Ordering Protoalgebraic Logics

Pedro Filipe

Instituto Superior Técnico

LisMath Seminar, 2018

Table of Contents

Table of Contents

Introduction - Why Logic?

Introduction - Why Logic?

- As a way of axiomatizing mathematics

Introduction - Why Logic?

- As a way of axiomatizing mathematics
- As a way of reasoning about systems

Introduction - Why Logic?

- As a way of axiomatizing mathematics
- As a way of reasoning about systems

Wait... logic's'?

A little bit of universal algebra

A little bit of universal algebra

Definition (Similarity type)

A similarity type (or algebraic language) $\mathbf{L}$ *is a pair* $\langle L, \mathrm{ar} \rangle$*, where* L *is a set of symbols and* $\mathrm{ar} : L \to \mathbb{N}$ *is a function.*

A little bit of universal algebra

Definition (Similarity type)

A similarity type (or algebraic language) $\mathbf{L}$ *is a pair* $\langle L, \mathrm{ar}\rangle$, *where* L *is a set of symbols and* $\mathrm{ar} : L \to \mathbb{N}$ *is a function.*

Definition ($\mathcal{L}$-algebra)

Let $\mathbf{L}$ *be a similarity type, a* $\mathbf{L}$*-algebra* $\mathcal{A}$ *is a pair* $\langle A, (-)^{\mathcal{A}}\rangle$, *where* A *is a set and, for each* $\lambda \in L$, *if* $\mathrm{ar}(\lambda) = n$, *then* $\lambda^{\mathcal{A}} : A^n \to A$ *is a function.*

A little bit of universal algebra

Definition (Similarity type)

A similarity type (or algebraic language) $\mathbf{L}$ *is a pair* $\langle L, \mathrm{ar}\rangle$, *where* L *is a set of symbols and* $\mathrm{ar} : L \to \mathbb{N}$ *is a function.*

Definition ($\mathcal{L}$-algebra)

Let $\mathbf{L}$ *be a similarity type, a* $\mathbf{L}$*-algebra* $\mathcal{A}$ *is a pair* $\langle A, (-)^{\mathcal{A}}\rangle$, *where* A *is a set and, for each* $\lambda \in L$, *if* $\mathrm{ar}(\lambda) = n$, *then* $\lambda^{\mathcal{A}} : A^n \to A$ *is a function.*

Definition ($\mathcal{L}$-homomorphism)

Let $\mathcal{A}$ *and* $\mathcal{B}$ *be two* $\mathbf{L}$*- algebras, then a function* $h : A \to B$ *is said to be a* $\mathbf{L}$*-homomorphism if, for every* $\lambda \in L$, $\mathrm{ar}(\lambda) = n$ *and* $a_1, \dots, a_n \in A$

$$h(\lambda^{\mathcal{A}}(a_1, \dots, a_n) = \lambda^{\mathcal{B}}(h(a_1), \dots, h(a_n))$$

In this case we simply write $h : \mathcal{A} \to \mathcal{B}$.

The algebra of formulas

The algebra of formulas

Definition (Algebra of formulas over L)

Let $\mathbf{L}$ *be a similarity type and* V *a coutable set such that* $V \cap L = \varnothing$*, then* $\mathcal{F}m_{\mathbf{L}}$ *,the algebra of formulas over* $\mathbf{L}$*, is the free* $\mathbf{L}$*-algebra generated by* V*.*

The algebra of formulas

Definition (Algebra of formulas over L)

Let $\mathbf{L}$ *be a similarity type and* V *a coutable set such that* $V \cap L = \varnothing$, *then* $\mathcal{F}m_{\mathbf{L}}$ *,the algebra of formulas over* $\mathbf{L}$, *is the free* $\mathbf{L}$*-algebra generated by* V.

- $x \in \mathcal{F}m_{\mathbf{L}}$, for all $x \in V$

The algebra of formulas

Definition (Algebra of formulas over L)

Let **L** *be a similarity type and* V *a coutable set such that* $V \cap L = \varnothing$, *then* $\mathcal{F}m_{\mathbf{L}}$ *,the algebra of formulas over* **L**, *is the free* **L***-algebra generated by* V.

- $x \in \mathcal{F}m_{\mathbf{L}}$, for all $x \in V$
- if $\lambda \in L$, $\mathrm{ar}(\lambda) = n$ and $\varphi_1, \ldots, \varphi_n \in \mathcal{F}m_{\mathbf{L}}$, then $\lambda(\varphi_1, \ldots, \varphi_n) \in \mathcal{F}m_{\mathbf{L}}$

The algebra of formulas

Definition (Algebra of formulas over $\mathbf{L}$)

Let $\mathbf{L}$ be a similarity type and V a coutable set such that $V \cap L = \varnothing$, then $\mathcal{F}\boldsymbol{m}_{\mathbf{L}}$,the algebra of formulas over $\mathbf{L}$, is the free $\mathbf{L}$-algebra generated by V.

