A diagram model of linear dependent type theory

Martin Lundfall
Stockholm University
martin@dapp.org

Abstract

We present a type theory dealing with non-linear, "ordinary" dependent types (which we will call *cartesian*), and *linear types*, where both constructs may depend on terms of the former. In the interplay between these, we find the new type formers $\sqcap_{x:A}B$ and $\sqsubset_{x:A} B$, akin to Π and Σ, but where the dependent type B, (and therefore the resulting construct) is a linear type. These can be seen as internalizing universal and existential quantification over linear propositions, respectively. We also consider two modalities, M and L, transforming linear types into cartesian types and vice versa.

The theory is interpreted in a split comprehension category $\pi : \mathcal{T} \to \mathcal{C}^{\to}$ [4], accompanied by a split monoidal fibration, $q : \mathcal{L} \to \mathcal{C}$. We interpret $\mathcal{C}$ as a category of contexts, which for any $\Gamma \in \mathcal{C}$, determines the fibers $\mathcal{T}_\Gamma$ and $\mathcal{L}_\Gamma$. We interpret $\mathcal{T}_\Gamma$ as category of the cartesian types over Γ, and $\mathcal{L}_\Gamma$ as the monoidal category of linear types in Γ. In this setting, the type formers $\sqcap_{x:A}$ and $\sqsubset_{x:A}$ are understood as right and left adjoints of the monoidal reindexing functor $\pi_A^* : \mathcal{L}_\Gamma \to \mathcal{L}_{\Gamma.A}$ corresponding to the weakening projection $\pi_A : \Gamma.A \to \Gamma$ in $\mathcal{C}$. The operators M and L give rise to a fiberwise adjunction $L \dashv M$ between $\mathcal{L}$ and $\mathcal{T}$, where we understand the traditional exponential modality as the comonad $! = LM$.

We provide a model of this theory called the *Diagram model*, which extends the groupoid model of dependent type theory [3] to accomodate linear types. Here, cartesian types over a context Γ are interpreted as a family of groupoids indexed over the groupoid Γ, while linear types are interpreted as diagrams over groupoids, $A : \Gamma \to \mathcal{V}$ in any symmetric monoidal category $\mathcal{V}$. We show that the *diagrams model* can under certain conditions support a linear analogue of the univalence axiom, and provide some discussion on the higher-dimensional nature of linear dependent types.

2012 ACM Subject Classification Theory of computation $\to$ Linear logic; Theory of computation $\to$ Type theory

Keywords and phrases Dependent type theory, linear type theory, diagram model, monoidal categories, groupoid model

Digital Object Identifier 10.4230/LIPIcs.CVIT.2016.23

Related Version `http://kurser.math.su.se/pluginfile.php/16103/mod_folder/content/0/2017/2017_47_report.pdf`

Acknowledgements I want to thank Peter Lumsdaine for his constant guidance and inspiration.

Contents

42nd Conference on Very Important Topics (CVIT 2016).
Editors: John Q. Open and Joan R. Access; Article No. 23; pp. 23:1–23:19
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction & summary of results

Lately, there has been an increasing interest in combining linear and dependent types [12], [5], [13], [8]. The idea is that such a theory would inherit the higher-order nature of dependent types, while maintaining a careful account of how assumptions are used in a derivation. It is not completely clear, however, what the synthesis looks like, since in dependent type theory, variables may appear in both terms and types, but linear type theory only allows each variable to appear freely exactly once. Here, we take an approach inspired by [5] and [13], in which we distinguish between non-linear, dependent types (which we call *cartesian*), and linear types, and circumvent the issue by only allowing cartesian terms to appear in types (both cartesian and linear).

The theory splits contexts into two parts, divided by a semicolon, where the first part contains cartesian assumptions, for which weakening and contraction is admissible, while the second part contains linear assumptions, for which only exchange is allowed. We introduce two new type formers, $\sqcap_{x:A}B$ and $\sqsubset_{x:A} B$, akin to Π and Σ, but where the dependent type B (and therefore the resulting construct) is a linear. The traditional ! modality is deconstructed as a comonad arising from the adjoint pair $L \dashv M$, where L is a functor (or modality) sending cartesian types into linear, and M sends linear types to cartesian. We have $\Pi_{x:A}B_M \cong (\sqcap_{x:A}B)_M$, for linear B, and, assuming a few additional rules, a linear equivalence $(\Sigma_{x:A}C)_L \cong \sqsubset_{x:A} C_L$ for cartesian C.

Compared to ordinary depdenent type theory, we get additional elimination and computation rules for both Σ and Id-types when eliminating into a linear type.

We postulate the existence of two universes, L and U, containing codes of linear and cartesian types, respectively and assumed to be closed under all type formers.

We develop categorical semantics for the theory by defining a model as a *comprehension category* [4], $\pi : \mathcal{T} \to \mathcal{C}$ equipped with a *split monoidal fibration* $q : \mathcal{L} \to \mathcal{C}$ over the same base. A split monoidal fibration has just enough structure to make the fibers $\mathcal{L}_\Gamma$ over a context $\Gamma \in \mathcal{C}$ into monoidal categories, and reindexing functors (strict) monoidal functors. The traditional linear type formers $\&, \oplus, 0, \top, \multimap$ correspond to the existence of binary products and coproducts, initial and terminal object and internal homs in each fiber, such that these are preserved under reindexing. The new type formers $\sqcap_{x:A}B$ and $\sqsubset_{x:A} B$ correspond to right and left adjoints to the reindexing functor $\pi_A^* : \mathcal{L}_\Gamma \to \mathcal{L}_{\Gamma.A}$, while the modalities L and M give rise to a fiber adjunction between $\mathcal{L}$ and $\mathcal{T}$. The new rules for Σ are automatically satisfied by the semantic interpretation of Σ_A as a left adjoint to the reindexing functor $\pi_A^* : \mathcal{T}_\Gamma \to \mathcal{T}_{\Gamma.A}$. The new rules for Id-types impose an additional condition on the semantic interpretation of Id, which are always fulfilled if our identity types are extensional.

We consider two concrete models, the first being the *families model*, in which cartesian types consist of families of sets, indexed by their context set Γ, and a linear type in the context Γ is a Γ-indexed family of objects in a given symmetric monoidal category $\mathcal{V}$. Examples of suitable $\mathcal{V}$ supporting all type formers present in our syntax are **AbGrp**, $\mathbf{GCTop}_*$, $\mathbf{Vect}_F$, i.e. the category of abelian groups, the category of compact generated, pointed topological space and the category of vector spaces over a field F, respectively.

Generalizing the families model, we get the *diagrams model*, in which contexts are interpreted as groupoids, and cartesian types over a groupoid Γ are diagrams in **Gpd** over Γ, and linear types over Γ are diagrams in a given symmetric monoidal category $\mathcal{V}$ over Γ. Just as the groupoid model [3] can be shown to support a univalent universe, we construct a linear analogue of the univalence axiom and show that it holds in the diagrams model if the adjunction $L \dashv M$ factors through sets.

2 Syntax

As cartesian type formers, we use the standard Σ, Π, and identity type formers as well as universe types U and L for linear and cartesian types, respectively. The purely linear part of our type theory contains all the type formers of intuitionistic linear logic; the additive connectives $\&, \oplus, \mathbf{0}, \top$ and the multiplicatives $\otimes, \mathbf{1}, \multimap$. In addition to these, we have the new type formers $\sqsubset$, $\sqcap$, which play a role analogous to that of Σ and Π in the cartesian setting. Finally, we have the two modalities, M and L, which turns linear types into cartesian, and vice versa. A detailed presentation of our syntax can be found in [7]. For the familiar, "purely" dependent or linear type formers, our presentation offer no significant surprises, except for a couple of additional rules for Σ and the identity type (see figure 1). Therefore, we focus on presenting the syntax for the new type formers $\sqsubset$, $\sqcap$ and the modalities M and L.

