Linear Homotopy Type Theory

Mitchell Riley
Wesleyan University

jww. Dan Licata
Wesleyan University

20$^{\text{th}}$ Jan 2022

Intended Models

Space-parameterised families of Spectra

Or more generally:

$\mathcal{X}$-parameterised families of $\mathcal{C}$

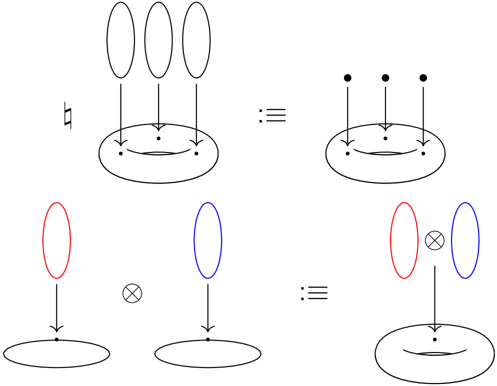
where

- $\mathcal{X}$ is an ∞-topos,
- $\mathcal{C}$ is a symmetric monoidal closed ∞-category *with a zero object*.

(A $\mathcal{C}$ for which $\mathcal{X}$-parameterised families form an ∞-topos is called an '∞-locus', Hoyois 2019)

Every object has a nonlinear aspect and a linear aspect.

Intended Models



- $\natural$: Extracts the nonlinear aspect of a type,
  - (R., Finster, and Licata 2021)
- $\otimes$: ‘Fibrewise’ tensor product
- $\mathbb{S}$: Unit of $\otimes$,
- $\multimap$: Right adjoint to $\otimes$.

Eg. (Co)homology

The *homology and cohomology of X with coefficients in E* can be defined by

$$E_n(X) :\equiv \pi_n^s(\Sigma^\infty(X) \otimes E)$$
$$E^n(X) :\equiv \pi_n^s(\Sigma^\infty(X) \multimap E)$$

where

$$\pi_n^s(E) :\equiv \natural(\mathbb{S} \to E)$$
$$\Sigma^\infty(X) :\equiv X \wedge \mathbb{S}$$

New Type Formers

We want the output of the type formers to be *ordinary types*.

Cannot use an indexed type theory (Vákár 2014; Krishnaswami, Pradic, and Benton 2015; Isaev 2021), or quantitative type theory (McBride 2016; Atkey 2018; Moon, Eades III, and Orchard 2021; Fu, Kishida, and Selinger 2020)

The Symmetry Proof We Want

Proposition

sym : $A \otimes B \simeq B \otimes A$

Proof.

To define sym : $A \otimes B \to B \otimes A$, suppose we have $p : A \otimes B$. Then $\otimes$-induction allows us to assume $p \equiv x \otimes y$, and we have $y \otimes x$.

$$\mathsf{sym} :\equiv \lambda p.\mathsf{let}\ x \otimes y = p\ \mathsf{in}\ y \otimes x$$

Then to prove $\prod_{(p:A \otimes B)} \mathsf{sym}(\mathsf{sym}(p)) = p$, use $\otimes$-induction again: the goal reduces to $x \otimes y = x \otimes y$ for which we have reflexivity.

$$\mathsf{inv} :\equiv \lambda p.\mathsf{let}\ x \otimes y = p\ \mathsf{in}\ \mathsf{refl}_{x \otimes y}$$

□

Colourful Variables

We need to prevent terms like $\lambda x.x \otimes x : A \to A \otimes A$, so variable use needs to be restricted somehow.

- Every variable x has a *colour* $\mathfrak{c}$.
- The relationships between colours are collected in a *palette*.