- $x \in \mathcal{F}\boldsymbol{m}_{\mathbf{L}}$, for all $x \in V$
- if $\lambda \in L$, $\mathrm{ar}(\lambda) = n$ and $\varphi_1, \ldots, \varphi_n \in \mathcal{F}\boldsymbol{m}_{\mathbf{L}}$, then $\lambda(\varphi_1, \ldots, \varphi_n) \in \mathcal{F}\boldsymbol{m}_{\mathbf{L}}$

Definition (Substitution)

A substitution is an endomorphism $\sigma : \mathcal{F}\boldsymbol{m}_{\mathbf{L}} \to \mathcal{F}\boldsymbol{m}_{\mathbf{L}}$, or equivalently, a function $\sigma : V \to \mathcal{F}\boldsymbol{m}_{\mathbf{L}}$.

Logic over a language

Logic over a language

Definition (Logic over a language)

Fix a similarity type $\mathbf{L}$*. A logic over* $\mathbf{L}$ *is a pair* $\mathcal{L} = \langle \mathbf{L}, \vdash_{\mathcal{L}} \rangle$*, where* $\vdash_{\mathcal{L}} \subseteq \mathcal{P}(\mathcal{F}\boldsymbol{m}) \times \mathcal{F}\boldsymbol{m}$ *is such that*

(E) *if* $\varphi \in \Gamma$*, then* $\Gamma \vdash_{\mathcal{L}} \varphi$

(M) *if* $\Gamma \vdash_{\mathcal{L}} \varphi$ *and* $\Gamma \subseteq \Delta$*, then* $\Delta \vdash_{\mathcal{L}} \varphi$

(I) *if* $\Gamma \vdash_{\mathcal{L}} \varphi$ *and* $\Delta \vdash_{\mathcal{L}} \psi$ *for every* $\psi \in \Gamma$*, then* $\Delta \vdash_{\mathcal{L}} \varphi$

(S) *if* $\Gamma \vdash_{\mathcal{L}} \varphi$*, then* $\sigma\Gamma \vdash_{\mathcal{L}} \sigma\varphi$*, for every substitution* σ

Logic over a language

Definition (Logic over a language)

Fix a similarity type $\mathbf{L}$. *A logic over* $\mathbf{L}$ *is a pair* $\mathcal{L} = \langle \mathbf{L}, \vdash_{\mathcal{L}} \rangle$, *where* $\vdash_{\mathcal{L}} \subseteq \mathscr{P}(\mathcal{F}\boldsymbol{m}) \times \mathcal{F}\boldsymbol{m}$ *is such that*

(E) *if* $\varphi \in \Gamma$, *then* $\Gamma \vdash_{\mathcal{L}} \varphi$

(M) *if* $\Gamma \vdash_{\mathcal{L}} \varphi$ *and* $\Gamma \subseteq \Delta$, *then* $\Delta \vdash_{\mathcal{L}} \varphi$

(I) *if* $\Gamma \vdash_{\mathcal{L}} \varphi$ *and* $\Delta \vdash_{\mathcal{L}} \psi$ *for every* $\psi \in \Gamma$, *then* $\Delta \vdash_{\mathcal{L}} \varphi$

(S) *if* $\Gamma \vdash_{\mathcal{L}} \varphi$, *then* $\sigma\Gamma \vdash_{\mathcal{L}} \sigma\varphi$, *for every substitution* σ

$$C_{\mathcal{L}}\Gamma = \{\varphi : \Gamma \vdash_{\mathcal{L}} \varphi\}$$

Logic over a language

Definition (Logic over a language)

Fix a similarity type $\mathbf{L}$. *A logic over* $\mathbf{L}$ *is a pair* $\mathcal{L} = \langle \mathbf{L}, \vdash_{\mathcal{L}} \rangle$, *where* $\vdash_{\mathcal{L}} \subseteq \mathcal{P}(\mathcal{F}\boldsymbol{m}) \times \mathcal{F}\boldsymbol{m}$ *is such that*

- *(E)* $\Gamma \subseteq C_{\mathcal{L}}\Gamma$
- *(M)* *if* $\Gamma \subseteq \Delta$, *then* $C_{\mathcal{L}}\Gamma \subseteq C_{\mathcal{L}}\Delta$
- *(I)* $C_{\mathcal{L}}C_{\mathcal{L}}\Gamma = C_{\mathcal{L}}\Gamma$
- *(S)* $\sigma C_{\mathcal{L}}\Gamma \subseteq C_{\mathcal{L}}\sigma\Gamma$

$$C_{\mathcal{L}}\Gamma = \{\varphi : \Gamma \vdash_{\mathcal{L}} \varphi\}$$

Logic over a language

Definition (Logic over a language)

Fix a similarity type $\mathbf{L}$. *A logic over* $\mathbf{L}$ *is a pair* $\mathcal{L} = \langle \mathbf{L}, \vdash_{\mathcal{L}} \rangle$, *where* $\vdash_{\mathcal{L}} \subseteq \mathcal{P}(\mathcal{F}m) \times \mathcal{F}m$ *is such that*