2.1 Auxillary elimination rules

Besides the traditional rules for Π, Σ and the identity type, we find that since we can now eliminate into linear types, we must introduce an extra elimination and computational rule for each one. We present these in the following figure next to the usual elimination and computation rules, for comparison.

$$\frac{\begin{array}{c}\Gamma, t : \Sigma_{x:A}B \vdash C \ \text{ type} \\ \Gamma, x : A, y : B \vdash c : C[(x,y)/t] \\ \Gamma \vdash s : \Sigma_{x:A}B\end{array}}{\Gamma \vdash \hat{c}[s] : C[s/t]}\ \Sigma\text{–E}_1 \qquad \frac{\begin{array}{c}\Gamma, t : \Sigma_{x:A}B \vdash C \ \text{ linear} \\ \Gamma, x : A, y : B; \Xi \vdash c : C[(x,y)/t] \\ \Gamma \vdash s : \Sigma_{x:A}B\end{array}}{\Gamma; \Xi[pr_1(s)/x][pr_2(s)/y] \vdash \hat{c}[s] : C[s/t]}\ \Sigma\text{-E}_2$$

$$\frac{\Gamma \vdash \hat{c}[(a,b)] : C[(a,b)/t]}{\Gamma \vdash \hat{c}[(a,b)] \equiv c[(a,b)/t] : C[(a,b)/t]}\ \Sigma\text{-C}_1 \qquad \frac{\Gamma; \Xi \vdash \hat{c}[(a,b)] : C[(a,b)/t]}{\Gamma; \Xi \vdash \hat{c}[(a,b)] \equiv c[(a,b)/t] : C[(a,b)/t]}\ \Sigma\text{-C}_2$$

Figure 1 Elimination and computation rules for Σ

Similarly, for the identity type, in addition to the usual elimination rule:

$$\frac{\begin{array}{c}\Gamma, x, y : A, p : x =_A y \vdash C \;\text{ type} \\ \Gamma, z : A \vdash c : C[z/x, z/y, refl(z)/p] \\ \Gamma \vdash M : A \\ \Gamma \vdash N : A \\ \Gamma \vdash P : M =_A N\end{array}}{\Gamma \vdash R^{Id}_{[x,y,p]}(c, M, N, P) : C[M/x, N/y, P/p]} \;=\text{-E}_1$$

we may also eliminate into linear types:

$$\frac{\begin{array}{c}\vdash \Gamma, x, y : A, p : x =_A y; \Xi \;\text{ctxt} \\ \Gamma, x, y : A, p : x =_A y \vdash C \;\text{ linear} \\ \Gamma, z : A; \Xi[z/x, z/y, refl(z)/p] \vdash c : C[z/x, z/y, refl(z)/p] \\ \Gamma \vdash M : A \\ \Gamma \vdash N : A \\ \Gamma \vdash P : M =_A N\end{array}}{\Gamma; \Xi[M/x, N/y, P/p] \vdash R^{Id}_{[x,y,p]}(c, M, N, P) : C[M/x, N/y, P/p]} \;=\text{-E}_2$$

2.2 The modalities M and L

We introduce two the modal operators M and L, which transfers a linear type/term to its cartesian counterpart and vice versa. Semantically, this will establish a fiberwise monoidal adjunction between the categories of linear and cartesian types:

$$\begin{tikzcd} \mathcal{L}_\Gamma \arrow[r, bend right, "M"'] \arrow[r, phantom, "\perp"] & \mathcal{T}_\Gamma \arrow[l, bend right, "L"'] \end{tikzcd}$$

where the exponential modality from traditional linear logic is understood as the comonad $! = LM$. The decomposition of the exponential into an adjunction goes back to at least [1], and is given an interesting new light in [6], where it is seen as a particular case of a more general procedure of encoding structure in contexts.

Below are the syntactic rules for the operators M and L[1].

---

1 We have not introduced universes yet, but there is a subtlety involved with how they interact with the operators M and L, which may warrant restricting the operators M and L to only act on small types. See section **??**

$$\frac{\Gamma \vdash A \ \text{type}}{\Gamma \vdash A_L \ \text{linear}} \text{ L-F}$$

$$\frac{\Gamma \vdash a : A}{\Gamma; \cdot \vdash a_L : A_L} \text{ L-I}$$

$$\frac{\begin{array}{c}(\Gamma \vdash B \ \text{linear}) \\ (\vdash \Gamma; \Xi' \ \text{ctxt}) \\ \Gamma; \Xi \vdash y : A_L \quad \Gamma, x : A; \Xi' \vdash t : B\end{array}}{\Gamma; \Xi, \Xi' \vdash \text{let } x \text{ be } y \text{ in } t : B} \text{ L-E}$$

$$\frac{\Gamma; \Xi \vdash \text{let } x \text{ be } s_L \text{ in } t : B}{\Gamma; \Xi \vdash \text{let } x \text{ be } s_L \text{ in } t \equiv t[s/x] : B} \text{ L-C}$$

$$\frac{\begin{array}{c}\Gamma; y : A_L, \Xi \vdash t : B \\ (\Gamma, x : A; \Xi \vdash t[x_L/y] : B) \\ \Gamma; \Xi' \vdash a : A_L\end{array}}{\Gamma; \Xi, \Xi' \vdash \text{let } x \text{ be } a \text{ in } t[x_L/y] \equiv t[a/y] : B} \text{ L-U}$$

$$\frac{\Gamma \vdash B \ \text{linear}}{\Gamma \vdash B_M \ \text{type}} \text{ M-F}$$

$$\frac{\Gamma; \cdot \vdash b : B}{\Gamma \vdash b_M : B_M} \text{ M-I}$$

$$\frac{\Gamma \vdash t : B_M}{\Gamma; \cdot \vdash \sigma(t) : B} \text{ M-E}$$

$$\frac{\Gamma \vdash b_M : B_M}{\Gamma; \cdot \vdash \sigma(b_M) \equiv b : B} \text{ M-C}_1$$

$$\frac{\Gamma \vdash \sigma(t)_M : B_M}{\Gamma \vdash \sigma(t)_M \equiv t : B_M} \text{ M-C}_2$$

The interpretation of L and M as an adjoint pair is already present at the syntactic level. We can show that they form instances of a Haskell-like **Functor** class, by constructing terms: `fmapM` $: (A \multimap B)_M \to A_M \to B_M$ and `fmapL` $: L(A \to B) \multimap (LA \multimap LB)$, satisfying the functor laws.

Furthermore, we can construct a "counit" $\epsilon : LM \implies 1$ satisfying the universal property of adjunction (thanks to L-U). The syntactic formulation of the statement becomes:

▶ **Theorem 1** ($L \dashv M$). *There is a term $\Gamma; \beta_1 : B_{LM} \vdash \epsilon_B : B$ with the following property: For any term: $\Gamma; y : A_L \vdash f : B$, there is a unique term $\Gamma, x : A \vdash g : B_M$ such that $\Gamma; y : A_L \vdash \epsilon_B[\mathit{let}\ x\ \mathit{be}\ y\ \mathit{in}\ g_L/\beta_1] \equiv f : B$.*

Based on this knowledge we expect the right adjoint M to preserve limits, and indeed we find an isomorphism: $A_M \times B_M \cong (A \& B)_M$. We can now also reformulate some common results about the exponential modality as $! = LM$, such as $(A \& B)_{LM} \cong A_{LM} \otimes B_{LM}$.

2.3 $\sqcap$ and $\sqsubset$

Since we allow linear types to depend on terms of cartesian types, we can form new versions of the Π- and Σ-types, denoted $\sqcap$ and $\sqsubset$, respectively.

$$\frac{\Gamma \vdash A \ \text{type} \quad \Gamma, x : A \vdash B \ \text{linear}}{\Gamma \vdash \sqcap_{x:A} B \ \text{linear}} \ \sqcap\text{-F} \qquad \frac{\Gamma \vdash A \ \text{type} \quad \Gamma, x : A \vdash B \ \text{linear}}{\Gamma \vdash \sqsubset_{x:A} B \ \text{linear}} \ \sqsubset\text{-F}$$

$$\frac{\begin{array}{c}\vdash \Gamma; \Xi \ \text{ctxt} \\ \Gamma, x : A; \Xi \vdash b : B\end{array}}{\Gamma; \Xi \vdash \lambda x.b : \sqcap_{x:A} B} \ \sqcap\text{-I} \qquad \frac{\Gamma \vdash s : A \quad \Gamma; \Xi \vdash b : B[s/x]}{\Gamma; \Xi \vdash (s, b) : \sqsubset_{x:A} B} \ \sqsubset\text{-I}$$

$$\frac{\Gamma; \Xi \vdash t : \sqcap_{x:A} B \quad \Gamma \vdash a : A}{\Gamma; \Xi \vdash t(a) : B[a/x]} \ \sqcap\text{-E} \qquad \frac{\begin{array}{c}\vdash \Gamma; \Xi' \ \text{ctxt} \\ \Gamma \vdash C \ \text{linear} \\ \Gamma; \Xi \vdash t : \sqsubset_{x:A} B \quad \Gamma, x : A; \Xi', y : B \vdash c : C\end{array}}{\Gamma; \Xi, \Xi' \vdash \text{let } x, y \text{ be } t \text{ in c : C}} \ \sqsubset\text{-E}$$

$$\frac{\Gamma; \Xi \vdash \lambda x.b(a) : \sqcap_{x:A} B}{\Gamma; \Xi \vdash \lambda x.b(a) \equiv b[a/x] : B[a/x]} \ \sqcap\text{-C} \qquad \frac{\Gamma; \Xi \vdash \text{let } x, y \text{ be } (s, t) \text{ in c : C}}{\Gamma; \Xi \vdash \text{let } x, y \text{ be } (s, t) \text{ in c} \equiv c[s/x][t/y] : C} \ \sqsubset\text{-C}$$

