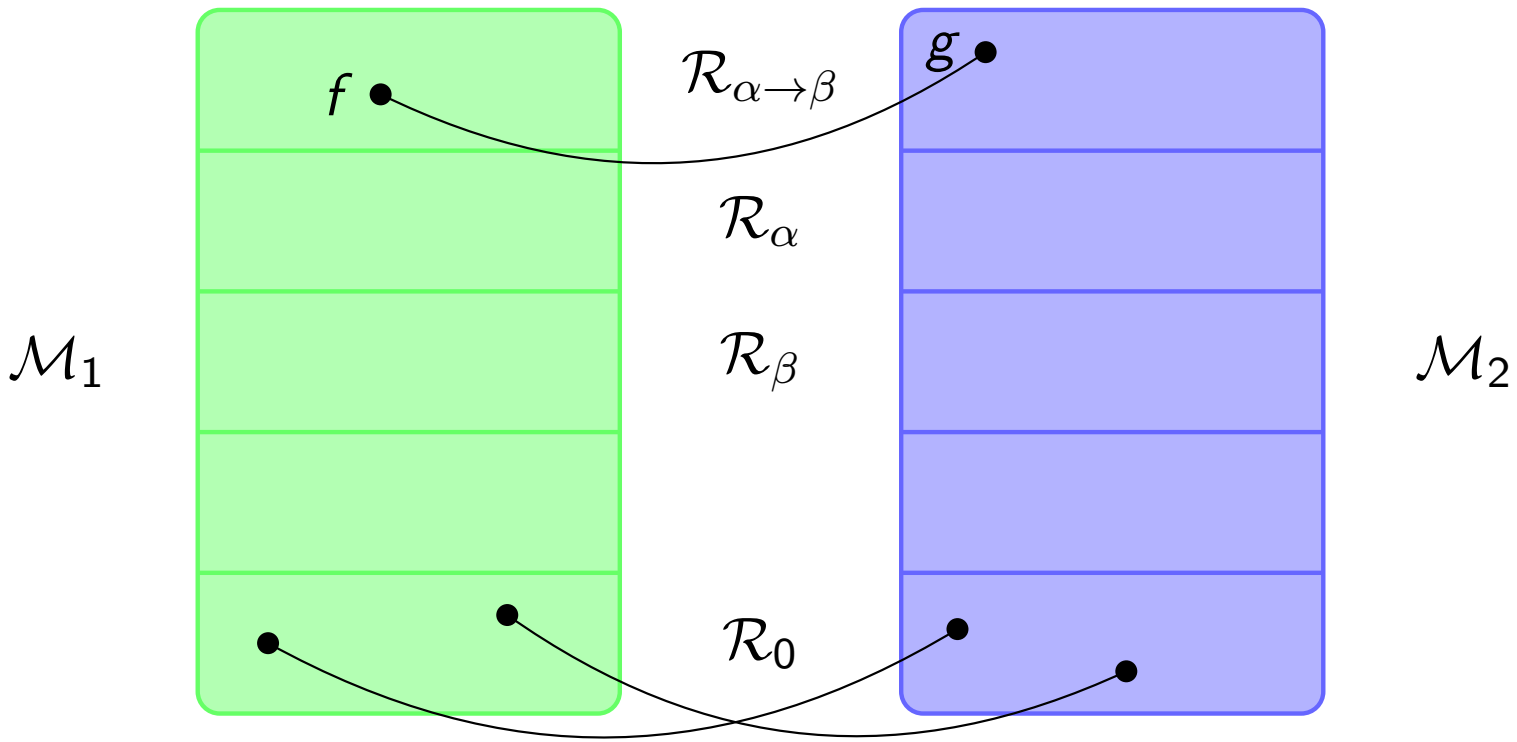
Logical relations



Lemma (Fundamental Lemma)

If $[\![Y]\!]^{\mathcal{M}_1} \mathcal{R} [\![Y]\!]^{\mathcal{M}_2}$, $[\![c]\!]^{\mathcal{M}_1} \mathcal{R} [\![c]\!]^{\mathcal{M}_2}$, ν_1 *and* ν_2 *so that for every* x, $\nu_1(x) \mathcal{R} \nu_2(x)$ *then for every* M: $[\![M]\!]^{\mathcal{M}_1}_{\nu_1} \mathcal{R} [\![M]\!]^{\mathcal{M}_2}_{\nu_2}$