(E) $\Gamma \subseteq C_{\mathcal{L}}\Gamma$

(M) *if* $\Gamma \subseteq \Delta$, *then* $C_{\mathcal{L}}\Gamma \subseteq C_{\mathcal{L}}\Delta$

(I) $C_{\mathcal{L}}C_{\mathcal{L}}\Gamma = C_{\mathcal{L}}\Gamma$

(S) $\sigma C_{\mathcal{L}}\Gamma \subseteq C_{\mathcal{L}}\sigma\Gamma$

Definition (Closure operator)

Let A *be a set, a closure operator on* A *is a function* $C : \mathcal{P}(A) \to \mathcal{P}(A)$ *such that, for every* $X, Y \subseteq A$

(E) $X \subseteq CX$

(M) *if* $X \subseteq Y$, *then* $CX \subseteq CY$

(I) $CCX = CX$

Theorems

Theorems

Theorem

Let $\mathcal{L}$ be a logic, then $C_{\mathcal{L}} : \mathscr{P}(\mathcal{F}\boldsymbol{m}) \to \mathscr{P}(\mathcal{F}\boldsymbol{m})$ defined as

$$C_{\mathcal{L}}\Gamma = \{\varphi \in \mathcal{F}\boldsymbol{m} : \Gamma \vdash_{\mathcal{L}} \varphi\}$$

is a structural closure operator.

Theorems

Theorem

Let $\mathcal{L}$ be a logic, then $C_{\mathcal{L}} : \mathcal{P}(\mathcal{F}m) \to \mathcal{P}(\mathcal{F}m)$ defined as

$$C_{\mathcal{L}}\Gamma = \{\varphi \in \mathcal{F}m : \Gamma \vdash_{\mathcal{L}} \varphi\}$$

is a structural closure operator.

Theorem

Let C be a closure operator on $\mathcal{F}m$, then $\vdash_C \subseteq \mathcal{P}(\mathcal{F}m) \times \mathcal{F}m$ defined as

$$\Gamma \vdash_C \varphi \qquad \text{iff} \qquad \varphi \in C\Gamma$$

is a closure relation. Furthermore, if C is structural then $\mathcal{L}_C = \langle \mathbf{L}, \vdash_C \rangle$ is a logic.

Closure systems

Closure systems

Definition (Closure system)

A closure system on a set A is a collection of subsets $\mathcal{C} \subseteq \mathcal{P}(A)$ that satisfies the following conditions:

- $A \in \mathcal{C}$
- *if $\mathcal{B} \subseteq \mathcal{C}$ is non-empty, then $\cap\mathcal{B} \in \mathcal{C}$*

Closure systems

Definition (Closure system)

A closure system on a set A is a collection of subsets $\mathcal{C} \subseteq \mathcal{P}(A)$ that satisfies the following conditions:

- $A \in \mathcal{C}$
- *if* $\mathcal{B} \subseteq \mathcal{C}$ *is non-empty, then* $\bigcap \mathcal{B} \in \mathcal{C}$

Theorem

Let C be a closure operator on A, then the collection

$$\mathcal{C} = \{X \subseteq A : CX = X\}$$

is a closure system on A.

Closure systems

Theorem

Let $\mathcal{C}$ be a closure system on A, then $C : \mathscr{P}(A) \to \mathscr{P}(A)$ defined as

$$CX = \cap\{F \in \mathcal{C} : X \subseteq F\}$$

is a closure operator on A whose closed sets are exactly the members of $\mathcal{C}$.

Triviality

Triviality

Definition (Trivial logic)

A logic $\mathcal{L}$ is said to be trivial if $x \vdash_{\mathcal{L}} y$ for some variables $x, y \in V$ (equivalently, when $\varphi \vdash_{\mathcal{L}} \psi$ for every $\varphi, \psi \in \mathcal{F}m$).

Triviality

Definition (Trivial logic)

A logic $\mathcal{L}$ is said to be trivial if $x \vdash_{\mathcal{L}} y$ for some variables $x, y \in V$ (equivalently, when $\varphi \vdash_{\mathcal{L}} \psi$ for every $\varphi, \psi \in \mathcal{F}m$).

There are exactly two trivial logics:

Triviality

Definition (Trivial logic)

A logic $\mathcal{L}$ is said to be trivial if $x \vdash_{\mathcal{L}} y$ for some variables $x, y \in V$ (equivalently, when $\varphi \vdash_{\mathcal{L}} \psi$ for every $\varphi, \psi \in \mathcal{F}m$).

There are exactly two trivial logics:

inconsistent logic $C_{\mathcal{L}}\Gamma = \mathcal{F}m$, for every $\Gamma \subseteq \mathcal{F}m$

Triviality

Definition (Trivial logic)

A logic $\mathcal{L}$ is said to be trivial if $x \vdash_{\mathcal{L}} y$ for some variables $x, y \in V$ (equivalently, when $\varphi \vdash_{\mathcal{L}} \psi$ for every $\varphi, \psi \in \mathcal{F}m$).