The sense in which $\sqcap$ and $\sqsubset$ are "linear analogues" of Π and Σ can be formalized in the following way:

▶ Proposition 2. For all $\Gamma \vdash A$ type and $\Gamma, x : A \vdash B$ linear, there is an isomorphism:

$$\Pi_{x:A} B_M \cong (\sqcap_{x:A} B)_M$$

We would like to show a similar result relating Σ and $\sqsubset$, but for this we need a couple of additional rules. First, we assume the following uniqueness rules for Σ and $\sqsubset$ [2]:

$$\frac{\Gamma \vdash p : \Sigma_{x:A} B}{\Gamma \vdash (\text{pr}_1(p), \text{pr}_2(p)) \equiv p} \ \Sigma\text{-U}$$

$$\frac{\Gamma; \Xi \vdash \text{let } x, y \text{ be } t \text{ in } (x, y) : A}{\Gamma; \Xi \vdash \text{let } x, y \text{ be } t \text{ in } (x, y) \equiv t : A} \ \sqsubset\text{-U}$$

Second, we assume a kind of naturality rule for the L modality:

$$\frac{\begin{array}{c}\Gamma; \Xi, y : B \vdash e : C \\ \Gamma, x : A; \Xi' \vdash u : B \\ \Gamma; \Xi'' \vdash t : A_L\end{array}}{\Gamma; \Xi, \Xi', \Xi'' \vdash e[\text{let } x \text{ be } t \text{ in } u/y] \equiv \text{let } x \text{ be } t \text{ in } e[u/y] : C} \ \text{Nat}_L$$

▶ Proposition 3. Assuming the Nat_L and the uniqueness rules for Σ and $\sqsubset$, there is a linear isomorphism:

$$(\Sigma_{x:A} B)_L \cong \sqsubset_{x:A} B_L$$

As outlined in section 3.2, the semantic interpretation of the type formers Π, $\sqcap$ and Σ, $\sqsubset$ are as left and right adjoints to reindexing functors respectively, which can by lemma **??** be constructed via limits and colimits, these results reflect the fact that the left adjoint L preserves colimits and the right adjoints M preserves limits.

[2] The former is provable as a propositional identity, see for example [10, Corollary 2.7.3]. Perhaps it is possible to obtain a similar result for $\sqsubset$, using the "surrogate equality" described in the end of Section 2.2

3 Semantics

3.1 Structural semantic core

Our semantic exploration of linear dependent type theory begins with the notion of a model. For the cartesian fragment of our theory, we follow [4] and ask for a *comprehension category*, $\pi : \mathcal{T} \to \mathcal{C}^{\to}$, where $\mathcal{C}$ is a category of context with terminal object, and the fibrations $\mathcal{T}_\Gamma$ contains the cartesian types over Γ. For the linear fragment of our theory, we would like a fibration $q : \mathcal{L} \to \mathcal{C}$ where each fiber $\mathcal{L}_\Gamma$ is a symmetric monoidal category and the reindexing functors are symmetric monoidal. This is captured in the notion of a *(lax) monoidal fibration*:

▶ **Definition 4.** A **lax monoidal fibration** [14] is a fibration $p : E \to B$ along with

1. Two functors $\otimes : E \times_B E \to E$ and $I : B \to E$ fitting into the following diagram:

$$\begin{tikzcd}
E \times_B E \arrow[r, "\otimes"] \arrow[dr] & E \arrow[d, "p"] & B \arrow[l, "I"'] \arrow[dl, "1_B"] \\
& B &
\end{tikzcd}$$

.

2. Three fibred natural isomorphisms α, λ and ρ associated with the diagrams:

$$\begin{tikzcd}
E \times_B E \times_B E \arrow[r, "1_E \times_B \otimes"] \arrow[d, "\otimes \times_B 1_E"'] \arrow[dr, phantom, "\swarrow\alpha"] & E \times_B E \arrow[d, "\otimes"'] \\
E \times_B E \arrow[r, "\otimes"] & E
\end{tikzcd}$$

and

$$\begin{tikzcd}
B \times_B E \arrow[r, "I \times_B 1_E"] \arrow[dr, "\pi_2"'] & E \times_B E \arrow[d, "\otimes"] \arrow[dl, phantom, "\lambda"] \arrow[dr, phantom, "\rho"] & E \times_B B \arrow[l, "1_E \times I"'] \arrow[dl, "\pi_1"] \\
& E &
\end{tikzcd}$$

3. such that α, λ and ρ satisfies the pentagon and triangle identities in each fiber.
4. for every $b \in B$, $\rho_{I_b} = \lambda^{-1}_{I_b} : I_b \otimes I_b \to I_b$

To avoid any coherence problems, we require the both the comprehension category and the monoidal fibration to be *split*.

▶ **Definition 5.** A **model for linear dependent type theory** consists of a split comprehension category $\pi : \mathcal{T} \to \mathcal{C}^{\to}$ and a split symmetric monoidal fibration $q : \mathcal{L} \to \mathcal{C}$, as illustrated in the following picture:

$$\begin{tikzcd}
\mathcal{L} \arrow[dr, "q"'] & \mathcal{T} \arrow[r, "\pi"] \arrow[d, "p"] & \mathcal{C}^{\to} \arrow[dl, "\mathrm{cod}"] \\
& \mathcal{C} &
\end{tikzcd}$$

This provides the necessary machinery to interpret all the structural rules of our theory as well as the rules for $\otimes$ and I, by constructing an interpretation function $[[-]]$, which sends:

- Cartesian contexts Γ to objects of $\mathcal{C}$, considered up to definitional equality and renaming of bound variables.
- Linear contexts $\Xi = a_1 : A_1, a_2 : A_2, \ldots a_n : A_N$ in Γ to objects $[[\Xi]] = \bigotimes_{i=1}^{n} [[A_i]]$ of $\mathcal{L}_{[[\Gamma]]}$.
- Cartesian types A in Γ to objects of $\mathcal{T}_{[[\Gamma]]}$.
- Linear types B in Γ to objects of $\mathcal{L}_{[[\Gamma]]}$.
- Cartesian terms $M : A$ in Γ to sections of the projection morphism $\pi([[A]]) : [[\Gamma, A]] \to [[\Gamma]]$.
- Linear terms $b : B$ in $\Gamma; \Xi$ to morphisms $[[b]] : [[\Xi]] \to [[B]]$.

3.2 Semantic type formers

Equipped with the baseline structure of a model, in which we can interpret the structural rules of our theory, we formulate the conditions under which such models support various type formers. From now on, we will assume that the comprehension category comprising the core of our syntax is full, i.e. that the functor $\pi : \mathcal{T} \to \mathcal{C}^{\to}$ is full and faithful. This simplifies the semantic interpretation of many type formers.

The interpretation of the purely linear type formers $\otimes, I, \multimap, \&, \oplus, \top$ and 0 in symmetric monoidal categories is well known. See for instance [9]. Notice that $\otimes$ and I types are supported in any model. For a model to support the type formers $\multimap, \&, \oplus, \top$ and 0, correspond to the condition that the fibers of $\mathcal{L}$ have weak versions of internal homs, binary products and coproducts, and terminal and initial object, and that these are stable under reindexing functors.

3.2.1 Π and Σ

What it means for a model of linear dependent type theory to *support* Π-types is directly inherited from the standard, non-linear case; we require right adjoints to reindexing functors satisfying a Beck-Chevalley condition.

