Implicative algebras II: completeness w.r.t. Set-based triposes

Alexandre Miquel

UNIVERSIDAD DE LA REPUBLICA URUGUAY
FACULTAD DE INGENIERIA
EQUIPO DE LÓGICA UDELAR

June 3th, 10th, 2022 – IMERL

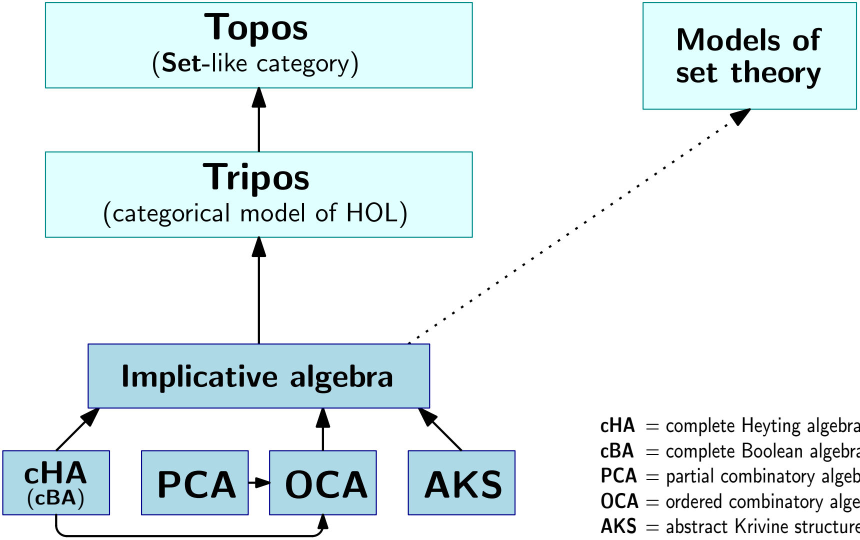
The categorical tradition of realizability

- **Categorical logic** [Lawvere, Tierney '70]
  - Hyperdoctrines = models of 1st order theories
    (Slogan: ∃/∀ are left/right adjoints!)
  - Modern definition of the notion of topos
    (generalizes Grothendieck's definition)

- **Categorical realizability** [Hyland, Johnstone, Pitts '80]
  - Major input from forcing and Boolean-valued models [Scott]
  - The effective topos [Hyland]
  - Notion of tripos and tripos-to-topos construction [Pitts]
  - Generalization to partial combinatory algebras (PCAs)
    ... but incompatible with classical logic

- **Categorical classical realizability**
  - Classical realizability from a categorical perspective [Streicher '13]
  - Ordered combinatory algebras and realizability [Ferrer *et al.* '17]
  - Implicative algebras [Miquel '20]

The categorical problem



cHA = complete Heyting algebra
cBA = complete Boolean algebra
PCA = partial combinatory algebra
OCA = ordered combinatory algebra
AKS = abstract Krivine structure

Unifying all kinds of models

- **Implicative algebras:** a simple algebraic structure that encompasses:
  - Complete Heyting Algebras (for Heyting-valued models, Kripke forcing)
  - Complete Boolean Algebras (for Boolean-valued models, Cohen forcing)
  - Partial Combinatory Algebras (for Intuitionistic realizability)
  - Ordered Combinatory Algebras (for Intuitionistic realizability)
  - Abstract Krivine Structures (for Classical realizability)
- Implicative algebras can be used to construct implicative triposes, thus encompassing all the currently known ($\mathbf{Set}$-based) triposes
- But do implicative triposes encompass all ($\mathbf{Set}$-based) triposes?
- Yes! The aim of this talk is to prove the

Theorem (Completeness)

Every ($\mathbf{Set}$-based) tripos is isomorphic to an implicative tripos

Plan

1. Introduction
2. Set-based triposes
3. Anatomy of a Set-based tripos
4. Extracting the implicative algebra

Plan

1. Introduction
2. Set-based triposes
3. Anatomy of a Set-based tripos
4. Extracting the implicative algebra

Some notations

- In what follows, we write:
  - $\mathbf{Set}$ the category of sets equipped with all maps
  - $\mathbf{Pos}$ the category of posets equipped with monotonic functions
  - $\mathbf{HA}$ the category of Heyting algebras equipped with the morphisms of Heyting algebras (i.e. that commute with $\bot$, $\top$, $\wedge$, $\vee$, $\to$)
- In the category $\mathbf{Set}$, we write:
  - 1 the terminal object (i.e. a fixed singleton)
  - $1_X : X \to 1$ the unique map from a given set X to 1
  - $X \times Y$ the Cartesian product of two sets X and Y, with the associated projections $\pi_{X,Y} : X \times Y \to X$ and $\pi'_{X,Y} : X \times Y \to Y$
  - Given maps $f : Z \to X$ and $g : Z \to Y$, we write $\langle f, g\rangle : Z \to X \times Y$ the unique map such that $\pi_{X,Y} \circ \langle f, g\rangle = f$ and $\pi'_{X,Y} \circ \langle f, g\rangle = g$

Set-based triposes

Definition 1.1 (Set-based tripos)

A (**Set**-based) tripos is a contravariant functor $\mathbf{P} : \mathbf{Set}^{\mathrm{op}} \to \mathbf{HA}$ such that:

1. For each map $f : X \to Y$ (in $\mathbf{Set}$), the associated map $\mathbf{P}f : \mathbf{P}Y \to \mathbf{P}X$ (in $\mathbf{HA}$) has left & right adjoints $\exists f, \forall f : \mathbf{P}X \to \mathbf{P}Y$ (in $\mathbf{Pos}$)
2. Beck-Chevalley condition: Each pullback square in $\mathbf{Set}$ (on the lhs) induces the following two commutative diagrams in $\mathbf{Pos}$ (on the rhs):

$$\begin{array}{ccc} X & \xrightarrow{f_1} & X_1 \\ {\scriptstyle f_2}\downarrow & \lrcorner & \downarrow{\scriptstyle g_1} \\ X_2 & \xrightarrow[g_2]{} & Y \end{array} \quad\Rightarrow\quad \begin{array}{ccc} \mathbf{P}X & \xrightarrow{\exists f_1} & \mathbf{P}X_1 \\ {\scriptstyle \mathbf{P}f_2}\uparrow & & \uparrow{\scriptstyle \mathbf{P}g_1} \\ \mathbf{P}X_2 & \xrightarrow[\exists g_2]{} & \mathbf{P}Y \end{array} \qquad \begin{array}{ccc} \mathbf{P}X & \xrightarrow{\forall f_1} & \mathbf{P}X_1 \\ {\scriptstyle \mathbf{P}f_2}\uparrow & & \uparrow{\scriptstyle \mathbf{P}g_1} \\ \mathbf{P}X_2 & \xrightarrow[\forall g_2]{} & \mathbf{P}Y \end{array}$$

3. The functor $\mathbf{P} : \mathbf{Set}^{\mathrm{op}} \to \mathbf{HA}$ has a generic predicate $tr_\Sigma \in \mathbf{P}\Sigma$ (for some set Σ), i.e. such that for all sets X, the following map is surjective:

$$\begin{array}{rcl} \Sigma^X & \to & \mathbf{P}X \\ \sigma & \mapsto & \mathbf{P}\sigma(tr_\Sigma) \end{array}$$

On the definitions of the notion of tripos

- The above definition is the initial definition of triposes, such as introduced in Hyland, Johnstone, Pitts: *Tripos theory* (1980)

- Pitts' PhD *The Theory of Triposes* (1981) generalizes the notion of tripos in essentially two directions:

  1. The category **Set** is replaced by an arbitrary Cartesian category $\mathbf{C}$ (intuitively: a category of 'contexts'), and the generic predicate is replaced by a more general membership predicate[1]

  2. The Beck-Chevalley condition is only required for certain pullback squares (the projection squares), and may not hold for all

- However, all forcing/realizability/implicative triposes are triposes in the sense of the initial definition (i.e. Set-based triposes); therefore we shall only consider these

[1] Due to the fact that the Cartesian category $\mathbf{C}$ is not necessarily closed. But when $\mathbf{C}$ is a ccc, the existence of the generic predicate is sufficient

Isomorphism of (Set-based) triposes

Definition 1.2 (Isomorphism of triposes)