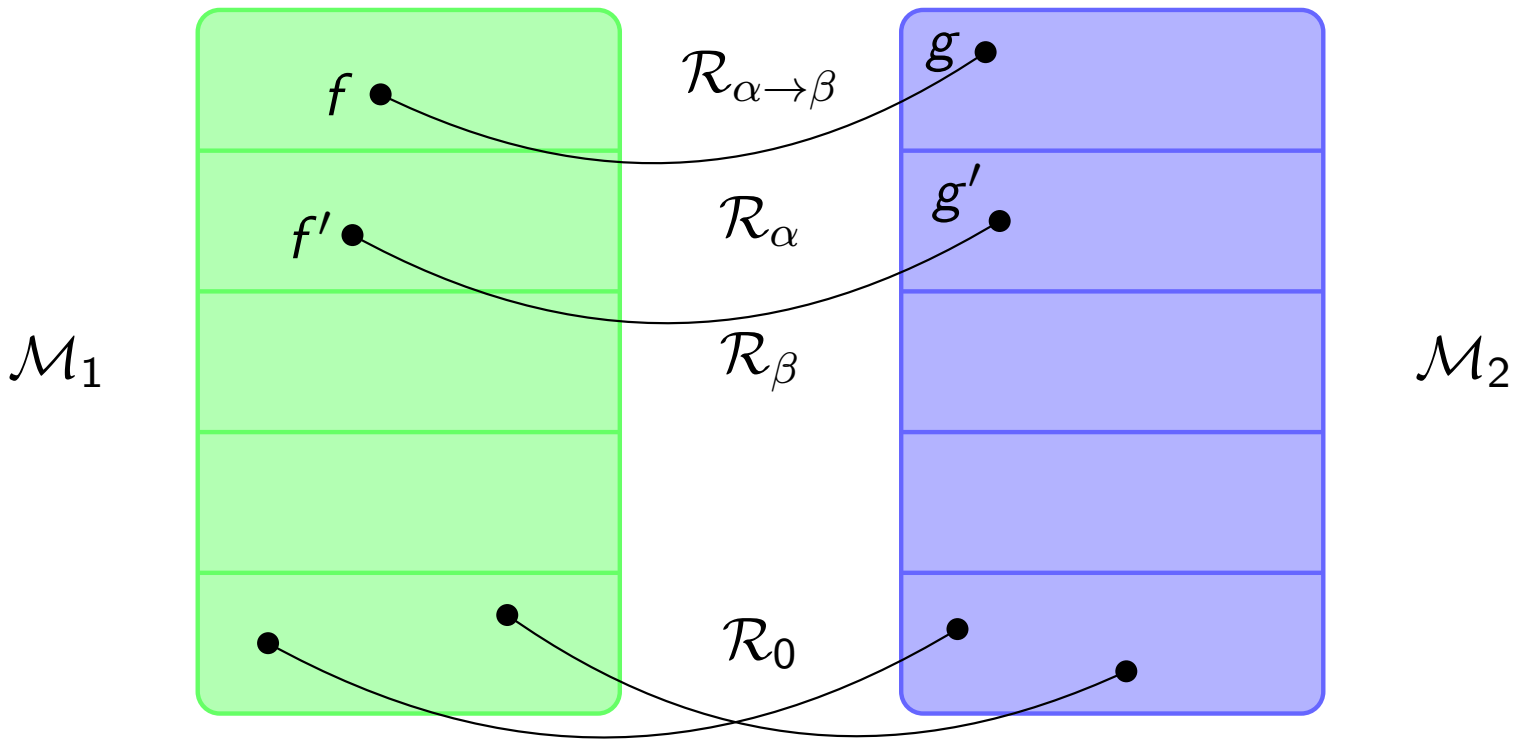
Logical relations



Lemma (Fundamental Lemma)

If $[\![Y]\!]^{\mathcal{M}_1} \mathcal{R} [\![Y]\!]^{\mathcal{M}_2}$, $[\![c]\!]^{\mathcal{M}_1} \mathcal{R} [\![c]\!]^{\mathcal{M}_2}$, ν_1 *and* ν_2 *so that for every* x, $\nu_1(x) \mathcal{R} \nu_2(x)$ *then for every* M: $[\![M]\!]^{\mathcal{M}_1}_{\nu_1} \mathcal{R} [\![M]\!]^{\mathcal{M}_2}_{\nu_2}$

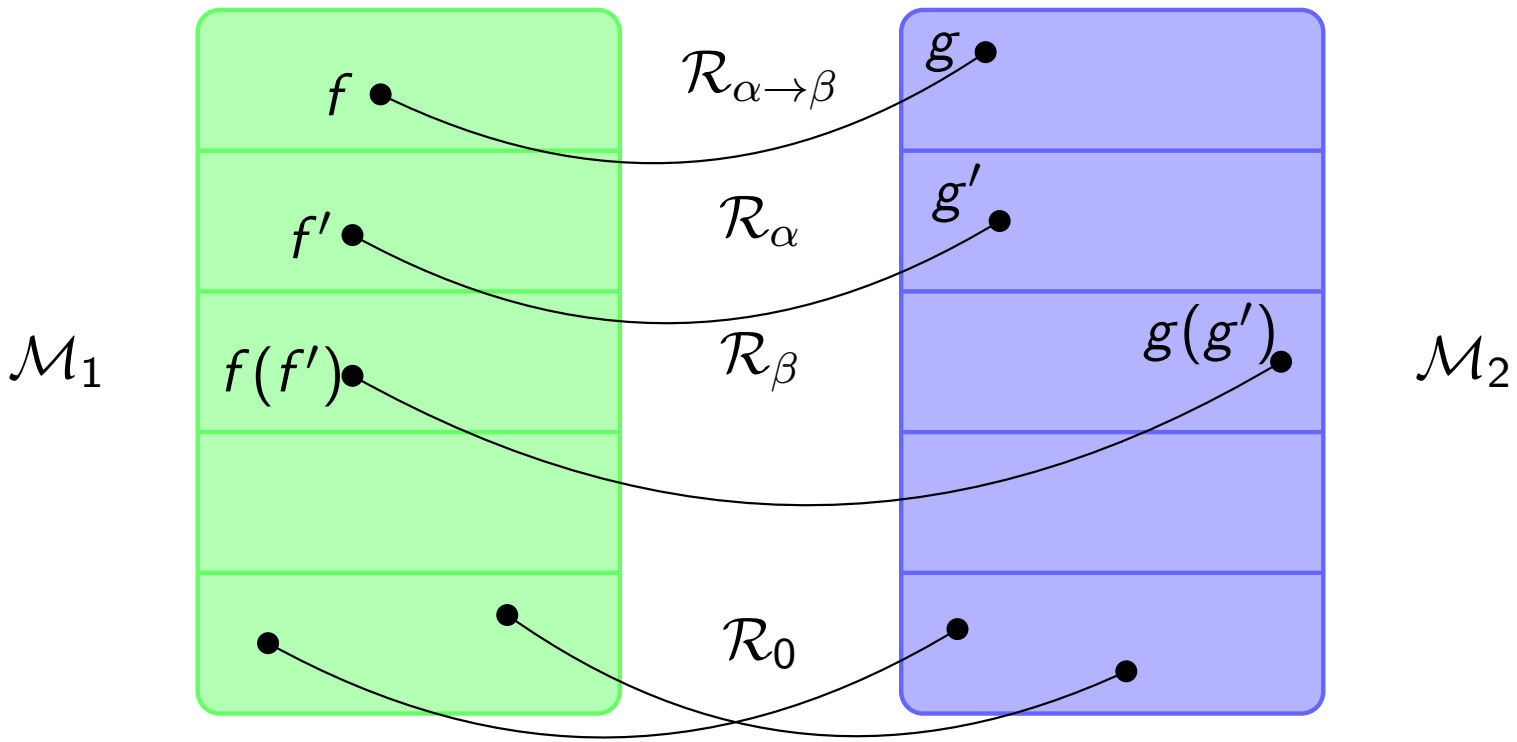
Logical relations



Lemma (Fundamental Lemma)

