Synthetic Spectra via a Monadic and Comonadic Modality

Mitchell Riley¹
jww. Dan Licata¹
Eric Finster²

Wesleyan University¹
University of Birmingham²

18th June 2020
Recall:

**Definition**

- A pointed type is a pair of $A : \text{Type}$ and $a : A$.
- A pointed function $(A, a) \rightarrow (B, b)$ is a function $f : A \rightarrow B$ and path $p : f(a) = b$.

Carrying these paths $p$ through constructions can be tedious.

We might prefer to talk about functions that preserve the point *strictly*. But we cannot arrange this in ordinary type theory.
Spectra

Definition

▶ A *prespectrum* $E$ is a sequence of pointed types $E : \mathbb{N} \to \text{Type}_*$ together with pointed maps $\alpha_n : E_n \to_\ast \Omega E_{n+1}$.

▶ A *spectrum* is a prespectrum such that the $\alpha_n$ are pointed equivalences.

Examples

▶ Each abelian group $G$ yields a spectrum with $E_n \equiv K(G, n)$, the ‘Eilenberg-MacLane spaces’.
▶ The *zero spectrum* with $E_n \equiv 1$.
▶ The *sphere prespectrum* has $E_n \equiv S^n$, with $\alpha_n$ the transpose of $\Sigma S^n \to_\ast S^{n+1}$.
**Definition**
A map of spectra $f : E \to F$ is a sequence of pointed maps $f_n : E_n \to \ast F_n$ that commute with the structure maps of $E$ and $F$.

Not many operations on spectra have been defined in type theory!
Can we find a model where functions automatically respect the point?

Pointed spaces or spectra don’t form a good model of type theory.

Space indexed families of pointed spaces/spectra do!
Parameterised Pointed Spaces

Definition
A *parameterised pointed space* is a space-indexed family of pointed spaces.

Theorem
The $\infty$-category of parameterised pointed spaces, $\mathcal{P}S_\star$, is an $\infty$-topos.
Parameterised Spectra

Definition
A parameterised spectrum is a space-indexed family of spectra.

Theorem (Joyal 2008, jww. Biedermann)
The $\infty$-category of parameterised spectra, $\mathcal{P}\text{Spec}$, is an $\infty$-topos.
Types As

HoTT
Types as $\infty$-groupoids.

In This Talk
Types as $\infty$-groupoids indexing a family of pointed things.

Spatial Type Theory (Shulman 2018)
Types as $\infty$-groupoids equipped with additional topological structure.
For every parameterised family, there is an operation that forgets the family.

And given a space, we can equip it with the trivial family.
Underlying Space

As a diagram of categories:

\[
\begin{array}{c}
\text{PC} \\
\downarrow 0 \\
S \\
\end{array}
\]

Let $\sharp$ be the round-trip on $PC$, this is an idempotent monad and comonad that is adjoint to itself.

**Goal:**
We want an extension of HoTT with a type former $\sharp$ that captures this situation.
Review: Spatial Type Theory

The □ Modality

Axioms

A Synthetic Smash Product
Review: Spatial Type Theory

The ⊤ Modality

Axioms

A Synthetic Smash Product
Spatial Type Theory

The $♭/♯$ fragment of cohesive type theory (Shulman 2018).

The intended models are ‘local toposes’:

$$
\begin{array}{ccc}
\mathcal{E} & \overset{♭}{\to} & \mathcal{S} \\
\downarrow & & \downarrow \\
\Gamma & \overset{♭}{\to} & \text{CoDisc}
\end{array}
$$

with the outer functors fully faithful.

$♭ :\equiv \text{Disc} \circ \Gamma$ is a lex idempotent comonad,

$♯ :\equiv \text{CoDisc} \circ \Gamma$ is an idempotent monad,

with $♭ \dashv ♯$.

We want $♭$ and $♯$ as unary type formers in our theory.
Following the pattern of adjoint logic, we put in a judgemental version of $♭$ and have the type formers interact with it.

$$\Delta \mid \Gamma \vdash a : A \quad \text{corresponds to} \quad a : b\Delta \times \Gamma \rightarrow A$$

We need two variable rules:

**VAR**

$$\Delta \mid \Gamma, x : A, \Gamma' \vdash x : A$$

**VAR-CRISP**

$$\Delta, x :: A, \Delta' \mid \Gamma \vdash x : A$$

The second rule comes from the counit $bA \rightarrow A$. 
The unary type former $\#$ is supposed to be right adjoint to $♭$, so we make it adjoint to the judgemental context $♭$.

