Fixed Point Models for Theories of Properties and Classes

Greg Restall

THE UNIVERSITY OF MELBOURNE

CLMPS 2015 · HELSINKI · 5 AUGUST 2015

Our Target

Model Construction

Classifying Class Theories

Order and Continuity

Order Models

OUR TARGET

Class Abstraction

$$a \in \{x : \phi(x)\} \text{ iff } \phi(a)$$

$$a \mathrel{\varepsilon} \lambda x.\phi(x) \text{ iff } \phi(a)$$

Russell's Paradox

$$\{x : x \notin x\} \in \{x : x \notin x\} \text{ iff } \{x : x \notin x\} \notin \{x : x \notin x\}$$

$$\{x : x \notin x\} \in \{x : x \notin x\} \text{ iff } \{x : x \notin x\} \notin \{x : x \notin x\}$$

In general,

$$\{x : F(x \in x)\} \in \{x : F(x \in x)\} \text{ iff}$$
$$F\big(\{x : F(x \in x)\} \in \{x : F(x \in x)\}\big)$$

The Heterological Paradox

$$\lambda x.(x \not\varepsilon x)\ \varepsilon\ \lambda x.(x \not\varepsilon x)\ \text{ iff }\ \lambda x.(x \not\varepsilon x) \not\varepsilon \lambda x.(x \not\varepsilon x)$$

$$\lambda x.(x \not\varepsilon x) \;\varepsilon\; \lambda x.(x \not\varepsilon x) \text{ iff } \lambda x.(x \not\varepsilon x) \not\varepsilon \lambda x.(x \not\varepsilon x)$$

In general,

$$\lambda x.F(x \;\varepsilon\; x) \;\varepsilon\; \lambda x.F(x \;\varepsilon\; x) \text{ iff }$$
$$F\big(\lambda x.F(x \;\varepsilon\; x) \;\varepsilon\; \lambda x.F(x \;\varepsilon\; x)\big)$$

Extensionality

If a and b have the same members, then $a = b$.

If a and b have the same members, then $a = b$.

$$\frac{\Gamma, x \in a \vdash x \in b, \Delta \quad \Gamma, x \in b \vdash x \in a, \Delta}{\Gamma \vdash a = b, \Delta}$$

Extensionality

If a and b have the same members, then $a = b$.

$$\frac{\Gamma, x \in a \vdash x \in b, \Delta \quad \Gamma, x \in b \vdash x \in a, \Delta}{\Gamma \vdash a = b, \Delta}$$

(Extensionality will not play a significant role in what follows.)

MODEL CONSTRUCTION

Defining validity.

Defining validity.

Providing *counterexamples*,
including *proving non-triviality*.

Defining validity.

Providing *counterexamples*,
including *proving non-triviality*.

Relating theories.

Defining validity.

Providing *counterexamples*,
including *proving non-triviality*.

Relating theories.

Giving a sense of what
the theory can be *about*.

Defining validity.

Providing *counterexamples*,
including *proving non-triviality*.

Relating theories.

Giving a sense of what
the theory can be *about*.

Motivating the theory.

Defining validity.

Providing *counterexamples*,
including *proving non-triviality*.

Relating theories.

Giving a sense of what
the theory can be *about*.

Motivating the theory.

Defining validity.

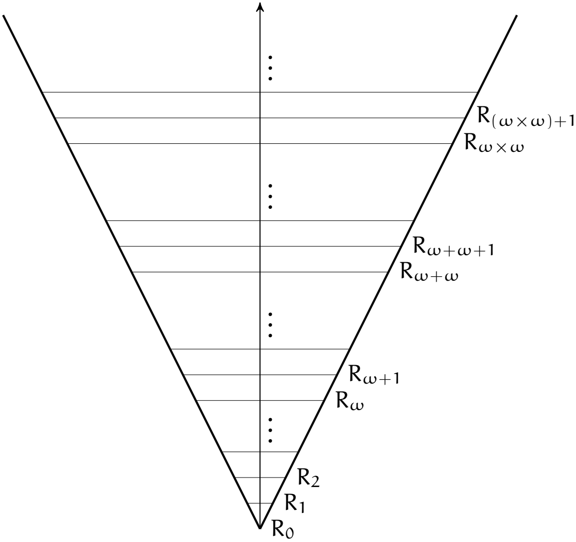
Providing *counterexamples*,
including *proving non-triviality*.

Relating theories.

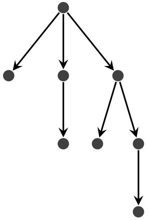
Giving a sense of what
the theory can be *about*.

Motivating the theory.