If $[\![Y]\!]^{\mathcal{M}_1} \mathcal{R} [\![Y]\!]^{\mathcal{M}_2}$, $[\![c]\!]^{\mathcal{M}_1} \mathcal{R} [\![c]\!]^{\mathcal{M}_2}$, ν_1 *and* ν_2 *so that for every* x, $\nu_1(x) \mathcal{R} \nu_2(x)$ *then for every* M: $[\![M]\!]^{\mathcal{M}_1}_{\nu_1} \mathcal{R} [\![M]\!]^{\mathcal{M}_2}_{\nu_2}$

Logical relations



Lemma (Fundamental Lemma)

If $[\![Y]\!]^{\mathcal{M}_1} \mathcal{R} [\![Y]\!]^{\mathcal{M}_2}$, $[\![c]\!]^{\mathcal{M}_1} \mathcal{R} [\![c]\!]^{\mathcal{M}_2}$, ν_1 *and* ν_2 *so that for every* x, $\nu_1(x) \mathcal{R} \nu_2(x)$ *then for every* M: $[\![M]\!]^{\mathcal{M}_1}_{\nu_1} \mathcal{R} [\![M]\!]^{\mathcal{M}_2}_{\nu_2}$

A model for detecting Ω

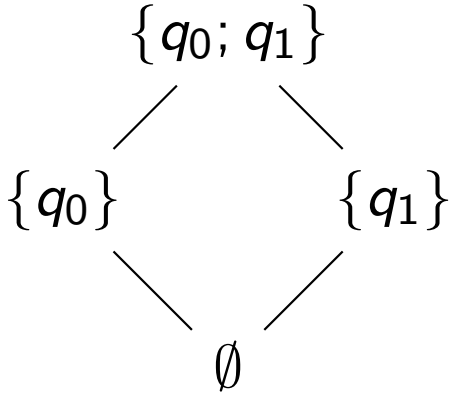
The monotone model $\mathcal{D}$ where Y is interpreted as least fixpoint and which is generated by the lattice:

$$[\![a]\!](x, y) = \top$$

is so that $BT(M) = \Omega$ iff $[\![M]\!]^{\mathcal{D}} = \bot$.

Constructing a model for insightful automata

We want to accept Ω only with state q_0

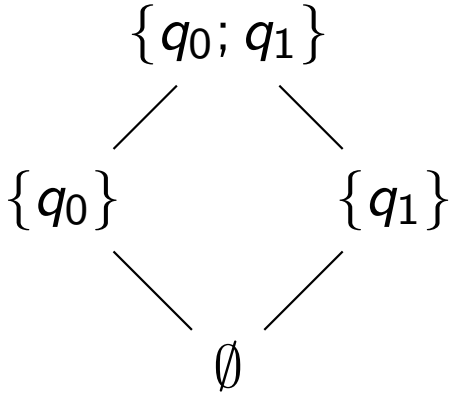


$$[\![a]\!](x, y) = \top \qquad [\![a]\!](Q, Q') = \{q \mid \delta(a, q) \subseteq Q \times Q'\}$$

- we need to interpret Ω as $(\bot, \{q_0\})$,
- make that compatible with higher-order and Y interpretation

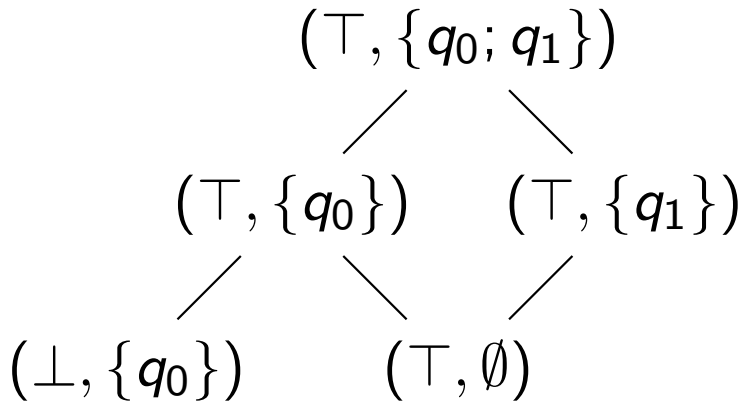
Constructing a model for insightful automata

We want to accept Ω only with state q_0



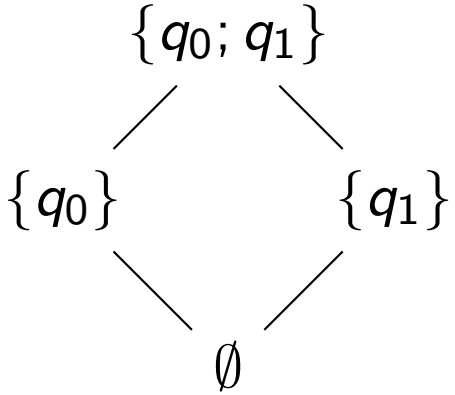
$$[\![a]\!](x, y) = \top \qquad [\![a]\!](Q, Q') = \{q \mid \delta(a, q) \subseteq Q \times Q'\}$$

- we need to interpret Ω as $(\bot, \{q_0\})$,
- make that compatible with higher-order and Y interpretation