Two triposes $\mathbf{P}, \mathbf{P}' : \mathbf{Set}^{\mathrm{op}} \to \mathbf{HA}$ are isomorphic when there is a natural isomorphism $\varphi : \mathbf{P} \Rightarrow \mathbf{P}'$, i.e. a family of isos $\varphi_X : \mathbf{P}X \to \mathbf{P}'X$ ($X \in \mathbf{Set}$) such that the following diagram commutes

$$\begin{array}{ccc} \mathbf{P}X & \xrightarrow[\sim]{\varphi_X} & \mathbf{P}'X \\ {\scriptstyle \mathbf{P}f}\uparrow & & \uparrow{\scriptstyle \mathbf{P}'f} \\ \mathbf{P}Y & \xrightarrow[\varphi_Y]{\sim} & \mathbf{P}'Y \end{array} \qquad \text{for all maps} \qquad \begin{array}{c} X \\ {\scriptstyle f}\downarrow \\ Y \end{array}$$

- The notion of iso can be taken indifferently in $\mathbf{HA}$ or in $\mathbf{Pos}$, since a map $\varphi_X : \mathbf{P}X \to \mathbf{P}'X$ is an iso in $\mathbf{HA}$ iff it is an iso in $\mathbf{Pos}$
- There is no need to take care about generic predicates!

  Reason: A natural iso will automatically map any generic predicate of $\mathbf{P}$ to a generic predicate of $\mathbf{P}'$ (generic predicates are not unique)

Some properties of triposes (1/2)

Lemma 1.3 (Functoriality of $\forall, \exists$)

The correspondences $f \mapsto \exists f$ and $f \mapsto \forall f$ are functorial:

$$\begin{array}{rclcrcl} \exists \mathrm{id}_X & = & \mathrm{id}_{\mathbf{P}X} & \qquad & \exists(g \circ f) & = & \exists g \circ \exists f \\ \forall \mathrm{id}_X & = & \mathrm{id}_{\mathbf{P}X} & \qquad & \forall(g \circ f) & = & \forall g \circ \forall f \end{array}$$

for all maps $f : X \to Y$ and $g : Y \to Z$ (in $\mathbf{Set}$)

Lemma 1.4 (Commutation with finite joins/meets)

Left adjoints $\exists f : \mathbf{P}X \to \mathbf{P}Y$ commute with all finite joins whereas right adjoints $\forall f : \mathbf{P}X \to \mathbf{P}Y$ commute with all finite meets:

$$\begin{array}{rclcrcl} \exists f(\bot_X) & = & \bot_Y & \qquad & \exists f(p \vee p') & = & \exists f(p) \vee \exists f(p') \\ \forall f(\top_X) & = & \top_Y & \qquad & \forall f(p \wedge p') & = & \forall f(p) \wedge \forall f(p') \end{array}$$

for all maps $f : X \to Y$ (in $\mathbf{Set}$) and for all predicates $p, p \in \mathbf{P}X$

Remark: The same property holds more generally for infinitary joins/meets (when they exist), but we shall never use this generalization

Some properties of triposes (2/2)

Lemma 1.5 (Adjoints of inverses)

Given a map $f : X \to Y$ (in $\mathbf{Set}$):

1. If f has an inverse, then $\exists f, \forall f$ are the inverse of $\mathbf{P}f$:
$$\exists f \;=\; \forall f \;=\; (\mathbf{P}f)^{-1} \;=\; \mathbf{P}f^{-1}$$
2. If f has a right inverse, then $\exists f$ and $\forall f$ are left inverses of $\mathbf{P}f$:
$$\exists f \circ \mathbf{P}f \;=\; \forall f \circ \mathbf{P}f \;=\; \mathrm{id}_{\mathbf{P}Y}$$
3. If f has a left inverse, then $\exists f$ and $\forall f$ are right inverses of $\mathbf{P}f$:
$$\mathbf{P}f \circ \exists f \;=\; \mathbf{P}f \circ \forall f \;=\; \mathrm{id}_{\mathbf{P}X}$$

Recall that in $\mathbf{Set}$, a map $f : X \to Y$

- has an inverse iff it is bijective
- has a right inverse iff it is surjective (AC)
- has a left inverse iff it is injective and $(X = \varnothing \Rightarrow Y = \varnothing)$

Implicative algebras

Definition 1.6 (Implicative algebra)

1. An implicative structure is a complete lattice $(\mathscr{A}, \preccurlyeq)$ equipped with a binary operation $(\to) : \mathscr{A}^2 \to \mathscr{A}$ such that:

   (1) If $a' \preccurlyeq a$ and $b \preccurlyeq b'$, then $(a \to b) \preccurlyeq (a' \to b')$

   (2) For all $a \in \mathscr{A}$ and $B \subseteq \mathscr{A}$, we have: $a \to \bigcurlywedge_{b \in B} b = \bigcurlywedge_{b \in B} (a \to b)$

2. A separator of $(\mathscr{A}, \preccurlyeq, \to)$ is a subset $S \subseteq \mathscr{A}$ such that:

   (1) If $a \in S$ and $a \preccurlyeq a'$, then $a' \in S$

   (2) $\bigcurlywedge_{a,b \in \mathscr{A}} (a \to b \to c) \ (= \mathbf{K}^{\mathscr{A}}) \in S$ and $\bigcurlywedge_{a,b,c \in \mathscr{A}} ((a \to b \to c) \to (a \to b) \to a \to c) \ (= \mathbf{S}^{\mathscr{A}}) \in S$

   (3) If $(a \to b) \in S$ and $a \in S$, then $b \in S$

3. An implicative algebra is an implicative structure $(\mathscr{A}, \preccurlyeq, \to)$ together with a separator $S \subseteq \mathscr{A}$

Construction of the implicative tripos

Let $(\mathscr{A}, \preccurlyeq, \to, S)$ be an implicative algebra

- For each set X, we write $\mathbf{P}X := \mathscr{A}^X/S[X]$ the poset reflection of the preordered set $(\mathscr{A}^X, \vdash_{S[X]})$, where

$$a \vdash_{S[X]} b \qquad \text{iff} \qquad \bigwedge_{x \in X}(a_x \to b_x) \in S \qquad \text{(for all } a, b \in \mathscr{A}^X\text{)}$$

  (By construction, $\mathbf{P}X$ is a Heyting algebra)

- For each map $f : X \to Y$, we write $\mathbf{P}f : \mathbf{P}Y \to \mathbf{P}X$ the unique map that factors the map $\mathscr{A}^f = (a \mapsto a \circ f) : \mathscr{A}^Y \to \mathscr{A}^X$ through the quotients $\mathbf{P}Y := \mathscr{A}^Y/S[Y]$ and $\mathbf{P}X := \mathscr{A}^X/S[X]$.

  (By construction, $\mathbf{P}f : \mathbf{P}Y \to \mathbf{P}X$ is a morphism of Heyting algebras)

Theorem 1.7 (Implicative tripos)

The functor $\mathbf{P} : \mathbf{Set}^{\mathsf{op}} \to \mathbf{HA}$ is a ($\mathbf{Set}$-based) tripos

Remark: A generic predicate of $\mathbf{P}$ is given by $\Sigma := \mathscr{A}$ and $tr_\Sigma := [\mathrm{id}_{\mathscr{A}}]_{/S[\mathscr{A}]}$

The completeness theorem

Each tripos $\mathbf{P} : \mathbf{Set}^{\mathsf{op}} \to \mathbf{HA}$ constructed from an implicative algebra $(\mathscr{A}, \preccurlyeq, \to, S)$ as shown above is called an implicative tripos

The aim of this talk is to show that:

Main Theorem (Completeness)

Each $\mathbf{Set}$-based tripos is (isomorphic to) an implicative tripos

- This theorem explains *a fortiori* why we succeeded to turn all the well-known triposes (induced by HA/AKS/PCA/OCA/etc.) into implicative triposes
- From the point of view of foundations, the above theorem expresses that the whole structure of a tripos (a structured proper class) can be described by a single implicative algebra (a structured set)

  $\Rightarrow$ Reduction of complexity

Architecture of the proof (1/2)

The proof is organized in two parts:

Part 1: Anatomy of a tripos ("reducing the complexity")[2]

