Modalities and Parametric Adjoints

DANIEL GRATZER, Aarhus University, Denmark
EVAN CAVALLO, Stockholm University, Sweden
G. A. KAVVOS, University of Bristol, United Kingdom
ADRIEN GUATTO, Université de Paris, CNRS, IRIF, France
LARS BIRKEDAL, Aarhus University, Denmark

Birkedal et al. recently introduced dependent right adjoints as an important class of (non-fibered) modalities in type theory. We observe that several aspects of their calculus are left underdeveloped and that it cannot serve as an internal language. We resolve these problems by assuming that the modal context operator is a parametric right adjoint. We show that this hitherto unrecognized structure is common. Based on these discoveries we present a new well-behaved Fitch-style multimodal type theory, which can be used as an internal language. Finally, we apply this syntax to guarded recursion and parametricity.

CCS Concepts: • **Theory of computation → Modal and temporal logics**; **Type theory**; **Proof theory;**

Additional Key Words and Phrases: Modal types, dependent types, Fitch-style modalities, categorical semantics, parametricity, guarded recursion

ACM Reference format:
Daniel Gratzer, Evan Cavallo, G. A. Kavvos, Adrien Guatto, and Lars Birkedal. 2022. Modalities and Parametric Adjoints. *ACM Trans. Comput. Logic* 23, 3, Article 18 (April 2022), 29 pages.
https://doi.org/10.1145/3514241

1 INTRODUCTION

18

When using **Martin-Löf Type Theory (MLTT)**, we often wish to reason about structure present in specific classes of models. Much of this structure—such as notions of time, cohesion, truncation, proof-irrelevance, and globality—can be captured through the addition of unary *modal operators* on types. Unfortunately, the development of *modal type theories* is fraught with difficulties. The overwhelming majority of the modalities we are interested in are *non-fibered*: They send types in

The first, third, and fifth authors were supported in part by a Villum Investigator grant (no. 25804), Center for Basic Research in Program Verification (CPV), from the VILLUM Foundation. The second author was supported in part by the Knut and Alice Wallenberg foundation (no. 2020.0266) and the Air Force Office of Scientific Research under MURI grants FA9550-15-1-0053 and FA9550-19-1-0216. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of any sponsoring institution, the U.S. government, or any other entity.
Authors' addresses: D. Gratzer and L. Birkedal, Aarhus University, Aabogade 34, Aarhus, Denmark, 8200; emails: {gratzer, birkedal}@cs.au.dk; E. Cavallo, Stockholm University, SE-106 91, Stockholm, Sweden; email: evan.cavallo@math.su.se; G. A. Kavvos, University of Bristol, Merchant Venturers Building, Bristol, United Kingdom, BS8 1UB; email: alex.kavvos @bristol.ac.uk; A. Guatto, Université de Paris, CNRS, IRIF, F-75006, Paris, France; email: guatto@irif.fr.

1529-3785/2022/04-ART18 $15.00
https://doi.org/10.1145/3514241

one context to types in a different context, disrupting the usual "context-agnostic" structure of type theory. Thus, all but a few modal operators require extensive changes to the rules of type theory.

The alteration of the judgmental structure of type theory to account for new modal operators is no small task, and various methods have been used in the past. Here we focus on *Fitch-style modal type theories* [9, 18, 25]. In broad strokes, the modal operators of Fitch-style type theories are functors that are *right adjoints*. This criterion is frequently satisfied, so we might expect Fitch-style type theories to find particular use as internal languages. Unfortunately, while their theory has absorbed considerable effort, many technical aspects of the Fitch style remain unsatisfactory. In particular, there seem to be some inexplicably delicate problems relating to substitution. The purpose of this article is to research the origin of these problems, highlight a key property that is missing, and use it to resolve them.