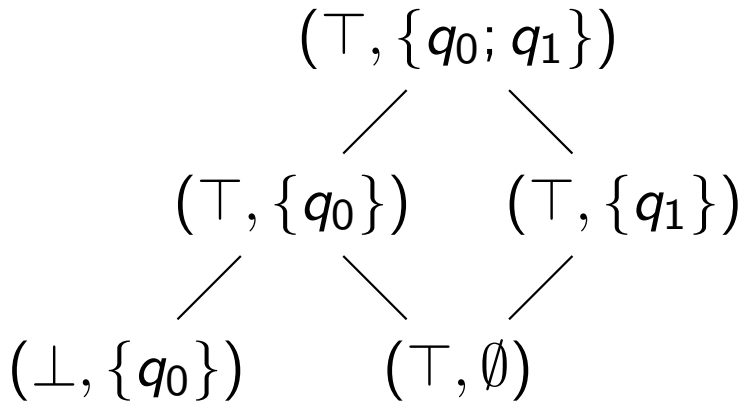
Constructing a model for insightful automata

We want to accept Ω only with state q_0



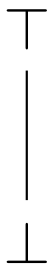
$$[\![a]\!](x, y) = \top \qquad [\![a]\!](Q, Q') = \{q \mid \delta(a, q) \subseteq Q \times Q'\}$$

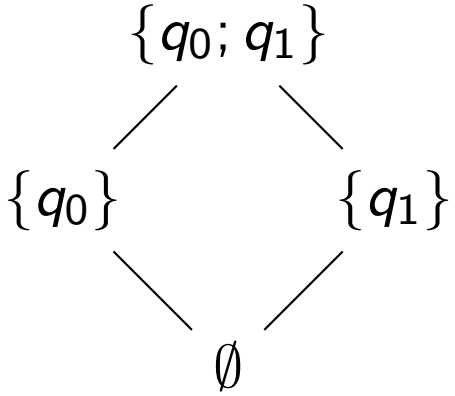
- we need to interpret Ω as $(\bot, \{q_0\})$,
- make that compatible with higher-order and Y interpretation



Constructing a model for insightful automata

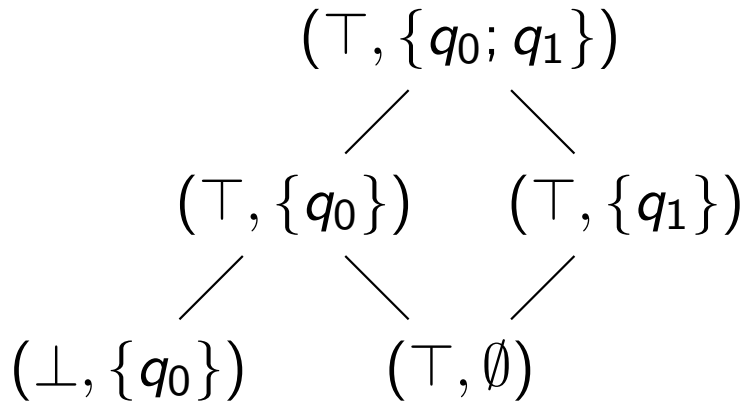
We want to accept Ω only with state q_0



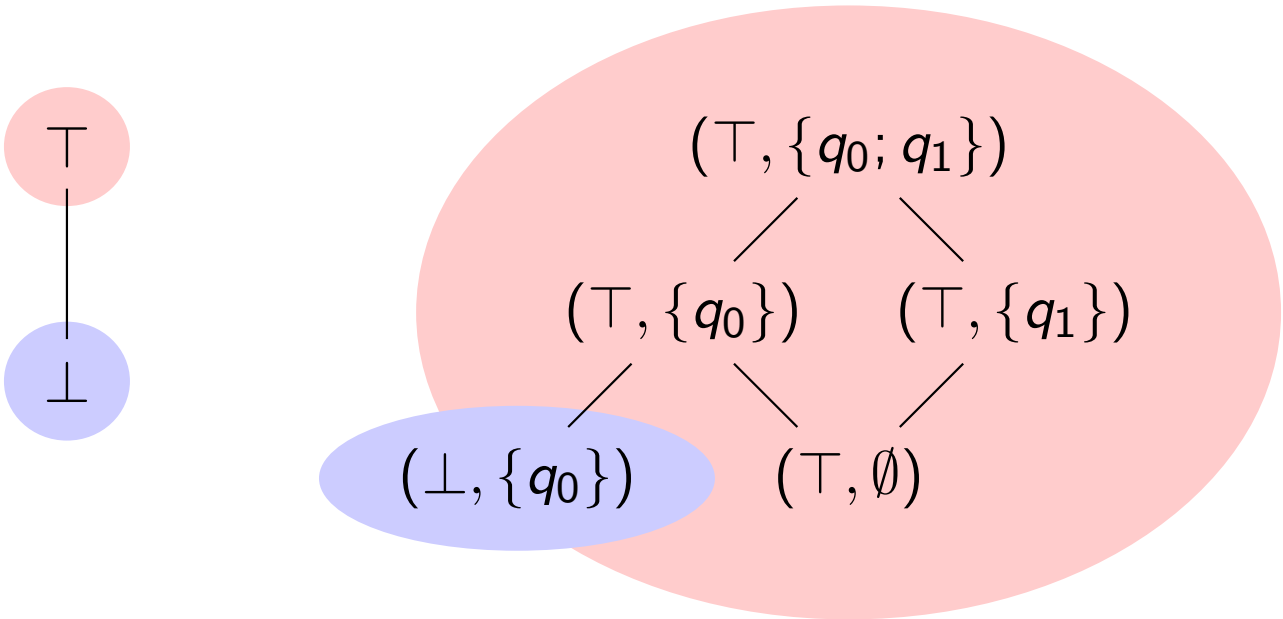


$$[\![a]\!](x, y) = \top \qquad [\![a]\!](Q, Q') = \{q \mid \delta(a, q) \subseteq Q \times Q'\}$$

- we need to interpret Ω as $(\bot, \{q_0\})$,
- make that compatible with higher-order and Y interpretation