As the rules Σ contains one more eliminator than usual (Σ-E$_2$ in Figure 1), one might wonder whether this poses additional clauses in the definition of the semantic type former. But as it turns out, the relevant condition will always hold in any model supporting Σ-types:

▶ **Definition 6.** A model of LDTT **supports Σ-types** if it satisfies the following:
1. For all $A \in \mathcal{T}_\Gamma$, the induced functor $\pi_A^* : \mathcal{T}_\Gamma \to \mathcal{T}_{\Gamma.A}$ has a left adjoint, Σ_A,
2. such that for all pullbacks (as in **??**), these satisfy the Beck-Chevalley condition, i.e. the natural transformation: $\Sigma_E q^* \to f^* \Sigma_{E'}$ is a natural isomorphism, and
3. the induced map $pair_{A,B} : \Gamma.A.B \to \Gamma.\Sigma_A B$ is an isomorphism

This structure is sufficient to support new elimination rule (Σ-E$_2$):

▶ **Theorem 7.** *If a model of LDTT supports Σ-types, then for every object $C \in \mathcal{L}_{\Gamma.\Sigma_A B}$ and morphism $c : \Xi \to C\{pair_{A,B}\}$ in $\mathcal{L}_{\Gamma.A.B}$ and section $s : \Gamma \to \Gamma.\Sigma_A B$, there exists a morphism $\hat{c}_s : \Xi\{(pr_1, pr_2\} \to C\{s\}$ such that given sections $a : \Gamma \to \Gamma.A$ and $b : \Gamma.A \to \Gamma.A.B$, then $\hat{c}_{(a,b)} = c\{ba\} : \Xi\{ba\} \to C\{ba\}$.*

Proof. The situation is illustrated in the following diagram:

$$\begin{tikzcd}
\Gamma.A.B \arrow[r, "pair_{A,B}"'] \arrow[d, "\pi_{A,B}"] & \Gamma.\Sigma_A B \arrow[l, bend right, "(pr_1, pr_2)"'] \arrow[dl, "\pi_{\Sigma_A B}"] \\
\Gamma &
\end{tikzcd}$$

Let $\hat{c}_s = c\{((pr_1, pr_2)s)\}$. First, this morphism has the correct target since we have

$$C\{((pr_1, pr_2)s)\}\{pair_{A,B}\} = C\{(pair_{A,B}(pr_1, pr_2)s)\} = C\{s\}$$

, relying on the fact our lax monoidal fibration is split. Secondly, we need to show that given sections $a : \Gamma \to \Gamma.A$ and $b : \Gamma.A \to \Gamma.A.B$, we have $c\{((pr_1, pr_2)(a, b))\} = c\{a\}\{b\}$:

$$\{(pr_1, pr_2)(a, b)\} = \{(pr_1, pr_2)pair_{A,B}ba\} = \{ba\} = \{a\}\{b\}$$

◀

3.2.2 Identity types

When it comes to Id-types, the situation is not as fortunate. If one wants to keep the theory intensional, we need to add condition (2) to make sure that the semantic identity types satisfy the added elimination and computation rule, $=$-E$_2$ and $=$-C$_2$ in Figure **??**.

▶ **Definition 8** (Id-types). A model of LDTT **supports Id-types** if, for all $A \in \mathcal{T}_\Gamma$, there exists an object $Id_A \in \mathcal{T}_{\Gamma.A.A\{\pi_A\}}$ and a morphism $r_A : \Gamma.A \to \Gamma.A.A\{\pi_A\}.Id_A$ such that $\pi_{Id_A} \circ r_A = v_A$, and:

1. For any commutative diagram:

$$\begin{tikzcd} \Gamma.A \arrow[r] \arrow[d, "r_A"] & \Delta.C \arrow[d, "\pi_C"] \\ \Gamma.A.A\{\pi_A\}.Id_A \arrow[r] & \Delta \end{tikzcd}$$

there exists a lift $J : \Gamma.A.A\{\pi_A\}.Id_A \to \Delta.C$ making the two triangles commute.

2. For any pair of objects, $C, \Xi \in \mathcal{L}_{\Gamma.A.A^+.Id}$, sections $M, N : \Gamma \to \Gamma.A$, $P : \Gamma \to \Gamma.Id_A\{N^+\}\{M\}$, and morphism $c : \Xi\{r_A\} \to C\{r_A\}$, there exists a morphism $\hat{c}_{[M,N,P]} : \Xi\{P^+\}\{N^+\}\{M\} \to C\{P^+\}\{N^+\}\{M\}$ such that $\hat{c}_{[M,M,refl]} = c\{M\}$.

Notice that if our type theory has extensional id-types, in the sense that $a =_A b$ implies $a \equiv b$, then the third condition is always met.

3.2.3 $\sqcap$- and $\sqsubset$-types

The semantic type formers for the linear dependent $\sqcap$ and $\sqsubset$ is akin to that of Π and Σ. They are given by adjoints to the functors between fibers of $\mathcal{L}$ induced by the projection maps in $\mathcal{C}$.

▶ **Definition 9.** A model of LDTT **supports $\sqcap$-types** if, for all $A \in \mathcal{T}_\Gamma$, the induced monoidal functor $\pi_A^* : \mathcal{L}_\Gamma \to \mathcal{L}_{\Gamma.A}$ has a monoidal right adjoint, $\sqcap_A$ satisfying the following Beck-Chevalley condition:

For all pullback squares in $\mathcal{C}$ of the following form:

$$\begin{tikzcd} \Gamma.E \arrow[r, "q_{E,E'}"] \arrow[d, "\pi_E"] & \Delta.E' \arrow[d, "\pi_{E'}"] \\ \Gamma \arrow[r, "f"] & \Delta \end{tikzcd}$$

the canonical natural transformation $f^* \sqcap_{E'} \to \sqcap_E q^*_{E,E'}$ is a natural isomorphism.

▶ **Definition 10.** It **supports $\sqsubset$-types** if, for all $A \in \mathcal{T}_\Gamma$, the functor every π_A^* has a monoidal left adjoint, satisfying the following:

1. (Beck-Chevalley): For all pullbacks squares as above, the natural transformation $\sqsubset_E q^* \to f^* \sqsubset_{E'}$ is a natural isomorphism.
2. (Frobenius reciprocity): For all objects $\Xi \in \mathcal{L}_\Gamma$ and $B \in \mathcal{L}_{\Gamma.A}$, the canonical morphism $\sqsubset_A (\Xi\{\pi_A\} \otimes B) \to \Xi \otimes \sqsubset_A B$ is an isomorphism.

3.2.4 The operators M and L

▶ **Definition 11.** A model of LDTT with unit **supports the operators M and L** if there exists functors $M : \mathcal{L} \leftrightarrow \mathcal{T} : L$ which are cartesian with respect to the fibrations $p : \mathcal{T} \to \mathcal{C}$ and $q : \mathcal{L} \to \mathcal{C}$, such that $L \dashv M$ is a fibred adjunction and there is an isomorphism of hom-sets:

$$\mathcal{L}_{\Gamma.A}(\pi_A^*(\Xi'), \pi_A^*(B)) \cong \mathcal{L}_\Gamma(LA \otimes \Xi', B)$$

.

Recall that a fibred adjunction implies that there are natural isomorphisms making the following diagram commute:

$$\begin{tikzcd}
\mathcal{L}_\Gamma \arrow[r, bend right, "M_\Gamma"'] \arrow[d, "\pi_A^{*(\mathcal{L})}"] & \mathcal{T}_\Gamma \arrow[l, bend right, "L_\Gamma"'] \arrow[d, "\pi_A^{*(\mathcal{T})}"] \\
\mathcal{L}_{\Gamma.A} \arrow[r, bend right, "M_{\Gamma.A}"'] & \mathcal{T}_{\Gamma.A} \arrow[l, bend right, "L_{\Gamma.A}"']
\end{tikzcd}$$

which from a syntactic perspective ensures that M and L commute with substitution.

