Domain Theory in Logical Form Revisited: A 20 Year Retrospective

Samson Abramsky

Oxford University Computing Laboratory

The Big Picture

A Basic Intuition

Consider the relation $P \models \phi$, where P is a program, ϕ a property.

We can look at this in two ways:

A Basic Intuition

Consider the relation $P \models \phi$, where P is a program, ϕ a property.
We can look at this in two ways:

- The **denotational view**: $[\![P]\!]$ is a "point" in a mathematical space $\mathcal{D}(\sigma)$, where σ is the type of P. Then we interpret ϕ extensionally as $[\![\phi]\!] \subseteq \mathcal{D}(\sigma)$, and $P \models \phi$ means $[\![P]\!] \in [\![\phi]\!]$.

A Basic Intuition

Consider the relation $P \models \phi$, where P is a program, ϕ a property.
We can look at this in two ways:

- The **denotational view**: $[\![P]\!]$ is a "point" in a mathematical space $\mathcal{D}(\sigma)$, where σ is the type of P. Then we interpret ϕ extensionally as $[\![\phi]\!] \subseteq \mathcal{D}(\sigma)$, and $P \models \phi$ means $[\![P]\!] \in [\![\phi]\!]$.

- The **axiomatic view**: we axiomatize $P \models \phi$ as a logical theory of which properties P satisfies. Thinking of properties as "observations", we can then seek to construct the meaning of P out of the properties it satisfies:

$$[\![P]\!] = \{\phi \mid P \models \phi\}$$

  Then $P \models \phi$ means $\phi \in [\![P]\!]$.

A Basic Intuition

Consider the relation $P \models \phi$, where P is a program, ϕ a property.
We can look at this in two ways:

- The **denotational view**: $[\![P]\!]$ is a "point" in a mathematical space $\mathcal{D}(\sigma)$, where σ is the type of P. Then we interpret ϕ extensionally as $[\![\phi]\!] \subseteq \mathcal{D}(\sigma)$, and $P \models \phi$ means $[\![P]\!] \in [\![\phi]\!]$.
- The **axiomatic view**: we axiomatize $P \models \phi$ as a logical theory of which properties P satisfies. Thinking of properties as "observations", we can then seek to construct the meaning of P out of the properties it satisfies:

$$[\![P]\!] = \{\phi \mid P \models \phi\}$$

  Then $P \models \phi$ means $\phi \in [\![P]\!]$.

How can we reconcile these views, and indeed bring them into exact correspondence? An elegant and robust mathematical framework for these ideas is provided by **Stone Duality**.

Stone Duality

The Classical Stone Representation Theorem (1931):

Stone Duality

The Classical Stone Representation Theorem (1931):

- The Problem: given an abstract Boolean algebra B, represent it as a concrete algebra of sets.

Stone Duality

The Classical Stone Representation Theorem (1931):

- The Problem: given an abstract Boolean algebra B, represent it as a concrete algebra of sets.
- Form the space of ultrafilters over B: *i.e.* $h^{-1}(1)$ for homomorphisms $h : B \longrightarrow \mathbf{2}$. This space $\mathsf{Pt}(B)$ is naturally topologized by taking basic opens

$$U_b = \{x \mid b \in x\}$$

Then $B \cong \mathsf{Clop}(\mathsf{Pt}(B))$.

Stone Duality

The Classical Stone Representation Theorem (1931):

- The Problem: given an abstract Boolean algebra B, represent it as a concrete algebra of sets.
- Form the space of ultrafilters over B: *i.e.* $h^{-1}(1)$ for homomorphisms $h : B \longrightarrow \mathbf{2}$. This space $\mathsf{Pt}(B)$ is naturally topologized by taking basic opens
$$U_b = \{x \mid b \in x\}$$
Then $B \cong \mathsf{Clop}(\mathsf{Pt}(B))$.
- The spaces that arise this way (totally disconnected compact Hausdorff spaces) are the **Stone spaces**. For every Stone space S,
$$S \cong \mathsf{Pt}(\mathsf{Clop}(S)).$$

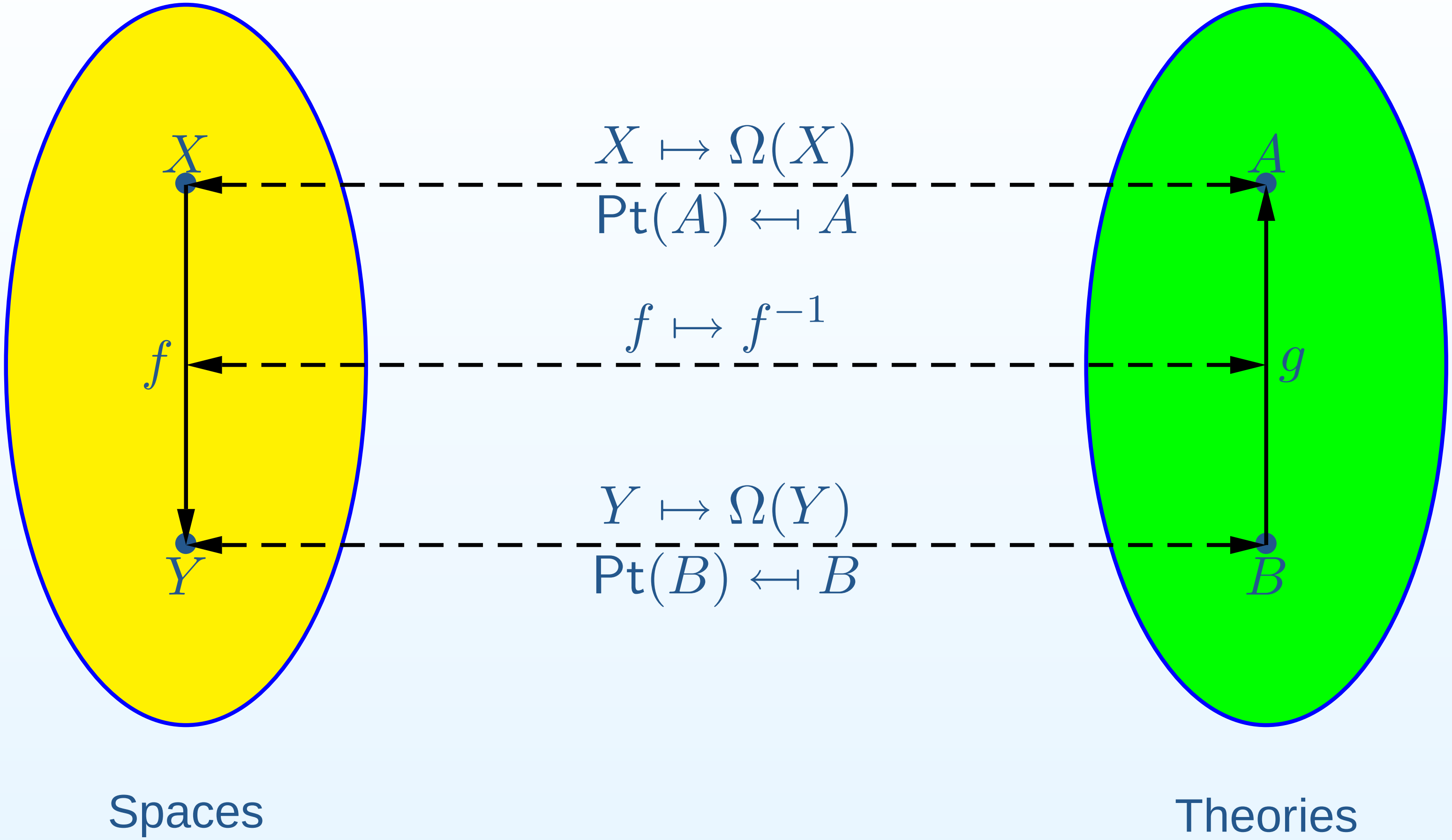
Stone Duality Logically

In logical terms:

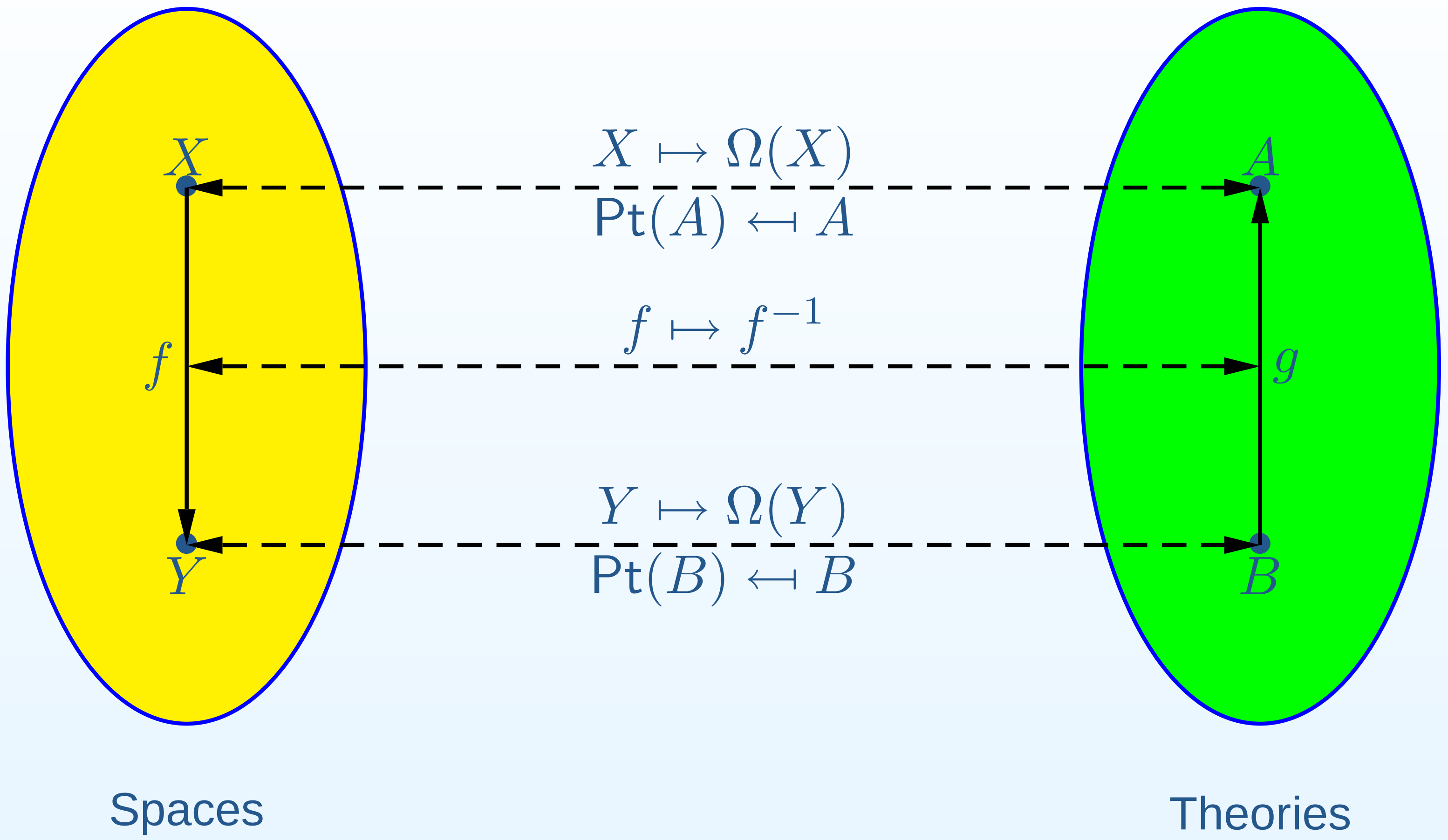
- B is the Lindenbaum algebra of a propositional theory
- Points are models
- The compactness of $\mathsf{Pt}(S)$ subsumes the Compactness Theorem
- The existence of enough points to achieve the above isomorphisms subsumes the Completeness Theorem — *i.e.*

$$\phi \not\leq \psi \implies \exists x \in \mathsf{Pt}(B).\, x \models \phi \ \wedge \ x \not\models \psi.$$

Stone Duality Generalized



Stone Duality Generalized



Note that **maps** are part of the picture, as well as spaces. This is important from the CS point of view — it generalizes the duality of **state transformers** and **predicate transformers**.

What It's About (As CS)

Connecting Program Logics and Semantics:

What It's About (As CS)

Connecting Program Logics and Semantics:

- Deriving a program logic from a semantics and vice versa, in such a way that each uniquely determines the other.

What It's About (As CS)

Connecting Program Logics and Semantics:

- Deriving a program logic from a semantics and vice versa, in such a way that each uniquely determines the other.
- Compositionally (and effectively) deriving program logics for complex semantic domains.

What It's About (As CS)

Connecting Program Logics and Semantics:

- Deriving a program logic from a semantics and vice versa, in such a way that each uniquely determines the other.
- Compositionally (and effectively) deriving program logics for complex semantic domains.
- A generalized Dynamic Logic covering all the constructs of denotational semantics — higher order functions, recursive types, non-determinism, etc.

What It's About (As CS)

Connecting Program Logics and Semantics:

- Deriving a program logic from a semantics and vice versa, in such a way that each uniquely determines the other.
- Compositionally (and effectively) deriving program logics for complex semantic domains.
- A generalized Dynamic Logic covering all the constructs of denotational semantics — higher order functions, recursive types, non-determinism, etc.
- Using the logical form to unpack the structure of complex, recursively defined semantic domains.

Domain Theory in Logical Form

Denotational Metalanguage

Syntax of Types:

$$\sigma \quad ::= \quad \sigma \times \tau \mid \sigma \rightarrow \tau \mid \sigma \oplus \tau \mid \sigma_\perp \mid P\sigma \mid X \mid \mathsf{rec}\, X.\, \sigma \mid \ldots$$

Typed terms $t : \sigma$.

Domain Theory in Logical Form

Denotational Metalanguage

Syntax of Types:

$$\sigma \quad ::= \quad \sigma \times \tau \mid \sigma \rightarrow \tau \mid \sigma \oplus \tau \mid \sigma_{\perp} \mid P\sigma \mid X \mid \mathsf{rec}\, X.\, \sigma \mid \ldots$$

Typed terms $t : \sigma$.