- Given
  - an arbitrary tripos $\mathbf{P} : \mathbf{Set}^{\text{op}} \to \mathbf{HA}$
  - a generic predicate $tr_\Sigma \in \mathbf{P}\Sigma$ (over some set of propositions Σ)

  we show that the whole structure of the functor $\mathbf{P}$ can be derived from suitable (and non canonical) "connectives" and "quantifiers"

  $$(\dot{\vee}), (\dot{\wedge}), (\dot{\to}) \ : \ \Sigma^2 \to \Sigma \qquad (\dot{\bigvee}), (\dot{\bigwedge}) \ : \ \mathfrak{P}(\Sigma) \to \Sigma$$

  via a suitable (and canonical) "filter" $\Phi \subseteq \Sigma$
- So that, morally: "everything happens in Σ"
- However, the set Σ equipped with these operations has no good algebraic properties (it only looks like a complete Heyting algebra)

[2] This part is essentially taken from [Hyland, Johnstone, Pitts, 1980]

Architecture of the proof (2/2)

Part 2: Extracting the implicative algebra (“regularizing Σ”)

- We first observe that the whole structure of the functor $\mathbf{P}$ can be derived from the only (non canonical) operations

$$(\dot{\to}) \;:\; \Sigma^2 \to \Sigma, \qquad (\dot{\bigwedge}) \;:\; \mathfrak{P}(\Sigma) \to \Sigma \qquad \text{via} \qquad \Phi \subseteq \Sigma$$

(the other connectives/quantifier being irrelevant)

- Using domain-theoretic techniques, we turn Σ into an implicative algebra $\mathscr{A} = (\mathscr{A}, \preccurlyeq, \to, S)$, that becomes a new set of propositions, with its own generic predicate $tr_{\mathscr{A}} \in \mathbf{P}\mathscr{A}$
- However, when passing from Σ to $\mathscr{A}$, the non canonical operations on Σ are turned into the canonical operations

$$(\to) \;:\; \mathscr{A}^2 \to \mathscr{A}, \qquad (\curlywedge) \;:\; \mathfrak{P}(\mathscr{A}) \to \mathscr{A} \qquad \text{via} \qquad S \subseteq \mathscr{A}$$

(The “algebraic regularization” is achieved by the very construction of $\mathscr{A}$)

- Hence the functor $\mathbf{P}$ can be derived from $(\mathscr{A}, \to, \curlywedge, S)$, which precisely means that $\mathbf{P}$ is the implicative tripos induced by $\mathscr{A}$

Plan

1 Introduction

2 Set-based triposes

3 Anatomy of a Set-based tripos

4 Extracting the implicative algebra

The generic predicate (1/3)

From now on, we work with a fixed tripos $\mathbf{P} : \mathbf{Set}^{\mathrm{op}} \to \mathbf{HA}$

- We take a generic predicate $tr_\Sigma \in \mathbf{P}\Sigma$ (for some set Σ). This means that for each set X, the 'decoding map'

$$\begin{array}{rcll} [\![\_]\!]_X : \Sigma^X & \to & \mathbf{P}X & \\ \sigma & \mapsto & \mathbf{P}\sigma(tr_\Sigma) & \quad \text{is surjective} \end{array}$$

- Intuitively, Σ is the set of (codes of) propositions, whereas Σ^X is the set of propositional functions over X
- The condition of surjectivity expresses that each predicate $p \in \mathbf{P}X$ is represented by *at least* one propositional function $\sigma \in \Sigma^X$ such that $[\![\sigma]\!]_X = p$, which we call a code for the predicate p
- **Remark:** Since codes for predicates are not unique, all the constructions involving such codes will be non canonical

The generic predicate (2/3)

In a tripos $\mathbf{P} : \mathbf{Set}^{\mathsf{op}} \to \mathbf{HA}$, the generic predicate is never unique!

Lemma 2.1 (Non-uniqueness of generic predicates)

Given a generic predicate $tr_\Sigma \in \mathbf{P}\Sigma$ and a surjection $h : \Sigma' \to \Sigma$, the predicate $tr_{\Sigma'} := \mathbf{P}h(tr_\Sigma) \in \mathbf{P}\Sigma'$ is another generic predicate of $\mathbf{P}$

Proof: Uses the fact that $h : \Sigma' \to \Sigma$ has a right inverse, by (AC). The same result holds without (AC) by replacing 'surjective' by 'having a right inverse'

More generally:

Lemma 2.2 (Conversion between generic predicates)

If $tr_\Sigma \in \mathbf{P}\Sigma$ and $tr_{\Sigma'} \in \mathbf{P}\Sigma'$ are two generic predicates of the tripos $\mathbf{P}$, then there exist two conversion maps $h : \Sigma' \to \Sigma$ and $h' : \Sigma \to \Sigma'$ such that $tr_{\Sigma'} = \mathbf{P}h(tr_\Sigma)$ and $tr_\Sigma = \mathbf{P}h'(tr_{\Sigma'})$

In what follows, we work with a fixed generic predicate $tr_\Sigma \in \mathbf{P}\Sigma$

The generic predicate (3/3)

Recall that for each set X, the 'decoding function' $[\![\_]\!]_X : \Sigma^X \to \mathbf{P}X$ is defined by $\quad [\![\sigma]\!]_X = \mathbf{P}\sigma(tr_\Sigma) \quad$ for all $\sigma \in \Sigma^X$

Proposition 2.3 (Naturality of $[\![\_]\!]_X$)

The decoding map $\;[\![\_]\!]_X : \Sigma^X \to \mathbf{P}X\;$ is natural in X:

$$\begin{array}{ccc} \Sigma^X & \xrightarrow{[\![\_]\!]_X} & \mathbf{P}X \\ {\scriptstyle \_\circ f}\uparrow & & \uparrow{\scriptstyle \mathbf{P}f} \\ \Sigma^Y & \xrightarrow[{[\![\_]\!]_Y}]{} & \mathbf{P}Y \end{array} \qquad \text{commutes for all} \qquad \begin{array}{c} X \\ {\scriptstyle f}\downarrow \\ Y \end{array}$$

Notations. Given $\sigma = (\sigma)_{x\in X} \in \Sigma^X$, we write $\;[\![\sigma_x]\!]_{x\in X} := [\![\sigma]\!]_X \in \mathbf{P}X$
In particular, given an individual code $\xi \in \Sigma$, we write

- $(\xi)_{\_\in 1} \in \Sigma^1$ the 1-element family formed by ξ
- $[\![\xi]\!]_{\_\in 1} \in \mathbf{P}1$ the associated predicate

Defining connectives in Σ (1/2)

- Writing $\pi, \pi' : \Sigma \times \Sigma \to \Sigma$ the two projections from $\Sigma \times \Sigma$ to Σ, we choose codes $(\dot\wedge), (\dot\vee), (\dot\to) \in \Sigma^{\Sigma\times\Sigma}$ such that

$$\begin{aligned} [\![\dot\wedge]\!]_{\Sigma\times\Sigma} &= [\![\pi]\!]_{\Sigma\times\Sigma} \wedge [\![\pi']\!]_{\Sigma\times\Sigma} && (\in \mathbf{P}(\Sigma\times\Sigma)) \\ [\![\dot\vee]\!]_{\Sigma\times\Sigma} &= [\![\pi]\!]_{\Sigma\times\Sigma} \vee [\![\pi']\!]_{\Sigma\times\Sigma} && (\in \mathbf{P}(\Sigma\times\Sigma)) \\ [\![\dot\to]\!]_{\Sigma\times\Sigma} &= [\![\pi]\!]_{\Sigma\times\Sigma} \to [\![\pi']\!]_{\Sigma\times\Sigma} && (\in \mathbf{P}(\Sigma\times\Sigma)) \end{aligned}$$

Proposition 2.4

For all sets X and for all codes $\sigma, \tau \in \Sigma^X$, we have

$$\begin{aligned} [\![\sigma_x \mathbin{\dot\wedge} \tau_x]\!]_{x\in X} &= [\![\sigma]\!]_X \wedge [\![\tau]\!]_X && (\in \mathbf{P}X) \\ [\![\sigma_x \mathbin{\dot\vee} \tau_x]\!]_{x\in X} &= [\![\sigma]\!]_X \vee [\![\tau]\!]_X && (\in \mathbf{P}X) \\ [\![\sigma_x \mathbin{\dot\to} \tau_x]\!]_{x\in X} &= [\![\sigma]\!]_X \to [\![\tau]\!]_X && (\in \mathbf{P}X) \end{aligned}$$