Palettes Φ are constructed by

$$1 \qquad \Phi_1 \otimes \Phi_2 \qquad \Phi_1, \Phi_2 \qquad \mathfrak{c} \qquad \mathfrak{c} \prec \Phi$$

Typical palettes:

$$\mathfrak{p} \prec \mathfrak{r} \otimes \mathfrak{b} \qquad \mathfrak{w} \prec (\mathfrak{p} \prec \mathfrak{r} \otimes \mathfrak{b}) \otimes \mathfrak{y} \qquad \mathfrak{p} \prec (\mathfrak{r} \otimes \mathfrak{b}, \mathfrak{r}' \otimes \mathfrak{b}')$$

(Similar to 'bunched' type theory P. W. O'Hearn and Pym 1999; P. O'Hearn 2003)

Using Colourful Variables

Building a term, we need to keep track of the current 'top colour'. Suppose the palette is $\mathfrak{p} \prec \mathfrak{r} \otimes \mathfrak{b}$, and we have variables

$$x^{\mathfrak{r}} : A, y^{\mathfrak{b}} : B, z^{\mathfrak{p}} : C.$$

- The top colour here is $\mathfrak{p}$.
- The only variable that can be used currently is $z : C$. (Using x here would correspond to a projection from one side of a tensor.)
- Ordinary type formers bind variables with the current top colour:

$$\textstyle\sum_{(x:A)} B(x) \qquad \prod_{(x:A)} B(x) \qquad (\lambda x.b)$$

$$\mathrm{ind}_{+}(z.C, x.c_1, y.c_2, p) \qquad \mathrm{ind}_{=}(x.x'.p.C, x.c, p)$$

- The rules for $\otimes$ will grant us access to the other variables.

Rules for ⊗, Take 1

Let $\mathfrak{p}$ be the top colour.

- **Formation:** For closed* $A : \mathcal{U}$ and $B : \mathcal{U}$, there is a type $A \otimes B : \mathcal{U}$.
- **Introduction:** In palette* $\mathfrak{p} \prec \mathfrak{r} \otimes \mathfrak{b}$, for any terms $a : A$ with colour $\mathfrak{r}$ and $b : B$ with colour $\mathfrak{b}$, there is a term

$$a \,{}_{\mathfrak{r}}\!\otimes_{\mathfrak{b}}\, b : A \otimes B$$

- **Elimination:** Any term $p : A \otimes B$ may be assumed to be of the form $x \,{}_{\mathfrak{r}}\!\otimes_{\mathfrak{b}}\, y$ for some variables $x^{\mathfrak{r}} : A, y^{\mathfrak{b}} : B$ with $\mathfrak{p} \prec \mathfrak{r} \otimes \mathfrak{b}$, in a term $c : C[x \,{}_{\mathfrak{r}}\!\otimes_{\mathfrak{b}}\, y/z]$.

$$(\text{let } x \,{}_{\mathfrak{r}}\!\otimes_{\mathfrak{b}}\, y = p \text{ in } c) : C[p/z]$$

- **Computation:** If the term really is of the form $a \,{}_{\mathfrak{r}'}\!\otimes_{\mathfrak{b}'}\, b$, then

$$(\text{let } x \,{}_{\mathfrak{r}}\!\otimes_{\mathfrak{b}}\, y = a \,{}_{\mathfrak{r}'}\!\otimes_{\mathfrak{b}'}\, b \text{ in } c) \equiv c[\mathfrak{r}'/\mathfrak{r} \otimes \mathfrak{b}'/\mathfrak{b} \mid a/x, b/y]$$

Eg: Symmetry

Proposition

There is a function $\mathsf{sym} : A \otimes B \to B \otimes A$

Proof.

Suppose have $p : A \otimes B$. Then $\otimes$-induction on p gives $x^{\mathfrak{r}} : A$ and $y^{\mathfrak{b}} : B$, where $\mathfrak{p} \prec \mathfrak{r} \otimes \mathfrak{b}$.

We need to form a purple term of $B \otimes A$, so 'split $\mathfrak{p}$ into $\mathfrak{b}$ and $\mathfrak{r}$'. Then we can form $y\ {}_{\mathfrak{b}}{\otimes}_{\mathfrak{r}}\ x : B \otimes A$.

$$\mathsf{sym} :\equiv \lambda p.\mathsf{let}\ x\ {}_{\mathfrak{r}}{\otimes}_{\mathfrak{b}}\ y = p\ \mathsf{in}\ y\ {}_{\mathfrak{b}}{\otimes}_{\mathfrak{r}}\ x$$

□

But we don't have $\mathfrak{p} \prec \mathfrak{b} \otimes \mathfrak{r}$ literally, we need to build in the symmetric monoidal structure.