ZFC and its Cousins: The Iterative Conception of Set

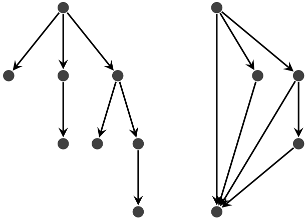


ZFC and its Cousins: Anti-Foundation



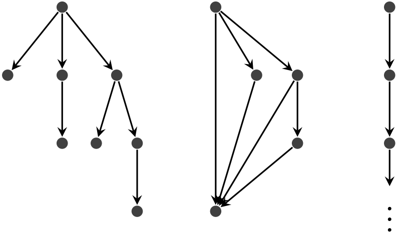
$$\{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\}$$

ZFC and its Cousins: Anti-Foundation



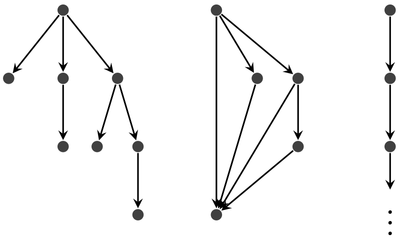
$$\{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\}$$

ZFC and its Cousins: Anti-Foundation



$$\{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\}$$

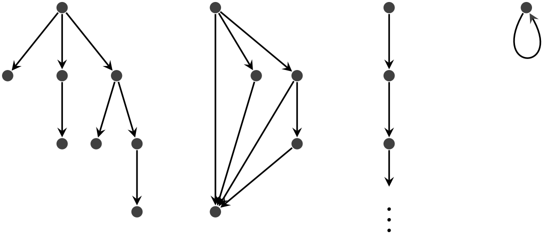
ZFC and its Cousins: Anti-Foundation



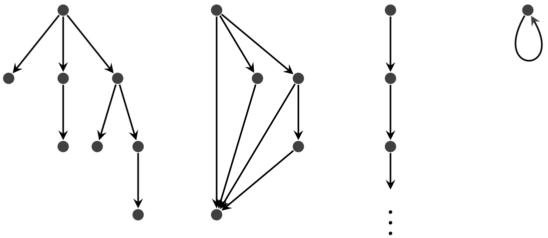
$\{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\}$

an a where $a = \{a\}$

ZFC and its Cousins: Anti-Foundation

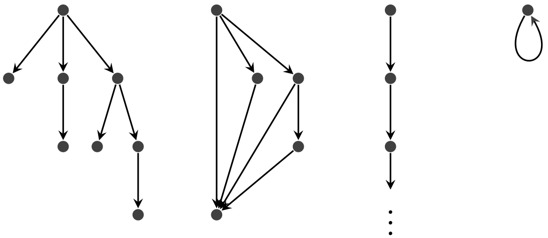


$\{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\}$ an a where $a = \{a\}$



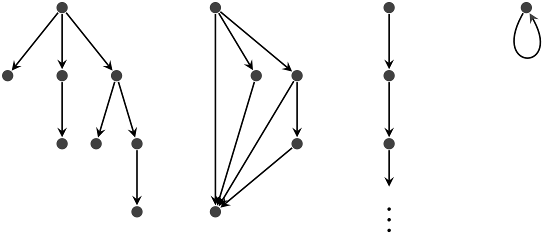
$\{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\}$ an a where $a = \{a\}$

These models are good for (1) *relating* ZFC to AFA,



$\{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\}$ an a where $a = \{a\}$

These models are good for (1) *relating* ZFC to AFA, (2) motivating a choice of the anti-foundation axiom,



$\{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\}$ an a where $a = \{a\}$

These models are good for (1) *relating* ZFC to AFA, (2) motivating a choice of the anti-foundation axiom, and (3) explaining what the theory could be *about*.

If x is a *variable* and M is a *term*, $\lambda x.M$ is a *term*.

If x is a *variable* and M is a *term*, $\lambda x.M$ is a *term*.

For any terms, M and N, MN is M *applied to* N.

If x is a *variable* and M is a *term*, $\lambda x.M$ is a *term*.

For any terms, M and N, MN is M *applied to* N.

$$(\lambda x.M)N = M[x := N].$$

Models of the Untyped λ Calculus

Models of the Untyped λ Calculus

$$D \qquad D \to D$$

$$D \cong D \to D$$

You bump up against *Cantor's Theorem*.

Models of the Untyped λ Calculus

$$D \cong [D \to D]$$

$[D \to E]$: the *order preserving functions* from $(D, \sqsubseteq)$ to $(E, \sqsubseteq)$.

$[D \to E]$: the *order preserving functions* from $(D, \sqsubseteq)$ to $(E, \sqsubseteq)$.

It's ordered too: $f \sqsubseteq g$ iff $(\forall x)(f(x) \sqsubseteq g(x))$.

$[D \to E]$: the *order preserving functions* from $(D, \sqsubseteq)$ to $(E, \sqsubseteq)$.

It's ordered too: $f \sqsubseteq g$ iff $(\forall x)(f(x) \sqsubseteq g(x))$.

Embed D_i into $[D_i \to D_i] = D_{i+1}$
(Use the constant functions.)

The Scott Construction

$[D \to E]$: the *order preserving functions* from $(D, \sqsubseteq)$ to $(E, \sqsubseteq)$.

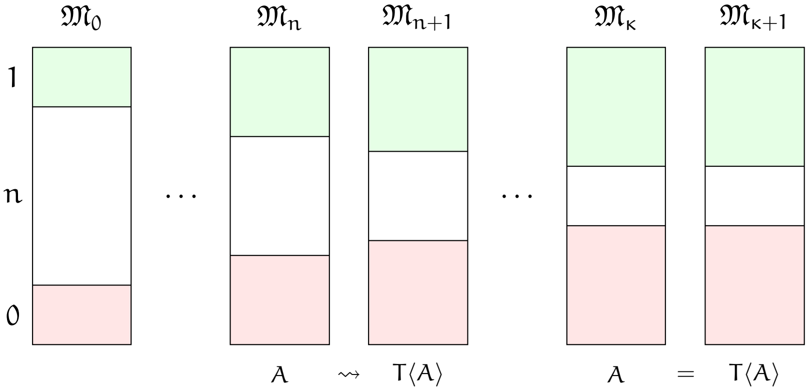
It's ordered too: $f \sqsubseteq g$ iff $(\forall x)(f(x) \sqsubseteq g(x))$.

