

Spaces of model for CFOL \longleftrightarrow Algebras of CFOL

TO SIMPLIFY $\left\{ \begin{array}{l} \rightarrow \text{only relation symbols (no function symbols)} \\ \rightarrow \text{one sort. (NO VECTOR SPACES } \mathbb{K}, \mathbb{V} \text{)} \\ \rightarrow \text{no equality.} \end{array} \right.$

EXAMPLE: preorders

- \leq
- 1) $\forall x (x \leq x)$
 - 2) $\forall x \forall y \forall z ((x \leq y) \wedge (y \leq z) \rightarrow (x \leq z))$

directed graphs

(X, R)
 $\overset{\text{reflexive}}{\underset{\text{transitive}}{R}}$

We start with a (relational, single-sorted, first-order) language L :

in a set L , equipped with a function

$$\text{ar: } L \rightarrow \mathbb{N} = \{0, 1, 2, 3, \dots\} \quad \text{'the arity function'}$$

E.g.: $L = \{ \leq \}$

$$\text{ar: } L \rightarrow \mathbb{N}$$

$$\leq \mapsto 2$$

Moreover, we assume we have a ^{infinite} set Var of variables x, y, z
 x_1, x_2, x_3, \dots

FORMULAS IN THE LANGUAGE L : defined by induction

① ATOMIC FORMULAS: $R(x_1, \dots, x_n)$, where $R \in L$, of arity n

$x_1, \dots, x_n \in \text{Var}$ (possibly coinciding)

$$\text{EX: } x \leq y \quad \text{or} \quad x \leq x$$

② \perp and \top are formulas.

$\perp \vee (x \leq x)$

If φ is a formula, then $\neg\varphi$ is a formula

If φ and ψ are formulas, then $\varphi \wedge \psi$ and $\varphi \vee \psi$ are formulas.

③ If φ is a formula and $x \in \text{Var}$, then $(\exists x)\varphi$ and $(\forall x)\varphi$ are formulas.

That's it for formulas.

Given a formula φ , we define the set of free variables of φ

$FV(\varphi)$

by induction

- $FV(R(x_1, \dots, x_n)) = \{x_1, \dots, x_n\}$
- $FV(\perp) = \emptyset$

- $FV(T) = \emptyset$
- $FV(\varphi \vee \psi) = FV(\varphi) \cup FV(\psi)$
- $FV(\varphi \wedge \psi) = FV(\varphi) \cup FV(\psi)$
- $FV(\neg \varphi) = FV(\varphi)$
- $FV(\exists x \varphi) = FV(\varphi) \setminus \{x\}$
- $FV(\forall x \varphi) = FV(\varphi) \setminus \{x\}$

DEF A sentence is a formula φ with no free variables
 i.e. such that $FV(\varphi) = \emptyset$.

E.g. $\forall x x \leq x$ sentence

$x \leq x$ not a sentence
 $\uparrow \uparrow$

DEF. If L is a (relational, single-sorted) language, then an L -structure is a set A equipped with a subset

$$R^A \subseteq A^{\text{ar}(R)} \quad \text{for each } R \in L.$$

$$\underbrace{A \times \dots \times A}_{\text{ar}(R) \text{ times}}$$

arity 2
↓

E.g: $L = \{ \leq \}$

L -structure
is a set equipped
with a binary
relation.

Def of TRUTH/VALIDITY

Given an L -structure A , a formula φ , a finite set X of variables s.t. $FV(\varphi) \subseteq X$, and a function $\nu: X \rightarrow A$, we define

the expression

$$A, \nu \models \varphi$$

" φ holds in A under the assignment ν "

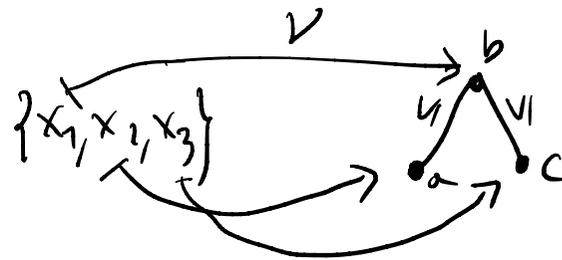
by induction on the complexity of φ .