The programme:

Domain Theory in Logical Form

Denotational Metalanguage

Syntax of Types:

$$\sigma \quad ::= \quad \sigma \times \tau \mid \sigma \rightarrow \tau \mid \sigma \oplus \tau \mid \sigma_\perp \mid P\sigma \mid X \mid \text{rec}\, X.\, \sigma \mid \ldots$$

Typed terms $t : \sigma$.

The programme:

- We assign, in a syntax-directed (*i.e.* compositional) fashion, a propositional theory $\mathcal{L}(\sigma)$ to each type σ, such that (the Lindenbaum algebra of) $\mathcal{L}(\sigma)$ is the **Stone dual** of $\mathcal{D}(\sigma)$, the domain associated (in conventional denotational semantics) to σ.

Domain Theory in Logical Form

Denotational Metalanguage

Syntax of Types:

$$\sigma \quad ::= \quad \sigma \times \tau \mid \sigma \to \tau \mid \sigma \oplus \tau \mid \sigma_\perp \mid P\sigma \mid X \mid \text{rec}\, X.\, \sigma \mid \ldots$$

Typed terms $t : \sigma$.

The programme:

- We assign, in a syntax-directed (*i.e.* compositional) fashion, a propositional theory $\mathcal{L}(\sigma)$ to each type σ, such that (the Lindenbaum algebra of) $\mathcal{L}(\sigma)$ is the **Stone dual** of $\mathcal{D}(\sigma)$, the domain associated (in conventional denotational semantics) to σ.
- We axiomatize the meaning of terms $t : \sigma \to \tau$ as
  - Endogenous version: $\phi\{t\}\psi$
  - Exogenous version: $\phi \leq [t]\psi$

  with the intended meaning

$$[\![\phi]\!] \subseteq [\![t]\!]^{-1}([\![\psi]\!]).$$

Domain as Theories

This requires giving constructions on theories which yield the Stone duals of denotational constructions such as function spaces and powerdomains:

$$\begin{aligned}\mathcal{L}(\sigma \to \tau) &= \mathcal{L}(\sigma) \to_{\mathcal{L}} \mathcal{L}(\tau) \\ \mathcal{L}(P\sigma) &= P_{\mathcal{L}}(\mathcal{L}(\sigma))\end{aligned}$$

Recursive types are handled by inductive definitions of the logics — in effect, allowing arbitrary nesting of (generalized) modalities. Cf. current work on "coalgebraic logic".

Domain as Theories

This requires giving constructions on theories which yield the Stone duals of denotational constructions such as function spaces and powerdomains:

$$\begin{aligned} \mathcal{L}(\sigma \to \tau) &= \mathcal{L}(\sigma) \to_{\mathcal{L}} \mathcal{L}(\tau) \\ \mathcal{L}(P\sigma) &= P_{\mathcal{L}}(\mathcal{L}(\sigma)) \end{aligned}$$

Recursive types are handled by inductive definitions of the logics — in effect, allowing arbitrary nesting of (generalized) modalities. Cf. current work on "coalgebraic logic".

In this way, we can **read off presentations of complex semantic domains as propositional theories**, hence unpacking their structure.

Some Details

Stone Duality For Domains

"Domains" here are ω-algebraic cpo's. Can be viewed as topological spaces under the Scott topology.

We are interested in categories of such domains which are **cartesian closed**, and closed under various constructions, e.g. **powerdomains**:

- Scott domains
- SFP

Stone duality for these categories can be seen as restrictions for the Stone duality between distributive lattices and "coherent" or "spectral" spaces — those for which the compact-open sets generate the topology.

Some Trade Secrets

The lattices of compact-opens for domains are **coprime-generated**, *i.e.* every element is a finite join of coprimes, where a is **coprime** if:

$$a \leq \bigvee_{i \in I} b_i \;\; \Rightarrow \;\; \exists i. \, a \leq b_i.$$

(These correspond to “2/3 SFP domains”. SFP or Scott domains require additional axioms, which in the case of SFP are **non-elementary**.)

Coprimes correspond to the basic opens $\uparrow(a)$, a a compact element of D, in the Scott topology on D.

Our axiomatization involves a predicate $\mathbf{C}(a)$ for coprimeness. The key point in proving completeness (and effectiveness) of the axiomatization is to show that there are **effective coprime normal forms**.

Philosophical Aside

How should we interpret the basic concepts of Domain Theory:

$$d \sqsubseteq e$$

and “partial elements”, e.g. in $\mathbb{N}_\bot$, Σ^∞?

Two views:

- **Ontological**: Partial elements are possible states of the computation system, independently of any observer: necessary extensions to our universe of discourse.
- **Epistemic**: We (implicitly) assume an observer; (compact) partial elements are **observable properties**.

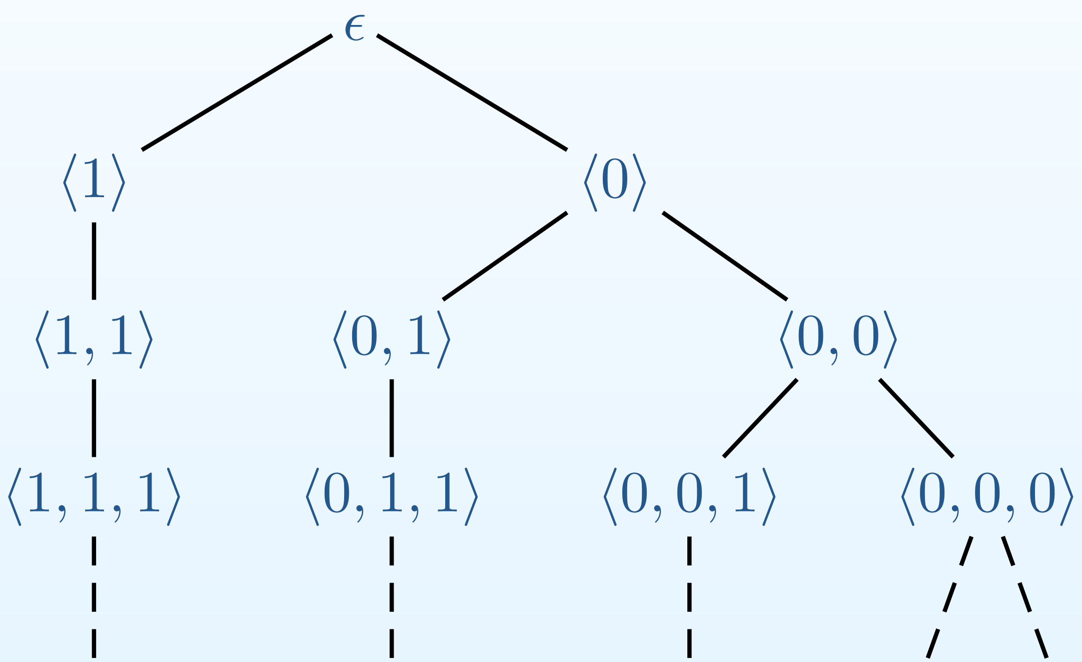
In fact, both readings are useful. The particular feature of domains which allows this creative ambiguity between points and properties to be used so freely is that **basic points and basic properties (or observations) are essentially the same things**. E.g. finite streams.