There are exactly two trivial logics:

inconsistent logic $C_{\mathcal{L}}\Gamma = \mathcal{F}m$, for every $\Gamma \subseteq \mathcal{F}m$

almost-inconsistent logic $C_{\mathcal{L}}\Gamma = \mathcal{F}m$, for every non-empty $\Gamma \subseteq \mathcal{F}m$ and $C_{\mathcal{L}}\varnothing = \varnothing$

Ordering **Log**

Definition

A logic $\mathcal{L}$ is weaker than a logic $\mathcal{L}'$, and we write $\mathcal{L} \leq \mathcal{L}'$, when $\vdash_{\mathcal{L}} \subseteq \vdash_{\mathcal{L}'}$. In this case, $\mathcal{L}'$ is said to be an extension of $\mathcal{L}$.

Theorem (The lattice of logics)

The set **Log** *of all logics, ordered under the relation $\leq$, is a complete lattice:*

Meet if $\mathcal{L} = \wedge_{i \in I} \mathcal{L}_i$ then $\vdash_{\mathcal{L}} = \cap_{i \in I} \vdash_{\mathcal{L}_i}$

Join if $\mathcal{L} = \vee_{i \in I} \mathcal{L}_i$ then $Th\mathcal{L} = \cap_{i \in I} Th\mathcal{L}_i$

Also, it has a maximum (the inconsistent logic) and a minimum (the identity logic).

Table of Contents

Protoalgebraic logics

Protoalgebraic logics

Definition (Protoalgebraic logic)

A logic $\mathcal{L}$ is protoalgebraic if there is a set $\Delta(x, y)$ of formulas, such that

$$(I_\Delta) \qquad \vdash_{\mathcal{L}} \Delta(x, x)$$

$$(MP_\Delta) \qquad x, \Delta(x, y) \vdash_{\mathcal{L}} y$$

In this case, we say that the set Δ witnesses the protoalgebraicity of $\mathcal{L}$.
The set of all protoalgebraic logics is denoted by **Prot**.

Protoalgebraic logics

Definition (Protoalgebraic logic)

A logic $\mathcal{L}$ is protoalgebraic if there is a set $\Delta(x,y)$ of formulas, such that

$$(I_\Delta) \qquad \vdash_{\mathcal{L}} \Delta(x,x)$$
$$(MP_\Delta) \qquad x, \Delta(x,y) \vdash_{\mathcal{L}} y$$

In this case, we say that the set Δ witnesses the protoalgebraicity of $\mathcal{L}$. The set of all protoalgebraic logics is denoted by **Prot**.

Theorem

If $\Delta = \varnothing$ or $\Delta = \Delta(x,x)$ for some variable x, then any logic witnessed by it is trivial. Furthermore, the only protoalgebraic logic without theorems is the almost-inconsistent logic.

Δ-logics

Δ-logics

Definition (Δ-logics)

Let $\Delta(x, y) \subseteq \mathcal{F}m(2)$ be a non-empty set of formulas then the logic $\mathcal{L}_\Delta$ is the logic defined by the following axiomatic system:

axioms $\Delta(x, x)$

rules $x, \Delta(x, y) \vdash_\Delta y$

Coherent sets

Coherent sets

Definition (Coherent sets)

A non-empty set $\Delta(x, y) \subseteq \mathcal{F}m(2)$ *is coherent when* $\delta(x, x) = \delta'(x, x)$ *for all* $\delta, \delta' \in \Delta(x, y)$*. Furthermore, if* $\Delta(x, y), \Delta'(x, y) \subseteq \mathcal{F}m(2)$ *are non-empty, then* Δ *is said to be coherent with* Δ' *when* $\Delta \cup \Delta'$ *is coherent.*

Coherent sets

Definition (Coherent sets)

A non-empty set $\Delta(x, y) \subseteq \mathcal{F}m(2)$ is coherent when $\delta(x, x) = \delta'(x, x)$ for all $\delta, \delta' \in \Delta(x, y)$. Furthermore, if $\Delta(x, y), \Delta'(x, y) \subseteq \mathcal{F}m(2)$ are non-empty, then Δ is said to be coherent with Δ' when $\Delta \cup \Delta'$ is coherent.

Theorem

Let $\delta_1(x, y), \delta_2(x, y) \in \mathcal{F}m(2)$ be different and coherent with each other. If $\delta_1(\alpha, \beta) = \delta_2(\gamma, \epsilon)$ then $\alpha = \epsilon$ and $\beta = \gamma$. In particular, if $\delta_1(\alpha, \beta) = \delta_2(\alpha, \beta)$, then $\alpha = \beta$.