$$A, \nu \models R(x_1, \dots, x_n)$$

iff

$$(\nu(x_1), \dots, \nu(x_n)) \in R^A$$

n times
 $Ax \dots xA$
 \cup



? NO.
 $A, \nu \models x_1 \leq x_2$

$(\nu(x_1), \nu(x_2)) \in \leq^A$

$\nu(x_1) \leq \nu(x_2)$
 " NO! "

$A, \nu \models 1$ iff True (i.e. always)

$A, \nu \models 0$ iff false (i.e. never)

$A, \nu \models \varphi \vee \psi$ iff $A, \nu \models \varphi$ or $A, \nu \models \psi$

$A, \nu \models \varphi \wedge \psi$ iff $A, \nu \models \varphi$ and $A, \nu \models \psi$

$A, \nu \models \neg \varphi$ iff not $A, \nu \models \varphi$

$A, \nu \models \exists x \varphi$
 $\begin{matrix} \uparrow \\ x \rightarrow A \end{matrix}$

iff... \rightarrow (if $x \notin X$)

...exists $a \in A$ s.t. $A, \nu' \models \varphi$

$\nu': X \cup \{x\} \rightarrow A$

$y \mapsto \begin{cases} \nu(y) & \text{if } y \in X \\ a & \text{if } y = x \end{cases}$

\rightarrow (if $x \in X$) . Pick any variable $z \notin X$

... exists $a \in A$ s.t. $A, \nu' \models \varphi[z/x]$, where

$$v': X \cup \{z\} \rightarrow A$$

$$y \mapsto \begin{cases} v(y) & \text{if } y \in X \\ a & \text{if } y = z \end{cases}$$

$A, v \models \forall x \varphi$ iff (pick any variable $z \notin X$)

for all all $a \in A$ we have $A, v' \models \varphi$

$$v': X \cup \{z\} \rightarrow A$$

$$y \mapsto \begin{cases} v(y) & \text{if } y \in X \\ a & \text{if } y = z \end{cases}$$

Given a sentence φ , we write

$$A \models \varphi \text{ iff } A, \emptyset \models \varphi$$

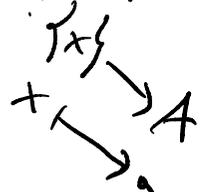
$\emptyset \rightarrow A$

E.g.

$$A \models \exists x \forall y x \leq y \Leftrightarrow A, \emptyset \models \exists x \forall y x \leq y$$

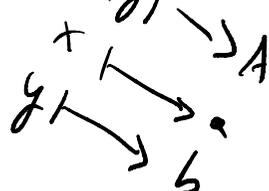
$$\Downarrow$$

exists $a \in A$ s.t. $A, \nu \models \forall y (x \leq y)$



$$\Downarrow$$

exists $a \in A$ s.t. for all $b \in A$

$$A, \nu \models x \leq y$$


$$\Downarrow$$

exists $a \in A$ s.t. for all $b \in A$
 $(\nu(x), \nu(y)) \in \leq^A$, i.e. $a \leq^A b$

PROPOSITIONAL SYMBOLS are the relation symbols of arity 0.

Def A Theory in a language L is a set of sentences in
the language L ↳ no free variables.

Def A Model of a theory T in a language L is
an L -structure s.t., for every $\varphi \in T$, $A \models \varphi$.

Def (SEMANTIC EQUIVALENCE)

Given a theory T , a finite set X of variables and two
formulas φ, ψ s.t. $FV(\varphi) \subseteq X$
 $FV(\psi) \subseteq X$,

We say that φ and ψ are SEMANTICALLY EQUIVALENT modulo
 T in context X , written

$$\varphi \equiv_x^T \psi$$

if, for every model A of T , for every function $\nu: X \rightarrow A$,

$$A, \nu \models \varphi \text{ iff } A, \nu \models \psi$$

Def (ESSENTIALLY, LAWVERE, 1970)

LAWVERE defined hyperdoctrines.
This is a variation.

A first-order Boolean doctrine (over $\text{FinSet}^{\text{op}}$) is
a functor

$$P: \text{FinSet} \rightarrow \text{BA}$$

s.t.