What does $♭$ do to contexts? Recall $\Delta \mid \Gamma$ means $♭\Delta \times \Gamma$.

$$♭(♭\Delta \times \Gamma) \cong b♭\Delta \times b\Gamma \cong b\Delta \times b\Gamma \cong b(\Delta \times \Gamma)$$

So applying $♭$ to $\Delta \mid \Gamma$ gives $\Delta, \Gamma \mid \cdot$.

### $\#$-FORM

$$\Delta, \Gamma \mid \cdot \vdash A \text{ type} \quad \frac{\Delta \mid \Gamma \vdash \#A \text{ type}}{\Delta \mid \Gamma \vdash \#A}$$

### $\#$-INTRO

$$\Delta, \Gamma \mid \cdot \vdash a : A \quad \frac{\Delta \mid \Gamma \vdash a^\# : \#A}{\Delta \mid \Gamma \vdash a^\# : \#A}$$
First go:

\[\text{♯-ELIM-v1?}\]
\[
\Delta \mid \Gamma \vdash s : \#A \\
\hline \\
\Delta, \Gamma \mid \cdot \vdash s_\# : A
\]

Going from the conclusion to the premise, demoting \(\Gamma\) only makes it more difficult to use:

\[\text{♯-ELIM-v2?}\]
\[
\Delta \mid \cdot \vdash s : \#A \\
\hline \\
\Delta \mid \cdot \vdash s_\# : A
\]

Context in the conclusion should be fully general:

\[\text{♯-ELIM}\]
\[
\Delta \mid \cdot \vdash s : \#A \\
\hline \\
\Delta \mid \Gamma \vdash s_\# : A
\]
Review: Spatial Type Theory

The ▼ Modality

Axioms

A Synthetic Smash Product
Comparing the setting of spatial type theory with ours:

We could use Spatial Type Theory to study our setting on the right, if we impose that $♭A \to A \to ♯A$ is always an equivalence.

But transport across equivalence this would need to occur everywhere. We want a version that captures such a modality directly.
The Roundtrip

- The primary difficulty is that the structure maps include a non-trivial round trip $A \to \exists A \to A$.
- In Spatial Type Theory the counit was silent, not annotated in the term.

\[
\Delta, x :: A, \Delta' \mid \Gamma \vdash x : A
\]

At least one of the unit or counit has to be explicit.
- We chose to make the counit explicit, and the unit silent.
Our contexts again have two zones, where $\Delta \mid \Gamma$ morally means \(\|\Delta \times \Gamma\).

<table>
<thead>
<tr>
<th>VAR</th>
<th>VAR-ZERO</th>
<th>VAR-ROUNDTRIP</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Delta \mid \Gamma, x : A, \Gamma' \vdash x : A$</td>
<td>$\Delta, \underline{x} :: A, \Delta' \mid \Gamma \vdash \underline{x} : A$</td>
<td>$\Delta \mid \Gamma, x : A, \Gamma' \vdash x : A$</td>
</tr>
</tbody>
</table>

- **VAR-ZERO** corresponds to a use of the counit,
- **VAR-ROUNDTRIP** corresponds to the unit followed by the counit.
- With this convention, whenever $x : A$ is used via $\|A$, it is marked.
What does \( \llbracket \) do to contexts? Like last time:

\[
\llbracket (\llbracket \Delta \times \Gamma) \rrbracket \cong \llbracket \Delta \rrbracket \times \llbracket \Gamma \rrbracket \cong \llbracket \Delta \rrbracket \times \llbracket \Gamma \rrbracket \cong \llbracket (\Delta \times \Gamma) \rrbracket
\]

But we can’t write \( \Delta, \Gamma \mid \cdot \) exactly, because the counit is not silent! The types in \( \Gamma \) have to have all uses of other variables from \( \Gamma \) marked.