Coherent Δ-logics

Coherent Δ-logics

Theorem

If $\Delta(x, y) \subseteq \mathcal{F}\boldsymbol{m}(2)$ is coherent and $\delta(x, y) \in \Delta(x, y)$, then the theorems of $\mathcal{L}_\Delta$ are all substitution instances of $\delta(x, x)$, i.e.

$$C_\Delta \varnothing = \{\delta(\varphi, \varphi) : \varphi \in \mathcal{F}\boldsymbol{m}\}$$

Coherent Δ-logics

Theorem

If $\Delta(x, y) \subseteq \mathcal{F}m(2)$ is coherent and $\delta(x, y) \in \Delta(x, y)$, then the theorems of $\mathcal{L}_\Delta$ are all substitution instances of $\delta(x, x)$, i.e.

$$C_\Delta \varnothing = \{\delta(\varphi, \varphi) : \varphi \in \mathcal{F}m\}$$

Theorem

Let $\Delta(x, y), \Delta'(x, y) \subseteq \mathcal{F}m(2)$ be two coherent sets, then $C_\Delta \varnothing = C_{\Delta'} \varnothing$ if and only if Δ is coherent with Δ'.

Coherent Δ-logics

Theorem

Let Δ be a coherent set. If $\Gamma \vdash_\Delta \varphi$ then φ is either a theorem or a subformula of some formula in Γ.

Coherent Δ-logics

Theorem

Let Δ be a coherent set. If $\Gamma \vdash_\Delta \varphi$ then φ is either a theorem or a subformula of some formula in Γ.

Corollary

For each coherent set Δ:

1. *If Γ is a set of variables, then $C_\Delta \Gamma = \Gamma \cup C_\Delta \varnothing$*
2. *If Γ is finite, then $C_\Delta \Gamma$ contains only a finite number of non-theorems*
3. *No finite set is inconsistent*
4. *$\phi \dashv\vdash_\Delta \psi$ if and only if ϕ and ψ are both theorems or $\phi = \psi$*

Coherent Δ-logics

Theorem

Let Δ be a coherent set. If $\Gamma \vdash_\Delta \varphi$ then φ is either a theorem or a subformula of some formula in Γ.

Corollary

For each coherent set Δ:

1. *If Γ is a set of variables, then $C_\Delta \Gamma = \Gamma \cup C_\Delta \varnothing$*
2. *If Γ is finite, then $C_\Delta \Gamma$ contains only a finite number of non-theorems*
3. *No finite set is inconsistent*
4. *$\phi \dashv\vdash_\Delta \psi$ if and only if ϕ and ψ are both theorems or $\phi = \psi$*

Theorem

Let $\Delta(x, y) \subseteq \mathcal{F}m(2)$ a coherent and Γ be a finite set with fewer elements than Δ. Then $C_\Delta \Gamma = \Gamma \cup C_\Delta \varnothing$.

Consequence in $\mathcal{L}_\delta$

Consequence in $\mathcal{L}_\delta$

Definition ($\tilde{\delta}$ notation)

Let $\delta \in \mathcal{F}m(2)$*,* β *be a formula and* $\overrightarrow{\alpha}$ *be a finite sequence of formulas. Then* $\tilde{\delta}(\overrightarrow{\alpha}, \beta)$ *denotes the formula inductively defined as follows:*

- $\tilde{\delta}(\varnothing, \beta) = \beta$
- $\tilde{\delta}(\langle\alpha\rangle, \beta) = \delta(\alpha, \beta)$
- $\tilde{\delta}(\overrightarrow{\alpha}, \beta) = \delta(\alpha_1, \tilde{\delta}(\overrightarrow{\alpha}_{[2)}, \beta))$

Consequence in $\mathcal{L}_\delta$

Definition ($\tilde{\delta}$ notation)

Let $\delta \in \mathcal{F}m(2)$, β *be a formula and* $\overrightarrow{\alpha}$ *be a finite sequence of formulas. Then* $\tilde{\delta}(\overrightarrow{\alpha}, \beta)$ *denotes the formula inductively defined as follows:*

- $\tilde{\delta}(\varnothing, \beta) = \beta$
- $\tilde{\delta}(\langle \alpha \rangle, \beta) = \delta(\alpha, \beta)$
- $\tilde{\delta}(\overrightarrow{\alpha}, \beta) = \delta(\alpha_1, \tilde{\delta}(\overrightarrow{\alpha}_{[2)}, \beta))$

Corollary

With the above notation, if $\delta \in \mathcal{F}m(2)$ *and* $\overrightarrow{\epsilon} = \langle \epsilon_1, \ldots, \epsilon_n \rangle$ *is a finite sequence of formulas, then* $\epsilon_1, \ldots, \epsilon_n, \tilde{\delta}(\overrightarrow{\epsilon}, \beta) \vdash_\delta \beta$.