Domain Constructions In Logical Form

The Plotkin Powerdomain

If D is a domain, we want to make a domain $P(D)$ of subsets of D, to represent **non-deterministic computation over** D.

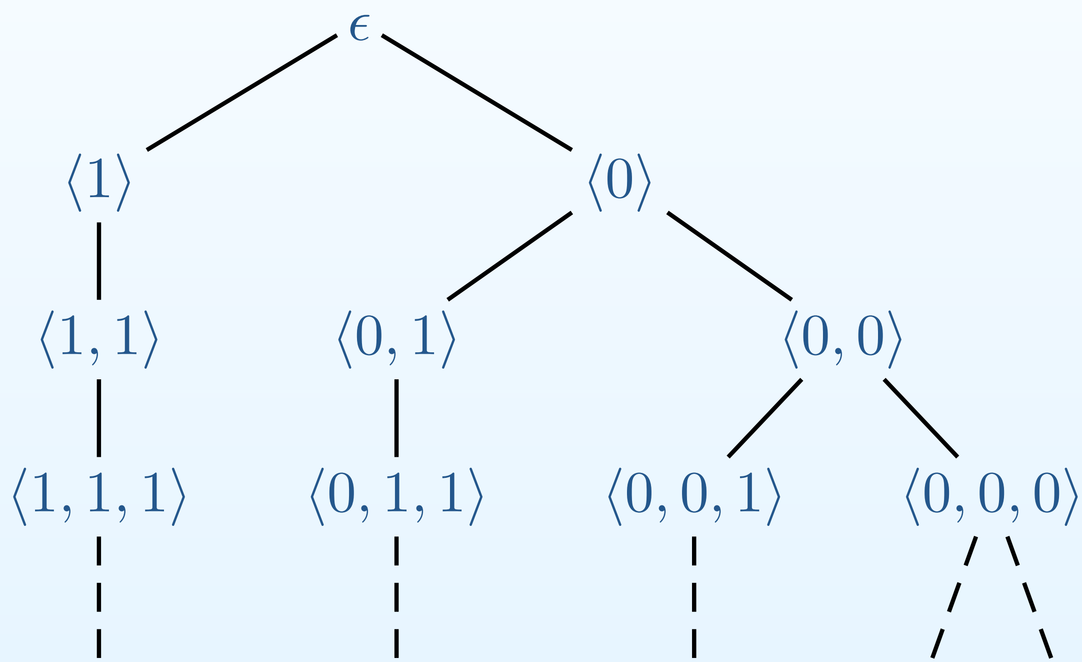
Which sets? the **finitely generable** ones.



The Plotkin Powerdomain

If D is a domain, we want to make a domain $P(D)$ of subsets of D, to represent **non-deterministic computation over** D.

Which sets? the **finitely generable** ones.

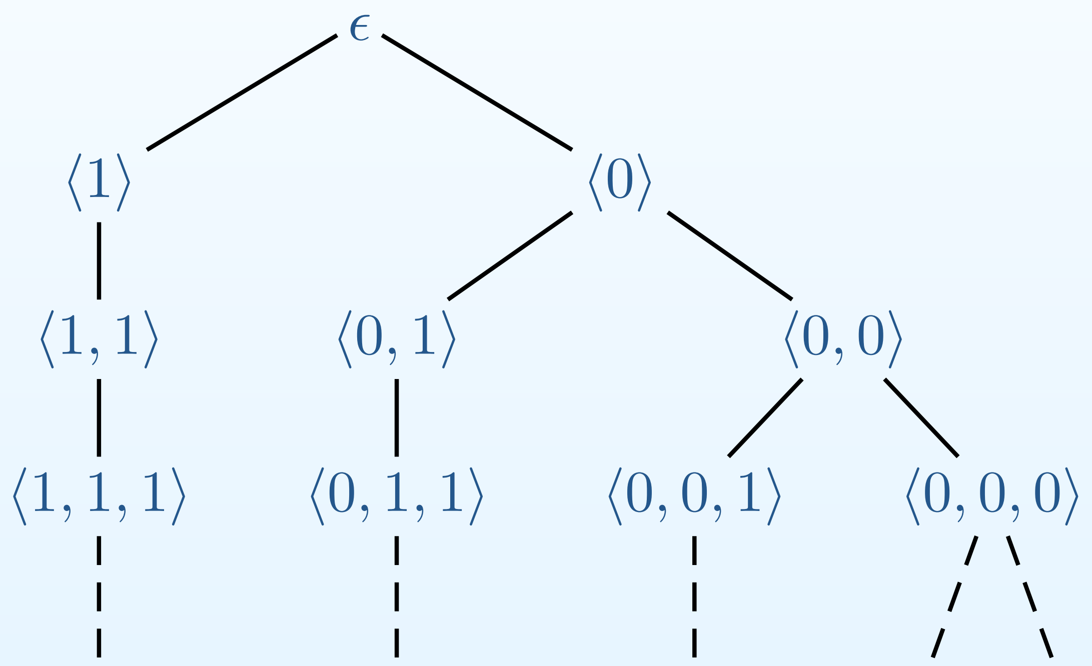


The set generated is $0^* 1^\omega \cup 0^\omega$.

The Plotkin Powerdomain

If D is a domain, we want to make a domain $P(D)$ of subsets of D, to represent **non-deterministic computation over** D.

Which sets? the **finitely generable** ones.



The set generated is $0^*1^\omega \cup 0^\omega$. Can we generate 0^*1^ω?

How to order sets

Consider two sets which could appear as cross-sections X_n, X_{n+1} of a generating tree. These are finite sets of finite elements.

How to order sets

Consider two sets which could appear as cross-sections X_n, X_{n+1} of a generating tree. These are finite sets of finite elements.

Note that:

- Each node labelled with b in X_n has one or more successors in X_{n+1}, each labelled with some b' such that $b \sqsubseteq b'$.
- Each node labelled with b' in X_{n+1} has an ancestor labelled with some b in X_n such that $b \sqsubseteq b'$.

How to order sets

Consider two sets which could appear as cross-sections X_n, X_{n+1} of a generating tree. These are finite sets of finite elements.

Note that:

- Each node labelled with b in X_n has one or more successors in X_{n+1}, each labelled with some b' such that $b \sqsubseteq b'$.
- Each node labelled with b' in X_{n+1} has an ancestor labelled with some b in X_n such that $b \sqsubseteq b'$.

Abstracting from this situation, we have sets X and Y such that:

- $\forall x \in X. \exists y \in Y. x \sqsubseteq y$
- $\forall y \in Y. \exists x \in X. x \sqsubseteq y$

We write this as $X \sqsubseteq_{\mathrm{EM}} Y$: the **Egli-Milner order**.