Let’s write \( \Delta, 0\Gamma \mid \cdot \) for this.

E.g.: \( x :: A \mid y : B, z : C(y) \) becomes \( x :: A, y :: B, z :: C(y) \mid \cdot \).
Precomposition with the structural rules can be extended to terms:

\[
\begin{align*}
\Delta | \Gamma \vdash a : A & \quad \Delta, 0\Gamma | \cdot \vdash a : A \\
\text{COUNIT} & \quad \text{UNIT} \\
\Delta, 0\Gamma | \cdot \vdash a : A & \quad \Delta | \Gamma \vdash a : A
\end{align*}
\]

When using \(x : A\) via the round-trip, also have to round-trip the type:

\[
\text{VAR-ROUNDTTRIP}
\]

\[
\Delta | \Gamma, x : A, \Gamma' \vdash x : A
\]
Here we don’t have to drop $\Gamma$ as we did with $\#$, instead we can precompose the result with the unit:

\[ \text{♭-ELIM-V1?} \]
\[
\Delta, 0\Gamma \vdash a : \text{♭} A \\
\Delta \vdash a_{\#} : A
\]
Rules for $\mathfrak{f}$

$\mathfrak{f}$-FORM

\[
\frac{\Delta, 0\Gamma \vdash A \text{ type}}{\Delta \vdash \Gamma \vdash \mathfrak{f}A \text{ type}}
\]

$\mathfrak{f}$-INTRO

\[
\frac{\Delta, 0\Gamma \vdash a : A}{\Delta \vdash \Gamma \vdash a^\mathfrak{f} : \mathfrak{f}A}
\]

$\mathfrak{f}$-ELIM

\[
\frac{\Delta \vdash \Gamma \vdash v : \mathfrak{f}A}{\Delta \vdash \Gamma \vdash v^\mathfrak{f} : A}
\]

$\mathfrak{f}$-BETA

\[
\frac{\Delta, 0\Gamma \vdash a : A}{\Delta \vdash \Gamma \vdash a^\mathfrak{f}^\mathfrak{f} \equiv a : A}
\]

$\mathfrak{f}$-ETA

\[
\frac{\Delta \vdash \Gamma \vdash v : \mathfrak{f}A}{\Delta \vdash \Gamma \vdash v \equiv v^\mathfrak{f}^\mathfrak{f} : \mathfrak{f}A}
\]
Properties of $\sharp$

- $\sharp$ is a lex monadic modality in the sense of the HoTT book, like $\#$.
- $\sharp$ is also comonadic, like $\flat$.
- $\sharp$ is self-adjoint: $\sharp(\sharp A \to B) \simeq \sharp(A \to \sharp B)$

Definition

- A type $X$ is a *space* if $(\lambda x.x^{\sharp}) : X \to \sharp X$ is an equivalence.
- A type $E$ is a *spectrum* if $\sharp E$ is contractible.

(To be more model agnostic you might call these ‘modal’ and ‘reduced’)
Proposition

For any $A$, the type $\mathbb{A}$ is a space.

Proof.

We have to show that $(\lambda v. v : \mathbb{A}) : \mathbb{A} \rightarrow \mathbb{\mathbb{A}}$ is an equivalence. For an inverse, use the counit $(\lambda z. z : \mathbb{\mathbb{A}}) : \mathbb{\mathbb{A}} \rightarrow \mathbb{A}$.

In one direction:

$$z : \mathbb{A} \equiv z : \mathbb{\mathbb{A}} \equiv z$$

and in the other:

$$v : \mathbb{\mathbb{A}} \equiv v \equiv v : \mathbb{\mathbb{A}} \equiv v : \mathbb{\mathbb{A}} \equiv v.$$
Review: Spatial Type Theory

The □ Modality

Axioms

A Synthetic Smash Product
Our spectra don’t behave much like actual spectra yet.

**Axiom S**
For any ‘dull’ spectra $E$ and $F$, the wedge inclusion

$\iota_{E,F} : E \vee F \to E \times F$ is an equivalence.

(The ‘spectra’ don’t form a stable category in every slice, only in slices over spaces!)

**Proposition**

* A dull square of spectra is a pushout square iff it is a pullback square.