Embed D_i into $[D_i \to D_i] = D_{i+1}$
(Use the constant functions.)

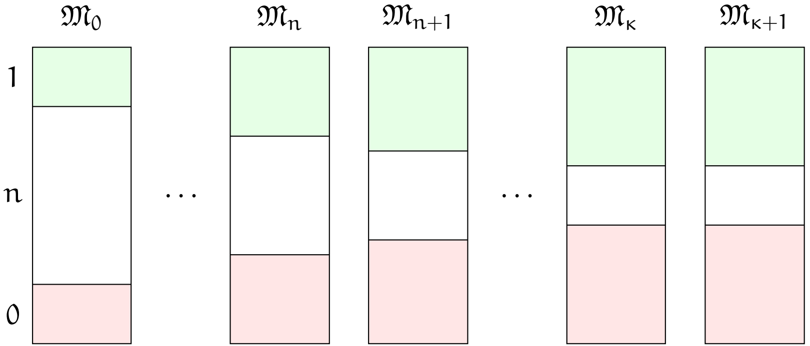
Let D_∞ be the limit: $D_\infty \cong [D_\infty \to D_\infty]$.

This is a model of the untyped λ calculus.

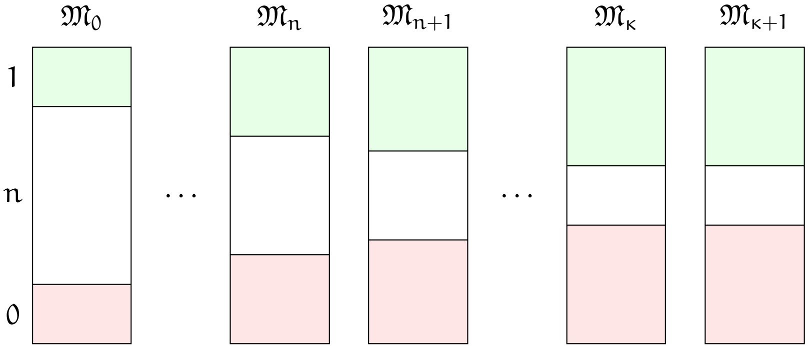
Truth Theories: Kripke, Woodruff, Gilmore, Brady



Class Theories

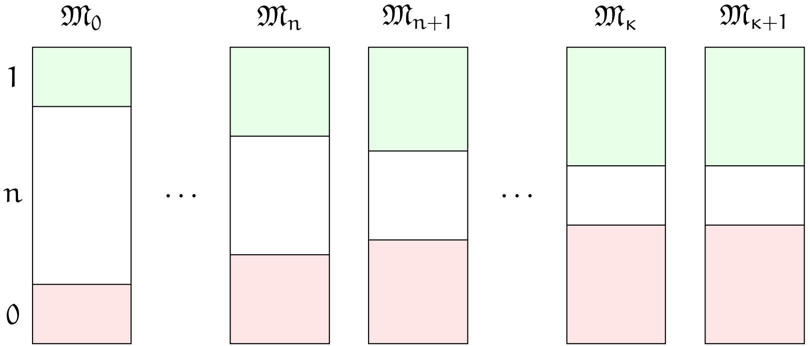


$\phi(a) \rightsquigarrow a \in \{x : \phi(x)\}$ $\qquad$ $\phi(a) = a \in \{x : \phi(x)\}$



$\phi(a) \rightsquigarrow a \in \{x : \phi(x)\}$ $\qquad$ $\phi(a) = a \in \{x : \phi(x)\}$

This is *not* like the other model constructions: the domain is constant—the terms $\{x : \phi(x)\}$.



$\phi(a) \rightsquigarrow a \in \{x : \phi(x)\}$ $\qquad$ $\phi(a) = a \in \{x : \phi(x)\}$

This is *not* like the other model constructions: the domain is constant—the terms $\{x : \phi(x)\}$.

This shows what the theory is *about* in only a very weak sense.

CLASSIFYING CLASS THEORIES

Gaps or *Gluts*?

Underlying Logic: Negation

Gaps or *Gluts*?

Paraconsistent or *Paracomplete*?

Do we have a conditional in the language?

Underlying Logic: The Conditional

Do we have a conditional in the language?

And if so, what is it like?

Underlying Logic: Not *that* important

These decisions are not *that* important.

Underlying Logic: Not *that* important

These decisions are not *that* important.

The logic must allow for *fixed points*.

These decisions are not *that* important.

The logic must allow for *fixed points*.