1.1 On Algebra and Type Theory

To simplify our technical development, for the rest of this article we will systematically blur the distinction between a Martin-Löf type theory and the **generalized algebraic theory (GAT)** that presents it. GATs originate in the work of Cartmell [14] and are often used to present the semantics of type theory in the guise of **categories with families (CwFs)** [20, 27]. Our approach replaces the study of (variable-based) type-theoretic syntax with the study of the (variable-free) CwFs that support the appropriate connectives. The syntax itself can then be defined as the *free algebra* over the signature, and various theorems guarantee the existence and initiality of this object [28].

There are many technical benefits to this approach. Most importantly, it reifies substitutions as explicit parts of the calculus, which allows us to directly observe their structure rather than treating substitution as a series of meta-operations. This is particularly helpful in modal type theory where substitutions interact with the modalities in a highly nontrivial manner.

1.2 Type Theory and Substitution

The admissibility of substitution is a central property of type theory, and indeed of all logic. By way of example, suppose we have $\Gamma \vdash A \triangleq (x : A_0) \to A_1$ type, and a substitution $\sigma : \Delta \to \Gamma$. Consider the type $\Delta \vdash A[\sigma]$ type. At the very least, we expect that this is again a dependent product: there should exist σ_0 and σ_1 such that $A[\sigma] = (x : A_0[\sigma_0]) \to A_1[\sigma_1]$. In variable-based presentations of type theory, equations of this sort are part of the definition of an *external substitution operation* on syntax; this operation is then is validated by proving that *substitution is admissible* in the type system. In variable-free presentations, such as CwFs, such equations are part of the definition of the generalized algebraic theory, which postulates a number of *naturality equations* that allow the pushing of substitutions under connectives.

Each of the standard connectives of type theory is understood to satisfy a property of this sort. Collectively, these equations ensure that type theory behaves in a predictable and usable manner. This global property is variably referred to as *admissibility of substitution*, *naturality*, *associativity*, or *stability under substitution*. It is unimportant whether this property is realized through meta-operation for substitution or through a sufficient number of naturality equations governing explicit substitutions, but the property itself is of paramount importance. For example, it is a direct consequence that we can prove a theorem in one context and then use it in another without worrying that the "shape" of the theorem has unpredictably changed (e.g., from a universal to an existential quantification).

1.3 Substitution and the Fitch Style

It is therefore troubling that the admissibility of substitution for Fitch-style calculi, such as *calculus of dependent right adjoints* (*DRA*) [9] or $\mathsf{MLTT}_{\text{🔒}}$ [25], comes with a caveat. There is no obvious way to write down naturality equations for modal types akin to those for other connectives. Indeed, examining the proof of admissibility for both languages in *loc. cit.*, we discover something unusual: Substitution in a term is not defined by induction over the term, but over the substitution itself! In other words, where we would normally reduce $A[\sigma]$ by examining the form of A, here we must also examine the form of σ.

This might seem like a small technical point, but in practice it is a crucial failing. The problem arises when we try to use a Fitch-style type theory $\mathbb{T}$ as an internal language. Suppose we have a category C with enough structure to interpret the types of $\mathbb{T}$. Then any theorem we can prove in $\mathbb{T}$ holds in C. However, we may also wish to prove theorems that speak about particular morphisms of C. To do so, we can construct the free type theory $\mathbb{T}_C$, which in addition to the terms of $\mathbb{T}$ also includes the morphisms of C as substitutions. We call this the *internal language* of C. Using this extended theory we can reason about C in type-theoretic terms.

Nevertheless, if we try to adapt the admissibility proof to $\mathbb{T}_C$, then we find ourselves in a predicament: We can no longer induct on substitutions. While the substitutions of the free algebra over $\mathbb{T}$ by and large consist of a list of terms definable in $\mathbb{T}$, the substitutions of $\mathbb{T}_C$ also contain "exotic" cases arising from the morphisms of C. As C is a parameter to this construction, these need not be inductively generated. It is therefore no longer evident that the theorems of the logic retain their "shape" under substitution.