Intuitively, the (non-canonical) operations $(\dot\wedge), (\dot\vee), (\dot\to) : \Sigma^2 \to \Sigma$ allow to compute $\wedge$, $\vee$ and $\to$ in each Heyting algebra $\mathbf{P}X$ (for $X \in \mathbf{Set}$)

Defining connectives in Σ (2/2)

- Similarly, we choose codes $\dot{\bot}, \dot{\top} \in \Sigma$ such that

$$[\![\dot{\bot}]\!]_{\_\in 1} = \bot_1 \qquad \text{and} \qquad [\![\dot{\top}]\!]_{\_\in 1} = \top_1 \qquad\qquad (\in \mathbf{P}1)$$

Again, we easily check that:

Proposition 2.5

For each set X, we have:

$$[\![\dot{\bot}]\!]_{x\in X} = \bot_X \qquad \text{and} \qquad [\![\dot{\top}]\!]_{x\in X} = \top_X \qquad\qquad (\in \mathbf{P}X)$$

Beware! Although the "connectives" $(\dot{\wedge}), (\dot{\vee}), (\dot{\to}) \ : \ \Sigma^2 \to \Sigma$ reflect the corresponding operations on each Heyting algebra $\mathbf{P}X$ (for $X \in \mathbf{Set}$), they enjoy none of the expected algebraic properties:

$$\begin{array}{lll} \xi \mathrel{\dot{\wedge}} \xi \neq \xi & \xi \mathrel{\dot{\wedge}} \xi' \neq \xi' \mathrel{\dot{\wedge}} \xi & (\xi \mathrel{\dot{\wedge}} \xi') \mathrel{\dot{\wedge}} \xi'' \neq \xi \mathrel{\dot{\wedge}} (\xi' \mathrel{\dot{\wedge}} \xi'') \\ \xi \mathrel{\dot{\vee}} \xi \neq \xi & \xi \mathrel{\dot{\vee}} \xi' \neq \xi' \mathrel{\dot{\vee}} \xi & (\xi \mathrel{\dot{\vee}} \xi') \mathrel{\dot{\vee}} \xi'' \neq \xi \mathrel{\dot{\vee}} (\xi' \mathrel{\dot{\vee}} \xi'') \end{array} \qquad \text{etc.}$$

In particular, Σ (with these operations) is not a Heyting algebra!

Defining quantifiers in Σ (1/3)

- Consider the membership relation

$$E \;:=\; \{(\xi, s) \in \Sigma \times \mathfrak{P}(\Sigma) \;:\; \xi \in s\}$$

together with its projections $e_1 : E \to \Sigma$ and $e_2 : E \to \mathfrak{P}(\Sigma)$

- We now choose codes $(\dot{\bigvee}), (\dot{\bigwedge}) \;\in\; \Sigma^{\mathfrak{P}(\Sigma)}$ such that

$$[\![\dot{\bigvee}]\!]_{\mathfrak{P}(\Sigma)} \;=\; \exists e_2([\![e_1]\!]_E) \qquad (\in \mathbf{P}(\mathfrak{P}(\Sigma)))$$

$$[\![\dot{\bigwedge}]\!]_{\mathfrak{P}(\Sigma)} \;=\; \forall e_2([\![e_1]\!]_E) \qquad (\in \mathbf{P}(\mathfrak{P}(\Sigma)))$$

Proposition 2.6

Given a code $\sigma = (\sigma_x)_{x \in X} \in \Sigma^X$ and a map $f : X \to Y$, we have:

$$\Big[\!\!\Big[\dot{\bigvee}\{\sigma_x : x \in f^{-1}(y)\}\Big]\!\!\Big]_{y \in Y} \;=\; \exists f([\![\sigma]\!]_X) \qquad (\in \mathbf{P}Y)$$

$$\Big[\!\!\Big[\dot{\bigwedge}\{\sigma_x : x \in f^{-1}(y)\}\Big]\!\!\Big]_{y \in Y} \;=\; \forall f([\![\sigma]\!]_X) \qquad (\in \mathbf{P}Y)$$

Intuitively, the (non-canonical) operations $(\dot{\bigvee}), (\dot{\bigwedge}) \;:\; \mathfrak{P}(\Sigma) \to \Sigma$ allow to compute left and right adjoints along all maps $f : X \to Y$

Defining quantifiers in Σ (2/3)

Proof of Proposition 2.6.

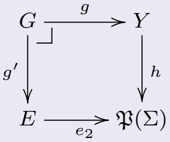
Define the map $h : Y \to \mathfrak{P}(\Sigma)$ by $h(y) := \{\sigma_x : x \in f^{-1}(y)\}$ for all $y \in Y$.
From this definition and from the definitions of $\dot{\bigvee}$, $\dot{\bigwedge}$, we get

$$\begin{aligned}
[\![\dot{\bigvee}\{\sigma_x : x \in f^{-1}(y)\}]\!]_{y\in Y} &= [\![\dot{\bigvee} \circ h]\!]_Y = \mathbf{P}h\big([\![\dot{\bigvee}]\!]_{\mathfrak{P}(\Sigma)}\big) = \mathbf{P}h\big(\exists e_2([\![e_1]\!]_E)\big) \\
[\![\dot{\bigwedge}\{\sigma_x : x \in f^{-1}(y)\}]\!]_{y\in Y} &= [\![\dot{\bigwedge} \circ h]\!]_Y = \mathbf{P}h\big([\![\dot{\bigwedge}]\!]_{\mathfrak{P}(\Sigma)}\big) = \mathbf{P}h\big(\forall e_2([\![e_1]\!]_E)\big)
\end{aligned}$$

Let us now consider the set $G \subseteq \Sigma \times Y$ defined by $G := \{(\sigma_x, f(x)) : x \in X\}$ as well as the two functions $g : G \to Y$ and $g' : G \to E$ given by

$$g(\xi, y) := y \qquad \text{and} \qquad g'(\xi, y) := (\xi, h(y)) \qquad (\text{for all } (\xi, y) \in G)$$

We observe that the following diagram is a pullback in $\mathbf{Set}$:



Hence $\mathbf{P}h \circ \exists e_2 = \exists g \circ \mathbf{P}g'$ and $\mathbf{P}h \circ \forall e_2 = \forall g \circ \mathbf{P}g'$ (Beck-Chevalley). (...)

Defining quantifiers in Σ (3/3)

Proof of Proposition 2.6 (continued).

From the equalities $\mathbf{P}h \circ \exists e_2 = \exists g \circ \mathbf{P}g'$ and $\mathbf{P}h \circ \forall e_2 = \forall g \circ \mathbf{P}g'$, we get:

$$\begin{array}{rcccl}
[\![\dot{\bigvee}\{\sigma_x : x \in f^{-1}(y)\}]\!]_{y \in Y} & = & (\mathbf{P}h \circ \exists e_2)([\![e_1]\!]_E) & = & (\exists g \circ \mathbf{P}g')([\![e_1]\!]_E) \\
[\![\dot{\bigwedge}\{\sigma_x : x \in f^{-1}(y)\}]\!]_{y \in Y} & = & (\mathbf{P}h \circ \forall e_2)([\![e_1]\!]_E) & = & (\forall g \circ \mathbf{P}g')([\![e_1]\!]_E)
\end{array}$$

Now we consider the map $q : X \to G$ defined by $q(x) := (\sigma_x, f(x))$ for all $x \in X$.

Since q is surjective, it has a right inverse by (AC), hence $\exists q$ and $\forall q$ are left inverses of $\mathbf{P}q$ (by Lemma 1.5 (2)), that is: $\exists q \circ \mathbf{P}q = \forall q \circ \mathbf{P}q = \mathrm{id}_{\mathbf{P}G}$. Therefore:

$$\begin{array}{rcl}
[\![\dot{\bigvee}\{\sigma_x : x \in f^{-1}(y)\}]\!]_{y \in Y} & = & (\exists g \circ \mathbf{P}g')([\![e_1]\!]_E) \\
& = & (\exists g \circ \exists q \circ \mathbf{P}q \circ \mathbf{P}g')([\![e_1]\!]_E) \\
& = & \big(\exists(g \circ q) \circ \mathbf{P}(g' \circ q)\big)([\![e_1]\!]_E) \\
& = & \exists f\big(\mathbf{P}(g' \circ q)([\![e_1]\!]_E)\big) \\
& = & \exists f\big([\![e_1 \circ g' \circ q]\!]_X\big) \\
& = & \exists f([\![\sigma]\!]_X)
\end{array}$$

(since $g \circ q = f$ and $e_1 \circ g' \circ q = \sigma$). And similarly for $\forall$. □

To sum up...