Consequence in $\mathcal{L}_\delta$

Theorem

If $\Gamma \vdash_\delta \beta$ *then* β *satisfies one of the following conditions:*

1. β *is a theorem*
2. $\beta \in \Gamma$
3. *there is a finite non-empty sequence of formulas* $\overrightarrow{\epsilon}$*, each satisfying one of these three conditions, such that* $\tilde{\delta}(\overrightarrow{\epsilon}, \beta) \in \Gamma$

Consequence in $\mathcal{L}_\delta$

Theorem

If $\Gamma \vdash_\delta \beta$ *then* β *satisfies one of the following conditions:*

1. β *is a theorem*
2. $\beta \in \Gamma$
3. *there is a finite non-empty sequence of formulas* $\overrightarrow{\epsilon}$*, each satisfying one of these three conditions, such that* $\tilde{\delta}(\overrightarrow{\epsilon}, \beta) \in \Gamma$

Theorem

$\alpha \vdash_\delta \beta$ *if and only if* β *satisfies one of the following conditions:*

1. β *is a theorem*
2. $\alpha = \beta$
3. $\alpha = \tilde{\delta}(\overrightarrow{\tau}, \beta)$ *for some finite non-empty sequence* $\overrightarrow{\tau}$ *of theorems*

Table of Contents

General results

General results

Theorem

*If $\mathcal{L} \in$ **Prot** and $\mathcal{L} \leq \mathcal{L}'$ then $\mathcal{L}' \in$ **Prot** and has the same witnessing set as $\mathcal{L}$. Thus **Prot** is an up-set of the lattice **Log**.*

General results

Theorem

If $\mathcal{L} \in \mathbf{Prot}$ *and* $\mathcal{L} \leq \mathcal{L}'$ *then* $\mathcal{L}' \in \mathbf{Prot}$ *and has the same witnessing set as* $\mathcal{L}$*. Thus* **Prot** *is an up-set of the lattice* **Log***.*

Corollary

If $\{\mathcal{L}_i : i \in I\} \subseteq \mathbf{Prot}$ *is non-empty, then* $\vee_{i\in I}\mathcal{L}_i \in \mathbf{Prot}$*. Thus, the poset* **Prot** *is a join-complete sub-semilattice of* **Log***.*

General results

Theorem

There is no weakest protoalgebraic logic, that is, **Prot** *has no minimum.*

General results

Theorem

There is no weakest protoalgebraic logic, that is, **Prot** *has no minimum.*

Corollary

Prot *is not a meet-complete semilattice.*

General results

Theorem

There is no weakest protoalgebraic logic, that is, **Prot** *has no minimum.*

Corollary

Prot *is not a meet-complete semilattice.*

Theorem

Prot *is not a meet-semilattice.*

Iteration of formulas

Iteration of formulas

Definition

Let $\delta(x, y) \in \mathcal{F}m(2)$, the formula $\delta^{\mathbf{i}}(x, y)$ (called the iteration of δ) is defined as follows

$$\delta^{\mathbf{i}}(x, y) = \delta(\delta(x, x), \delta(x, y))$$

This definition is extended to sets $\Delta(x, y) \subseteq \mathcal{F}m(2)$ as

$$\Delta^{\mathbf{i}}(x, y) = \{\delta'(\delta(x, x), \delta(x, y)) : \delta, \delta' \in \Delta\}$$

Iteration of formulas

Definition

Let $\delta(x, y) \in \mathscr{F}m(2)$, the formula $\delta^{\mathbf{i}}(x, y)$ (called the iteration of δ) is defined as follows

$$\delta^{\mathbf{i}}(x, y) = \delta(\delta(x, x), \delta(x, y))$$

This definition is extended to sets $\Delta(x, y) \subseteq \mathscr{F}m(2)$ as

$$\Delta^{\mathbf{i}}(x, y) = \{\delta'(\delta(x, x), \delta(x, y)) : \delta, \delta' \in \Delta\}$$

Theorem

If $\Delta(x, y)$ is coherent, then $\Delta^{\mathbf{i}}(x, y)$ is coherent as well. More precisely, if $\Delta(x, y)$ is coherent with $\delta(x, y)$, then $\Delta^{\mathbf{i}}(x, y)$ is coherent with $\delta^{\mathbf{i}}(x, y)$.

Iteration of formulas

Theorem

Let $\Delta(x, y) \subseteq \mathcal{F}m(2)$ *be a coherent set, then* $\mathcal{L}_{\Delta^{\mathsf{i}}} < \mathcal{L}_{\Delta}$.

Iteration of formulas

Theorem

Let $\Delta(x, y) \subseteq \mathcal{F}m(2)$ be a coherent set, then $\mathcal{L}_{\Delta^{\mathsf{i}}} < \mathcal{L}_{\Delta}$.

Corollary

There are infinitely many strictly descending chains in **Prot**, *having no lower bound.*

Ordering the logics $\mathcal{L}_\Delta$

Ordering the logics $\mathcal{L}_\Delta$

Theorem

Let $\Delta(x, y) \subseteq \mathcal{F}m(2)$ *be a coherent set and* $\Delta', \Delta'' \subseteq \Delta$ *be non-empty. Then* $\Delta' \subseteq \Delta''$ *if and only if* $\mathcal{L}_{\Delta''} \leq \mathcal{L}_{\Delta'}$.

Ordering the logics $\mathcal{L}_\Delta$

Theorem

Let $\Delta(x, y) \subseteq \mathcal{F}m(2)$ be a coherent set and $\Delta', \Delta'' \subseteq \Delta$ be non-empty. Then $\Delta' \subseteq \Delta''$ if and only if $\mathcal{L}_{\Delta''} \leq \mathcal{L}_{\Delta'}$.