This difficulty is compounded in multimodal type theories: While there have been a few type theories equipped with a single Fitch-style modality, none have managed to support more than one. Even one of the simplest multimodal settings that are desired in practice, i.e., the combination of the $\square$ and $\triangleright$ modalities of guarded type theory [19], has so far resisted attempts to be reformulated into a Fitch-style system that admits substitution.

1.4 Rectifying the Problem

This deficiency of Fitch-style systems is well known, and Gratzer et al. [22] attempted to address it by introducing a multimodal type theory called MTT, which replaces the elimination rule by an indirect variant. The resultant theory is weaker: Neither of the aforementioned Fitch-style theories, namely DRA [9] nor $\mathsf{MLTT}_{🔒}$ [25], can be embedded in it. The present article aims to rectify the issue of substitution without weakening the intuitive and powerful Fitch-style elimination rules.

We begin by observing that the secret ingredient that makes substitution admissible in the free model of DRA is that the modal context operator is a *parametric right adjoint (PRA)*. To illustrate the mechanics of this we show that the admissibility of substitution for (non-dependent) function types follows from a similar property. Based on this and the other examples provided by prior Fitch-style type theories, we observe that PRAs underlie the stability of a wide variety of rules that modify the context by applying an operator.

Starting from this idea, we introduce a new Fitch-style type theory that assumes from the outset that its modal context operator is a PRA. This new system, which we call FitchTT, can readily be used as an internal language: As the assumption of PRA structure is baked into the theory, any model C will also support it, and so the same proof of admissibility that applies to FitchTT can be "replayed" for the internal language $\mathsf{FitchTT}_C$.

The type-theoretic rules we introduce for parametric right adjoints appear unusual at first sight, in part because they involve strange substitutions of type $\Gamma \longrightarrow F(\mathbf{1})$ whose codomain is a functor applied to a terminal object. However, we discover that "extra-logical" structures found in several prior type theories can be seen as instances of these rules. For example, we show that the extension of a context by an affine dimension variable, as used in internalizing parametricity [7, 16], forms a PRA. Furthermore, the "tick variables" used in clocked type theory [4] can also be seen as arising from a PRA. We show that recognition of this fact can be used to provide a "rational reconstruction" of the rule for the tick constant that is simpler, and evidently implementable.

Both of these applications—viz. parametricity and guarded recursion—essentially require the formulation of the internal language of a categorical model: most non-trivial results depend on "importing" structure from the model into the type theory, which is then used for further reasoning. For instance, while the modal apparatus of guarded recursion can be captured by FitchTT directly, it is necessary to extend the theory to include the Löb induction principle to perform definition by guarded recursion. Previous Fitch-style type theories were incompatible with this step, and thus could not be used for either application. Thus, our work is more than just a technical exercise on the admissibility of substitution: it is a necessary stepping stone for the formulation of richer, practical type theories that encapsulate the logical principles found in many models of interest.

Finally, and most remarkably, we see that the structure of PRAs immediately scales to multiple modalities. Thus, FitchTT can be effortlessly parameterized by an arbitrary collection of modalities and natural transformations between them, with only minimal changes to the rules.

1.5 Contributions

In summary, we make the following contributions:

- We recognize parametric right adjoints as the key ingredient for validating substitution in modal type theories.
- We propose a new modal dependent type theory FitchTT that uses parametric right adjoints to generalize DRA to a setting with multiple modes and modalities.
- We prove that an appropriate instance of FitchTT constitutes a conservative extension of DRA, and investigate its more complex relationship to MLTT🔒 [25].
- We show that this extra structure of parametric right adjoints allows FitchTT to emulate the convenient syntax of handcrafted type theories for internalized parametricity [7, 16] and guarded recursion [4, 5].