We introduced codes $(\dot\wedge), (\dot\vee), (\dot\to) \;:\; \Sigma^2 \to \Sigma$ and $\dot\bigvee, \dot\bigwedge \;:\; \mathfrak{P}(\Sigma) \to \Sigma$ such that for all sets X and for all predicates $p, q \in \mathbf{P}X$:

- If $\sigma, \tau \in \Sigma^X$ are codes for $p, q \in \mathbf{P}X$, respectively, then:

$$(\sigma_x \mathbin{\dot\wedge} \tau_x)_{x\in X} \quad (\in \Sigma^X) \quad \text{is a code for} \quad p \wedge q \quad (\in \mathbf{P}X)$$
$$(\sigma_x \mathbin{\dot\vee} \tau_x)_{x\in X} \quad (\in \Sigma^X) \quad \text{is a code for} \quad p \vee q \quad (\in \mathbf{P}X)$$
$$(\sigma_x \mathbin{\dot\to} \tau_x)_{x\in X} \quad (\in \Sigma^X) \quad \text{is a code for} \quad p \to q \quad (\in \mathbf{P}X)$$

- If $\sigma \in \Sigma^X$ is a code for $p \in \mathbf{P}X$ and $f : X \to Y$ any map, then:

$$\Big(\dot\bigvee\{\sigma_x : x \in f^{-1}(y)\}\Big)_{y\in Y} \quad (\in \Sigma^Y) \quad \text{is a code for} \quad \exists f(p) \quad (\in \mathbf{P}Y)$$
$$\Big(\dot\bigwedge\{\sigma_x : x \in f^{-1}(y)\}\Big)_{y\in Y} \quad (\in \Sigma^Y) \quad \text{is a code for} \quad \forall f(p) \quad (\in \mathbf{P}Y)$$

Beware! As for the "connectives" $\dot\wedge$, $\dot\vee$ and $\dot\to$, the "quantifiers" $\dot\bigvee$ and $\dot\bigwedge$ enjoy no good algebraic properties:

$$\dot\bigwedge\{\xi\} \neq \xi, \qquad \dot\bigwedge\{\xi, \dot\bigwedge\{\xi', \xi''\}\} \neq \dot\bigwedge\{\dot\bigwedge\{\xi, \xi'\}, \xi''\} \neq \dot\bigwedge\{\xi, \xi', \xi''\} \qquad \text{etc.}$$

Defining the "filter" $\Phi \subseteq \Sigma$

It now remains to characterize the ordering on each $\mathbf{P}X$. For that we let

$$\Phi \ := \ \{\xi \in \Sigma : [\![\xi]\!]_{\_\in 1} = \top_1\} \qquad \text{(where } \top_1 = \max(\mathbf{P}1)\text{)}$$

Proposition 2.7 (Characterizing the order in $\mathbf{P}X$)

For all $X \in \mathbf{Set}$ and $\sigma, \tau \in \Sigma^X$, we have

$$[\![\sigma]\!]_X \leq [\![\tau]\!]_X \qquad \text{iff} \qquad \dot{\bigwedge}\{\sigma_x \dot{\to} \tau_x : x \in X\} \in \Phi$$

- This result implies that for each set X, the Heyting algebra $\mathbf{P}X$ is (isomorphic to) the poset reflection of the preordered set $(\Sigma^X, \vdash_X)$, writing $\vdash_X$ the preorder (on Σ^X) defined by

$$\sigma \vdash_X \tau \qquad \text{iff} \qquad \dot{\bigwedge}\{\sigma_x \dot{\to} \tau_x : x \in X\} \in \Phi$$

  (for all $\sigma, \tau \in \Sigma^X$)

- **Conclusion:** The tripos $\mathbf{P}$ is completely characterized by the set Σ together with the operations $\dot{\to}$, $\dot{\bigwedge}$ and the "filter" $\Phi \subseteq \Sigma$

Other properties (1/2)

Proposition 2.8 (Merging quantifications)

$$\begin{aligned} [\![\dot{\bigvee}\{\dot{\bigvee}s : s \in S\}]\!]_{S\in\mathfrak{P}(\mathfrak{P}(\Sigma))} &= [\![\dot{\bigvee}(\bigcup S)]\!]_{S\in\mathfrak{P}(\mathfrak{P}(\Sigma))} \\ [\![\dot{\bigwedge}\{\dot{\bigwedge}s : s \in S\}]\!]_{S\in\mathfrak{P}(\mathfrak{P}(\Sigma))} &= [\![\dot{\bigwedge}(\bigcup S)]\!]_{S\in\mathfrak{P}(\mathfrak{P}(\Sigma))} \end{aligned}$$

Proof. Apply the Beck-Chevalley condition to the suitable pullback!

Proposition 2.9 (Distributivity $\rightarrow/\forall$)

$$[\![\dot{\bigwedge}\{\theta \dot{\rightarrow} \xi \ : \ \xi \in s\}]\!]_{(\theta,s)\in\Sigma\times\mathfrak{P}(\Sigma)} = [\![\theta \dot{\rightarrow} \dot{\bigwedge}s]\!]_{(\theta,s)\in\Sigma\times\mathfrak{P}(\Sigma)}$$

Corollary 2.10 (Distributivity $\rightarrow/\forall$)

Given a set X and two families $\sigma \in \Sigma^X$ and $t \in \mathfrak{P}(\Sigma)^X$, we have

$$[\![\dot{\bigwedge}\{\sigma_x \dot{\rightarrow} \xi \ : \ \xi \in t_x\}]\!]_{x\in X} = [\![\sigma_x \dot{\rightarrow} \dot{\bigwedge}t_x]\!]_{x\in X}$$

Other properties (2/2)

Consider the inclusion relation

$$F := \{(s, s') \in \mathfrak{P}(\Sigma) \times \mathfrak{P}(\Sigma) \ : \ s \subseteq s'\}$$

together with its two projections $f_1, f_2 : F \to \mathfrak{P}(\Sigma)$

Proposition 2.11

$$[\![\dot{\bigvee} \circ f_1]\!]_F \ \leq \ [\![\dot{\bigvee} \circ f_2]\!]_F \qquad \text{and} \qquad [\![\dot{\bigwedge} \circ f_1]\!]_F \ \geq \ [\![\dot{\bigwedge} \circ f_2]\!]_F$$

Corollary 2.12

Given a set X and families $a, b \in \mathfrak{P}(\Sigma)^X$ such that $a_x \subseteq b_x$ for all $x \in X$:

$$[\![\dot{\bigvee} \circ a]\!]_X \ \leq \ [\![\dot{\bigvee} \circ b]\!]_X \qquad \text{and} \qquad [\![\dot{\bigwedge} \circ a]\!]_X \ \geq \ [\![\dot{\bigwedge} \circ b]\!]_X$$

Plan

1 Introduction

2 Set-based triposes

3 Anatomy of a Set-based tripos

4 Extracting the implicative algebra

How to regularize the set Σ of propositional codes?