For *any* sentence context $F(-)$, we need to allow for some p to be *equivalent to* $F(p)$.
If $c =_{df} \{x : F(x \in x)\}$, then $c \in c$ *iff* $F(c \in c)$

$$D$$

- D: the *ordinary* domain.

$$D$$

- D: the *ordinary* domain.

$$D \qquad \Omega$$

- D: the *ordinary* domain.
- Ω: truth values.

$$\color{red}{C} \qquad\qquad D \to \Omega$$

- D: the *ordinary* domain.
- Ω: truth values.
- C: the classes

Semantic Values

$$C \qquad (C \cup D) \to \Omega$$

- D: the *ordinary* domain.
- Ω: truth values.
- C: the classes

$$C \cong (C \cup D) \to \Omega$$

- D: the *ordinary* domain.
- Ω: truth values.
- C: the classes

$$C \cong [(C \cup D) \to \Omega]$$

- D: the *ordinary* domain.
- Ω: truth values.
- C: the classes

Extensionality

We won't focus on extensionality here.

We won't focus on extensionality here.

But we'll *identify* classes by their extensions as much as possible.

Sharpening our Target

$$C \cong [C \cup D \to \Omega]$$

$$C \stackrel{\sim}{=} [C \cup D \to \Omega]$$

$\phi(x)$ gives a function $[C \cup D \to \Omega]$.
So we can find a class C to *match*.

$a \in \{x : \phi(x)\}$ has the *same* value in Ω as $\phi(a)$.

ORDER AND CONTINUITY

Underlying Logic: Preservation

$$\sqsubseteq$$

Underlying Logic: Preservation

$$\sqsubseteq$$

Ω is ordered by $\sqsubseteq$.

Underlying Logic: Preservation

$$\sqsubseteq$$

Ω is ordered by $\sqsubseteq$.

All connectives & quantifiers are $\sqsubseteq$-order preserving.

Underlying Logic: Preservation

$$\sqsubseteq$$

Ω is ordered by $\sqsubseteq$.

All connectives & quantifiers
are $\sqsubseteq$-order preserving.

(If $x \sqsubseteq x'$ and $y \sqsubseteq y'$ then $x \sharp y \sqsubseteq x' \sharp y'$, etc.)

Preservation on candidates for Ω

$$\begin{tikzcd} 0 & & 1 \\ & * \arrow[ul] \arrow[ur] & \end{tikzcd}$$

Preservation on candidates for Ω

$$\begin{tikzcd}
0 & & 1 \\
& * \arrow[ul] \arrow[ur] &
\end{tikzcd}$$

K_3 or LP

Preservation on candidates for Ω

$$\begin{tikzcd} 0 & & 1 \\ & * \arrow[ul] \arrow[ur] & \end{tikzcd}$$

K_3 or LP, but not Ł$_3$

In Ł$_3$, $* \rightarrow *$ is 1; but $1 \rightarrow 0$ is 0

Preservation on candidates for Ω

$$\begin{tikzcd} 0 & & 1 \\ & * \arrow[ul] \arrow[ur] & \end{tikzcd}$$

K_3 or LP, but not Ł$_3$ or RM_3

In Ł$_3$, $* \to *$ is 1; but $1 \to 0$ is 0

In RM_3, $1 \to *$ is 0; but $1 \to 1$ is 1

Preservation on candidates for Ω

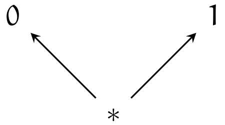
$$\begin{tikzcd}
0 & & 1 \\
& * \arrow[ul] \arrow[ur] &
\end{tikzcd}
\qquad\qquad
\begin{tikzcd}
& \star & \\
0 \arrow[ur] & & 1 \arrow[ul] \\
& * \arrow[ul] \arrow[ur] &
\end{tikzcd}$$

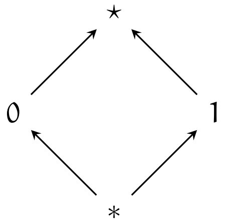
K_3 or LP, but not Ł$_3$ or RM_3

In Ł$_3$, $* \to *$ is 1; but $1 \to 0$ is 0

In RM_3, $1 \to *$ is 0; but $1 \to 1$ is 1

Preservation on candidates for Ω





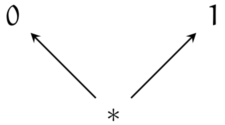
K_3 or LP, but not Ł$_3$ or RM_3

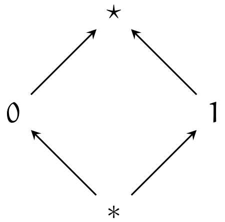
In Ł$_3$, $* \to *$ is 1; but $1 \to 0$ is 0

In RM_3, $1 \to *$ is 0; but $1 \to 1$ is 1

FDE, but no robust conditionals.

Preservation on candidates for Ω





K_3 or LP, but not Ł$_3$ or RM_3

In Ł$_3$, $* \rightarrow *$ is 1; but $1 \rightarrow 0$ is 0

In RM_3, $1 \rightarrow *$ is 0; but $1 \rightarrow 1$ is 1

FDE, but no robust conditionals.

Similar behaviour here.

Many other choices for Ω are possible.

Many other choices for Ω are possible.

Even $\{0, 1\}$ can be ordered: $0 \sqsubseteq 1$. Then $\wedge, \vee, 0, 1$ are order preserving, but $\neg$ and $\supset$ are *not* order preserving.