1.6 Notation

We will use standard notation for CwFs. We write Γ, Δ, and so on, for contexts and σ, γ, δ for substitutions $\Delta \longrightarrow \Gamma$. We also write $\mathbf{1}$ for the terminal context, and $\Gamma.A$ for the extension of Γ by $\Gamma \vdash A$ type. If $\sigma : \Delta \longrightarrow \Gamma$ and $\Delta \vdash M : A[\sigma]$, then we can extend σ to $\sigma.M : \Delta \longrightarrow \Gamma.A$. There is a weakening substitution $\uparrow : \Gamma.A \longrightarrow \Gamma$, and we write $\uparrow^n$ for the composition of n of them. The last element in an extended context is accessed by the term $\Gamma.A \vdash \mathbf{v}_0 : A[\uparrow]$. Finally, substitutions σ have an action on types and terms that is denoted by $A[\sigma]$ and $M[\sigma]$, respectively.

2 MODALITIES AND SUBSTITUTION

Suppose we have a type theory on a category $\mathcal{C}$, and some endofunctor $\square : \mathcal{C} \longrightarrow \mathcal{C}$ of interest. Our objective is to internalize $\square$ in the type theory. Adopting the rule

$$\frac{\Gamma \vdash A \text{ type}}{\Gamma \vdash \square A \text{ type}}$$

amounts to assuming that the functor $\square$ is *fibered* [37, Section 2], i.e., has an action on types whose output lives in the same context as its input. Most of the functors that we are interested in are not.

If we do not wish to assume that $\square$ is fibered, then we may formulate rules that only assume its functoriality, like so:

$$\frac{\Gamma \vdash A \text{ type}}{\square\Gamma \vdash \square A \text{ type}}\ \textsc{ty/functorial-form} \qquad \frac{\Gamma \vdash M : A}{\square\Gamma \vdash \mathsf{mod}(M) : \square A}\ \textsc{tm/functorial-intro}$$

Unfortunately, these rules do not admit substitution. Suppose that $\Gamma \vdash M : A$ and $\sigma : \Delta \longrightarrow \Box\Gamma$, so that $\Delta \vdash (\Box A)[\sigma]$ type. For $\Box$ to be natural, there should be a substitution σ' for which

$$\Delta \vdash (\Box A)[\sigma] = \Box(A[\sigma']) \text{ type}$$

This is constitutionally impossible: The right-hand side is typable only in a context of the form $\Box\Gamma'$, not a general Δ.

To obtain a usable type theory, one must repair this. There are three standard solutions.

Delay substitutions. Instead of propagating substitutions under modal constructs, we may choose to absorb them. We can do so by building a *delayed substitution* into the modal introduction rules for both types and terms:

$$\frac{\sigma : \Gamma \longrightarrow \Box\Delta \qquad \Delta \vdash M : A}{\Gamma \vdash \mathsf{mod}(M)^{\sigma} : \Box^{\sigma} A}$$

Substitution is then effected by absorbing a morphism into this cut: for any $\sigma' : \Gamma' \longrightarrow \Gamma$ we have

$$\Gamma' \vdash \mathsf{mod}(M)^{\sigma}[\sigma'] = \mathsf{mod}(M)^{\sigma\circ\sigma'} : \Box^{\sigma\circ\sigma'} A$$

This method was pioneered by Bierman and de Paiva [8].

Split the contexts. Another approach, originally due to Davies and Pfenning [34], replaces the usual judgments by a form that involves two or more contexts. The *dual context* $\Delta;\Gamma$ stands for the object $\Box\Delta \times \Gamma$. The introduction rules are

$$\frac{\mathbf{1};\Delta \vdash A \text{ type}}{\Delta;\Gamma \vdash \Box A \text{ type}} \qquad\qquad \frac{\mathbf{1};\Delta \vdash M : A}{\Delta;\Gamma \vdash M : \Box A}$$

The semantics of these rules is clear: If $\mathbf{1};\Delta \vdash A$ type is interpreted by a family $\pi_A : \Delta.A \longrightarrow \Delta$, then $\Box A$ is interpreted by the family $\Box\pi_A \times \mathsf{id}_\Gamma$, which is over the object $\Box\Delta \times \Gamma$. This rule is well behaved under substitution but with a caveat: We must adapt the structure of substitutions in a way that follows the structure of contexts. We must take our "primitive" substitutions $(\delta;\gamma) : \Delta';\Gamma \longrightarrow \Delta;\Gamma$ to be morphisms $F(\delta) \times \gamma : \Box\Delta' \times \Gamma' \longrightarrow \Box\Delta \times \Gamma$ of C.