Palette Splits

Need a more general judgement for when the palette linearly splits into two monoidally combined pieces: $\Phi \vdash \vec{\mathfrak{r}} \mid \vec{\mathfrak{b}}$ split

Symmetry: In palette $\mathfrak{p} \prec \mathfrak{r} \otimes \mathfrak{b}$,

$$\mathfrak{b} \mid \mathfrak{r} \text{ split}$$

Associativity: In palette $\mathfrak{w} \prec (\mathfrak{p} \prec \mathfrak{r} \otimes \mathfrak{b}) \otimes \mathfrak{y}$,

$$\mathfrak{r} \mid (\mathfrak{b} \otimes \mathfrak{y}) \text{ split}$$

Cartesian weakening: In palette $\mathfrak{p} \prec (\mathfrak{r} \otimes \mathfrak{b}, \mathfrak{r}' \otimes \mathfrak{b}')$,

$$\mathfrak{r}' \mid \mathfrak{b}' \text{ split}$$

Rules for $\otimes$, Take 2

Let $\mathfrak{p}$ be the top colour.

- **Formation:** For closed* $A : \mathcal{U}$ and $B : \mathcal{U}$, there is a type $A \otimes B : \mathcal{U}$.
- **Introduction:** For any palette split $\vec{\mathfrak{r}} \mid \vec{\mathfrak{b}}$ and terms $a : A$ with colour $\vec{\mathfrak{r}}$ and $b : B$ with colour $\vec{\mathfrak{b}}$, there is a term

$$a \;{}_{\vec{\mathfrak{r}}}{\otimes}_{\vec{\mathfrak{b}}}\; b : A \otimes B$$

- **Elimination:** Any term $p : A \otimes B$ may be assumed to be of the form $x \;{}_{\mathfrak{r}}{\otimes}_{\mathfrak{b}}\; y$ for some variables $x^{\mathfrak{r}} : A, y^{\mathfrak{b}} : B$ with $\mathfrak{p} \prec \mathfrak{r} \otimes \mathfrak{b}$ in a term $c : C[x \;{}_{\mathfrak{r}}{\otimes}_{\mathfrak{b}}\; y/z]$.

$$(\text{let } x \;{}_{\mathfrak{r}}{\otimes}_{\mathfrak{b}}\; y = p \text{ in } c) : C[p/z]$$

- **Computation:** If the term really is of the form $a \;{}_{\vec{\mathfrak{r}'}}{\otimes}_{\vec{\mathfrak{b}'}}\; b$, then

$$(\text{let } x \;{}_{\mathfrak{r}}{\otimes}_{\mathfrak{b}}\; y = a \;{}_{\vec{\mathfrak{r}'}}{\otimes}_{\vec{\mathfrak{b}'}}\; b \text{ in } c) \equiv c[\vec{\mathfrak{r}'}/\mathfrak{r} \otimes \vec{\mathfrak{b}'}/\mathfrak{b} \mid a/x, b/y]$$

Eg: Uniqueness principle for $\otimes$

Proposition

If $C : A \otimes B \to \mathcal{U}$ is a type family and $f : \prod_{(p:A\otimes B)} C(p)$, then for any $p : A \otimes B$ we have

$$(\mathrm{let}\ x \otimes y = p\ \mathrm{in}\ f(x \otimes y)) = f(p)$$

Proof.

By $\otimes$-induction we may assume $p \equiv x' \otimes y'$. Our goal is now

$$(\mathrm{let}\ x \otimes y = x' \otimes y'\ \mathrm{in}\ f(x \otimes y)) = f(x' \otimes y')$$

Which by computation reduces to $f(x' \otimes y') = f(x' \otimes y')$, for which we have reflexivity. □

(Cannot state this in indexed type or quantitative type theories)

Dependency in $\otimes$

From last time:

- Any assumption $x : A$ can be used 'marked': $\underline{x} : \underline{A}$.
- A $\underline{x}$ usage ignores the 'linear aspect' of x.
- A term a is *dull* if all free variables in a are marked.

Then we can allow the following dependency in $\otimes$:

- If $A : \mathcal{U}$ and $B : \mathcal{U}$ are *dull* types then $A \otimes B : \mathcal{U}$.
- If $A : \mathcal{U}$ is a dull type and B is a dull type assuming $x : A$, then $\bigotimes_{(\underline{x}:A)} B : \mathcal{U}$.