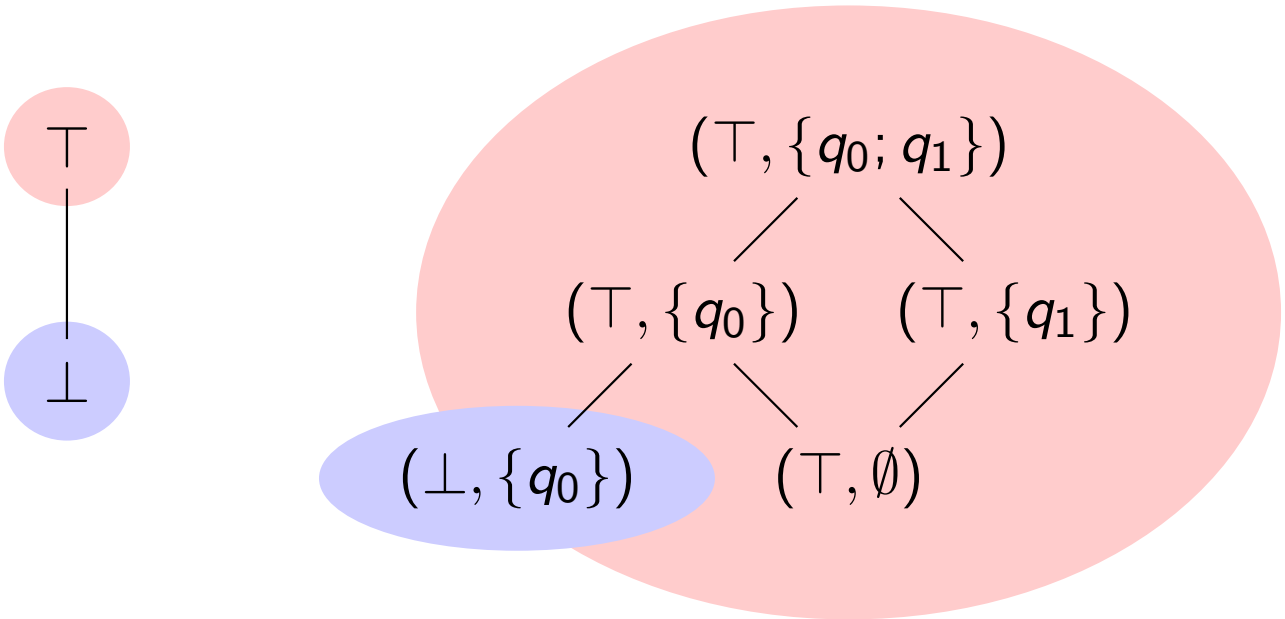


Going higher-order with a logical relation



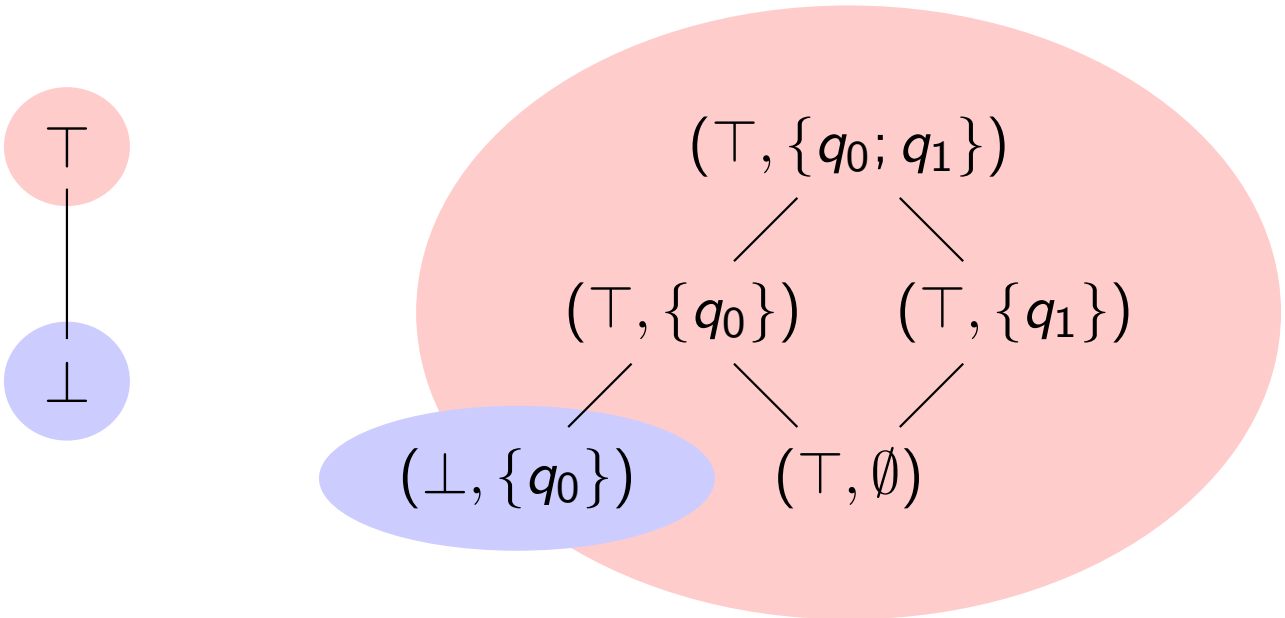
- $\mathcal{L}_0 = \{((d, P), d) \mid (d, P) \in \mathcal{K}_0\}$,
- $\mathcal{K}_{\alpha\to\beta} = \{f \in \mathrm{mon}[\mathcal{K}_\alpha \to \mathcal{K}_\beta] \mid \exists d \in \mathcal{D}_{\alpha\to\beta}. \forall (g, e) \in \mathcal{L}_\alpha. (f(g), d(e)) \in \mathcal{L}_\beta\}$,
- $\mathcal{L}_{\alpha\to\beta} = \{(f, d) \in \mathcal{K}_{\alpha\to\beta} \times \mathcal{D}_{\alpha\to\beta} \mid \forall (g, e) \in \mathcal{L}_\alpha. (f(g), d(e)) \in \mathcal{L}_\beta\}$.