Factorize the substitution. A third way to push a substitution $\sigma : \Delta \longrightarrow \Box\Gamma$ under a modality is to assume that it factorizes in a convenient way. For example, we may assume that for every Δ there is a *universal arrow* from Δ to $\Box$, i.e., an object $\Delta.\text{🔒}$ and a morphism $\eta_\Delta : \Delta \longrightarrow \Box(\Delta.\text{🔒})$ through which every substitution $\sigma : \Delta \longrightarrow \Box\Gamma$ into a modal context factorizes uniquely:

$$\begin{tikzcd}
\Delta \arrow[r, "\eta_\Delta"] \arrow[dr, "\sigma"'] & \Box(\Delta.\text{🔒}) \arrow[d, "\Box\hat{\sigma}"] & \Delta.\text{🔒} \arrow[d, dashed, "\hat{\sigma}"] \\
& \Box\Gamma & \Gamma
\end{tikzcd}$$

This does not quite solve the substitution problem for TY/FUNCTORIAL-FORM, but it canonically simplifies it: It allows us to find a "maximal" substitution $\hat{\sigma}$ that we can push under the modality, so that $\Delta \vdash (\Box A)[\sigma] = \Box(A[\hat{\sigma}])[\eta_\Delta]$ type. In a sense, this is a case of carrying a "canonical delayed substitution" η_Δ.

A simple observation allows us to make η invisible. It is a well-known fact from category theory that if such a universal arrow exists for every Δ, then $-.\text{🔒}$ extends to an endofunctor that is left adjoint to $\Box$. We can promote this to an additional *operator on contexts*, and replace the introduction rules with

$$\frac{\Gamma.\text{🔒} \vdash A \text{ type}}{\Gamma \vdash \Box A \text{ type}} \qquad \frac{\Gamma.\text{🔒} \vdash M : A}{\Gamma \vdash \mathsf{mod}(M) : \Box A}$$

These are called *Fitch-style rules* [18].

All three approaches have their strengths and weaknesses. The Bierman-de Paiva style of delayed substitutions is conceptually clear, but difficult to use and implement. Moreover, it does not readily adapt to support multiple modalities, at least not when they interact in a nontrivial way. However, the split-context approach has proven practical whenever the modalities interact in certain convenient ways (see, e.g., Shulman [36]). However, this is the exception and not the rule.

In contrast, the Fitch-style approach is supported by a single *universal property* that fully determines the modality up to isomorphism—just as with standard connectives, like dependent products and sums. Thus, one might be led to believe that Fitch-style calculi are the preferred formalism. Alas, it is not difficult to see that they suffer from a number of technical disadvantages. We illustrate these using a specific theory, viz. the calculus of dependent right adjoints.

2.1 The Calculus of Dependent Right Adjoints

The calculus of dependent right adjoints (DRA) [9] consists of standard Martin-Löf type theory extended with an operation on contexts—denoted by 🔒—and a single modality $\Box$ on types. In addition to the usual CwF structure, DRA postulates a *dependent adjunction.*

Definition 1. A *dependent adjunction* consists of

(1) an endofunctor $-.\text{🔒}$ on the category of contexts
(2) an assignment $\Box$ from types to types, such that

$$\textsc{dra/ty/mod}\ \frac{\Gamma.\text{🔒} \vdash A \text{ type}}{\Gamma \vdash \Box A \text{ type}}$$

(3) a bijection $\mathsf{mod}(-)/\mathsf{unmod}(-)$ on terms, such that

$$\textsc{dra/tm/mod}\ \frac{\Gamma.\text{🔒} \vdash M : A}{\Gamma \vdash \mathsf{mod}(M) : \Box A} \qquad \textsc{dra/tm/unmod}\ \frac{\Gamma \vdash M : \Box A}{\Gamma.\text{🔒} \vdash \mathsf{unmod}(M) : A}$$

All of $\Box$, $\mathsf{mod}(-)$, and $\mathsf{unmod}(-)$ must be natural in Γ.