Axiomatizing The Plotkin Powerdomain

(i) The generators:

$$G(P(A)) \equiv \{\Box a : a \in |A|\} \cup \{\Diamond a : a \in |A|\}$$

(ii) Axioms:

$$\begin{array}{ll}
(\Box - \wedge) & \Box \bigwedge_{i \in I} a_i = \bigwedge_{i \in I} \Box a_i \\
(\Diamond - \vee) & \Diamond \bigvee_{i \in I} a_i = \bigvee_{i \in I} \Diamond a_i \\
(\Box - \vee) & \Box (a \vee b) \leq \Box a \vee \Diamond b \\
(\Diamond - \wedge) & \Box a \wedge \Diamond b \leq \Diamond (a \wedge b) \\
(\Box - 0) & \Box 0 = 0
\end{array}$$

(iii) Rules:

$$(\Box - \leq) \quad \frac{a \leq b}{\Box a \leq \Box b}$$

$$(\mathbf{C} - \Box - \Diamond) \quad \frac{\{\mathbf{C}_A(a_i)\}_{i \in I} \quad (I \neq \varnothing)}{\mathbf{C}(\Box \bigvee_{i \in I} a_i \wedge \bigwedge_{i \in I} \Diamond a_i)}$$

Remarks on the Axiomatization

Remarks on the Axiomatization

- The axiomatization (aside from coprimeness) is that of the **Vietoris construction** on (coherent) locales. The Hoare and Smyth powerdomains arise by omitting the $\Box$ and $\Diamond$ parts respectively.

Remarks on the Axiomatization

- The axiomatization (aside from coprimeness) is that of the **Vietoris construction** on (coherent) locales. The Hoare and Smyth powerdomains arise by omitting the $\Box$ and $\Diamond$ parts respectively.
- The coprimeness axiom corresponds to the **nabla modality** — picking out "point-like properties".

Remarks on the Axiomatization

- The axiomatization (aside from coprimeness) is that of the **Vietoris construction** on (coherent) locales. The Hoare and Smyth powerdomains arise by omitting the $\Box$ and $\Diamond$ parts respectively.
- The coprimeness axiom corresponds to the **nabla modality** — picking out "point-like properties".
- There is a tight link between **bisimulation** and the Egli-Milner ordering, or the Vietoris construction, first identified in this setting, in my paper: "A Domain Equation for Bisimulation".

Remarks on the Axiomatization

- The axiomatization (aside from coprimeness) is that of the **Vietoris construction** on (coherent) locales. The Hoare and Smyth powerdomains arise by omitting the $\Box$ and $\Diamond$ parts respectively.
- The coprimeness axiom corresponds to the **nabla modality** — picking out "point-like properties".
- There is a tight link between **bisimulation** and the Egli-Milner ordering, or the Vietoris construction, first identified in this setting, in my paper: "A Domain Equation for Bisimulation".
- Note that this Vietoris construction gives "one-level" or "flat" modalities. To get the usual iterated modalities we must combine this with a **recursive domain equation** — yielding a finer analysis.

Axiomatizing Function Spaces

(i) The generators:

$$G(A \to B) \;\equiv\; \{(a \to b) : a \in |A|, b \in |B|\}.$$

(ii) Relations:

$$(\to -\wedge) \qquad (a \to \bigwedge_{i\in I} b_i) = \bigwedge_{i\in I}(a \to b_i)$$

$$(\to - \vee -L) \qquad (\bigvee_{i\in I} a_i \to b) = \bigwedge_{i\in I}(a_i \to b)$$

$$(\to - \vee -R) \qquad \frac{\mathbf{C}_A(a)}{(a \to \bigvee_{i\in I} b_i) = \bigvee_{i\in I}(a \to b_i)}$$

$$(\to - \leq) \qquad \frac{a' \leq a,\ b \leq b'}{(a \to b) \leq (a' \to b')}$$

Axiomatizing Function Spaces

(i) The generators:

$$G(A \to B) \;\equiv\; \{(a \to b) : a \in |A|, b \in |B|\}.$$

(ii) Relations:

$$(\to -\wedge) \qquad (a \to \bigwedge_{i\in I} b_i) = \bigwedge_{i\in I}(a \to b_i)$$

$$(\to -\vee -L) \qquad (\bigvee_{i\in I} a_i \to b) = \bigwedge_{i\in I}(a_i \to b)$$

$$(\to -\vee -R) \qquad \frac{\mathbf{C}_A(a)}{(a \to \bigvee_{i\in I} b_i) = \bigvee_{i\in I}(a \to b_i)}$$

$$(\to -\leq) \qquad \frac{a' \leq a,\; b \leq b'}{(a \to b) \leq (a' \to b')}$$

Note the key use of coprimeness in $(\to -\vee -R)$.

Axiomatizing Function Spaces

(i) The generators:

$$G(A \to B) \;\equiv\; \{(a \to b) : a \in |A|, b \in |B|\}.$$

(ii) Relations:

$$\begin{array}{ll} (\to -\wedge) & (a \to \bigwedge_{i\in I} b_i) = \bigwedge_{i\in I}(a \to b_i) \\ (\to - \vee -L) & (\bigvee_{i\in I} a_i \to b) = \bigwedge_{i\in I}(a_i \to b) \end{array}$$

$$(\to - \vee -R) \quad \frac{\mathbf{C}_A(a)}{(a \to \bigvee_{i\in I} b_i) = \bigvee_{i\in I}(a \to b_i)}$$

$$(\to - \leq) \quad \frac{a' \leq a, \; b \leq b'}{(a \to b) \leq (a' \to b')}$$

Note the key use of coprimeness in $(\to - \vee -R)$.

Note also the resemblance to **intersection types** and **filter models** — again on one level, combining with recursive types to yield a finer analysis.

Coprimeness

The coprimeness axiom:

$$(\mathbf{C}{-}{\longrightarrow}) \quad \frac{\{\mathbf{C}_A(a_i)\}_{i\in I} \quad \{\mathbf{C}_B(b_i)\}_{i\in I} \qquad \forall J \subseteq I.\, \exists K \subseteq I.\, [\bigwedge_{j\in J} a_j =_A \bigvee_{k\in K} a_k \ \&\ [\forall j \in J, k \in K.\, b_k \leq_B b_j]]}{\mathbf{C}(\bigwedge_{i\in I}(a_i \rightarrow b_i))}$$

Coprimeness

The coprimeness axiom:

$$(\mathbf{C}{-}{\rightarrow}) \qquad \frac{\{\mathbf{C}_A(a_i)\}_{i\in I} \quad \{\mathbf{C}_B(b_i)\}_{i\in I} \quad \forall J \subseteq I.\, \exists K \subseteq I.\, [\bigwedge_{j\in J} a_j =_A \bigvee_{k\in K} a_k \;\&\; [\forall j \in J, k \in K.\, b_k \leq_B b_j]]}{\mathbf{C}(\bigwedge_{i\in I}(a_i \rightarrow b_i))}$$

Horrific — but **effective**.

Examples

Examples

- A Domain Equation for Bisimulation

$$\mathsf{ST} = P_0(\Sigma_{a \in \mathsf{Act}} \mathsf{ST})$$

Denotational semantics for process calculi, fully abstract wrt strong bisimulation, connection to Hennessy-Milner logic.