- We have seen that the structure of the tripos $\mathbf{P} : \mathbf{Set}^{\mathsf{op}} \to \mathbf{HA}$ is fully characterized from
  - the "implication" $(\dot{\to}) \; : \; \Sigma^2 \to \Sigma$
  - the "universal quantifier" $(\dot{\bigwedge}) \; : \; \mathfrak{P}(\Sigma) \to \Sigma$
  - the "filter" $\Phi \; \subseteq \; \Sigma$

  via the equivalence

$$[\![\sigma]\!]_X \leq [\![\tau]\!]_X \qquad \text{iff} \qquad \dot{\bigwedge}\{\sigma_x \dot{\to} \tau_x : x \in X\} \in \Phi$$

- However, due to the non canonical definition of the operations $\dot{\to}$ and $\dot{\bigwedge}$ (as codes for certain predicates), the structure $(\Sigma, \dot{\to}, \dot{\bigwedge}, \Phi)$ has no good algebraic properties

- **Question:** How to make these operations more regular?
  1. Via a suitable quotient? (I tried but did not find...)
  2. By encapsulating codes into a larger well-behaved structure?
     $\Rightarrow$ embedding Σ into an implicative structure $(\mathscr{A}, \preccurlyeq, \to)$

The set of atoms $\mathscr{A}_0$

- We first define a set $\mathscr{A}_0$ of atoms from the grammar:

Atoms $\quad \alpha, \beta \quad ::= \quad \dot{\xi} \quad | \quad s \mapsto \alpha \qquad (\xi \in \Sigma,\ s \in \mathfrak{P}(\Sigma))$

Each atom is a finite list/stack of the form $s_1 \mapsto \cdots \mapsto s_n \mapsto \dot{\xi}$, where $s_1, \ldots, s_n \in \mathfrak{P}(\Sigma)$ and $\xi \in \Sigma$

- The set $\mathscr{A}_0$ is equipped with the preorder $\alpha \leq \alpha'$ that is inductively defined from the two rules

$$\frac{}{\dot{\xi} \leq \dot{\xi}} \qquad\qquad \frac{s \subseteq s' \qquad \alpha \leq \alpha'}{s \mapsto \alpha \ \leq \ s' \mapsto \alpha'}$$

- The set $\mathscr{A}_0$ of atoms is also equipped with a conversion function $\varphi_0 : \mathscr{A}_0 \to \Sigma$, defined by

$$\varphi_0(\dot{\xi}) \ := \ \xi \qquad \text{and} \qquad \varphi_0(s \mapsto \alpha) \ := \ (\dot{\bigwedge} s) \mathrel{\dot{\to}} \varphi_0(\alpha)$$

(By construction, the function $\varphi_0 : \mathscr{A}_0 \to \Sigma$ is surjective)

The complete lattice $(\mathscr{A}, \preccurlyeq)$

From the set $\mathscr{A}_0$ of atoms equipped with the preorder $\alpha \leq \alpha'$, we let:

- $\mathscr{A} \;:=\; \mathfrak{P}_{\uparrow}(\mathscr{A}_0)$ (set of upwards closed subsets of $\mathscr{A}_0$, w.r.t. $\leq$)
- $a \preccurlyeq b \;:\equiv\; a \supseteq b$ for all $a, b \in \mathscr{A}$ (reverse inclusion)

Fact: $(\mathscr{A}, \preccurlyeq)$ is a complete lattice

In this complete lattice, we have: $\curlywedge = \bigcup$, $\bot_{\mathscr{A}} = \mathscr{A}_0$, $\top_{\mathscr{A}} = \varnothing$

Intuitions:

- Each atom $\alpha \;=\; (s_1 \mapsto \cdots \mapsto s_n \mapsto \dot{\xi})$ represents the code

$$\varphi_0(\alpha) \;:=\; (\dot{\bigwedge} s_1) \dot{\to} \cdots \dot{\to} (\dot{\bigwedge} s_n) \dot{\to} \xi$$

- Each (upwards-closed) set of atoms $a \in \mathscr{A}$ represents the code

$$\varphi(a) \;:=\; \dot{\bigwedge} \{\varphi_0(\alpha) \;:\; \alpha \in a\}$$

Relationship with graph models (1/2)

Remark. The above construction is reminiscent from the construction of graph models of the λ-calculus [Engeler '81]

- In the context of graph models, the set $\mathscr{A}_0$ would be defined from the grammar

$$\alpha, \beta \in \mathscr{A}_0 \quad ::= \quad \dot{\xi} \quad | \quad \{\alpha_1, \ldots, \alpha_n\} \mapsto \beta \qquad (\xi \in \Sigma)$$

  that is, as the least solution of the set-theoretic equation

$$\mathscr{A}_0 \ = \ \Sigma + \mathfrak{P}_{\mathsf{fin}}(\mathscr{A}_0) \times \mathscr{A}_0$$

- However, the set $\mathscr{A} := \mathfrak{P}_{\uparrow}(\mathscr{A}_0)$ induced by this $\mathscr{A}_0$ would be a (D_∞-like) model of the λ-calculus, but not an implicative structure

- **Reason:** The application $(a, b \mapsto ab) : \mathscr{A}^2 \to \mathscr{A}$ that naturally comes with this definition of $\mathscr{A}_0$ has no right adjoint, due to the finiteness of the l.h.s. in the construct $\{a_1, \ldots, \alpha_n\} \mapsto \beta$. So that there is no implication in $\mathscr{A}$

Relationship with graph models (2/2)

- To fix this problem, it would be natural to relax the condition of finiteness, by considering instead the equation

$$\mathscr{A}_0 \;=\; \Sigma + \mathfrak{P}(\mathscr{A}_0) \times \mathscr{A}_0$$

Alas, this equation has no solution! (for obvious cardinality reasons)

- **Trick:** Replace arbitrary subsets $a \subseteq \mathscr{A}_0$ (in the l.h.s. of $a \mapsto \beta$) by arbitrary subsets of Σ, using the fact that subsets of $\mathscr{A}_0$ can be converted (element-wise) into subsets of Σ, via $\varphi_0 : \mathscr{A}_0 \to \Sigma$

- So that in the end, we obtain the set-theoretic equation

$$\mathscr{A}_0 \;=\; \Sigma + \mathfrak{P}(\Sigma) \times \mathscr{A}_0,$$

whose least solution is generated from the grammar

$$\alpha, \beta \in \mathscr{A}_0 \quad ::= \quad \dot{\xi} \quad | \quad s \mapsto \alpha \qquad\qquad (\xi \in \Sigma,\ s \in \mathfrak{P}(\Sigma))$$

Defining the implication in $(\mathscr{A}, \preccurlyeq)$

- Recall that $(\mathscr{A}, \preccurlyeq) \;=\; (\mathfrak{P}_{\uparrow}(\mathscr{A}_0),\ \supseteq)$, where:

$$\alpha, \beta \in \mathscr{A}_0 \quad ::= \quad \dot{\xi} \quad | \quad s \mapsto \alpha \qquad\qquad (\xi \in \Sigma,\ s \in \mathfrak{P}(\Sigma))$$

$$\frac{}{\dot{\xi} \leq \dot{\xi}} \qquad\qquad \frac{s \subseteq s' \qquad \alpha \leq \alpha'}{s \mapsto \alpha \;\leq\; s' \mapsto \alpha'}$$

- The conversion function $\varphi_0 \;:\; \mathscr{A}_0 \to \Sigma$ naturally extends to a function $\tilde{\varphi}_0 \;:\; \mathscr{A} \to \mathfrak{P}(\Sigma)$ (element-wise)

- Given $a, b \in \mathscr{A}$ $(= \mathfrak{P}_{\uparrow}(\mathscr{A}_0))$, we let

$$a \to b \;:=\; \{s \mapsto \beta \;:\; s \in \tilde{\varphi}_0(a)^{\subseteq},\ \beta \in b\} \qquad\qquad (\in \mathscr{A})$$

writing $\tilde{\varphi}_0(a)^{\subseteq} \;:=\; \{s \in \mathfrak{P}(\Sigma) : \tilde{\varphi}_0(a) \subseteq s\}$

Proposition 3.1

The triple $(\mathscr{A}, \preccurlyeq, \to)$ is an implicative structure

Viewing $\mathscr{A}$ as a new set of propositions

Let us now define:

- $\varphi : \mathscr{A} \to \Sigma$ by $\varphi(a) := \dot{\bigwedge} \tilde{\varphi}_0(a)$ (for all $a \in \mathscr{A}$)
- $\psi : \Sigma \to \mathscr{A}$ by $\psi(\xi) := \{\dot{\xi}\}$ (for all $\xi \in \Sigma$)
- $\mathit{tr}_{\mathscr{A}} \in \mathbf{P}\mathscr{A}$ by $\mathit{tr}_{\mathscr{A}} := \mathbf{P}\varphi(\mathit{tr}_\Sigma)$

Lemma 3.2

1. $\varphi(\psi(\xi)) = \dot{\bigwedge}\{\xi\}$ for all $\xi \in \Sigma$
2. Therefore: $\mathit{tr}_\Sigma = \mathbf{P}\psi(\mathit{tr}_{\mathscr{A}})$