While $-.\text{🔒}$ has an action on the entire category of contexts, the modality $\Box$ acts *only on types*, which are a distinct sort (depending on contexts).[1] The fact that $\mathsf{mod}(-)$ and $\mathsf{unmod}(-)$ form a bijection yields the following β and η laws for $\Box$:

$$\frac{\Gamma.\text{🔒} \vdash M : A}{\Gamma.\text{🔒} \vdash \mathsf{unmod}(\mathsf{mod}(M)) = M : A} \qquad \frac{\Gamma \vdash M : \Box A}{\Gamma \vdash \mathsf{mod}(\mathsf{unmod}(M)) = M : \Box A}$$

Do these rules admit substitution? In the case of the introduction rule DRA/TM/MOD, the naturality required of $\Box$ and $\mathsf{mod}(-)$ solves the problem: For any $\Gamma.\text{🔒} \vdash M : A$ and $\sigma : \Delta \to \Gamma$ it implies that

$$\Delta \vdash \mathsf{mod}(M)[\sigma] = \mathsf{mod}(M[\sigma.\text{🔒}]) : \Box(A[\sigma.\text{🔒}])$$

[1] This gap disappears if we can blend types and contexts. For example, if the CwF is *democratic*, i.e., if for every context Γ there is a $\vdash \tilde{\Gamma}$ type such that $\Gamma \cong \mathbf{1}.\tilde{\Gamma}$, then $\Box$ can be extended to a right adjoint of $-.\text{🔒}$ [9, Section 4.1].

where $\sigma.\text{🔒}$ is the action of the functor $-.\text{🔒}$ on σ. The same cannot be said of the elimination rule DRA/TM/UNMOD: there is no evident way to commute a substitution with $\mathsf{unmod}(-)$. Indeed, we cannot use naturality, as a general substitution $\sigma : \Delta \to \Gamma.\text{🔒}$ need not be of the form $\gamma.\text{🔒}$.

To address this the original paper on DRA replaces DRA/TM/UNMOD with a rule involving additional weakening:

$$\textsc{dra/tm/unmod}^{*}\quad \frac{\Gamma \vdash M : \Box A}{\Gamma.\text{🔒}.A_0.\cdots.A_{n-1} \vdash \mathsf{unmod}(M) : A[\uparrow^n]}$$

This rule has an "exorbitant privilege": It is stable under substitution in the free algebra. Every $\sigma : \Delta \to \Gamma$ in the free algebra is a substitution that is *definable* in the pure type theory with no constants. One can then prove that for every such $\sigma : \Delta \to \Gamma.\text{🔒}.A$ there is a Δ' such that $\Delta = \Delta'.\text{🔒}.A$, and a $\sigma' : \Delta' \to \Gamma$ such that $\sigma = \sigma'.\text{🔒}.\mathrm{v}_0$. This enables us to extract σ' from σ, and push that under $\mathsf{unmod}(-)$.

This is all well and good if we just want a syntax for dependent adjunctions: we can write proofs in the free algebra, and interpret them in any dependent adjunction [9, Section 3.1]. One can even implement this syntax, following an approach similar to that of [25] for $\mathsf{MLTT}_{\text{🔒}}$. Nevertheless, there is something unsatisfying about this state of affairs. The aforementioned factorization property of substitutions is proven by performing an *induction on the substitution* σ. As a consequence, it only works in the free algebra: It is not in general possible to decompose substitutions by induction in an arbitrary CwF. In other words, the stability of $\mathsf{unmod}(-)$ depends on the *absence* of certain substitutions.