Corollary

Every finite Boolean lattice is isomorphic to a lattice of protoalgebraic logics.

Table of Contents

The Lindenbaum-Tarski process

The Lindenbaum-Tarski process

$$\Gamma \vdash_{\mathcal{C}\ell} \varphi \Longleftrightarrow \Gamma \vdash_{\mathbf{2}} \varphi$$

The Lindenbaum-Tarski process

$$\Gamma \vdash_{\mathcal{C}\ell} \varphi \Longleftrightarrow \Gamma \vdash_{\mathbf{2}} \varphi$$

1. Suppose $\Gamma \nvdash_{\mathcal{C}\ell} \varphi$

The Lindenbaum-Tarski process

$$\Gamma \vdash_{\mathcal{C}\ell} \varphi \Longleftrightarrow \Gamma \vdash_{\mathbf{2}} \varphi$$

1. Suppose $\Gamma \nvdash_{\mathcal{C}\ell} \varphi$
2. Show that there is a maximally consistent theory Γ' such that $\Gamma \subseteq \Gamma'$ and $\varphi \notin \Gamma'$ (Lindenbaum Lema)

The Lindenbaum-Tarski process

$$\Gamma \vdash_{\mathcal{C}\ell} \varphi \Longleftrightarrow \Gamma \vdash_{\mathbf{2}} \varphi$$

1. Suppose $\Gamma \nvdash_{\mathcal{C}\ell} \varphi$
2. Show that there is a maximally consistent theory Γ' such that $\Gamma \subseteq \Gamma'$ and $\varphi \notin \Gamma'$ (Lindenbaum Lema)
3. Define the function $h : \mathcal{F}m \to \mathbf{2}$ as $h(\varphi) = 1$ if and only if $\varphi \in \Gamma'$

The Lindenbaum-Tarski process

$$\Gamma \vdash_{\mathcal{C}\ell} \varphi \Longleftrightarrow \Gamma \vdash_{\mathbf{2}} \varphi$$

1. Suppose $\Gamma \nvdash_{\mathcal{C}\ell} \varphi$
2. Show that there is a maximally consistent theory Γ' such that $\Gamma \subseteq \Gamma'$ and $\varphi \notin \Gamma'$ (Lindenbaum Lema)
3. Define the function $h : \mathcal{F}m \to \mathbf{2}$ as $h(\varphi) = 1$ if and only if $\varphi \in \Gamma'$
4. Prove that the above function is an homomorphism

The Lindenbaum-Tarski process

$$\Gamma \vdash_{\mathcal{C}\ell} \varphi \Longleftrightarrow \Gamma \vdash_{\mathbf{2}} \varphi$$

1. Suppose $\Gamma \nvdash_{\mathcal{C}\ell} \varphi$
2. Show that there is a maximally consistent theory Γ' such that $\Gamma \subseteq \Gamma'$ and $\varphi \notin \Gamma'$ (Lindenbaum Lema)
3. Define the relation $\boldsymbol{\Omega}\Gamma'$ in $\mathcal{F}\boldsymbol{m}$ as: $\varphi \equiv \psi(\boldsymbol{\Omega}\Gamma')$ if and only if $\varphi \leftrightarrow \psi \in \Gamma'$

The Lindenbaum-Tarski process

$$\Gamma \vdash_{\mathcal{C}\ell} \varphi \Longleftrightarrow \Gamma \vdash_{\mathbf{2}} \varphi$$

1. Suppose $\Gamma \nvdash_{\mathcal{C}\ell} \varphi$
2. Show that there is a maximally consistent theory Γ' such that $\Gamma \subseteq \Gamma'$ and $\varphi \notin \Gamma'$ (Lindenbaum Lema)
3. Define the relation $\boldsymbol{\Omega}\Gamma'$ in $\mathcal{F}\boldsymbol{m}$ as: $\varphi \equiv \psi(\boldsymbol{\Omega}\Gamma')$ if and only if $\varphi \leftrightarrow \psi \in \Gamma'$
4. Show that $\boldsymbol{\Omega}\Gamma'$ is a congruence in $\mathcal{F}\boldsymbol{m}$

The Lindenbaum-Tarski process

$$\Gamma \vdash_{\mathcal{C}\ell} \varphi \Longleftrightarrow \Gamma \vdash_{\mathbf{2}} \varphi$$

1. Suppose $\Gamma \nvdash_{\mathcal{C}\ell} \varphi$
2. Show that there is a maximally consistent theory Γ' such that $\Gamma \subseteq \Gamma'$ and $\varphi \notin \Gamma'$ (Lindenbaum Lema)
3. Define the relation $\boldsymbol{\Omega}\Gamma'$ in $\boldsymbol{\mathcal{F}m}$ as: $\varphi \equiv \psi(\boldsymbol{\Omega}\Gamma')$ if and only if $\varphi \leftrightarrow \psi \in \Gamma'$
4. Show that $\boldsymbol{\Omega}\Gamma'$ is a congruence in $\boldsymbol{\mathcal{F}m}$
5. Show that for any $\varphi \in \boldsymbol{\mathcal{F}m}$, $\varphi \in \Gamma'$ if and only if $\varphi/\boldsymbol{\Omega}\Gamma' \in \Gamma'/\boldsymbol{\Omega}\Gamma'$