Eg. Associativity

Like dependent associativity of $\times$,

$$\mathsf{assoc} : \left(\sum_{(x:A)} \sum_{(y:B(x))} C(x)(y)\right) \simeq \left(\sum_{(v:\sum_{(x:A)} B(x))} C(\mathsf{pr}_1 v)(\mathsf{pr}_2 v)\right)$$

There is dependent associativity of $\otimes$:

$$\mathsf{assoc} : \left(\bigcirc\!\!\!\!\!\!\sum_{(\underline{x}:A)} \bigcirc\!\!\!\!\!\!\sum_{(\underline{y}:B(\underline{x}))} C(\underline{x})(\underline{y})\right) \simeq \left(\bigcirc\!\!\!\!\!\!\sum_{(\underline{v}:\bigcirc\!\!\!\!\!\!\sum_{(\underline{x}:A)} B(\underline{x}))} \mathsf{let}\ x \otimes y = \underline{v}\ \mathsf{in}\ C(\underline{x})(\underline{y})\right)$$

Hom

$$\frac{\Gamma \times A \vdash B}{\Gamma \vdash A \to B}$$

$$\frac{\Gamma \otimes A \vdash B}{\Gamma \vdash A \multimap B}$$

Hom

$$\frac{\Gamma \times (x : A) \vdash b : B}{\Gamma \vdash \lambda x.b : \prod_{(x:A)} B}$$

$$\frac{\Gamma \otimes (y : A) \vdash b : B}{\Gamma \vdash \partial y.b : \bigcirc\!\!\!\!\prod_{(y:A)} B}$$

Hom

$$\frac{\mathfrak{r} \mid \Gamma, x^{\mathfrak{r}} : A \vdash b : B}{\mathfrak{r} \mid \Gamma \vdash \lambda x.b : \prod_{(x:A)} B}$$

$$\frac{\mathfrak{p} \prec \mathfrak{r} \otimes \mathfrak{b} \mid \Gamma, y^{\mathfrak{b}} : A \vdash b : B}{\mathfrak{r} \mid \Gamma \vdash \partial y.b : \textcircled{$\prod$}_{(y^{\mathfrak{b}}:A)} B}$$

Hom Extensionality

Axiom Homext

For any $f, g : \mathop{\bigcirc\!\!\!\!\!\!\prod}_{(x:\underline{A})} B\langle x\rangle$, the function

$$(f = g) \to \mathop{\bigcirc\!\!\!\!\!\!\prod}_{(x:\underline{A})} f\langle x\rangle = g\langle x\rangle$$

is an equivalence.

Theorem

Univalence implies hom extensionality.

Bigger Picture

Applications

- Formalising some arguments in synthetic homotopy theory: (Schreiber 2017, Section 5.5)
- Acting as a specification language for quantum circuits: (Fu, Kishida, Ross, et al. 2020; Fu, Kishida, and Selinger 2020)

Modal Type Theories

- **Specialised modal extensions of MLTT:** (Shulman 2018; Birkedal et al. 2020; Gratzer, Sterling, and Birkedal 2019; Zwanziger 2019; Bizjak et al. 2016)
- **MTT Framework:** Adjoint Modalities, Dependent Types, No Substructural Types (Gratzer, Kavvos, et al. 2020; Gratzer, Cavallo, et al. 2021)
- **Fibrational Framework:** Any Modalities, Non-dependent Types, Substructural Types (Licata and Shulman 2016; Licata, Shulman, and R. 2017)

Linear HoTT does not currently fit into either framework!

References I

Robert Atkey (2018). "Syntax and Semantics of Quantitative Type Theory". In: *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science*. DOI: 10.1145/3209108.3209189.

Lars Birkedal et al. (2020). "Modal dependent type theory and dependent right adjoints". In: *Mathematical Structures in Computer Science* 30.2. DOI: 10.1017/S0960129519000197.

Aleš Bizjak et al. (2016). "Guarded dependent type theory with coinductive types". In: *Foundations of software science and computation structures*. Vol. 9634. DOI: 10.1007/978-3-662-49630-5_2.

Peng Fu, Kohei Kishida, Neil J. Ross, et al. (2020). "A Tutorial Introduction to Quantum Circuit Programming in Dependently Typed Proto-Quipper". In: *Reversible Computation*. DOI: 10.1007/978-3-030-52482-1_9.

