Synthetic topology in Homotopy Type Theory for probabilistic programming

Martin Bidlingmaier   Florian Faissole   Bas Spitters

1907.10674

Monadic programming with effects

Moggi's computational λ-calculus
Kleisli category of a monad:

- $Obj(\mathcal{C}_T) = Obj(\mathcal{C})$;
- $\mathcal{C}_T(A, B) = \mathcal{C}(A, T(B))$.

Used for:
Partial functions: $X + \bot$
State: $(X \times S)^S$
Non-determinism: $\mathcal{P}(X)$
Discrete probabilities: $convex(X)$

Probability theory

- Classical probability: measures on σ-algebras *of sets*
  σ-algebra: collection closed under countable $\bigcup, \bigcap$
  measure: σ-additive map to $\mathbb{R}$.
- Giry monad:
  $X \mapsto Meas(X)$ is a monad...
  ...on measurable spaces,
  ...on subcategories of topological spaces or domains.

valuations restrict measures to open sets.

Problem 1: Meas is not CCC
Problem 2: Not a monad on Set

Use a synthetic approach

Plan

Plan:

- Develop a richer semantics using topos theory
- Synthetic topology and its models
- Probability theory using synthetic topology
- Use HoTT to formalize this

Both computable and topological semantics

Synthetic topology

Synthetic topology

Scott: Synthetic domain theory
Domains as sets in a topos (Hyland, Rosolini, ...)
By adding axioms to the topos we make a DSL for domains.
Synthetic topology
(Brouwer, ..., Escardo, Taylor, Vickers, Bauer, ..., Lešnik)
Every object carries a topology, all maps are continuous
Idea: Sierpinski space $\mathbb{S} = \left(\substack{\odot \\ \cdot}\right)$ classifies opens:

$$O(X) \cong X \to \mathbb{S}$$

Convenient category of/type theory for 'topological' spaces.
Synthetic (real) computability
semi-decidable truth values $\mathbb{S}$ classify semi-decidable subsets.

Common generalization based on abstract properties for $\mathbb{S} \subset \Omega$:
Dominance axiom: monos classified by $\mathbb{S}$ compose.

Synthetic topology

Ambient logic: predicative topos (hSets).

Assumption: free ω-cpo completions exist.

This follows from:

- QIITs [ADK16]
- countable choice
- impredicativity
- classical logic

The ω-cpo completion of 1 is a dominance.

More axioms for synthetic topology

Definition

A space X is metrizable if its intrisic topology, given by $X \to \mathbb{S}$, coincides with a metric topology.

The fan principle:
Fan: $2^{\mathbb{N}}$ is metrizable and compact

Intuitionistic, will be used for the synthetic Lebesgue measure.

Fix such a topos where every object comes with a topology.

Models for synthetic topology

Standard axioms for continuous computations:
Brouwer, Kleene-Vesley K_2-realizability (TTE)
Gives a realizability topos

CAC $\vdash \mathbb{S}$ is the set of increasing binary sequences modulo

$$\alpha \sim \beta \text{ iff there exists } n,\ \alpha n = \beta n = 1.$$

Big Topos

Topological site:
A category of topological spaces closed under open inclusions
Covering by jointly epi open inclusions
Big topos: sheaves over such a site
$\mathbb{S}$ is Yoneda of the Sierpinski space

Fourman: Model for intuitionism: all maps are continuous

Convenient category: Nice category vs nice objects

Valuation monad

Valuations and Lower integrals

Lower Reals:

$r : \mathbb{R}_l := \mathbb{Q} \to \mathbb{S}$

$\forall p, r(p) \iff \exists q, (p < q) \wedge r(q).$

$\rightsquigarrow$ lower semi-continuous topology.

Dedekind Reals:

$$\mathbb{R}_D \subset \underbrace{(\mathbb{Q} \to \mathbb{S})}_{lower\ real} \times \underbrace{(\mathbb{Q} \to \mathbb{S})}_{upper\ real}$$

Valuations:

Valuations on $A : Set$:

$Val(A) = (A \to \mathbb{S}) \to \mathbb{R}_l^+$

- $\mu(\emptyset) = 0$
- Modularity
- Monotonicity
- ω-continuity

Integrals:

Positive integrals:

$Int^+(A) = (A \to \mathbb{R}_D^+) \to \mathbb{R}_D^+$

- $\int(\lambda x.0) = 0$
- Additivity
- Monotonicity
- ω-continuity

Riesz theorem: homeomorphism between integrals and valuations.

Constructive proof (Coquand/S): A regular compact locale.

Valuations and Lower integrals

Lower Reals:

$r : \mathbb{R}_l := \mathbb{Q} \to \mathbb{S}$

$\forall p, r(p) \iff \exists q, (p < q) \wedge r(q).$

$\rightsquigarrow$ lower semi-continuous topology.

Dedekind Reals:

$$\mathbb{R}_D \subset \underbrace{(\mathbb{Q} \to \mathbb{S})}_{lower\ real} \times \underbrace{(\mathbb{Q} \to \mathbb{S})}_{upper\ real}$$

Valuations:

Valuations on $A : Set$:

$Val(A) = (A \to \mathbb{S}) \to \mathbb{R}_l^+$

- $\mu(\emptyset) = 0$
- Modularity
- Monotonicity
- ω-continuity

Lower integrals:

Positive integrals:

$Int^+(A) = (A \to \mathbb{R}_l^+) \to \mathbb{R}_l^+$

- $\int(\lambda x.0) = 0$
- Additivity
- Monotonicity
- ω-continuity

Riesz theorem: homeomorphism between integrals and valuations.