Proposition 3.3

For each set X, the function

$$\begin{array}{rccl} \langle\!\langle \_ \rangle\!\rangle_X : & \mathscr{A}^X & \to & \mathbf{P}X \\ & a & \mapsto & \mathbf{P}a(\mathit{tr}_{\mathscr{A}}) \end{array}$$ is surjective

Which means that $\mathit{tr}_{\mathscr{A}} \in \mathbf{P}\mathscr{A}$ is a generic predicate of the tripos $\mathbf{P}$

Relating both generic predicates

From now on, we have:

- Two sets of (codes of) propositions: Σ and $\mathscr{A}$
- Two generic predicates $tr_\Sigma \in \mathbf{P}\Sigma$ and $tr_{\mathscr{A}} \in \mathbf{P}\mathscr{A}$
- For each set X, two (surjective) decoding functions

$$\begin{array}{rcrcl} [\![\text{-}]\!]_X & : & \Sigma^X & \to & \mathbf{P}X \\ & & \sigma & \mapsto & \mathbf{P}\sigma(tr_\Sigma) \end{array} \qquad \begin{array}{rcrcl} \langle\!\langle\text{-}\rangle\!\rangle_X & : & \mathscr{A}^X & \to & \mathbf{P}X \\ & & a & \mapsto & \mathbf{P}a(tr_{\mathscr{A}}) \end{array}$$

Proposition 3.4

For each set X, the following two diagrams commute:

$$\begin{array}{ccc} \Sigma^X & \xrightarrow{[\![\text{-}]\!]_X} & \mathbf{P}X \\ {\scriptstyle \varphi^X}\uparrow & & \Vert \\ \mathscr{A}^X & \xrightarrow[\langle\!\langle\text{-}\rangle\!\rangle_X]{} & \mathbf{P}X \end{array} \qquad \begin{array}{ccc} \Sigma^X & \xrightarrow{[\![\text{-}]\!]_X} & \mathbf{P}X \\ {\scriptstyle \psi^X}\downarrow & & \Vert \\ \mathscr{A}^X & \xrightarrow[\langle\!\langle\text{-}\rangle\!\rangle_X]{} & \mathbf{P}X \end{array}$$

To sum up...

- We started from Σ, equipped with non canonical operations

$$(\dot{\to}) \;:\; \Sigma^2 \to \Sigma \qquad \text{and} \qquad (\dot{\bigwedge}) \;:\; \mathfrak{P}(\Sigma) \to \Sigma$$

  acting as codes for implication and universal quantification, in the sense of the initial generic predicate $tr_\Sigma \in \mathbf{P}\Sigma$.

- From Σ, $\dot{\to}$ and $\dot{\bigwedge}$, we defined an implicative structure $(\mathscr{A}, \preccurlyeq, \to)$ whose implication encapsulates both operations $\dot{\to}$ and $\dot{\bigwedge}$:

$$a \to b \;:=\; \{s \mapsto \beta \;:\; s \in \tilde{\varphi}_0(a)^{\subseteq},\ \beta \in b\} \qquad (\in \mathscr{A})$$

(Recall that $\varphi_0 : \mathscr{A} \to \Sigma$ is defined from $\dot{\to}$ and $\dot{\bigwedge}$)

- We now want to show that the canonical operations

$$(\to) \;:\; \mathscr{A}^2 \to \mathscr{A} \qquad \text{and} \qquad (\curlywedge) \;:\; \mathfrak{P}(\mathscr{A}) \to \mathscr{A}$$

  still act as codes for implication and universal quantification, but now in the sense of the new generic predicate $tr_{\mathscr{A}} \in \mathbf{P}\mathscr{A}$

Universal quantification in $\mathscr{A}$ (1/2)

- Recall that we chose $(\dot{\bigwedge}) \in \Sigma^{\mathfrak{P}(\Sigma)}$ such that $[\![\dot{\bigwedge}]\!]_{\mathfrak{P}(\Sigma)} = \forall e_2([\![e_1]\!]_E)$, where $e_1 : E \to \Sigma$ and $e_2 : E \to \mathfrak{P}(\Sigma)$ are the projections associated with the membership relation $E := \{(\xi, s) \in \Sigma \times \mathfrak{P}(\Sigma) \ : \ \xi \in s\}$
- Let us now consider the membership relation
$$E' \ := \ \{(a, A) \in \mathscr{A} \times \mathfrak{P}(\mathscr{A}) \ : \ a \in A\}$$
together with its projections $e'_1 : E' \to \mathscr{A}$ and $e'_2 : E' \to \mathfrak{P}(\mathscr{A})$

Proposition 3.5

We have: $\langle\!\langle \curlywedge A \rangle\!\rangle_{A \in \mathfrak{P}(\mathscr{A})} \ = \ \forall e'_2(\langle\!\langle e'_1 \rangle\!\rangle_{E'})$

Corollary 3.6

Given a code $a = (a_x)_{x \in X} \in \mathscr{A}^X$ and a map $f : X \to Y$, we have:

$$\langle\!\langle \curlywedge \{a_x : x \in f^{-1}(y)\} \rangle\!\rangle_{y \in Y} \ = \ \forall f(\langle\!\langle a \rangle\!\rangle_X) \qquad (\in \mathbf{P}Y)$$

Universal quantification in $\mathscr{A}$ (2/2)

Proof of Proposition 3.5.

$$
\begin{aligned}
\langle\!\langle \curlywedge A \rangle\!\rangle_{A\in\mathfrak{P}(\mathscr{A})} &= \langle\!\langle \bigcup A \rangle\!\rangle_{A\in\mathfrak{P}(\mathscr{A})} && (\text{since } \curlywedge = \bigcup)\\
&= [\![\varphi(\bigcup A)]\!]_{A\in\mathfrak{P}(\mathscr{A})} && (\text{Prop. 3.4})\\
&= [\![\dot{\bigwedge}\tilde{\varphi}_0(\bigcup A)]\!]_{A\in\mathfrak{P}(\mathscr{A})} && (\text{Def. of } \varphi)\\
&= [\![\dot{\bigwedge}\bigcup\mathfrak{P}\tilde{\varphi}_0(A)]\!]_{A\in\mathfrak{P}(\mathscr{A})} && (\text{Def. of } \tilde{\varphi}_0)\\
&= \mathbf{P}(\mathfrak{P}\tilde{\varphi}_0)([\![\dot{\bigwedge}\bigcup S]\!]_{S\in\mathfrak{P}(\mathfrak{P}(\Sigma))}) && (\text{Naturality of } [\![\_]\!])\\
&= \mathbf{P}(\mathfrak{P}\tilde{\varphi}_0)([\![\dot{\bigwedge}\{\dot{\bigwedge}s : s\in S\}]\!]_{S\in\mathfrak{P}(\mathfrak{P}(\Sigma))}) && (\textbf{Prop. 2.8})\\
&= [\![\dot{\bigwedge}\{\dot{\bigwedge}\tilde{\varphi}_0(a) : a\in A\}]\!]_{A\in\mathfrak{P}(\mathscr{A})} && (\text{Naturality of } [\![\_]\!])\\
&= [\![\dot{\bigwedge}\{\varphi(a) : a\in A\}]\!]_{A\in\mathfrak{P}(\mathscr{A})} && (\text{Def. of } \varphi)\\
&= [\![\dot{\bigwedge}\{\varphi(e'_1(p)) : p\in {e'}_2^{-1}(A)\}]\!]_{A\in\mathfrak{P}(\mathscr{A})} && (\text{Def. of } e'_1, e_2)\\
&= \forall e'_2([\![\varphi\circ e'_1]\!]_{E'}) && (\textbf{Prop. 2.6})\\
&= \forall e'_2(\langle\!\langle e'_1\rangle\!\rangle_{E'}) && (\text{Prop. 3.4})
\end{aligned}
$$

□

Implication in $\mathscr{A}$ (1/2)

- Recall that we chose $(\dot{\to}) \in \Sigma^{\Sigma\times\Sigma}$ such that $[\![\dot{\to}]\!]_{\Sigma^2} = [\![\pi]\!]_{\Sigma^2} \to [\![\pi']\!]_{\Sigma^2}$, where $\pi, \pi' : \Sigma^2 \to \Sigma$ are the two projections from Σ^2 to Σ