This may seem like a small price to pay, but in fact it has grave consequences: It prohibits the use of DRA as the internal language of an arbitrary dependent adjunction. In models of DRA, DRA/TM/UNMOD* may not respect substitution, leading to unwelcome surprises. For example, the truth of a theorem of the type theory might depend on the precise context in which it is proven. Unfortunately, such models are not uncommon: For instance, $\mathsf{MLTT}_{\text{🔒}}$ [25] is a proper extension (and hence a model) of DRA, yet the $\mathsf{unmod}(-)$ form of DRA is not stable under substitution in $\mathsf{MLTT}_{\text{🔒}}$. In short, Fitch-style type theories à la DRA cannot play the rôle of internal languages.

2.2 Parametric Right Adjoints

It is natural to wonder if there is a special property of the pure syntax that confers stability under substitution. If we were to identify and axiomatize it, then we would have a chacterization of dependent adjunctions that support it.

To this end, it is instructive to consider a particular example. Suppose that $\vdash \mathfrak{A}$ type, i.e., that $\mathfrak{A}$ is a *closed type*. (What follows does not work if $\mathfrak{A}$ is not closed.) Then context extension by $\mathfrak{A}$ coincides with $- \times \mathfrak{A}$, and has a dependent right adjoint $\mathfrak{A} \to (-)$ [9, Section 5]. Writing out the rule DRA/TM/MOD yields the usual introduction rule for the function space:

$$\textsc{tm/lam}\quad \frac{\Gamma.\mathfrak{A} \vdash M : B}{\Gamma \vdash \lambda(M) : \mathfrak{A} \to B}$$

However, the elimination rule DRA/TM/UNMOD looks unfamiliar:

$$\textsc{tm/unlam}\quad \frac{\Gamma \vdash M : \mathfrak{A} \to B}{\Gamma.\mathfrak{A} \vdash \mathsf{unlam}(M) : B}$$

This rule suffers from the same issues as the more general DRA/TM/UNMOD. Given a closed term $\vdash N : \mathfrak{A}$, there is no evident way to push the corresponding substitution $\mathbf{1} \longrightarrow \mathbf{1}.\mathfrak{A}$ under $\mathsf{unlam}(-)$. In fact, the traditional elimination rule

$$\frac{\Gamma \vdash M : \mathfrak{A} \to B \qquad \Gamma \vdash N : \mathfrak{A}}{\Gamma \vdash M(N) : B[\mathsf{id}.N]}\ \textsc{tm/app}$$

is a kind of closure of $\mathsf{unlam}(-)$ under substitution: we may define $M(N) \triangleq \mathsf{unlam}(M)[\mathsf{id}.N]$. Conversely, using TM/APP we can define $\mathsf{unlam}(M) \triangleq M[\uparrow](\mathbf{v}_0)$. Thus, TM/UNLAM and TM/APP are interderivable rules, but we can only write a naturality equation for the latter.

On the surface, TM/APP does not seem to be the most general closure of TM/UNLAM under substitution: That would include an arbitrary $\sigma : \Delta \to \Gamma.\mathfrak{A}$ in the premise, which the conclusion would carry it in a delayed form. However, this is not necessary: As $\mathfrak{A}$ is closed, every such σ is determined by a substitution $\Delta \longrightarrow \Gamma$ and a term $\Delta \vdash N : \mathfrak{A}$.

Lemma 1. *Let* $\vdash \mathfrak{A}$ type. *Any substitution* $\sigma \colon \Delta \longrightarrow \Gamma.\mathfrak{A}$ *can be uniquely decomposed into a pair of substitutions*

$$\sigma_0 \triangleq \uparrow \circ\, \sigma : \Delta \to \Gamma \qquad\qquad r \triangleq (!_\Gamma)^+ \circ \sigma : \Delta \to \mathbf{1}.\mathfrak{A}$$

where $\gamma^+ \triangleq (\gamma \circ \uparrow).\mathbf{v}_0 : \Delta.\mathfrak{A} \to \Gamma.\mathfrak{A}$ *for any* $\gamma : \Delta \longrightarrow \Gamma$.