**Proposition**

* Dull spectra and dull maps between them are $\infty$-connected.
Normalisation

Fix a distinguished spectrum $\mathcal{S} : \text{Type}$.

We can use this to build an adjunction

\[
\begin{array}{c}
\Sigma^\infty \\
\Downarrow \\
\text{Space}_* \\
\Uparrow \\
\Omega^\infty
\end{array}
\begin{array}{c}
\Sigma^\infty X : \equiv X \land \mathcal{S} \\
\Omega^\infty E : \equiv \sharp(\mathcal{S} \to_\ast E)
\end{array}
\]

Definition

The homotopy groups of a spectrum $E$ are

\[
\pi_n^s(E) : \equiv \pi_n(\Omega^\infty E)
\]
Normalisation

In fact, this factors into a sequence of adjunctions:

\[
\begin{array}{cccc}
\text{Space}_* & \xrightarrow{\text{susp}} & \text{SeqPreSpec} & \xrightarrow{\text{spec}} & \text{SeqSpec} & \xrightarrow{L} & \text{Spec} \\
\quad & \xleftarrow{0th} & \quad & \xleftarrow{\iota} & \quad & \xleftarrow{R} & \\
\end{array}
\]

where SeqPreSpec and SeqSpec are the types of sequential prespectra and spectra described earlier.

\[
LJ \equiv \text{colim} (\Sigma^\infty J_0 \to \Omega \Sigma^\infty J_1 \to \Omega^2 \Sigma^\infty J_2 \to \ldots )
\]

\[
(RE)_n \equiv \Omega^\infty \Sigma^n E
\]

(The details of the SeqPreSpec \to SeqSpec adjunction have not yet been done in type theory)
Normalisation

Axiom N
The $L \dashv R$ adjunction between $\text{SeqSpec}$ and $\text{Spec}$ is a (dull) adjoint equivalence: $\text{Mor}(J, RE) \simeq \mathbb{T}(LJ \to_* E)$

Proposition
$\pi^s_n(S) \simeq \text{colim}_k \pi_{n+k}(S^k)$

Proof.

$$\pi^s_n(S) \equiv \pi_n(\Omega^\infty S) \simeq \pi_n(\Omega^\infty (S^0 \wedge S)) \simeq \pi_n(\Omega^\infty \Sigma^\infty S^0)$$
$$\simeq \pi_n(\text{colim}_k \Omega^k \Sigma^k S^0) \simeq \text{colim} \pi_n(\Omega^k \Sigma^k S^0)$$
$$\simeq \text{colim}_k \pi_{n+k}(\Sigma^k S^0) \simeq \text{colim}_k \pi_{n+k}(S^k)$$
Review: Spatial Type Theory

The \( \exists \) Modality

Axioms

A Synthetic Smash Product
For two types $A$ and $B$ there should be a type $A \otimes B$ corresponding to the ‘external smash product’.

This is a symmetric monoidal product with no additional structural rules. (i.e., no weakening or contraction)
We can take a cue from ‘bunched logics’, where there are two ways of combining contexts, an ordinary cartesian one and a linear one.

\[
\begin{array}{c}
\Gamma_1 \text{ ctx} \quad \Gamma_2 \text{ ctx} \\
\overline{\Gamma_1, \Gamma_2 \text{ ctx}} \\
\end{array}
\quad
\begin{array}{c}
\Gamma_1 \text{ ctx} \quad \Gamma_2 \text{ ctx} \\
\overline{(\Gamma_1)(\Gamma_2) \text{ ctx}} \\
\end{array}
\]

For the comma *only*, we have weakening and contraction as normal.
Smash and Dependency

▶ When does a ‘dependent external smash’ \((x : A) \otimes B(x)\) make sense?

▶ When \(B(x)\) only depends on the base space of \(x : A\), so when we have \((x : A) \otimes B(x)\).

▶ Having the modality first is critical for dependent smash to work!
Thank You!

- Described a human-usable type theory for a $\Box$ modality with the correct properties.
- Gave an axiom making synthetic spectra form a stable category, and another for ‘normalisation’ of $\mathcal{S}$.
- Hinted at how the smash type former will work.

Questions?