Examples

- A Domain Equation for Bisimulation

$$\mathsf{ST} = P_0(\Sigma_{a \in \mathsf{Act}} \mathsf{ST})$$

  Denotational semantics for process calculi, fully abstract wrt strong bisimulation, connection to Hennessy-Milner logic.

- The Lazy Lambda Calculus.

$$D = [D \longrightarrow D]_{\perp}$$

  Connections with ideas from filter models and intersection types.

Examples

- A Domain Equation for Bisimulation

$$\mathsf{ST} = P_0(\Sigma_{a \in \mathsf{Act}} \mathsf{ST})$$

  Denotational semantics for process calculi, fully abstract wrt strong bisimulation, connection to Hennessy-Milner logic.
- The Lazy Lambda Calculus.

$$D = [D \longrightarrow D]_{\perp}$$

  Connections with ideas from filter models and intersection types.
- The Finitary Non-Well Founded Sets

$$S = V(S)$$

  The Stone Space of the free modal algebra! Carries an interesting set-theory, in which the universe is a set.

Example: The Finitary Non-Well Founded Sets

The Finitary Non-Well-Founded Sets

A quiz for modal logicians:

What is the Stone space of the free modal algebra?

Which kind of set theory does it provide a model for?

The Finitary Non-Well-Founded Sets

A quiz for modal logicians:

What is the Stone space of the free modal algebra?
Which kind of set theory does it provide a model for?

We study an example: the space $\mathbb{F}$ of **finitary non-well-founded sets**. By finitary we mean:

- not the strictly finite
- not the unboundedly infinite
- but the **finitary** *i.e.* those objects appearing as "limits" of finite ones.

The Finitary Non-Well-Founded Sets

A quiz for modal logicians:

What is the Stone space of the free modal algebra?
Which kind of set theory does it provide a model for?

We study an example: the space $\mathbb{F}$ of **finitary non-well-founded sets**. By finitary we mean:

- not the strictly finite
- not the unboundedly infinite
- but the **finitary** *i.e.* those objects appearing as "limits" of finite ones.

We can describe $\mathbb{F}$ as **the Stone space of the free modal algebra (on no generators)**. Here we take a modal algebra to be a Boolean algebra B equipped with a unary operator $\Diamond$ satisfying the axioms

$$\text{(MA)} \quad \Diamond(a \vee b) = \Diamond a \vee \Diamond b \qquad\qquad \Diamond 0 = 0.$$

This is the algebraic variety corresponding to the minimal normal modal logic $\mathbf{K}$. The Boolean algebra is equipped with a constant 0, so the free algebra over no generators can be non-trivial. We shall show that it is indeed non-trivial!

First description of $\mathbb{F}$

We firstly describe $\mathbb{F}$, *qua* topological space, as the solution of a domain equation in $\mathbf{Stone}$. We use the **Vietoris construction** $\mathcal{P}_V$. Given a Stone space S, $\mathcal{P}_V(S)$ is the set of all compact (which since S is compact Hausdorff, is equivalent to closed) subsets of S, with topology generated by

$$\Box U = \{C \mid C \subseteq U\} \qquad (1)$$
$$\Diamond U = \{C \mid C \cap U \neq \varnothing\} \qquad (2)$$

where U ranges over the open sets of S. We can read $\Box U$ as the set of all C such that C **must** satisfy U, and $\Diamond U$ as the set of C such that C **may** satisfy U. The allusion to modal logic notation is thus deliberate, and we shall shortly see a connection to standard modal notions.

Interlude: Domain Equations

We consider domain equations $X \cong F(X)$ for endofunctors $F : \mathcal{C} \longrightarrow \mathcal{C}$. We want **extremal solutions** of such an equation: either an **initial algebra** $\alpha : FA \rightarrow A$, or a **final coalgebra** $\beta : A \rightarrow FA$. (The Lambek lemma then guarantees that the arrow is an isomorphism). These concepts generalize the lattice-theoretic notions of least and greatest fixpoint. In most cases of interest, initial algebras can be constructed as colimits:

$$\lim_{\rightarrow}(\mathbf{0} \rightarrow F\mathbf{0} \rightarrow F^2\mathbf{0} \rightarrow \cdots)$$

generalizing the construction of the least fixpoint as $\bigvee_k F^k \bot$, while final coalgebras can be constructed as limits:

$$\lim_{\leftarrow}(\mathbf{1} \leftarrow F\mathbf{1} \leftarrow F^2\mathbf{1} \leftarrow \cdots)$$

generalizing the construction of the greatest fixpoint as $\bigwedge_k F^k \top$. (In the domain theoretic case, the **limit-colimit coincidence** means that the two constructions coincide, and we obtain **both** an initial algebra **and** a final coalgebra.) For the finitary case we are considering, the functors will be ω-continuous in the appropriate sense, and the limit or colimit can be taken with respect to the ω-chain of finite iterations.

First description

We define $\mathbb{F}$ as the **final coalgebra of the Vietoris functor on $\mathbf{Stone}$**. Since $\mathcal{P}_V$ is cocontinuous, $\mathbb{F}$ is constructed as the limit of the ω^{op}-chain

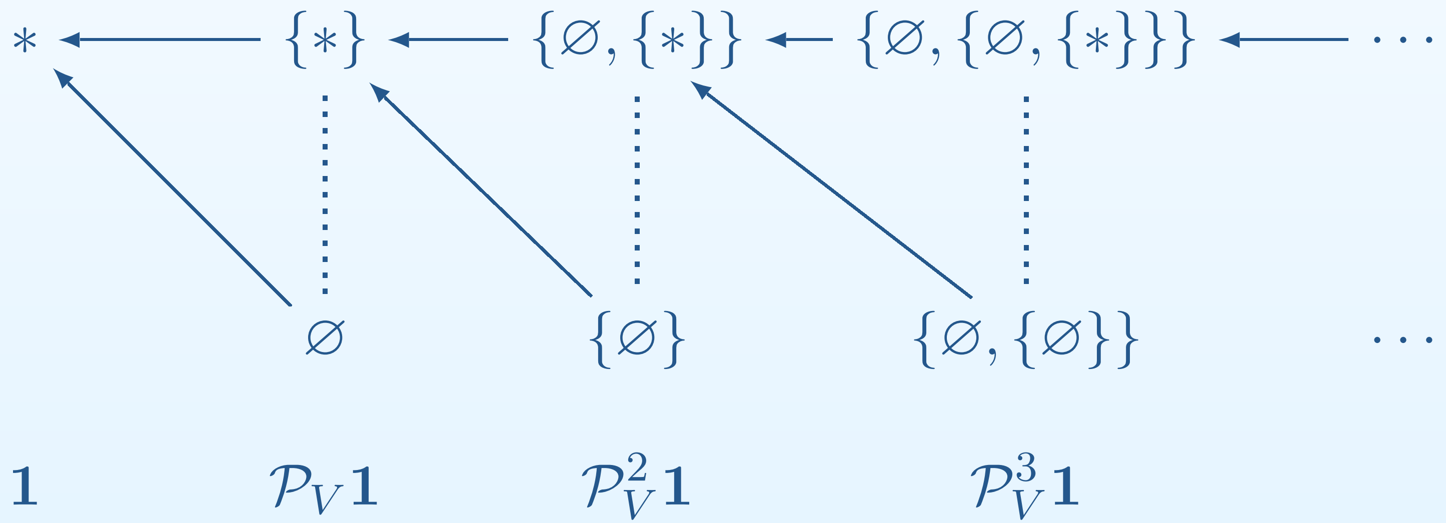
$$\varprojlim(\mathbf{0} \leftarrow \mathcal{P}_V\mathbf{0} \leftarrow \mathcal{P}_V^2\mathbf{0} \leftarrow \cdots)$$