Proposition 3.6

We have: $\langle\!\langle a \to b \rangle\!\rangle_{(a,b)\in\mathscr{A}^2} \;=\; \langle\!\langle \pi \rangle\!\rangle_{\mathscr{A}^2} \to \langle\!\langle \pi' \rangle\!\rangle_{\mathscr{A}^2}$,

writing $\pi, \pi' : \mathscr{A}^2 \to \mathscr{A}$ the two projections from $\mathscr{A}^2$ to $\mathscr{A}$

Proof of Prop. 3.6 relies on the following technical lemma:

Lemma 3.7

$$\left[\!\!\left[\dot{\bigwedge}\left\{(\dot{\bigwedge} s')\dot{\to}\xi \;:\; s' \in s^{\subseteq},\ \xi \in t\right\}\right]\!\!\right]_{(s,t)\in\mathfrak{P}(\Sigma)^2} = \left[\!\!\left[\dot{\bigwedge}\left\{(\dot{\bigwedge} s)\dot{\to}\xi \;:\; \xi \in t\right\}\right]\!\!\right]_{(s,t)\in\mathfrak{P}(\Sigma)^2}$$

Corollary 3.8

Let X be a set. For all codes $a, b \in \mathscr{A}^X$, we have

$$\langle\!\langle a_x \to b_x \rangle\!\rangle_{x\in X} \;=\; \langle\!\langle a \rangle\!\rangle_X \to \langle\!\langle b \rangle\!\rangle_X \qquad (\in \mathbf{P}X)$$

Implication in $\mathscr{A}$ (2/2)

Proof of Proposition 3.6.

$$
\begin{aligned}
& \langle\!\langle a \to b \rangle\!\rangle_{(a,b)\in\mathscr{A}^2} \\
= {} & [\![\varphi(a \to b)]\!]_{(a,b)\in\mathscr{A}^2} && \text{(Prop. 3.4)} \\
= {} & [\![\varphi(\{s' \mapsto \beta \;:\; s' \in \tilde{\varphi}_0(a)^{\subseteq},\ \beta \in b\})]\!]_{(a,b)\in\mathscr{A}^2} && \text{(Def. of } a \to b\text{)} \\
= {} & [\![\dot{\bigwedge}\{(\dot{\bigwedge} s') \dot{\to} \xi \;:\; s' \in \tilde{\varphi}_0(a)^{\subseteq},\ \xi \in \tilde{\varphi}_0(b)\}]\!]_{(a,b)\in\mathscr{A}^2} && \text{(Def. of } \varphi\text{)} \\
= {} & \mathbf{P}(\tilde{\varphi}_0 \times \tilde{\varphi}_0)\Big([\![\dot{\bigwedge}\{(\dot{\bigwedge} s') \dot{\to} \xi \;:\; s' \in s^{\subseteq},\ \xi \in t\}]\!]_{(s,t)\in\mathfrak{P}(\Sigma)^2}\Big) && \text{(Nat. of } [\![\_]\!]\text{)} \\
= {} & \mathbf{P}(\tilde{\varphi}_0 \times \tilde{\varphi}_0)\Big([\![\dot{\bigwedge}\{(\dot{\bigwedge} s) \dot{\to} \xi \;:\; \xi \in t\}]\!]_{(s,t)\in\mathfrak{P}(\Sigma)^2}\Big) && \text{(Lemma 3.7)} \\
= {} & \mathbf{P}(\tilde{\varphi}_0 \times \tilde{\varphi}_0)\Big([\![(\dot{\bigwedge} s) \dot{\to} (\dot{\bigwedge} t)]\!]_{(s,t)\in\mathfrak{P}(\Sigma)^2}\Big) && \text{(Prop. 2.9)} \\
= {} & [\![(\dot{\bigwedge}\tilde{\varphi}_0(a)) \dot{\to} (\dot{\bigwedge}\tilde{\varphi}_0(b))]\!]_{(a,b)\in\mathscr{A}^2} && \text{(Nat. of } [\![\_]\!]\text{)} \\
= {} & [\![\varphi(a) \dot{\to} \varphi(b)]\!]_{(a,b)\in\mathscr{A}^2} && \text{(Def. of } \varphi\text{)} \\
= {} & [\![\varphi(a)]\!]_{(a,b)\in\mathscr{A}^2} \to [\![\varphi(b)]\!]_{(a,b)\in\mathscr{A}^2} && \text{(Prop. 2.4)} \\
= {} & [\![\varphi \circ \pi]\!]_{\mathscr{A}^2} \to [\![\varphi \circ \pi']\!]_{\mathscr{A}^2} && \text{(Def. of } \pi, \pi'\text{)} \\
= {} & \langle\!\langle \pi \rangle\!\rangle_{\mathscr{A}^2} \to \langle\!\langle \pi' \rangle\!\rangle_{\mathscr{A}^2} && \text{(Prop. 3.4)}
\end{aligned}
$$

□

Defining the separator $S \subseteq \mathscr{A}$

By analogy with the definition of the "filter" $\Phi \subseteq \Sigma$, we let

$$S \ := \ \{a \in \mathscr{A} : \langle\!\langle a \rangle\!\rangle_{\_\in 1} = \top_1\} \qquad \text{(where } \top_1 = \max(\mathbf{P}1)\text{)}$$

- By construction, we have

$$\begin{aligned} S \ &= \ \{a \in \mathscr{A} \ : \ [\![\varphi(a)]\!]_{\_\in 1} = \top_1\} \\ &= \ \{a \in \mathscr{A} \ : \ \varphi(a) \in \Phi\} \ = \ \varphi^{-1}(\Phi) \end{aligned}$$

Proposition 3.9 (S is a separator)

The subset $S \subseteq \mathscr{A}$ is a separator of $(\mathscr{A}, \preccurlyeq, \to)$

Proposition 3.10 (Characterizing the ordering in $\mathbf{P}X$)

For all sets X and for all codes $a, b \in \mathscr{A}^X$, we have:

$$\langle\!\langle a \rangle\!\rangle_X \leq \langle\!\langle b \rangle\!\rangle_X \qquad \text{iff} \qquad \bigwedge_{x \in X} (a_x \to b_x) \ \in \ S$$

Constructing the isomorphism

Let us now write $\mathbf{P}' : \mathbf{Set}^{\mathrm{op}} \to \mathbf{HA}$ the implicative tripos induced by the implicative algebra $(\mathscr{A}, \preccurlyeq, \to, S)$

Proposition (Conclusion of the Main Theorem)

The implicative tripos $\mathbf{P}'$ is isomorphic to the initial tripos $\mathbf{P}$

Proof.

Let us consider the family of maps $\rho_X := \langle\!\langle \_ \rangle\!\rangle_X : \mathscr{A}^X \to \mathbf{P}X$ (which is natural in X). From Prop. 3.10, we observe that for all $a, b \in \mathscr{A}^X$, we have

$$a \vdash_{S[X]} b \qquad \text{iff} \qquad \bigcurlywedge_{x \in X} (a_x \to b_x) \in S \qquad \text{iff} \qquad \rho_X(a) \leq \rho_X(b)\,.$$

Hence $\rho_X : \mathscr{A}^X \to \mathbf{P}X$ induces a natural embedding of posets $\hat{\rho}_X : \mathbf{P}'X \to \mathbf{P}X$ through the quotient $\mathbf{P}'X := \mathscr{A}^X / S[X]$. Moreover, the embedding $\hat{\rho}_X$ is surjective (since $\rho_X = \langle\!\langle \_ \rangle\!\rangle_X$ is), therefore it is an isomorphism. □

The case of classical triposes

- Recall that:

Theorem (Classical implicative triposes)

Each tripos induced by a classical implicative algebra $(\mathscr{A}, \preccurlyeq, \to, S)$ is isomorphic to a tripos induced by an abstract Krivine structure

- We also easily see that an implicative tripos is classical (as a tripos) iff the underlying implicative algebra is. Therefore:

Theorem

Every ($\mathbf{Set}$-based) tripos is isomorphic to a Krivine tripos

- In conclusion:
  - Classical triposes
  - Classical implicative triposes
  - Krivine/Streicher triposes

  are one and the same thing