Constructive proof by Vickers: A locale. Here: synthetically.

Analysis based on $\mathbb{S}$

HoTT book: 'one experiment with QIITs is enough. . . '

We've done the experiment:
We've learned:

- the lower reals are the ω-cpo completion of $\mathbb{Q}$
- avoid countable choice by indexing by $\mathbb{S}$
- similarity with geometric reasoning (open power set, no choice)

Lebesgue valuation

Fubini: the monad is (almost) commutative

So far, classically, ω-supported discrete valuations.

To construct the Lebesgue valuation we use
the fan principle: the locale 2^ω is spatial.

Probabilistic programming

Monadic semantics

Kleisli category:

Giry monad: (space) $\rightsquigarrow$ (space of its valuations):

- functor $\mathcal{M} : Space \to Space$.
- unit operator $\eta_x = \delta_x$ (Dirac)
- bind operator $(\mu >>= M)(f) = \underline{\int_\mu} \lambda x.(Mx)f$.

$$(>>=) :: \mathcal{M}A \to (A \to \mathcal{M}B) \to \mathcal{M}B.$$

Function types

To interpret the full computational λ-calculus we need T-exponents $(A \to TB)$ as objects.
The standard Giry monads do not support this.
hSet is cartesian closed, so we obtain a higher order language.

Moreover, the Kleisli category is ω-cpo enriched (subprobability valuations), so we can interpret PCF with fix [Plotkin-Power].

Rich semantics for a programming language.

Unfolding

Huang developed an efficient compiled higher order probabilistic programming language: augur/v2

Semantics in topological domains
(domains with computability structure)

Theorem (Huang/Morrisett/S)

Markov's Principle $\vdash$
The interpretation of the monadic calculus in the K2-realizability topos gives the same interpretation as in topological domains.

Finally: HoTT. . .

Type theory

Formalizing this construction in homotopy type theory.

- Correctness, proof assistant for continuous probabilistic programs
- Programming language with an expressive type system
- Potentially: type theory based on K2 (as in PrI)

Discrete probabilities : ALEA library

ALEA library (Audebaud, Paulin-Mohring) basis for CertiCrypt

- Discrete measure theory in Coq;
- Monadic approach (Giry, Jones/Plotkin, ...):
  - CPS: $\overbrace{\underbrace{(A \to [0,1])}_{\text{`}meas.\ functions'} \to [0,1]}^{\text{`}measures'}$
  - submonad: monotonicity, summability, linearity.

Example: flip coin : $Mbool$

$$\lambda\ (f : bool \to [0,1]).(0.5 \times f(true) + 0.5 \times f(false))$$

Univalent homotopy type theory

Coq lacks quotient types and functional extensionality.
ALEA uses setoids, $(T, \equiv)$. ('exact completion')

Use Univalent homotopy type theory as an internal type theory for a generalization of setoids, groupoids, ...

We use Coq's HoTT library.
(CPP: Bauer, Gross, Lumsdaine, Shulman, Sozeau, Spitters)

Toposes and types

How to formalize toposes in type theory?
Rijke/S: hSets in HoTT form a (predicative) topos:
large power objects.

Shulman: HoTT can interpreted in higher toposes.
Here: higher topos over a topological site.
hSets coincide with the 1-topos

Constructive model: Cubical stacks (Coquand)
Cubical assemblies (Uemura)...
...However, hSet logic is different from the 1-topos

HoTT for predicative constructive maths without countable choice.

Implementation in HoTT/Coq

Our basis: Cauchy reals in HoTT as QIIT (book, Gilbert)

- HoTTClasses: like MathClasses but for HoTT
- Experimental Induction-Recursion branch by Sozeau

Partiality (ADK): Construction in HoTT:
free ω-cpo completion as a higher inductive inductive type:

$$A_\perp : hSet \qquad \perp : A_\perp \qquad \eta : A \to A_\perp$$

$$\subseteq_{A_\perp} : A_\perp \to A_\perp \to Type$$

$$\bigcup : \prod_{f:\mathbb{N}\to A_\perp} \left(\prod_{n:\mathbb{N}} f(n) \subseteq_{A_\perp} f(n+1)\right) \to A_\perp$$

$\subseteq$ must satisfy the expected relations.

$\mathbb{S}$:=Partial(1).

Higher order probabilistic computation (Related work)

Compare: Top is not Cartesian closed.
1. Define a convenient super category. E.g. quasi-topological spaces: concrete sheaves over compact Hausdorff spaces.
This is a quasi-topos which models synthetic topology.
Even: big topos
2. Add probabilities inside this setting.

Staton, Yang, Heunen, Kammar, Wood model for higher order probabilistic programming has the same ingredients (but in opposite direction):
1. Standard Giry model for probabilistic computation
2. Obtain higher order by (a tailored) Yoneda

Conclusions

- Probabilistic computation with continuous data types
- Formalization in HoTT
- Experiment with synthetic topology in HoTT
- Extension of the Giry monad from locales to synthetic topology
- Model for higher order probabilistic computation: Augur/v2