The final condition of the definition is what yields the elimination and computation rules L-U, and while it might appear somewhat unnatural semantically, it does turn out to hold automatically in a large class of models, due to the following result:

▶ **Theorem 12.** *In a model of LDTT that supports $\multimap$ type formers, then any fibred adjunction $L \dashv M$ where $L(1) \cong I$ satisfies $\mathcal{L}_{\Gamma.A}(\pi_A^*(\Xi'), \pi^*(B)) \cong \mathcal{L}_\Gamma(LA \otimes \Xi', B)$.*

Proof. A model supporting internal homs must have reindexing functions which preserve these. That is, we have an isomorphism $\pi_A^*[\Xi, B] \cong [\pi_A^*\Xi, \pi_A^*B]$. We get a chain of isomorphisms:

$$\mathcal{L}_\Gamma(LA \otimes \Xi, B) \cong \mathcal{L}_\Gamma(LA, [\Xi, B]) \cong \mathcal{T}_\Gamma(A, M_\Gamma[\Xi, B]) \cong$$
$$\mathcal{T}_{\Gamma.A}(1, \pi_A^*(M_\Gamma[\Xi, B])) \cong \mathcal{T}_{\Gamma.A}(1, M_{\Gamma.A}\pi_A^*[\Xi, B])) \cong \mathcal{L}_{\Gamma.A}(L_{\Gamma.A}(1), \pi_A^*[\Xi, B])) \cong$$
$$\mathcal{L}_{\Gamma.A}(I, \pi_A^*[\Xi, B])) \cong \mathcal{L}_{\Gamma.A}(I, [\pi_A^*\Xi, \pi_A^*B])) \cong \mathcal{L}_{\Gamma.A}(\pi_A^*\Xi, \pi_A^*B).$$

◀

4 Diagram Model

The main novelty of this paper is the Diagram model of linear dependent type theory. This model extends the groupoid model of dependent type theory [3] to support linear types, while still maintaining a higher dimensional interpretation of the identity type. Most interestingly, perhaps, it provides a model in which we can support univalent universes, both for cartesian and linear types. The diagram model can be seen as a natural generalization of the set indexed families model described by [13]. We briefly recall the set indexed families model below as a useful comparison to the diagrams model.

4.1 Set indexed families

▶ **Definition 13** ($Fam(\mathcal{C})$)**.** For an arbitrary category $\mathcal{C}$, let $Fam(\mathcal{C})$ denote the category whose objects consists of pairs (S, f) where S is a set and f is a function $f : S \to Ob(\mathcal{C})$. Morphisms of $Fam(\mathcal{C})$ are pairs $(u, \alpha) : (S, f) \to (S', g)$ where $u : S \to S'$ and $\alpha : S \to \mathrm{Mor}(\mathcal{C})$ such that $\alpha(s) : f(s) \to g(u(s))$ for all $s \in S$.

By projecting a family to its indexing set, we get a fibration $p : Fam(\mathcal{C}) \to \mathbf{Set}$ and a comprehension category by defining $\pi(S, f) = fst : \{(s, t) \mid s \in S, t : \top \to f(s)\} \to S$ [3]

Letting $\mathcal{C} = \mathbf{Set}$ thus gives us a (full, split) comprehension category, forming the cartesian part of our model. For the linear part, we can for any symmetric monoidal category $\mathcal{V}$ form a monoidal fibration by a simple pointwise construction, giving us the following picture:

$$\begin{tikzcd}
Fam(\mathcal{V}) \arrow[dr, "q"] & Fam(\mathbf{Set}) \arrow[r, "\pi"] \arrow[d, "p"] & \mathbf{Set}^{\to} \arrow[dl, "cod"] \\
 & \mathbf{Set} &
\end{tikzcd}$$

In this setting, most type formers will be given by a simple pointwise construction which are preserved under reindexing. It turns out that the families model supports the type formers $\otimes, I, \multimap, \oplus, 0, \&,$ and $\top$ if $\mathcal{V}$ is a monoidal category which is closed, has binary coproducts, initial object, binary products and terminal object respectively.

It supports $\sqcap$-types if $\mathcal{V}$ has small products, and $\sqsubset$ if $\mathcal{V}$ has small coproducts that distribute over $\otimes$ (Frobenius reciprocity).

The families model of course also supports Π and Σ-types, and since its identity types are extensional, the extra condition posed on our semantic identity types poses no additional difficulty.

Whenever $\mathcal{V}$ is a concrete category, the adjunction $F \dashv U$ will induce a fiber adjunction between the corresponding fibrations, which forms support for the operators M and L (as long as $F(\mathbf{1}) \cong I$).

4.2 Diagrams in monoidal categories

For any category $\mathcal{C}$, there is a fibration $cod : \mathbf{Diag}(\mathcal{C}) \to \mathbf{Cat}$, whereby $\mathbf{Diag}(\mathcal{C})$ refers to the category of diagrams in $\mathcal{C}$, i.e. consisting of objects $J : \mathcal{A} \to \mathcal{C}$, and morphisms between $J : \mathcal{A} \to \mathcal{C}$ and $J' : \mathcal{B} \to \mathcal{C}$ being functors $F : \mathcal{A} \to \mathcal{B}$ equipped with a natural transformation $J \implies J' \circ F$. In other words, the fibers of $\mathbf{Diag}(\mathcal{C})$ are functor categories, which we write

[3] As long as $\mathcal{C}$ has a terminal object and the hom-sets $\mathcal{C}(\top, A)$ are small for all $A \in \mathcal{C}$

$[\Gamma, \mathcal{C}]$, for any small category Γ. Any functor $F : \mathcal{A} \to \mathcal{B}$ in the base induces a canonical lift $F^* : [\mathcal{B}, \mathcal{C}] \to [\mathcal{A}, \mathcal{C}]$ simply given by precomposition.

Again, when $\mathcal{C}$ has a terminal object $\top$ such that the collections $\mathcal{C}(\top, A)$ are small for any $A \in \mathcal{C}$, then this forms a comprehension category,

$$\begin{tikzcd} \mathbf{Diag}(\mathcal{C}) \arrow[r, "\pi"] \arrow[d, "dom"] & \mathbf{Cat}^{\to} \arrow[dl, "cod"] \\ \mathbf{Cat} & \end{tikzcd}$$

where the functor π is defined as follows. Given any diagram $A : \Gamma \to \mathcal{C}$, we let the category $\Gamma.A$ be the **Grothendieck construction** for A, in other words, the category whose objects are pairs (γ, t_γ) where $\gamma \in \Gamma$ and $t_\gamma : \top \to A(\gamma)$. Morphisms $(\gamma, t_\gamma) \to (\gamma', t'_{\gamma'})$ consists of morphisms $u : \gamma \to \gamma'$ such that $A(u) \circ t_\gamma = t'_{\gamma'}$. If $\mathcal{C}$ is a 2-category, this can be weakened so that morphisms $(\gamma, t_\gamma) \to (\gamma', t'_{\gamma'})$ are pairs (u, α), where $u : \gamma \to \gamma'$ and α is a 2-cell $\alpha : A(u) \circ t_\gamma \implies t'_{\gamma'}$.

When $\mathcal{C}$ is any symmetric monoidal category $\mathcal{V}$, there is an obvious symmetric monoidal structure on each fiber $[\Gamma, \mathcal{V}]$, given pointwise. Again, reindexing functors are given by precomposition and therefore strictly monoidal. Restricting the base of the fibration to groupoids instead of categories, and setting $\mathcal{C} = \mathbf{Gpd}$ we get a model of linear dependent type theory which expands the groupoid model by Hofmann and Streicher [3]:

$$\begin{tikzcd} \mathbf{Diag}(\mathcal{V}) \arrow[dr, "dom"] & \mathbf{Diag}\,(\mathbf{Gpd}) \arrow[r, "\pi"] \arrow[d, "dom"] & \mathbf{Gpd}^{\to} \arrow[dl, "cod"] \\ & \mathbf{Gpd} & \end{tikzcd}$$

Since $\top$ is the groupoid $\mathbf{1}$ consisting of a single object, we will equate the functor $t_\gamma : 1 \to A(\gamma)$ with an object a_γ of $A(\gamma)$, and the natural transformation $\alpha : A(u) \circ t_\gamma \implies t'_{\gamma'}$ with a morphism $\alpha_\gamma : A(u)(a_\gamma) \to a'_{\gamma'}$.

As shown in [3], this model supports Π and Σ, and provides an important example of Id-types where there might be more terms of $x =_A x$ than $refl(x)$, setting the stage for the homotopy interpretation of dependent type theory. The groupoid model interpretation of the identity type Id_A is the functor $Id_A : \Gamma.A.A^+ \to \mathbf{Gpd}$ which sends an object $(\gamma, a_\gamma, b_\gamma)$ to the discrete groupoid $\delta\,\mathrm{Hom}_{A(\gamma)}(\gamma)(a, b)$, and a morphism $(u, \alpha, \beta) : (\gamma, a_\gamma, b_\gamma) \to (\gamma', a'_{\gamma'}, b'_{\gamma'})$ to the functor $\alpha^{-1} \circ A(u)(-) \circ \beta : \delta(\mathrm{Hom}_A(a, b)) \to \delta(\mathrm{Hom}_A(a', b'))$, which sends $f : a \to b$ to the composite $\alpha^{-1} \circ A(u)(f) \circ \beta : a' \to A(u)a \to A(u)b \to b'$.