Going higher-order with a logical relation



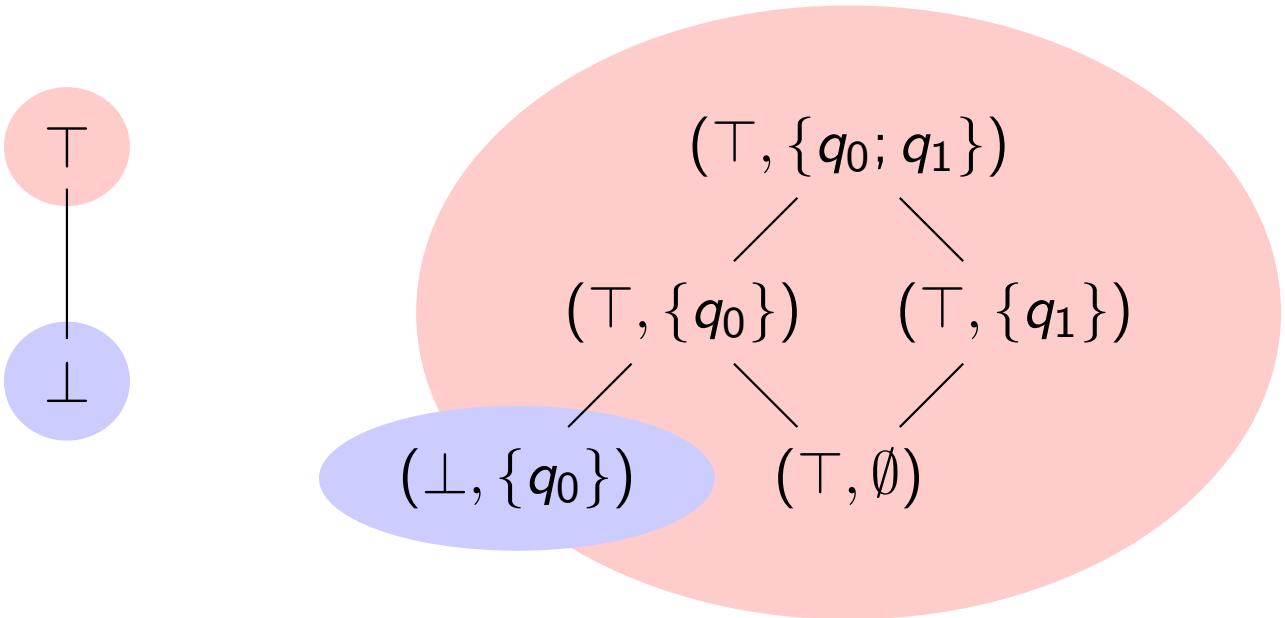
- $\mathcal{L}_0 = \{((d, P), d) \mid (d, P) \in \mathcal{K}_0\}$,
- $\mathcal{K}_{\alpha\to\beta} = \{f \in \mathrm{mon}[\mathcal{K}_\alpha \to \mathcal{K}_\beta] \mid \exists d \in \mathcal{D}_{\alpha\to\beta}. \forall (g, e) \in \mathcal{L}_\alpha. (f(g), d(e)) \in \mathcal{L}_\beta\}$,
- $\mathcal{L}_{\alpha\to\beta} = \{(f, d) \in \mathcal{K}_{\alpha\to\beta} \times \mathcal{D}_{\alpha\to\beta} \mid \forall (g, e) \in \mathcal{L}_\alpha. (f(g), d(e)) \in \mathcal{L}_\beta\}$.

Going higher-order with a logical relation



- $\mathcal{L}_0 = \{((d, P), d) \mid (d, P) \in \mathcal{K}_0\}$,
- $\mathcal{K}_{\alpha\to\beta} = \{f \in \mathrm{mon}[\mathcal{K}_\alpha \to \mathcal{K}_\beta] \mid \exists d \in \mathcal{D}_{\alpha\to\beta}.\forall (g, e) \in \mathcal{L}_\alpha.(f(g), d(e)) \in \mathcal{L}_\beta\}$,
- $\mathcal{L}_{\alpha\to\beta} = \{(f, d) \in \mathcal{K}_{\alpha\to\beta} \times \mathcal{D}_{\alpha\to\beta} \mid \forall (g, e) \in \mathcal{L}_\alpha.(f(g), d(e)) \in \mathcal{L}_\beta\}$.

Going higher-order with a logical relation



- $\mathcal{L}_0 = \{((d, P), d) \mid (d, P) \in \mathcal{K}_0\}$,
- $\mathcal{K}_{\alpha\to\beta} = \{f \in \mathrm{mon}[\mathcal{K}_\alpha \to \mathcal{K}_\beta] \mid \exists d \in \mathcal{D}_{\alpha\to\beta}.\forall (g, e) \in \mathcal{L}_\alpha.(f(g), d(e)) \in \mathcal{L}_\beta\}$,
- $\mathcal{L}_{\alpha\to\beta} = \{(f, d) \in \mathcal{K}_{\alpha\to\beta} \times \mathcal{D}_{\alpha\to\beta} \mid \forall (g, e) \in \mathcal{L}_\alpha.(f(g), d(e)) \in \mathcal{L}_\beta\}$.

A Galois connection

The logical relation $\mathcal{L}$ induces two functors:

- for every f in $\mathcal{K}_\alpha$ there is a unique $\overline{f}$ in $\mathcal{D}_\alpha$ so that $f\ \mathcal{L}_\alpha\ \overline{f}$,
- for d in $\mathcal{D}_\alpha$, let $d^\uparrow = \bigvee\{f \mid d\ \mathcal{L}_\alpha\ f\}$.

Functoriality

- $\overline{f(g)} = \overline{f}(\overline{g})$ and $f \leq g$ implies $\overline{f} \leq \overline{g}$,
- $d(e)^\uparrow = d^\uparrow(e^\uparrow)$, $d\ \mathcal{L}\ d^\uparrow$ and $d \leq e$ implies $d^\uparrow \leq e^\uparrow$.