FinSet = category
of finite
sets and
functions

T Theory in \mathcal{L}

$P(X) :=$ set of formulae φ
s.t. $\text{FV}(\varphi) \subseteq X$,
modulo \equiv_x^T

Given a function $f: X \rightarrow Y$

$$P(f): P(X) \rightarrow P(Y)$$

$$\alpha \mapsto \alpha \left[\frac{f(x)}{x}, x \in X \right]$$

① For every $X \in \text{FinSet}$, for every $y \notin X$,
 for every $\varphi \in P(X \cup \{y\})$ there
 is an element $(\exists y)_x \varphi \in P(X)$ s.t.

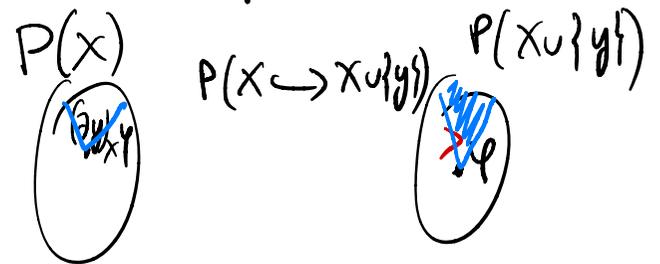
for every $\psi \in P(X)$

$$(\exists y)_x \varphi \leq \psi \iff \varphi \leq P(X \hookrightarrow X \cup \{y\})(\psi)$$

in $P(X)$
in $P(X \cup \{y\})$

(note that one such element is
 necessarily unique)

i.e. $(\exists y)_x \varphi$ is the smallest ψ s.t.
 $\varphi \leq P(X \hookrightarrow X \cup \{y\})(\psi)$



E.g.: $f: \{x, y\} \rightarrow \{y, z\}$
 $x \mapsto y$
 $y \mapsto z$

$$P(\{x, y\}) \rightarrow P(\{y, z\})$$

$$x \leq y \mapsto y \leq z$$

$$x \leq x \mapsto y \leq y$$

$$\forall z (y \leq z) \mapsto \forall w (z \leq w)$$

$$\forall w (y \leq w) \mapsto \forall w (z \leq w)$$

$\varphi \mapsto (\exists y)_x \varphi$
 \uparrow
 $P(X \cup \{y\})$

E.g. $f: \{x, y\} \rightarrow \{z\}$

$$P(f): x \leq y \mapsto z \leq z$$

$$P(X \hookrightarrow X \cup \{y\}): P(X) \rightarrow P(X \cup \{y\})$$

$$\alpha \mapsto \alpha \left[\begin{array}{c} \text{C}(x) \\ \text{X} \end{array} \right]$$

Equivalently: the function

$$\alpha \left[\frac{x}{x} \cdot x \right]$$

$P(X \hookrightarrow X \cup \{y\})$ has a

left adjoint $(\exists y)_x : P(X \cup \{y\}) \rightarrow P(X)$ (as a function between posets)

Equivalently it has a right adjoint

$$(\forall y)_x : P(X \cup \{y\}) \rightarrow P(X) \text{ i.e.}$$

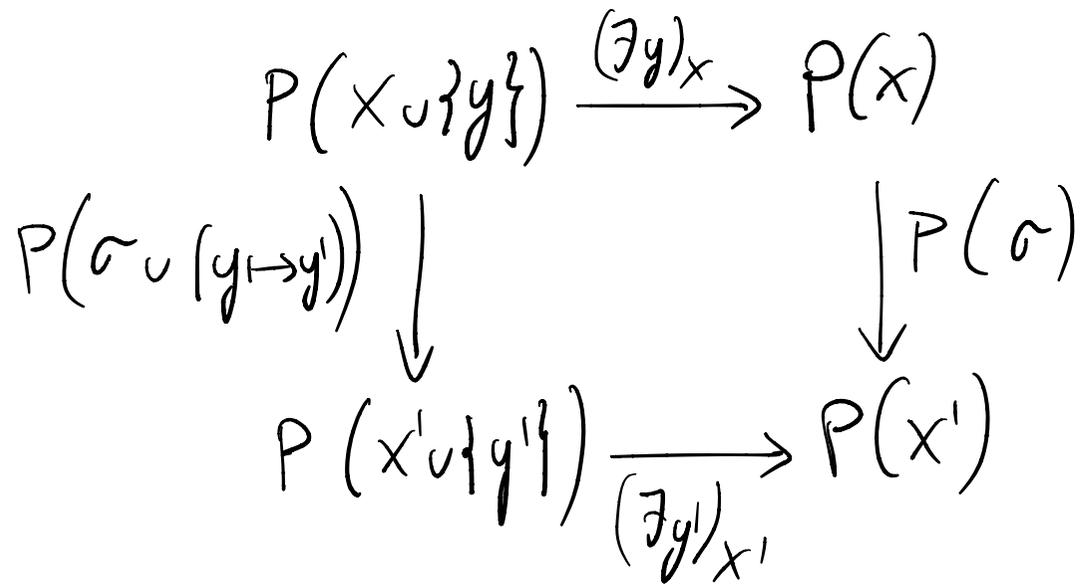
for every $\varphi \in P(X \cup \{y\})$ there is an element $(\forall y)_x \varphi \in P(X)$