We see that the extended context $\Gamma.A.A^+.Id_A$ is equivalent to $\Gamma.A^{\to}$, and we define $r_A : (\Gamma.A) \to (\Gamma.A^{\to})$ to be the functor sending morphisms:

$$(u, \alpha) : (\gamma, a_\gamma) \to (\gamma', a'_{\gamma'})$$

to squares:

$$\begin{tikzcd} (\gamma, a_\gamma) \arrow[r, "(u,\alpha)"] \arrow[d, "(1_\gamma, 1_{a_\gamma})"] & (\gamma', a'_{\gamma'}) \arrow[d, "(1_{\gamma'}, 1_{a'_{\gamma'}})"] \\ (\gamma, a_\gamma) \arrow[r, "(u,\alpha)"] & (\gamma', a'_{\gamma'}) \end{tikzcd}$$

This construction is sufficient to support identity types in our linear dependent setting.

▶ **Theorem 14.** *For any indexed groupoid A in Γ, the construction Id_A described above forms support for Id-types in the diagrams model.*

Proof. That Id_A satisfies condition (1) of Definition 8 is proved in [3]. It remains to show that condition (2) is satisfied by this construction. Let $C, \Xi \in \mathcal{L}_{\Gamma.A.A^+.Id_A}$, and assume we have sections $M, N : \Gamma \to \Gamma.A$ and $P : \Gamma \to \Gamma.Id_A\{M\}\{N^+\}$. We are given a natural transformation $c : \Xi\{r_A\} \to C\{r_A\}$, between between the two functors $\Xi \circ r_A : \Gamma.A \to \Gamma.A.A^+.Id_A \to \mathcal{V}$ and $C \circ r_A : \Gamma.A \to \Gamma.A.A^+.Id_A \to \mathcal{V}$ and need to display a natural transformation $\hat{c}_{[M,N,P]} : \Xi \circ P^+ \circ N^+ \circ M \implies C \circ P \circ N^+ \circ M$ such that $\hat{c}_{[M,M,refl(M)]} = c\{M\}$.

The key point to observe is that there is always an isomorphism $(\gamma, P_\gamma : M_\gamma \to N_\gamma) \cong (\gamma, 1_{M_\gamma} : M_\gamma \to M_\gamma)$ given by the commutative diagram:

$$\begin{tikzcd}
(\gamma, M_\gamma) \arrow[r, "1_M"] \arrow[d, "P"] & (\gamma, M_\gamma) \arrow[d, "1_M"] \\
(\gamma, N_\gamma) \arrow[r, "P^{-1}"] & (\gamma, M_\gamma)
\end{tikzcd}$$

in the groupoid $\Gamma.A^{\to}$, giving rise to a natural isomorphism $\phi : r_A \circ M \cong P^+ \circ N^+ \circ M$. We define $\hat{c}[M, N, P]$ as the composite:

$$\hat{c}[M, N, P] :\equiv \Xi_\phi \circ c\{M\} \circ C_{\phi^{-1}} : \Xi \circ P^+ \circ N^+ \circ M \to \Xi \circ r_A \circ M \to C \circ r_A \circ M \to C \circ P^+ \circ N^+ \circ M$$

◀

As in the families model, limits and colimits are constructed pointwise, and preserved by precomposition, so the model supports &, $\top$, $\oplus$, 0, if $\mathcal{V}$ has binary products, terminal object, binary coproducts and initial object respectively.

When it comes to $\multimap$, we utilize the following result:

▶ **Theorem 15.** *If $\mathcal{V}$ has internal homs and is complete, $[\mathcal{C}, \mathcal{V}]$ also has internal homs, defined for $F, G \in [\mathcal{C}, \mathcal{V}]$ by:*

$$[F, G] := \int_{x \in \mathcal{C}} [Fx, Gx])$$

These internal homs are preserved under reindexing. Given any functor $p : \mathcal{D} \to \mathcal{C}$ in the base, we have:

$$p^*([G, F]) = \lambda x. \int_{p(x) \in \mathcal{C}} [G(p(x)), F(p(x))]) = \int_{x \in \mathcal{D}} [G(p(x)), F(p(x))] = [p^* G, p^* F]$$

so we get that the diagrams model supports $\multimap$ if $\mathcal{V}$ is monoidal closed and complete.

▶ **Definition 16.** For any functor $p : \mathcal{A} \to \mathcal{B}$ in the base, a left or right adjoint to the induced functor $p^* : [\mathcal{B}, \mathcal{V}] \to [\mathcal{A}, \mathcal{V}]$ is called a **left or right Kan extension** of p.

Recall the following general fact about Kan extensions:

▶ **Theorem 17.** *Left (right) Kan extensions along $p : \mathcal{A} \to \mathcal{B}$ between two arbitrary small categories $\mathcal{A}$ and $\mathcal{B}$ exists if and only if $\mathcal{V}$ has all colimits (limits).*

The result above ensures the existence of left and right adjoints to reindexing functors in the diagrams model as long as $\mathcal{V}$ is co-complete or complete, respectively. But since our main concern are diagrams over groupoids, it suffices to consider the case where $\mathcal{V}$ has limits and colimits for these. Similarly to the families model, the fact that our reindexing functors are given by precomposition ensures that these always satisfy the Beck-Chevalley condition. In order to interpret M and L in this model, the same argument as in lemma **??** can be used to show that an adjunction between $\mathcal{V}$ and **Gpd** induces a fiber adjunction between the respective diagram categories. Therefore, for any diagrams model which supports $\multimap$ to support M and L, it suffices to display an adjunction

$$\begin{tikzcd} \mathcal{V} \arrow[rr, bend right, "L_0"'] & & \mathbf{Gpd} \arrow[ll, bend right, "M_0"'] \end{tikzcd}$$

such that $L(1) \cong I$.

▶ Remark. When $\mathcal{V}$ is a representable concrete category, we will often find that support for M and L comes "for free". Since the composition of two adjunctions is again an adjunction, we get that whenever the functor $\mathcal{V}(I, -) : \mathcal{V} \to \mathbf{Set}$ has a left adjoint F, then there exists an adjunction between $\mathbf{Diag}(\mathcal{V})$ and $\mathbf{Diag}(\mathbf{Gpd})$, arising out of:

$$\begin{tikzcd} \mathcal{V} \arrow[rr, bend right, "\mathcal{V}(I,-)"'] \arrow[rr, phantom, "\perp"] & & \mathbf{Set} \arrow[ll, bend right, "F"'] \arrow[rr, bend right, "\delta"'] \arrow[rr, phantom, "\perp"] & & \mathbf{Gpd} \arrow[ll, bend right, "\pi_0"'] \end{tikzcd}$$

where π_0 is the functor sending a groupoid to its set of connected components.

▶ **Theorem 18.** *There are models in which M is not faithful.*

Proof. Let $\mathcal{V}$ to be **Gpd** so that $L = \delta\pi_0$ and $M = \delta\mathbf{Gpd}(1, -)$. This induces a fiber adjunction $L \dashv M$ where $L(1) = 1$, but M is not faithful. ◀

4.2.1 Universes in the diagrams model

To support universes, assuming one inaccessible cardinal allows us to shift our perspective to from the category of small groupoids, **Gpd**, to the category **GPD** of all groupoids. Among the objects of **GPD** we find the large groupoid $\mathbf{Gpd}^{core}$, and $\mathcal{V}^{core}$, where $(-)^{core}$ denotes the core (i.e. maximal sub-groupoid) of a category. This allows us to define our cartesian and linear universes in any context Γ as the functors:

$$\mathbb{U} : \Gamma \to \mathbf{GPD}$$
$$\mathbb{L} : \Gamma \to \mathbf{GPD}$$

which are constant at $\mathbf{Gpd}^{core}$ and $\mathcal{V}^{core}$, respectively. Any section $s : \Gamma \to \Gamma.\mathbb{U}$ will determine a functor $\hat{s} : \Gamma \to \mathbf{Gpd}$, which will we can embed via the full subcategory embedding $\mathbf{Gpd} \to \mathbf{GPD}$ to get an interpretation of $El(s)$. Similarly, we get from each section $s : \Gamma \to \Gamma.\mathbb{L}$, a functor $El(s) : \Gamma \to \mathcal{V}$. Restricting the type formers $\sqcap$, $\sqsubset$, M and L to small types only, this linear universe is all-encompassing. It is easily seen that defining the linear universe via the core of $\mathcal{V}$ gives rise to the following interesting property, hinting at the possibility of a linear univalence axiom:

▶ **Corollary 19.** *For a linear universe defined as above via $\mathcal{V}^{core}$, and two sections $s, t : \Gamma \to \Gamma.\mathbb{L}$, an isomorphism $\alpha : El(t) \cong El(s)$ gives rise to a section $p : \Gamma \to \Gamma.Id_{\mathbb{L}}\{s\}\{t\}$.*

4.3 Univalence in linear dependent types

A key feature of the groupoid model is that it provides a model of dependent type theory where there might be nontrivial terms of the identity type. A natural question to ask is whether this higher dimensional feature of type theory can be extended to the linear dependent setting. In particular, we want to explore a model in which the following, linear analogue to the univalence axiom, would hold:

$$\frac{\begin{array}{c}\Gamma \vdash A : \mathbb{L} \\ \Gamma \vdash B : \mathbb{L} \\ \Gamma; \cdot \vdash f : El(A) \multimap El(B) \\ \Gamma; \cdot \vdash g : El(B) \multimap El(A) \\ \Gamma; \cdot \vdash h : El(B) \multimap El(A) \\ \Gamma \vdash p : (g \circ f)_M =_{(El(A) \multimap El(A))_M} (id_A)_M \\ \Gamma \vdash q : (f \circ h)_M =_{(El(B) \multimap El(B))_M} (id_B)_M\end{array}}{\Gamma \vdash ua(f)_{[g,h,p,q]} : A =_{\mathbb{L}} B} \text{ L-ua-I}$$

To define the corresponding computation rule, we will the make use of a linear version of `transport`, which is easily defineable through identity elimination. Given $\Gamma, x : C \vdash D$ linear, and an identity $p : a =_C b$, we get a function:

$$p^* : D[a/x] \multimap D[b/x]$$

which we call the linear **transport along p**. Taking $C \equiv \mathbb{L}$, and $D \equiv El(x)$ for any $x : \mathbb{L}$ in the procedure above yields, from any $q : A =_L B$ functions $q^* : El(A) \multimap El(B)$ and $sym(q)^* : El(B) \multimap El(A)$, which form a linear equivalence $El(A) \cong El(B)$. The two computation rules for the linear univalence axiom states that the procedure of generating an equivalence from an identity is itself an equivalence. In other words, given a linear function $f : El(A) \multimap El(B)$ which gives rise to a linear equivalence of $El(A)$ and $El(B)$, turning the equivalence f into an identity and then back into an equivalence should return f again:

$$\frac{\Gamma \vdash R^{Id}_{[x,y,p]}(\lambda u.u, A, B, ua(f)_{g,h,p,q}) : El(A) \multimap El(B)}{\Gamma \vdash R^{Id}_{[x,y,p]}(\lambda u.u, A, B, ua(f)_{g,h,p,q}) \equiv f : El(A) \multimap El(B)} \text{ L-ua-C}_1$$

and in the other direction, given an identity $p : A =_{\mathbb{L}} B$, turning it to an equivalence and then back into an identity should return p:

$$\frac{\Gamma \vdash ua(R^{Id}_{[x,y,p]}(\lambda u.u, A, B, p))_{[-,-,-,-]} : A =_{\mathbb{L}} B}{\Gamma \vdash ua(R^{Id}_{[x,y,p]}(\lambda u.u, A, B, p))_{[-,-,-,-]} \equiv p : A =_{\mathbb{L}} B} \text{ L-ua-C}_2$$

4.3.1 Semantic justification

The semantic interpretation of the procedure of turning an identity to an equivalence above is the following:

▶ **Lemma 20.** *Given two sections $A, B : \Gamma \to \Gamma.\mathbb{L}$, and a section $p : \Gamma \to \Gamma.Id_{\mathbb{L}}\{A\}\{B^+\}$, there is an natural isomorphism $IdToEquiv(p) : El(A) \cong El(B)$ in $[\Gamma, \mathcal{V}]$.*

Proof. The section $p : \Gamma \to \Gamma.Id_{\mathbb{L}}\{A\}\{B^+\}$ must for every Γ, select an object $(a \in A(\gamma), b \in B(\gamma), f : a \to b)$ of the groupoid $Id_{\mathbb{L}}.\{A\}\{B^+\}(\gamma) = \delta\mathcal{V}^{core}(A, B)$, and map morphisms to commutative squares, defining a natural isomorphism $El(A) \cong El(B)$ between the diagrams $El(A) : \Gamma \to \mathcal{V}$ and $El(B) : \Gamma \to \mathcal{V}$. ◀

▶ **Theorem 21.** *For $L \dashv M$ factoring through* **Sets** *as in Remark 4.2, the linear univalence axiom holds in the diagrams model. That is, given the following data:*

- *sections: $A, B : \Gamma \to \Gamma.\mathbb{L}$*
- *morphisms: $f : I \to [El(A), El(B)]$ and $g, h : I \to El(B)$ in $[\Gamma, \mathcal{V}]$,*
- *and sections: $p : \Gamma \to \Gamma.Id_{(M[El(A),El(A)])}\{M(g \circ f)\}\{M(id_A)\}$ and $q : \Gamma \to \Gamma.Id_{(M[El(B),El(B)])}\{M(f \circ h)\}\{M(id_B)\}$*

Then there is a section $ua(f)_{g,h,p,q} : \Gamma \to \Gamma.Id_{\mathbb{L}}\{A\}\{B\}$ such that $IdToEquiv(ua(f)) = f$ and $ua(IdToEquiv(p)) = p$.

Proof. Since M factors through sets, the interpretation of any type in its image is a functor with values in discrete groupoids, so by lemma **??**, the existence of a section $p : \Gamma \to \Gamma.Id_{M[El(A),El(A)]}\{(g \circ f)_M\}\{(id_{El(A)})_M\}$ implies that the sections $(g \circ f)_M : 1 \to M[A, A]$ and $(id_{El(A)})_M : 1 \to M[A, A]$ coincide, which by the adjunction $L \dashv M$ implies that $[[(g \circ f)]] = id_{El(A)}$. From the interpretation of $\multimap$ as internal hom, the syntactic composition operation coincides with the composition of the corresponding morphisms. In other words, denoting by $\hat{f}$ the transport of a map via the isomorphism $\mathcal{L}_\Gamma(I, [A, B]) \cong \mathcal{L}_\Gamma(A, B)$, we have $\widehat{[[g \circ f]]} = \hat{g} \circ \hat{f}$ and $\widehat{id}_{El(A)} = 1_{El(A)}$. We therefore have an isomorphism $\hat{f} : El(A) \to El(B)$ with $\hat{g} = \hat{h}$ as inverse. Selecting this isomorphism in $\mathcal{V}^{core}$ gives rise to a section $P : \Gamma \to \Gamma.Id_{\mathbb{L}}\{A\}\{B\}$, which by the previous lemma can be transported back by $IdToEquiv(P) = \hat{f}$. ◀

So the linear univalence axiom holds in the diagrams model as long as M factors through sets, analogous to how one can prove the univalence axiom in the groupoid model for the universe which only contains discrete groupoids. This might not be completely satisfying from the perspective of homotopy type theory, as this essentially truncates any higher dimensional type into a set when transporting them via M. Instead, we could to imagine forming a model where our linear types have an inherently higher dimensional structure, which is preserved by M. One such model is given by letting the objects of $\mathcal{V}$ be categories, and M be the functor taking a category to its core, i.e. its maximal sub-groupoid. If we want $\mathcal{V}$ to have a non-cartesian monoidal structure the candidate choices **Cat** and **Gpd** won't do. As briefly outlined in in **??** and carefully described in [11], there is a symmetric monoidal structure on **SMCat**, the category of small symmetric monoidal categories, symmetric monoidal functors and monoidal natural transformations. [4]

▶ **Definition 22.** Let the **2-categorical model of LDTT** be given by the diagrams model where $\mathcal{V}$ is the 2-category of small symmetric monoidal categories, symmetric monoidal functors and monoidal natural transformations:

$$\begin{tikzcd}
\mathbf{Diag}(\mathbf{SMCat}) \arrow[drr, "cod"] & & \mathbf{Diag}(\mathbf{Gpd}) \arrow[r, "\pi"] \arrow[d, "cod"] & \mathbf{Gpd}^{\to} \arrow[dl, "dom"] \\
& & \mathbf{Gpd} &
\end{tikzcd}$$

For two symmetric monoidal categories $\mathcal{A}$ and $\mathcal{B}$, the category $[\mathcal{A}, \mathcal{B}]$ consisting of monoidal functors and monoidal transformations between them carries a natural monoidal structure

[4] Technically, the structure on **SMCat** is not quite symmetric monoidal, as the associators, unitors and symmetry functors are only invertible up to higher homotopy. However, if one applies these homotopies whenever necessary, one does get a model of linear dependent type theory.