3: our choice of Ω

$$\begin{tikzcd} 0 & & 1 \\ & * \arrow[ul] \arrow[ur] & \end{tikzcd}$$

3: our choice of Ω

$$\begin{tikzcd} 0 & & 1 \\ & * \arrow[ul] \arrow[ur] & \end{tikzcd}$$

(I *really* don't care if you think of $*$ as *true*, or as *untrue*.)

ORDER MODELS

Defining Order Models

Given an order algebra Ω, and a domain D of urelements

$\langle C, \sqsubseteq, \Uparrow, \Downarrow\rangle$ is a $\langle D, \Omega\rangle$ *order model* iff

Given an order algebra Ω, and a domain D of urelements

$$\langle C, \sqsubseteq, \Uparrow, \Downarrow \rangle \text{ is a } \langle D, \Omega \rangle \textit{ order model} \text{ iff}$$

- $\sqsubseteq$ is a partial order on C.

Given an order algebra Ω, and a domain D of urelements

$$\langle C, \sqsubseteq, \Uparrow, \Downarrow \rangle \text{ is a } \langle D, \Omega \rangle \textit{ order model} \text{ iff}$$

- $\sqsubseteq$ is a partial order on C.
- $\Uparrow : C \to [C \cup D \to \Omega]$ is order preserving and invertible.

Defining Order Models

Given an order algebra Ω, and a domain D of urelements

$$\langle C, \sqsubseteq, \Uparrow, \Downarrow \rangle \text{ is a } \langle D, \Omega \rangle \textit{ order model} \text{ iff}$$

- $\sqsubseteq$ is a partial order on C.
- $\Uparrow : C \to [C \cup D \to \Omega]$ is order preserving and invertible.
- $\Downarrow : [C \cup D \to \Omega] \to C$, where $\Downarrow = \Uparrow^{-1}$, is also order preserving.

Defining Order Models

Given an order algebra Ω, and a domain D of urelements

$$\langle C, \sqsubseteq, \Uparrow, \Downarrow \rangle \text{ is a } \langle D, \Omega \rangle \textit{ order model} \text{ iff}$$

- $\sqsubseteq$ is a partial order on C.
- $\Uparrow : C \to [C \cup D \to \Omega]$ is order preserving and invertible.
- $\Downarrow : [C \cup D \to \Omega] \to C$, where $\Downarrow = \Uparrow^{-1}$, is also order preserving.

– Write '$\Uparrow(c)$' as '$c_\Uparrow$' and '$\Downarrow(f)$' as '$f_\Downarrow$.' So $c_{\Uparrow\Downarrow} = c$ and $f_{\Downarrow\Uparrow} = f$.

Defining Order Models

Given an order algebra Ω, and a domain D of urelements

$$\langle C, \sqsubseteq, \Uparrow, \Downarrow \rangle \text{ is a } \langle D, \Omega \rangle \textit{ order model} \text{ iff}$$

- $\sqsubseteq$ is a partial order on C.
- $\Uparrow : C \to [C \cup D \to \Omega]$ is order preserving and invertible.
- $\Downarrow : [C \cup D \to \Omega] \to C$, where $\Downarrow = \Uparrow^{-1}$, is also order preserving.

– Write '$\Uparrow(c)$' as '$c_\Uparrow$' and '$\Downarrow(f)$' as '$f_\Downarrow$.' So $c_{\Uparrow\Downarrow} = c$ and $f_{\Downarrow\Uparrow} = f$.

– If $b \in C \cup D$ and $c \in C$, then $c_\Uparrow(b)$ tells you whether b is in c.

Membership is order preserving

If $x \sqsubseteq x'$ and $y \sqsubseteq y'$ then $x_{\Uparrow}(y) \sqsubseteq x'_{\Uparrow}(y')$.

Membership is order preserving

If $x \sqsubseteq x'$ and $y \sqsubseteq y'$ then $x_{\Uparrow}(y) \sqsubseteq x'_{\Uparrow}(y')$.

$x_{\Uparrow}(y) \sqsubseteq x_{\Uparrow}(y')$ — $y \sqsubseteq y'$ and $x_{\Uparrow}$ is order preserving.

$$\text{If } x \sqsubseteq x' \text{ and } y \sqsubseteq y' \text{ then } x_{\Uparrow}(y) \sqsubseteq x'_{\Uparrow}(y').$$

$$\begin{array}{rcl} x_{\Uparrow}(y) \sqsubseteq x_{\Uparrow}(y') & — & y \sqsubseteq y' \text{ and } x_{\Uparrow} \text{ is order preserving.} \\ x_{\Uparrow} \sqsubseteq x'_{\Uparrow} & — & x \sqsubseteq x' \text{ and } \Uparrow \text{ is order preserving.} \end{array}$$

Membership is order preserving

If $x \sqsubseteq x'$ and $y \sqsubseteq y'$ then $x_{\Uparrow}(y) \sqsubseteq x'_{\Uparrow}(y')$.

$x_{\Uparrow}(y) \sqsubseteq x_{\Uparrow}(y')$	—	$y \sqsubseteq y'$ and $x_{\Uparrow}$ is order preserving.
$x_{\Uparrow} \sqsubseteq x'_{\Uparrow}$	—	$x \sqsubseteq x'$ and $\Uparrow$ is order preserving.
$x_{\Uparrow}(y') \sqsubseteq x'_{\Uparrow}(y')$	—	by the definition of $\sqsubseteq$ for functions.