n.t. for all $\varphi \in P(X)$:

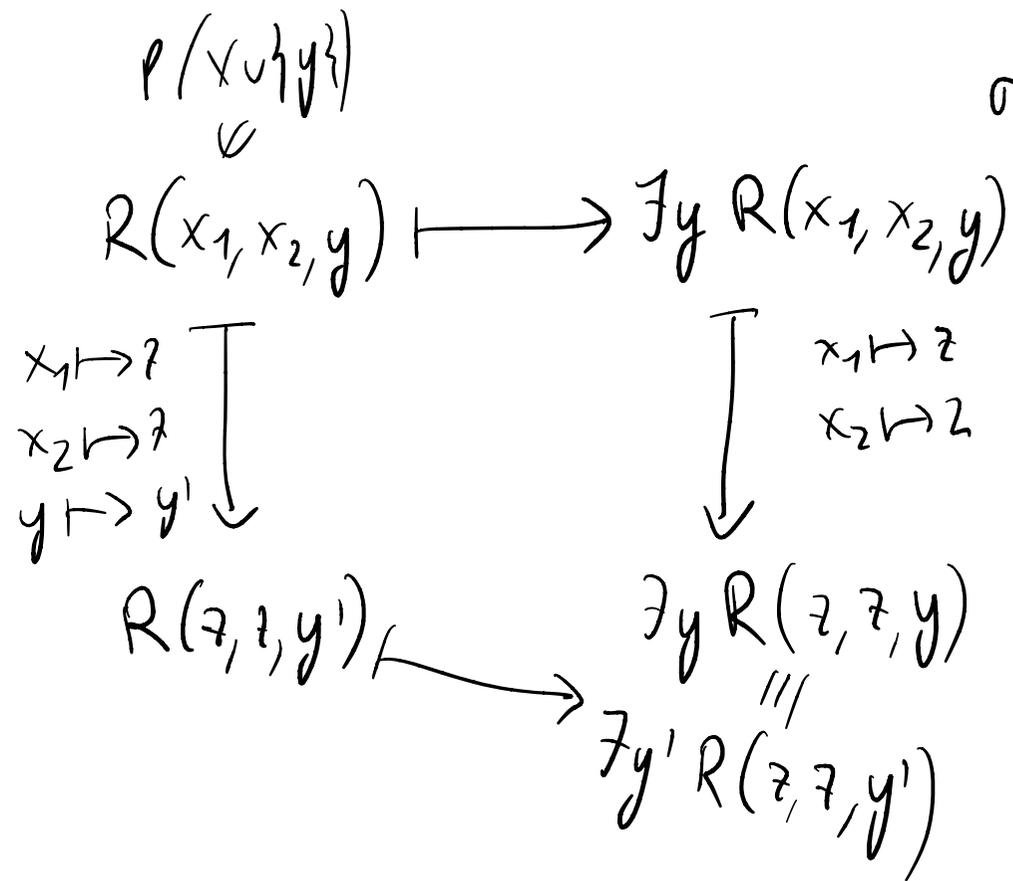
$$P(X \hookrightarrow X \cup \{y\})(\varphi) \geq \varphi \iff \varphi \leq (\forall y)_x \varphi$$

② Beck-Chevalley: SUBSTITUTIONS COMMUTE WITH QUANTIFIERS

For all $X, X' \in \text{FinSet}$, for every $\sigma : X \rightarrow X'$, for every $y \notin X, y' \notin X'$,
the following commutes



E.g.: $X = \{x_1, x_2\}$,
 $X' = \{z\}$
 $\sigma : \{x_1, x_2\} \rightarrow \{z\}$



Equivalently, I could ask that a similar diagram for \forall commutes.