Gallois connection

- $\overline{f} \leq d$ iff $f \leq d^\uparrow$,
- in particular $\overline{d^\uparrow} = d$

A Galois connection

The logical relation $\mathcal{L}$ induces two functors:

- for every f in $\mathcal{K}_\alpha$ there is a unique $\overline{f}$ in $\mathcal{D}_\alpha$ so that $f\ \mathcal{L}_\alpha\ \overline{f}$,
- for d in $\mathcal{D}_\alpha$, let $d^\uparrow = \bigvee\{f \mid d\ \mathcal{L}_\alpha\ f\}$.

Functoriality

- $\overline{f(g)} = \overline{f}(\overline{g})$ and $f \leq g$ implies $\overline{f} \leq \overline{g}$,
- $d(e)^\uparrow = d^\uparrow(e^\uparrow)$, $d\ \mathcal{L}\ d^\uparrow$ and $d \leq e$ implies $d^\uparrow \leq e^\uparrow$.

Gallois connection

- $\overline{f} \leq d$ iff $f \leq d^\uparrow$,
- in particular $\overline{d^\uparrow} = d$

A Galois connection

The logical relation $\mathcal{L}$ induces two functors:

- for every f in $\mathcal{K}_\alpha$ there is a unique $\overline{f}$ in $\mathcal{D}_\alpha$ so that $f\ \mathcal{L}_\alpha\ \overline{f}$,
- for d in $\mathcal{D}_\alpha$, let $d^\uparrow = \bigvee\{f \mid d\ \mathcal{L}_\alpha\ f\}$.

Functoriality

- $\overline{f(g)} = \overline{f}(\overline{g})$ and $f \leq g$ implies $\overline{f} \leq \overline{g}$,
- $d(e)^\uparrow = d^\uparrow(e^\uparrow)$, $d\ \mathcal{L}\ d^\uparrow$ and $d \leq e$ implies $d^\uparrow \leq e^\uparrow$.

Gallois connection

- $\overline{f} \leq d$ iff $f \leq d^\uparrow$,
- in particular $\overline{d^\uparrow} = d$

Fixpoint

Fixpoint definition

For $f \in \mathcal{K}_{\alpha\to\alpha}$, we let $Fix_\alpha(f) = \bigwedge_{n\in\mathbb{N}} f^n(fix_\alpha(\overline{f})^\uparrow)$ with $fix_\alpha(d) = \bigvee_{n\in\mathbb{N}} d^n(\bot_\alpha)$

Lemma ($\mathcal{K}$ is a model of ΛY)

Fix_α is a fixpoint and is in $\mathcal{K}_{(\alpha\to\alpha)\to\alpha}$.

Fixpoint

Fixpoint definition

For $f \in \mathcal{K}_{\alpha\to\alpha}$, we let $Fix_\alpha(f) = \bigwedge_{n\in\mathbb{N}} f^n(fix_\alpha(\overline{f})^\uparrow)$ with $fix_\alpha(d) = \bigvee_{n\in\mathbb{N}} d^n(\bot_\alpha)$

Lemma ($\mathcal{K}$ is a model of ΛY)

Fix_α is a fixpoint and is in $\mathcal{K}_{(\alpha\to\alpha)\to\alpha}$.

$\mathcal{K}$ does the job

Theorem

For a given insightful automaton $\mathcal{A}$, the construction of $\mathcal{K}$ for $\mathcal{A}$, is so that if $[\![M]\!]^{\mathcal{K}} = (d, Q)$, $BT(M)$ is accepted by $\mathcal{A}$ in state q iff $q \in Q$.

Weak MSO and weak parity automata

A weak parity automaton is $\mathcal{A} = (Q, \delta, \mathrm{rk})$ so that:
$\mathrm{rk} : Q \mapsto \mathbb{N}$, $(q_1, q_2) \in \delta(a, q)$ implies $\mathrm{rk}(q_1) \leq \mathrm{rk}(q)$ and $\mathrm{rk}(q_2) \leq \mathrm{rk}(q)$.

$\mathcal{A}$ accepts a tree t from state q if there is a run satisfying the usual parity condition.

Idea of the construction

- We construct the model by induction on the parity,
- At each step the model $\mathcal{M}_k$ at base type is $\mathcal{P}(Q_k)$ where $Q_k = \{q \in Q \mid \mathrm{rk}(q) \leq k\}$,
- and $q \in [\![M]\!]^{\mathcal{M}_k}$ iff $\mathcal{A}$ accepts $BT(M)$ from q.

Basic block: domain extension

Given finite sets X_1 and X_2, so that $X_1 \subseteq X_2$, a model $\mathcal{M}_1$ so that $\mathcal{M}_1^0 = \mathcal{P}(X_1)$, we define $\mathcal{M}_2 = \mathrm{ext}(\mathcal{M}_1, X_2)$ as:

- $\mathcal{L}^0 = \{(Q, P) \mid P = Q \cap X_1\}$
- $\mathcal{M}_2^0 = \mathcal{P}(X_2)$
- $\mathcal{M}_2^{\alpha\to\beta} = \{f \in \mathrm{mon}[\mathcal{M}_2^\alpha \mapsto \mathcal{M}_2^\beta] \mid \exists d \in \mathcal{M}_1^\alpha . \forall (g, e) \in \mathcal{L}^\alpha . (f(g), d(e)) \in \mathcal{L}^\beta\}$
- $\mathcal{L}^{\alpha\to\beta} = \{(f, d) \mid \forall (g, e) \in \mathcal{L}^\alpha . (f(g), d(e)) \in \mathcal{L}^\beta\}$

Two Galois connections

$\mathcal{L}$ induces three functors:

- for every f in $\mathcal{M}_2^\alpha$ there is a unique $\overline{f}$ in $\mathcal{M}_1^\alpha$ so that $f\ \mathcal{L}^\alpha\ \overline{f}$,
- for d in $\mathcal{M}_1^\alpha$, let $d^\uparrow = \bigvee\{f \mid d\ \mathcal{L}_\alpha\ f\}$,
- for d in $\mathcal{M}_1^\alpha$, let $d^\downarrow = \bigwedge\{f \mid d\ \mathcal{L}_\alpha\ f\}$.