[11], and serves as the internal hom of $\mathcal{A}$ and $\mathcal{B}$. Since **SMCat** is complete, with limits inherited from **Cat** equipped with a pointwise monoidal structure, we have support for $\sqcap$ and &, and theorem 15 gives us that this model supports $\multimap$ type formers. [5]

▶ Proposition 23. There is a functor $F : \mathbf{Cat} \to \mathbf{SMCat}$ which constructs the free symmetric monoidal category of any groupoid, i.e. is a left adjoint to the forgetful functor $U_{smcat} : \mathbf{SMCat} \to \mathbf{Cat}$, forgetting the monoidal structure.

Combining this adjunction with the familiar adjunction:

$$\begin{tikzcd} \mathbf{Cat} \arrow[rr, bend right, "\text{core}"'] \arrow[rr, phantom, "\perp"] & & \mathbf{Gpd} \arrow[ll, bend right, "U_{gpd}"'] \end{tikzcd}$$

where U_{gpd} is the forgetful functor and core is the functor sending a category to its underlying maximal sub-groupoid, we get an adjunction:

$$\begin{tikzcd} \mathbf{SMCat} \arrow[rr, bend right, "\text{core} \circ U_{smcat}"'] \arrow[rr, phantom, "\perp"] & & \mathbf{Gpd} \arrow[ll, bend right, "F \circ U_{gpd}"'] \end{tikzcd}$$

This adjunction lifts by lemma **??** to a fiber adjunction:

$$\begin{tikzcd} \mathbf{Diag}(\mathbf{SMCat}) \arrow[rr, "M"'] \arrow[rr, phantom, bend left, "\perp"] \arrow[dr, "cod"] & & \mathbf{Diag}(\mathbf{Gpd}) \arrow[ll, bend right, "L"'] \arrow[dl, "cod"] \\ & \mathbf{Gpd} & \end{tikzcd}$$

Furthermore, it is clear from the definition of F and the unit I of **SMCat**, defined at **??** that the image of $\mathbf{1}$ under F is precisely I, so the higher dimensional model supports M and L.

Here we have two choices for our linear universe.[6] If we want univalence to hold in this model we can let the linear universe $\mathbb{L}$ be the groupoid consisting of all *discrete* symmetric monoidal categories, and isomorphisms between them. This is equivalently the core of $CMon$, the category of small commutative monoids (in **Set**). $CMon$ is a full subcategory of $SMCat$, which allows us to let El be the function sending an object of $\mathbb{L}$ to an object of **SMCat**.

Another choice is to let $\mathbb{L} = \mathbf{SMCat}^{core}$, making it an all-encompassing universe. This universe is not univalent, however, as one-dimensional groupoids do not carry enough higher dimensional structure to capture the weakness distinguishing an equivalence of categories and an isomorphism of categories.

[5] Note, however, that we do not have all coproducts in **SMCat**. Therefore, we cannot support $\oplus$ or $\sqsubset$. An alternative to be explored is the category **Mult**, of multicategories, which is a symmetric monoidal closed, complete and co-complete [2]. More on this in section **??**

[6] In our syntax we only defined a single linear universe, but it can easily be extended into several universes, which may be subcategories of each other, or even an infinite hierarchy of universes

4.3.2 Examples

To summarize, these are the conditions imposed in order to support all of the type formers described in section 2 using the families model:

- A symmetric monoidal closed category $\mathcal{V}$, with small products and coproducts, which distribute over the monoidal structure.
- An adjunction $L \dashv M$ between $\mathcal{V}$ and **Sets**, such that $L\{*\}$ is isomorphic to the unit of the monoidal structure of $\mathcal{V}$.

Some concrete choices for $\mathcal{V}$ that fulfill these conditions are:

- The category **AbGroups** of abelian groups with the monoidal structure given by the tensor product of abelian groups. Here $L \dashv M$ arises from the free functor on abelian groups.
- More generally, for any commutative ring R, the category R-**Mod** of modules over R with the free functor/forgetful functor adjunction
- The category $\mathbf{CGTop}_*$, of pointed compactly generated topological spaces, with the smash product as monoidal structure. The functor M is here the forgetful functor which both forgets the base point and the topology, which has a left adjoint given by the discrete topology, and then taking the coproduct with the point to create a pointed space. The unit of $\mathbf{CGTop}_*$ is the two point discrete set S^0, which is precisely the image of the point in the adjunction above.

References

1. P. N. Benton. A mixed linear and non-linear logic: proofs, terms and models (extended abstract). In *Computer science logic (Kazimierz, 1994)*, volume 933 of *Lecture Notes in Comput. Sci.*, pages 121–135. Springer, Berlin, 1995. URL: `https://doi.org/10.1007/BFb0022251`.
2. A. D. Elmendorf and M. A. Mandell. Permutative categories, multicategories and algebraic K-theory. *Algebr. Geom. Topol.*, 9(4):2391–2441, 2009. URL: `https://doi.org/10.2140/agt.2009.9.2391`.
3. Martin Hofmann and Thomas Streicher. The groupoid interpretation of type theory. In *Twenty-five years of constructive type theory (Venice, 1995)*, volume 36 of *Oxford Logic Guides*, pages 83–111. Oxford Univ. Press, New York, 1998.
4. Bart Jacobs. Comprehension categories and the semantics of type dependency. *Theoret. Comput. Sci.*, 107(2):169–207, 1993. URL: `https://doi.org/10.1016/0304-3975(93)90169-T`.
5. Neelakantan R Krishnaswami, Pierre Pradic, and Nick Benton. Integrating linear and dependent types. In *ACM SIGPLAN Notices*, volume 50, pages 17–30. ACM, January 2015. URL: `http://www.cs.bham.ac.uk/~krishnan/dlnl-paper.pdf`.
6. Daniel R. Licata, Michael Shulman, and Mitchell Riley. A Fibrational Framework for Substructural and Modal Logics. In Dale Miller, editor, *2nd International Conference on Formal Structures for Computation and Deduction (FSCD 2017)*, volume 84 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 25:1–25:22, Dagstuhl, Germany, 2017. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. URL: `http://drops.dagstuhl.de/opus/volltexte/2017/7740`, `doi:10.4230/LIPIcs.FSCD.2017.25`.
7. Martin Lundfall. Models of linear dependent type theory. *Stockholm University*, 2017. URL: `http://kurser.math.su.se/pluginfile.php/16103/mod_folder/content/0/2017/2017_47_report.pdf`.

8 Conor McBride. *I got plenty o' nuttin'*, pages 207–233. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). Springer, 3 2016. `doi:10.1007/978-3-319-30936-1_12`.

9 Paul-André Melliès. Categorical semantics of linear logic. In *Interactive models of computation and program behavior*, volume 27 of *Panor. Synthèses*, pages 1–196. Soc. Math. France, Paris, 2009.

10 The Univalent Foundations Program. *Homotopy type theory—univalent foundations of mathematics*. The Univalent Foundations Program, Princeton, NJ; Institute for Advanced Study (IAS), Princeton, NJ, 2013.

11 Vincent Schmitt. Tensor product for symmetric monoidal categories. preprint, 2007. `arXiv:0711.0324`.

12 Urs Schreiber. Quantization via linear homotopy types. preprint, 2014. `arXiv:1402.7041`.

13 Matthijs Vákár. A categorical semantics for linear logical frameworks. In *Foundations of software science and computation structures*, volume 9034 of *Lecture Notes in Comput. Sci.*, pages 102–116. Springer, Heidelberg, 2015. URL: `https://doi.org/10.1007/978-3-662-46678-0_7`.

14 Marek Zawadowski. Lax monoidal fibrations. In *Models, logics, and higher-dimensional categories*, volume 53 of *CRM Proc. Lecture Notes*, pages 341–426. Amer. Math. Soc., Providence, RI, 2011.