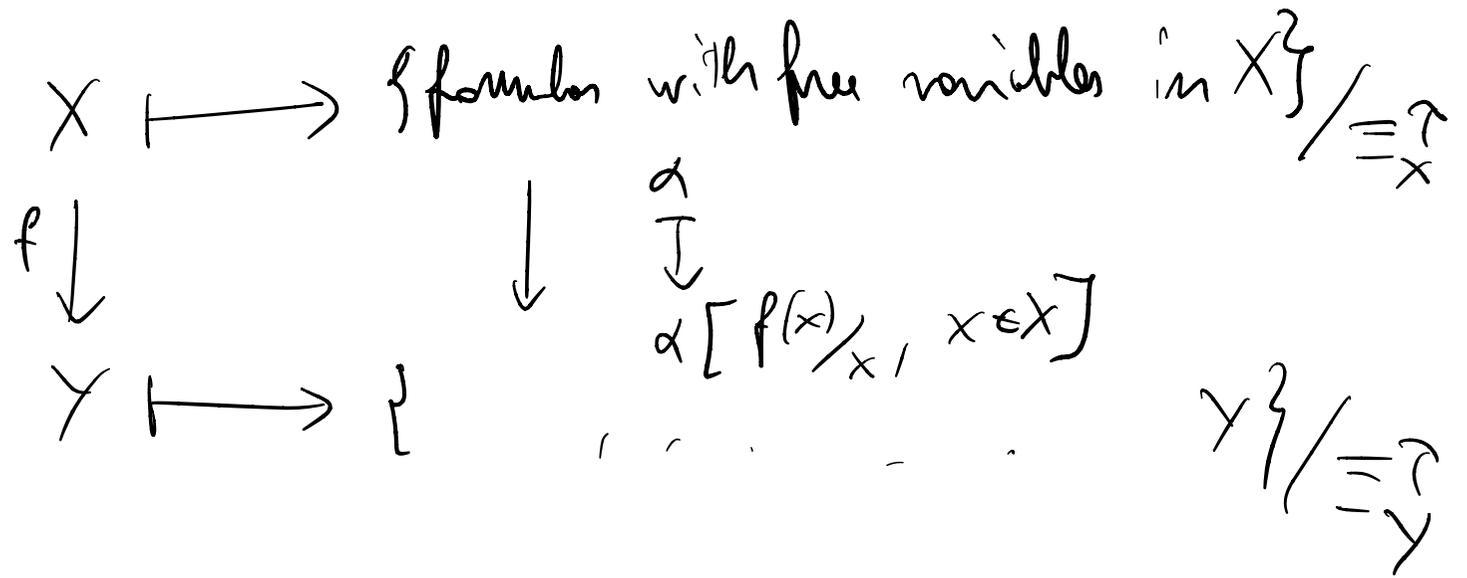
$$\begin{array}{ccc} P(x \cup \{y\}) & \xrightarrow{(\forall y)_x} & P(x) \\ P(\sigma \cup \{y \mapsto y'\}) & \downarrow & \downarrow P(\sigma) \\ P(x' \cup \{y'\}) & \xrightarrow{(\forall y')_{x'}} & P(x') \end{array}$$

END OF THE DEF.

Given a first-order theory T in a language L (relations
one-sorted
no eqs)

\mathcal{M} get - FOBD over $\text{FinSet}^{\text{PP}}$

$LT_T : \text{FinSet} \rightarrow \text{BA}$



One can prove that also the converse holds:

every FOBD over $\text{FinSet}^{\text{op}}$ is isomorphic to LT_T for some theory T in some language L_0 .

This is for the case \rightarrow NO FUNCT. SYMB, }
 SINGLE SORT }
 NO EQUALITY.

For equality: further conditions.

A first-order Bool. doctrine over a algebraic theory category \mathcal{C} with finite products
 is a functor

$$\mathcal{C}^{op} \rightarrow BA$$

s.t. ...

In our case,

$$\mathcal{C} = \text{FinSet}^{op}$$

ONLY RELATIONS, SINGLE SORT.

$$\text{FinSet} = (\text{FinSet}^{op})^{op}$$