Given a $\langle D, 3\rangle$ order model $\mathfrak{M} = \langle C, \sqsubseteq, \Uparrow, \Downarrow\rangle$,

Interpreting a Language

Given a $\langle D, 3\rangle$ order model $\mathfrak{M} = \langle C, \sqsubseteq, \Uparrow, \Downarrow\rangle$,

- An assignment α, takes variables to values in $C \cup D$.

Given a $\langle D, 3\rangle$ order model $\mathfrak{M} = \langle C, \sqsubseteq, \Uparrow, \Downarrow\rangle$,

- An assignment α, takes variables to values in $C \cup D$.

- $[\![x]\!]_{\mathfrak{M},\alpha} = \alpha(x)$ is the interpretation of the variable x.

Interpreting a Language

Given a $\langle D, 3\rangle$ order model $\mathfrak{M} = \langle C, \sqsubseteq, \Uparrow, \Downarrow\rangle$,

- An assignment α, takes variables to values in $C \cup D$.
- $[\![x]\!]_{\mathfrak{M},\alpha} = \alpha(x)$ is the interpretation of the variable x.
- (We abbreviate this $[\![x]\!]$ when $\mathfrak{M}$ and α is clear.)

Given a $\langle D, 3\rangle$ order model $\mathfrak{M} = \langle C, \sqsubseteq, \Uparrow, \Downarrow\rangle$,

– An assignment α, takes variables to values in $C \cup D$.

- $[\![x]\!]_{\mathfrak{M},\alpha} = \alpha(x)$ is the interpretation of the variable x.

– (We abbreviate this $[\![x]\!]$ when $\mathfrak{M}$ and α is clear.)

- $[\![s \in t]\!]_{\mathfrak{M},\alpha}$ is $[\![t]\!]_{\Uparrow}([\![s]\!])$ when $[\![t]\!] \in C$,
  and is 0 when $[\![t]\!] \in D$.

Given a $\langle D, 3\rangle$ order model $\mathfrak{M} = \langle C, \sqsubseteq, \Uparrow, \Downarrow\rangle$,

- An assignment α, takes variables to values in $C \cup D$.
- $[\![x]\!]_{\mathfrak{M},\alpha} = \alpha(x)$ is the interpretation of the variable x.
- (We abbreviate this $[\![x]\!]$ when $\mathfrak{M}$ and α is clear.)
- $[\![s \in t]\!]_{\mathfrak{M},\alpha}$ is $[\![t]\!]_{\Uparrow}([\![s]\!])$ when $[\![t]\!] \in C$,
  and is 0 when $[\![t]\!] \in D$.
- Connectives and quantifiers are interpreted as usual.

Given a $\langle D, 3\rangle$ order model $\mathfrak{M} = \langle C, \sqsubseteq, \Uparrow, \Downarrow\rangle$,

- An assignment α, takes variables to values in $C \cup D$.
- $[\![x]\!]_{\mathfrak{M},\alpha} = \alpha(x)$ is the interpretation of the variable x.
- (We abbreviate this $[\![x]\!]$ when $\mathfrak{M}$ and α is clear.)
- $[\![s \in t]\!]_{\mathfrak{M},\alpha}$ is $[\![t]\!]_{\Uparrow}([\![s]\!])$ when $[\![t]\!] \in C$, and is 0 when $[\![t]\!] \in D$.
- Connectives and quantifiers are interpreted as usual.
- (Connectives and quantifiers are order preserving functions on 3 or $[C \cup D \to 3]$.)

Extending the Language with Terms

$$\{x : \phi(x)\}$$

$$\{x : \phi(x)\}$$

Since $[\![\phi(x)]\!]_{\mathfrak{M},\alpha[x:=v]}$ is order preserving in v we can use that function, in $[C \cup D \rightarrow 3]$, to select the extension of $\{x : \phi(x)\}$.

$$\{x : \phi(x)\}$$

Since $[\![\phi(x)]\!]_{\mathfrak{M},\alpha[x:=v]}$ is order preserving in v
we can use that function, in $[C \cup D \to 3]$,
to select the extension of $\{x : \phi(x)\}$.

$$[\![\{x : \phi(x)\}]\!]_{\mathfrak{M},\alpha} = (\lambda v.[\![\phi(x)]\!]_{\mathfrak{M},\alpha[x:=v]})_{\Downarrow}$$

Strong Comprehension

$$[\![t \in \{x : \phi(x)\}]\!]_{\mathfrak{M},\alpha}$$

Strong Comprehension

$$[\![t \in \{x : \phi(x)\}]\!]_{\mathfrak{M},\alpha} \quad = \quad [\![\{x : \phi(x)\}]\!]_{\alpha_{\Uparrow}}([\![t]\!]_{\alpha})$$

(I've dropped reference to $\mathfrak{M}$ as it is constant throughout.)

Strong Comprehension

$$
\begin{aligned}
[\![t \in \{x : \phi(x)\}]\!]_{\mathfrak{M},\alpha} &= [\![\{x : \phi(x)\}]\!]_{\alpha_{\Uparrow}}([\![t]\!]_{\alpha}) \\
&= (\lambda v.[\![\phi(x)]\!]_{\alpha[x:=v]})_{\Downarrow\Uparrow}([\![t]\!]_{\alpha})
\end{aligned}
$$

(I've dropped reference to $\mathfrak{M}$ as it is constant throughout.)