References II

Peng Fu, Kohei Kishida, and Peter Selinger (2020). "Linear Dependent Type Theory for Quantum Programming Languages: Extended Abstract". In: *Proceedings of the 35th Annual ACM/IEEE Symposium on Logic in Computer Science*. DOI: 10.1145/3373718.3394765.

Daniel Gratzer, Evan Cavallo, et al. (2021). "Modalities and Parametric Adjoints". URL: https://jozefg.github.io/papers/modalities-and-parametric-adjoints.pdf.

Daniel Gratzer, G. A. Kavvos, et al. (2020). "Multimodal Dependent Type Theory". In: *Proceedings of the 35th Annual ACM/IEEE Symposium on Logic in Computer Science*. DOI: 10.1145/3373718.3394736.

Daniel Gratzer, Jonathan Sterling, and Lars Birkedal (2019). "Implementing a modal dependent type theory". In: *Proceedings of the 24th ACM SIGPLAN International Conference on Functional Programming*. DOI: 10.1145/3341711.

References III

Marc Hoyois (2019). "Topoi of parametrized objects". In: *Theory and Applications of Categories* 34. URL: http://www.tac.mta.ca/tac/volumes/34/9/34-09abs.html.

Valery Isaev (2021). "Indexed type theories". In: *Mathematical Structures in Computer Science* 31.1. DOI: 10.1017/S0960129520000092.

Neelakantan R. Krishnaswami, Pierre Pradic, and Nick Benton (2015). "Integrating Linear and Dependent Types". In: *Proceedings of the 42nd Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*. DOI: 10.1145/2676726.2676969.

Daniel R. Licata and Michael Shulman (2016). "Adjoint Logic with a 2-Category of Modes". In: *Logical foundations of computer science*. Vol. 9537. DOI: 10.1007/978-3-319-27683-0_16. URL: http://dlicata.web.wesleyan.edu/pubs/ls15adjoint/ls15adjoint.pdf.

References IV

Daniel R. Licata, Michael Shulman, and R. (2017). "A Fibrational Framework for Substructural and Modal Logics". In: *2nd International Conference on Formal Structures for Computation and Deduction*. Vol. 84. URL: http://dlicata.web.wesleyan.edu/pubs/lsr17multi/lsr17multi-ex.pdf.

Conor McBride (2016). "I Got Plenty o' Nuttin'". In: *A list of successes that can change the world*. Vol. 9600. DOI: 10.1007/978-3-319-30936-1_12.

Benjamin Moon, Harley Eades III, and Dominic Orchard (2021). "Graded Modal Dependent Type Theory". In: *Programming Languages and Systems*. DOI: 10.1007/978-3-030-72019-3_17.

Peter O'Hearn (2003). "On bunched typing". In: *Journal of Functional Programming* 13.4. DOI: 10.1017/S0956796802004495.

Peter W. O'Hearn and David J. Pym (1999). "The Logic of Bunched Implications". In: *Bulletin of Symbolic Logic* 5.2. DOI: 10.2307/421090.

References V

R., Eric Finster, and Daniel R. Licata (2021). *Synthetic Spectra via a Monadic and Comonadic Modality*. arXiv: 2102.04099 [math.CT].

Urs Schreiber (2017). *Differential cohomology in a cohesive infinity-topos*. URL: https://ncatlab.org/schreiber/files/dcct170811.pdf.

Michael Shulman (2018). "Brouwer's fixed-point theorem in real-cohesive homotopy type theory". In: *Mathematical Structures in Computer Science* 28.6. DOI: 10.1017/S0960129517000147.

Matthjis Vákár (2014). *Syntax and Semantics of Linear Dependent Types*. arXiv: 1405.0033 [cs.AT].

Colin Zwanziger (2019). "Natural Model Semantics for Comonadic and Adjoint Type Theory: Extended Abstract". In: *Preproceedings of the Applied Category Theory Conference 2019*. URL: http://www.cs.ox.ac.uk/ACT2019/preproceedings/Colin%20Zwanziger%20Natural%20Model%20Semantics%20for%20Comonadic%20and%20Adjoint%20Modal%20Type%20Theory.pdf.