First description

We define $\mathbb{F}$ as the **final coalgebra of the Vietoris functor on $\mathbf{Stone}$**. Since $\mathcal{P}_V$ is cocontinuous, $\mathbb{F}$ is constructed as the limit of the ω^{op}-chain

$$\varprojlim(\mathbf{0} \leftarrow \mathcal{P}_V\mathbf{0} \leftarrow \mathcal{P}_V^2\mathbf{0} \leftarrow \cdots)$$

We give a picture of the first few terms of the construction:

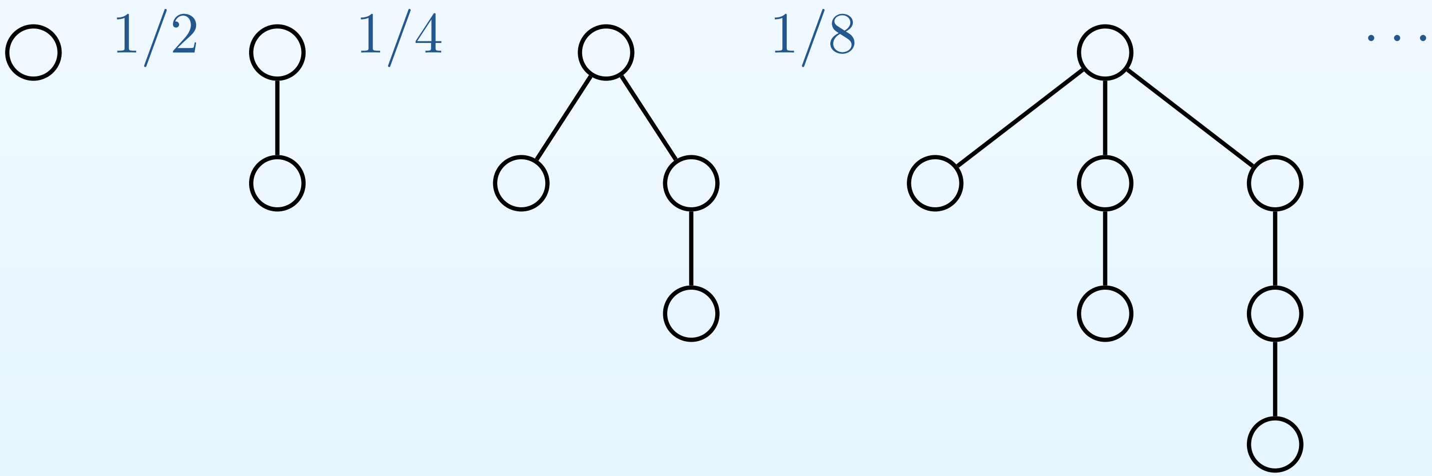


$\mathbb{F}$ As An Ultrametric Completion

$\mathbb{F}$ can equivalently be described as the **ultrametric completion of the hereditarily finite sets**.

$$d(S,T) \quad = \quad \begin{cases} 0, & S \sim T \\ 2^{-k}, \ \text{least } k \text{ such that } S \not\sim_k T & \text{otherwise} \end{cases}$$

Example of Cauchy Sequence:



The corresponding sequence of sets is

$$\varnothing, \quad \{\varnothing\}, \quad \{\varnothing, \{\varnothing\}\}, \quad \{\varnothing, \{\varnothing\}, \quad \{\{\varnothing\}\}\}, \ldots$$

Vietoris Logically

As we have seen, the Vietoris construction can be described **logically** as an operation on **theories**. For the coherent case, $V(L)$, for a distributive lattice L, is the distributive lattice generated by $\Box a$, $\Diamond a$, ($a \in L$), subject to the axioms:

$$\Box(a \wedge b) = \Box a \wedge \Box b \qquad \Diamond(a \vee b) = \Diamond a \vee \Diamond b \tag{3}$$

$$\Box 1 = 1 \qquad \Diamond 0 = 0 \tag{4}$$

$$\Box(a \vee b) \leq \Box a \vee \Diamond b \qquad \Diamond(a \wedge b) \geq \Diamond a \wedge \Box b. \tag{5}$$

In the boolean case, where we have a classical negation, $\Box$ and $\Diamond$ are inter-definable (e.g. $\Box a = \neg \Diamond \neg a$), and the axiomatization simplifies to (MA).

Free Modal Algebra As A Fixpoint

The construction $\mathsf{MA}(B)$ lifts to a functor on $\mathbf{Bool}$, the category of Boolean algebras. We can iterate this construction to get the initial solution of $\mathbb{B} = \mathsf{MA}(\mathbb{B})$ in $\mathbf{Bool}$:

$$\lim_{\rightarrow}(\mathbf{2} \hookrightarrow \mathsf{MA}(\mathbf{2}) \hookrightarrow \mathsf{MA}^2(\mathbf{2}) \hookrightarrow \cdots)$$

Concretely this is the Lindenbaum algebra of the propositional theory which is inductively generated by these iterates. This is the standard modal system $\mathbf{K}$—but with no propositional atoms. Thus another role for domain equations is revealed: systematizing the inductive definition of the formulas and inference rules of a logic.

To see how hereditarily finite sets can be completely characterized by modal formulas (the "master formula" of the set), we define:

$$\mathcal{F}(\varnothing) = \Box 0 \qquad (= \neg\Diamond 1) \tag{6}$$

$$\mathcal{F}(\{x_1, \ldots, x_n\}) = \Box \bigvee_{i=1}^{n} \mathcal{F}(x_i) \wedge \bigwedge_{i=1}^{n} \Diamond \mathcal{F}(x_i). \tag{7}$$

The Characterization

The link between $\mathbb{B}$ and $\mathbb{F}$ is given by **Stone duality**:

Proposition 1 $\mathbb{B}$ *is the Stone dual of* $\mathbb{F}$.

Again, this is an instance of very general results.

Relation To Domains

We define a domain D as the solution (both initial algebra and final coalgebra) of the equation

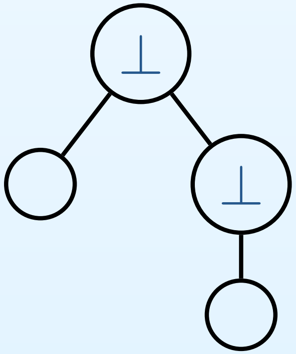
$$D \;=\; \mathcal{P}^0_P(D) = \; \mathbf{1}_\perp \oplus \mathcal{P}_P(D). \tag{8}$$

Here $\mathcal{P}_P(\cdot)$ is the Plotkin powerdomain.