Strong Comprehension

$$
\begin{aligned}
[\![t \in \{x : \phi(x)\}]\!]_{\mathfrak{M},\alpha} &= [\![\{x : \phi(x)\}]\!]_{\alpha_{\Uparrow}}([\![t]\!]_{\alpha}) \\
&= (\lambda v.[\![\phi(x)]\!]_{\alpha[x:=v]})_{\Downarrow\Uparrow}([\![t]\!]_{\alpha}) \\
&= (\lambda v.[\![\phi(x)]\!]_{\alpha[x:=v]})([\![t]\!]_{\alpha})
\end{aligned}
$$

(I've dropped reference to $\mathfrak{M}$ as it is constant throughout.)

Strong Comprehension

$$\begin{aligned}
[\![t \in \{x : \phi(x)\}]\!]_{\mathfrak{M},\alpha} &= [\![\{x : \phi(x)\}]\!]_{\alpha\Uparrow}([\![t]\!]_\alpha) \\
&= (\lambda v.[\![\phi(x)]\!]_{\alpha[x:=v]})_{\Downarrow\Uparrow}([\![t]\!]_\alpha) \\
&= (\lambda v.[\![\phi(x)]\!]_{\alpha[x:=v]})([\![t]\!]_\alpha) \\
&= [\![\phi(x)]\!]_{\alpha[x:=[\![t]\!]_\alpha]}
\end{aligned}$$

(I've dropped reference to $\mathfrak{M}$ as it is constant throughout.)

Strong Comprehension

$$
\begin{aligned}
[\![t \in \{x : \phi(x)\}]\!]_{\mathfrak{M},\alpha} &= [\![\{x : \phi(x)\}]\!]_{\alpha\Uparrow}([\![t]\!]_{\alpha}) \\
&= (\lambda v.[\![\phi(x)]\!]_{\alpha[x:=v]})_{\Downarrow\Uparrow}([\![t]\!]_{\alpha}) \\
&= (\lambda v.[\![\phi(x)]\!]_{\alpha[x:=v]})([\![t]\!]_{\alpha}) \\
&= [\![\phi(x)]\!]_{\alpha[x:=[\![t]\!]_{\alpha}]} \\
&= [\![\phi(t)]\!]_{\mathfrak{M},\alpha}
\end{aligned}
$$

(I've dropped reference to $\mathfrak{M}$ as it is constant throughout.)

Strong Comprehension

$$
\begin{aligned}
[\![t \in \{x : \phi(x)\}]\!]_{\mathfrak{M},\alpha} &= [\![\{x : \phi(x)\}]\!]_{\alpha\Uparrow}([\![t]\!]_\alpha) \\
&= (\lambda v.[\![\phi(x)]\!]_{\alpha[x:=v]})_{\Downarrow\Uparrow}([\![t]\!]_\alpha) \\
&= (\lambda v.[\![\phi(x)]\!]_{\alpha[x:=v]})([\![t]\!]_\alpha) \\
&= [\![\phi(x)]\!]_{\alpha[x:=[\![t]\!]_\alpha]} \\
&= [\![\phi(t)]\!]_{\mathfrak{M},\alpha}
\end{aligned}
$$

(I've dropped reference to $\mathfrak{M}$ as it is constant throughout.)

$0 \qquad\qquad 1$

$$0 \qquad * \qquad 1$$

Λ, V and $V\!\!\!\Lambda$

$$\Lambda \quad = \quad \{x : 0\}$$

$$\Lambda \quad = \quad \{x : 0\}$$

$x \in \Lambda$ is always false.

Λ, V and $\mathbb{X}$

$$\Lambda = \{x : 0\}$$

$x \in \Lambda$ is always false.

$$V = \{x : 1\}$$

$$\Lambda = \{x : 0\}$$

$x \in \Lambda$ is always false.

$$V = \{x : 1\}$$

$x \in V$ is always true.

$$\Lambda = \{x : 0\}$$

$x \in \Lambda$ is always false.

$$V = \{x : 1\}$$

$x \in V$ is always true.

$$\mathbb{X} = \{x : *\}$$

Λ, V and $\mathbb{X}$

$$\Lambda = \{x : 0\}$$

$x \in \Lambda$ is always false.

$$V = \{x : 1\}$$

$x \in V$ is always true.

$$\mathbb{X} = \{x : *\}$$

$x \in \mathbb{X}$ is always $*$.

Ordering the Classes

$$
\begin{tikzcd}
0 & & 1 \\
& * \arrow[ul] \arrow[ur] &
\end{tikzcd}
$$

Ordering the Classes

$$
\begin{tikzcd}
0 & & 1 & \Lambda & & V \\
& * \arrow[ul] \arrow[ur] & & & \Lambda\!\!\!\!V \arrow[ul] \arrow[ur] &
\end{tikzcd}
$$

Ordering the Classes

$$\begin{tikzcd} 0 & & 1 \\ & * \arrow[ul] \arrow[ur] & \end{tikzcd} \qquad \begin{tikzcd} \Lambda & & V \\ & \mathbb{X} \arrow[ul] \arrow[ur] & \end{tikzcd}$$

In fact, $[\![\mathbb{X}]\!] \sqsubseteq c$ for every class $c \in C$.