The Lindenbaum-Tarski process

$$\Gamma \vdash_{\mathcal{Cl}} \varphi \iff \Gamma \vdash_{\mathbf{2}} \varphi$$

1. Suppose $\Gamma \nvdash_{\mathcal{Cl}} \varphi$
2. Show that there is a maximally consistent theory Γ' such that $\Gamma \subseteq \Gamma'$ and $\varphi \notin \Gamma'$ (Lindenbaum Lema)
3. Define the relation $\boldsymbol{\Omega}\Gamma'$ in $\boldsymbol{\mathcal{F}m}$ as: $\varphi \equiv \psi(\boldsymbol{\Omega}\Gamma')$ if and only if $\varphi \leftrightarrow \psi \in \Gamma'$
4. Show that $\boldsymbol{\Omega}\Gamma'$ is a congruence in $\boldsymbol{\mathcal{F}m}$
5. Show that for any $\varphi \in \boldsymbol{\mathcal{F}m}$, $\varphi \in \Gamma'$ if and only if $\varphi/\boldsymbol{\Omega}\Gamma' \in \Gamma'/\boldsymbol{\Omega}\Gamma'$
6. Show that $\boldsymbol{\mathcal{F}m}/\boldsymbol{\Omega}\Gamma'$ is a boolean algebra

The Leibniz operator

Definition

A congruence θ in an algebra $\mathcal{A}$ is said to be compatible with a set $F \subseteq A$ when any of the following properties hold:

1. *For any $a, b \in A$, if $a \in F$ and $a \equiv b(\theta)$, then $b \in F$*
2. *For any $a \in A$, $a \in F$ if and only if $a/\theta \in F/\theta$*
3. $F = \cup_{a \in F} a/\theta$

The Leibniz operator

Definition

A congruence θ *in an algebra* $\mathcal{A}$ *is said to be compatible with a set* $F \subseteq A$ *when any of the following properties hold:*

1. *For any* $a, b \in A$*, if* $a \in F$ *and* $a \equiv b(\theta)$*, then* $b \in F$
2. *For any* $a \in A$*,* $a \in F$ *if and only if* $a/\theta \in F/\theta$
3. $F = \cup_{a \in F} a/\theta$

Definition (The Leibniz operator)

The Leibniz operator in an algebra $\mathcal{A}$ *is the function* $\mathbf{\Omega}^{\mathcal{A}} : \mathcal{P}(\mathcal{A}) \to Con\mathcal{A}$ *that associates to each* $F \subseteq A$ *the largest congruence on* $\mathcal{A}$ *compatible with* F*.*

The Leibniz operator

$\mathcal{L}$ is ...	if and only if over every $\boldsymbol{A}$, $\Omega^{\boldsymbol{A}}$ is ... (equivalently: over $\boldsymbol{Fm}$, $\boldsymbol{\Omega}$ is ...)
Protoalgebraic	monotone
Equivalential	monotone and commutes with endomorphisms
Finitely equivalential	continuous
Truth-equational	completely order-reflecting
Weakly algebraizable	monotone and injective (i.e., an isomorphism)
Algebraizable	an isomorphism that commutes with endomorphisms
Finitely algebraizable	a continuous isomorphism

The Leibniz hierarchy

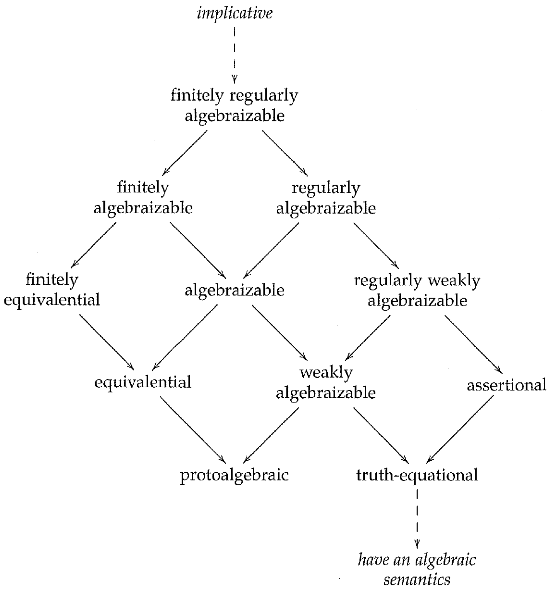

Table of Contents

References

1. Josep Maria Font, Ordering protoalgebraic logics, Journal of Logic and Computation 26 (2016)
2. Josep Maria Font, Abstract Algebraic Logic - An Introductory Textbook, 2016