Proposition 2 $\mathbb{F} \cong \mathsf{Max}(D)$, *where* D *is the solution of the domain equation (8).*

We note that D has "partial sets".

Example 3



$\leadsto \quad \{\perp, \varnothing, \{\perp, \varnothing\}\}$

Set Theory In $\mathbb{F}$

We now consider $\mathbb{F}$ as a set-theoretic universe $(\mathbb{F}, \in, =)$. Since we have

$$\mathbb{F} \cong \mathcal{P}_V(\mathbb{F}) \qquad \mathsf{unfold} : \mathbb{F} \xrightarrow{\cong} \mathcal{P}_V(\mathbb{F})$$

we can define

$$S \in T \quad \equiv \quad S \in \mathsf{unfold}(T).$$

Note that in this set theory, the universe V is a set!

Set Theory In $\mathbb{F}$

We now consider $\mathbb{F}$ as a set-theoretic universe $(\mathbb{F}, \in, =)$. Since we have

$$\mathbb{F} \cong \mathcal{P}_V(\mathbb{F}) \qquad \mathsf{unfold} : \mathbb{F} \xrightarrow{\cong} \mathcal{P}_V(\mathbb{F})$$

we can define

$$S \in T \quad \equiv \quad S \in \mathsf{unfold}(T).$$

Note that in this set theory, the universe V is a set!

- Using the monadic structure of the Vietoris construction, we can easily deduce that this structure satisfies the Empty Set, Union, Pairing, and Powerset axioms.

Set Theory In $\mathbb{F}$

We now consider $\mathbb{F}$ as a set-theoretic universe $(\mathbb{F}, \in, =)$. Since we have

$$\mathbb{F} \cong \mathcal{P}_V(\mathbb{F}) \qquad \mathsf{unfold} : \mathbb{F} \xrightarrow{\cong} \mathcal{P}_V(\mathbb{F})$$

we can define

$$S \in T \quad \equiv \quad S \in \mathsf{unfold}(T).$$

Note that in this set theory, the universe V is a set!

- Using the monadic structure of the Vietoris construction, we can easily deduce that this structure satisfies the Empty Set, Union, Pairing, and Powerset axioms.
- A suitable axiom of Infinity holds, since e.g. $x = \{\varnothing\} \cup \{\{y\} \mid y \in x\}$ has a (unique) solution in $\mathbb{F}$.

Set Theory In $\mathbb{F}$

We now consider $\mathbb{F}$ as a set-theoretic universe $(\mathbb{F}, \in, =)$. Since we have

$$\mathbb{F} \cong \mathcal{P}_V(\mathbb{F}) \qquad \mathsf{unfold} : \mathbb{F} \xrightarrow{\cong} \mathcal{P}_V(\mathbb{F})$$

we can define

$$S \in T \quad \equiv \quad S \in \mathsf{unfold}(T).$$

Note that in this set theory, the universe V is a set!

- Using the monadic structure of the Vietoris construction, we can easily deduce that this structure satisfies the Empty Set, Union, Pairing, and Powerset axioms.
- A suitable axiom of Infinity holds, since e.g. $x = \{\varnothing\} \cup \{\{y\} \mid y \in x\}$ has a (unique) solution in $\mathbb{F}$.
- Clearly, we cannot have full (classical) separation, since we have $V \in V$. We **do**, however, have **continuous** versions of Separation, Replacement, and Choice. Choice relies on standard topological results about **selection functions**. The continuity requirement for Separation enforces restrictions on the use of negation.

Set Theory In $\mathbb{F}$

We now consider $\mathbb{F}$ as a set-theoretic universe $(\mathbb{F}, \in, =)$. Since we have

$$\mathbb{F} \cong \mathcal{P}_V(\mathbb{F}) \qquad \mathsf{unfold} : \mathbb{F} \xrightarrow{\cong} \mathcal{P}_V(\mathbb{F})$$

we can define

$$S \in T \;\equiv\; S \in \mathsf{unfold}(T).$$

Note that in this set theory, the universe V is a set!

- Using the monadic structure of the Vietoris construction, we can easily deduce that this structure satisfies the Empty Set, Union, Pairing, and Powerset axioms.
- A suitable axiom of Infinity holds, since e.g. $x = \{\varnothing\} \cup \{\{y\} \mid y \in x\}$ has a (unique) solution in $\mathbb{F}$.
- Clearly, we cannot have full (classical) separation, since we have $V \in V$. We **do**, however, have **continuous** versions of Separation, Replacement, and Choice. Choice relies on standard topological results about **selection functions**. The continuity requirement for Separation enforces restrictions on the use of negation.

This set theory, and generalizations to "κ-finitary" universes, has been studied by Forti and Honsell.

In Conclusion

Precursors and Successors

Precursors and Successors

Some Precursors (among many):

- Scott information systems, intersection types and filter models, Martin-Löf, Plotkin, Kozen, . . .
- Mike Smyth, *Powerdomains and Predicate Transformers: A Topological View*, in ICALP 1983. Smyth's slogan: open sets "are" c.e. sets. My slogan: open sets are **finitely observable properties**.

Precursors and Successors

Some Precursors (among many):

- Scott information systems, intersection types and filter models, Martin-Löf, Plotkin, Kozen, . . .
- Mike Smyth, *Powerdomains and Predicate Transformers: A Topological View*, in ICALP 1983. Smyth's slogan: open sets "are" c.e. sets. My slogan: open sets are **finitely observable properties**.

Some Successors (among many):

- Thomas Jensen, Strictness Analysis in Logical Form
- Marcelo Bonsangue et al.
- Achim Jung and Drew Moshier
- Michael Huth, Marta Kwiatkowska et al.
- Coalgebraic Logic (e.g. Moss, Jacobs, Kurz, Venema et al.)
- Work on Quantales (Resende et al.)
- Work on Constructive Mathematics taking the observational point of view (e.g. Spitters)

References

These papers are all available from my web pages
`http://web.comlab.ox.ac.uk/oucl/work/samson.abramsky/`

- S. Abramsky, “Domain Theory snd the Logic of Observable Properties”, Ph.D. Thesis, University of London, 1988.
- S. Abramsky, “The Lazy λ-Calculus”, in *Research Topics in Functional Programming*, D. Turner, ed. (Addison Wesley) 1990, 65–117.
- S. Abramsky, “Domain Theory in Logical Form”, *Annals of Pure and Applied Logic* 51, (1991), 1–77.
- S. Abramsky, “A Domain Equation for Bisimulation”, *J. Information and Computation*, 92(2) (1991), 161–218.
- S. Abramsky and C.-H. L. Ong, “Full Abstraction in the Lazy λ-calculus”, *Information and Computation*, 105(2) (1993), 159–268.
- S. Abramsky, A Cook’s Tour of the Finitary Non-Well-Founded Sets, in *We Will Show Them: Essays in honour of Dov Gabbay*, edited by Sergei Artemov, Howard Barringer, Artur d’Avila Garcez, Luis C. Lamb and John Woods, College Publications, Vol. 1, 1–18, 2005.