Ordering the Classes

$$\begin{tikzcd} 0 & & 1 \\ & * \arrow[ul] \arrow[ur] & \end{tikzcd} \qquad \begin{tikzcd} \Lambda & & V \\ & \mathbb{X} \arrow[ul] \arrow[ur] & \end{tikzcd}$$

In fact, $[\![\mathbb{X}]\!] \sqsubseteq c$ for every class $c \in C$.

From now, we'll use '$\emptyset$', 'V' and '$\mathbb{X}$' as both the *class terms* in the language, and as their denotations, names for objects in C.

In a model $\mathfrak{M}$, a class c is SHARP iff
for each object b in $C \cup D$
$c_{\Uparrow}(b)$ takes the value 0 or 1

In a model $\mathfrak{M}$, a class c is SHARP iff
for each object b in $C \cup D$
$c_{\Uparrow}(b)$ takes the value 0 or 1

Λ and V are sharp.

In a model $\mathfrak{M}$, a class c is SHARP iff
for each object b in $C \cup D$
$c_{\Uparrow}(b)$ takes the value 0 or 1

Λ and V are sharp.

$\mathbb{X}$ is *not* sharp.

Almost No Classes are *Sharp*

If $c_{\Uparrow}(b) = 1$ and $c_{\Uparrow}(b') = 0$, then $c_{\Uparrow}(\mathfrak{X}) = *$.

$$\text{If } c_{\Uparrow}(b) = 1 \text{ and } c_{\Uparrow}(b') = 0, \text{ then } c_{\Uparrow}(\mathbb{X}) = *.$$

$$\mathbb{X} \sqsubseteq b, \text{ so } c_{\Uparrow}(\mathbb{X}) \sqsubseteq c_{\Uparrow}(b) = 1.$$

$$\text{If } c_{\Uparrow}(b) = 1 \text{ and } c_{\Uparrow}(b') = 0, \text{ then } c_{\Uparrow}(\mathfrak{X}) = *.$$

$$\mathfrak{X} \sqsubseteq b, \text{ so } c_{\Uparrow}(\mathfrak{X}) \sqsubseteq c_{\Uparrow}(b) = 1.$$

$$\mathfrak{X} \sqsubseteq b', \text{ so } c_{\Uparrow}(\mathfrak{X}) \sqsubseteq c_{\Uparrow}(b') = 0.$$

Almost No Classes are *Sharp*

If $c_{\Uparrow}(b) = 1$ and $c_{\Uparrow}(b') = 0$, then $c_{\Uparrow}(\mathbb{X}) = *$.

$\mathbb{X} \sqsubseteq b$, so $c_{\Uparrow}(\mathbb{X}) \sqsubseteq c_{\Uparrow}(b) = 1$.

$\mathbb{X} \sqsubseteq b'$, so $c_{\Uparrow}(\mathbb{X}) \sqsubseteq c_{\Uparrow}(b') = 0$.

It follows that $c_{\Uparrow}(\mathbb{X}) = *$

There is no *classical recapture* through crisp classes

Once a class *includes* something
and *excludes* something,
it is *indecisive* about $\mathbb{X}$.

There is no *classical recapture* through crisp classes

Once a class *includes* something
and *excludes* something,
it is *indecisive* about $\mathbb{X}$.

It follows that there are no *crisp singletons*:
objects $\{a\}$ for which $[\![a \in \{x\}]\!] = 1$
and $[\![b \in \{x\}]\!] = 0$ *for all other* b.

Singletons and Anti-Signetons: $\{t\}$ and $\}t\{$

- $[\![\{t\}]\!]_\alpha$: (the class representative of) the function that
  - assigns 1 to x iff $[\![t]\!]_\alpha \sqsubseteq x$,
  - and 0 to x iff there is no z where $x \sqsubseteq z$ and $[\![t]\!]_\alpha \sqsubseteq z$,
  - and $*$ otherwise.
- $[\![\}t\{]\!]_\alpha$: (the class representative of) the function that
  - assigns 0 to x iff $[\![t]\!]_\alpha \sqsubseteq x$, and
  - and 1 to x if there is no z where $x \sqsubseteq z$ and $[\![t]\!]_\alpha \sqsubseteq z$,
  - and $*$ otherwise.

From Here

From Here

- Study *pure* order models (where D is empty),

From Here

- Study *pure* order models (where D is empty),

  … and *impure* order models.

From Here

- Study *pure* order models (where D is empty),

  … and *impure* order models.
- Find perspicuous ways to *construct* order models.

From Here

- Study *pure* order models (where D is empty),

  ... and *impure* order models.
- Find perspicuous ways to *construct* order models.
- Relate these constructions to other known model constructions.

From Here

- Study *pure* order models (where D is empty),

  … and *impure* order models.
- Find perspicuous ways to *construct* order models.
- Relate these constructions to other known model constructions.
- *Axiomatise* the logic of order models.

From Here

- Study *pure* order models (where D is empty),

  … and *impure* order models.
- Find perspicuous ways to *construct* order models.
- Relate these constructions to other known model constructions.
- *Axiomatise* the logic of order models.
- Examine different *motivations* of order models.

THANK YOU!