Functoriality

- $\overline{f(g)} = \overline{f}(\overline{g})$, and $f \leq g$ implies $\overline{f} \leq \overline{g}$,
- $d(e)^\uparrow = d^\uparrow(e^\uparrow)$, $d\ \mathcal{L}\ d^\uparrow$ and $d \leq e$ implies $d^\uparrow \leq e^\uparrow$,
- $d(e)^\downarrow = d^\downarrow(e^\downarrow)$, $d\ \mathcal{L}\ d^\downarrow$ and $d \leq e$ implies $d^\downarrow \leq e^\downarrow$.

Galois connections

- $\overline{f} \leq d$ iff $f \leq d^\uparrow$,
- $d \leq \overline{f}$ iff $d^\downarrow \leq f$,
- in particular $d = \overline{d^\uparrow} = \overline{d^\downarrow}$.

Two Galois connections

$\mathcal{L}$ induces three functors:

- for every f in $\mathcal{M}_2^\alpha$ there is a unique $\overline{f}$ in $\mathcal{M}_1^\alpha$ so that $f\ \mathcal{L}^\alpha\ \overline{f}$,
- for d in $\mathcal{M}_1^\alpha$, let $d^\uparrow = \bigvee\{f \mid d\ \mathcal{L}_\alpha\ f\}$,
- for d in $\mathcal{M}_1^\alpha$, let $d^\downarrow = \bigwedge\{f \mid d\ \mathcal{L}_\alpha\ f\}$.

Functoriality

- $\overline{f(g)} = \overline{f}(\overline{g})$, and $f \leq g$ implies $\overline{f} \leq \overline{g}$,
- $d(e)^\uparrow = d^\uparrow(e^\uparrow)$, $d\ \mathcal{L}\ d^\uparrow$ and $d \leq e$ implies $d^\uparrow \leq e^\uparrow$,
- $d(e)^\downarrow = d^\downarrow(e^\downarrow)$, $d\ \mathcal{L}\ d^\downarrow$ and $d \leq e$ implies $d^\downarrow \leq e^\downarrow$.

Galois connections

- $\overline{f} \leq d$ iff $f \leq d^\uparrow$,
- $d \leq \overline{f}$ iff $d^\downarrow \leq f$,
- in particular $d = \overline{d^\uparrow} = \overline{d^\downarrow}$.

Two Galois connections

$\mathcal{L}$ induces three functors:

- for every f in $\mathcal{M}_2^\alpha$ there is a unique $\overline{f}$ in $\mathcal{M}_1^\alpha$ so that $f\ \mathcal{L}^\alpha\ \overline{f}$,
- for d in $\mathcal{M}_1^\alpha$, let $d^\uparrow = \bigvee\{f \mid d\ \mathcal{L}_\alpha\ f\}$,
- for d in $\mathcal{M}_1^\alpha$, let $d^\downarrow = \bigwedge\{f \mid d\ \mathcal{L}_\alpha\ f\}$.

Functoriality

- $\overline{f(g)} = \overline{f}(\overline{g})$, and $f \leq g$ implies $\overline{f} \leq \overline{g}$,
- $d(e)^\uparrow = d^\uparrow(e^\uparrow)$, $d\ \mathcal{L}\ d^\uparrow$ and $d \leq e$ implies $d^\uparrow \leq e^\uparrow$,
- $d(e)^\downarrow = d^\downarrow(e^\downarrow)$, $d\ \mathcal{L}\ d^\downarrow$ and $d \leq e$ implies $d^\downarrow \leq e^\downarrow$.

Galois connections

- $\overline{f} \leq d$ iff $f \leq d^\uparrow$,
- $d \leq \overline{f}$ iff $d^\downarrow \leq f$,
- in particular $d = \overline{d^\uparrow} = \overline{d^\downarrow}$.

Two possible fixpoints

Fixpoints definition

If fix_α is the interpretation of the fixpoint at type α in $\mathcal{M}_1$, given $f \in \mathcal{M}_2^{\alpha\to\alpha}$ we let:

- $Fix^0_\alpha(f) = \bigwedge_{n\in\mathbb{N}} f^n(fix_\alpha(\overline{f})^\uparrow)$
- $Fix^1_\alpha(f) = \bigvee_{n\in\mathbb{N}} f^n(fix_\alpha(\overline{f})^\downarrow)$

Lemma ($\mathcal{M}_2$ is a model of ΛY)

Fix^0_α and Fix^1_α are fixpoints and are in $\mathcal{M}_2^{(\alpha\to\alpha)\to\alpha}$.

Two possible fixpoints

Fixpoints definition

If fix_α is the interpretation of the fixpoint at type α in $\mathcal{M}_1$, given $f \in \mathcal{M}_2^{\alpha\to\alpha}$ we let:

- $Fix^0_\alpha(f) = \bigwedge_{n\in\mathbb{N}} f^n(fix_\alpha(\overline{f})^\uparrow)$
- $Fix^1_\alpha(f) = \bigvee_{n\in\mathbb{N}} f^n(fix_\alpha(\overline{f})^\downarrow)$

Lemma ($\mathcal{M}_2$ is a model of ΛY)

Fix^0_α and Fix^1_α are fixpoints and are in $\mathcal{M}_2^{(\alpha\to\alpha)\to\alpha}$.

The model for wMSO

- We let $\mathcal{M}(0)$ be the monotone model generated by $\mathcal{P}(Q_0)$ where $Q_0 = \{q \in Q \mid \mathrm{rk}(q) = 0\}$,
- $\mathcal{M}(k+1) = \mathrm{ext}(\mathcal{M}_k, Q_{k+1})$ where $Q_{k+1} = \{q \in Q \mid \mathrm{rk}(q) \leq k+1\}$, and $[\![Y]\!]^{\mathcal{M}(k+1)} = Fix^p$ where $p \equiv k+1[2]$.

Theorem

Given $q \in Q_k$, the following are equivalent:

- $q \in [\![M]\!]^{\mathcal{M}_k}$,
- *$\mathcal{A}$ accepts $BT(M)$ from the state q.*