Thus, if we fix a substitution $r : \Delta \to \mathbf{1}.\mathfrak{A}$ (i.e., a term), substitutions $\sigma : \Delta \to \Gamma.\mathfrak{A}$ such that $(!_\Gamma)^+ \circ \sigma = r$ correspond to substitutions $\Delta \to \Gamma$. Regarding r as an object in the slice over $\mathbf{1}.\mathfrak{A}$, this equation then shows that σ is a morphism $r \to (!_\Gamma)^+$ in the slice category over $\mathbf{1}.\mathfrak{A}$. Accordingly, we obtain a bijection between substitutions $\Delta \to \Gamma$ and morphisms $r \to (!_\Gamma)^+$ in the slice category. With this observation in hand, the decomposition provided by Lemma 1 can be sharpened into a more familiar categorical structure:

Definition 2. Let C have a terminal object $\mathbf{1}_C$. A functor $G \colon C \longrightarrow \mathcal{D}$ is a *parametric right adjoint* if the induced functor $G/\mathbf{1} \colon C \longrightarrow \mathcal{D}/G(\mathbf{1}_C)$ is a right adjoint.

See Carboni and Johnstone [13] and Weber [40, Section 2] for the origins of PRAs.

Specializing to $G = -.\mathfrak{A}$, the functor $G/\mathbf{1} = (-.\mathfrak{A})/\mathbf{1}$ maps Δ to $(!_\Delta)^+ : \Delta.\mathfrak{A} \to \mathbf{1}.\mathfrak{A}$, and this has a left adjoint given by $F(r : \Gamma \longrightarrow \mathbf{1}.\mathfrak{A}) \triangleq \Gamma$. Unfolding, we see that Lemma 1 precisely states that these functors are adjoints. The unit and counit of this adjunction have recognizable forms: the unit at $r \colon \Gamma \longrightarrow \mathbf{1}.\mathfrak{A}$ is the substitution $\eta[r] \triangleq \mathsf{id}.\mathbf{v}_0[r] \colon \Gamma \longrightarrow \Gamma.\mathfrak{A}$, and the counit at Γ is $\epsilon[\Gamma] \triangleq \uparrow \colon \Gamma.\mathfrak{A} \longrightarrow \Gamma$.

Using these parts we may precisely restate the application rule TM/APP without actually changing any of its ingredients:

$$\frac{r \colon \Gamma \longrightarrow \mathbf{1}.\mathfrak{A} \qquad F(r) \vdash M : \mathfrak{A} \to A}{\Gamma \vdash M\langle r\rangle : A[\eta[r]]}\ \textsc{tm/pra-app}$$

This formulation only uses two facts: that $\mathfrak{A} \to (-)$ is a dependent right adjoint to $(-).\mathfrak{A}$, and that $(-).\mathfrak{A}$ is itself a parametric right adjoint with left adjoint F. One naturally wonders whether we can adapt this maneuver to a general dependent adjunction: Can an ill-behaved elimination rule (like TM/UNLAM) always be replaced by an equivalent well-behaved rule (like TM/APP) if we assume that the modal context operator is a parametric right adjoint? The answer is positive: we will in fact show that the adjunctions automatically guarantee the admissibility of substitution!

Indeed, suppose $-.\text{🔒}$ has a dependent right adjoint $\Box$. Suppose furthermore that $-.\text{🔒}$ is a parametric right adjoint, and write Γ/r for the application of the left adjoint to $r : \Gamma \longrightarrow \mathbf{1}.\text{🔒}$. Recalling that $\eta[r] : \Gamma \longrightarrow (\Gamma/r).\text{🔒}$, we can write down a rule

$$\frac{r : \Gamma \longrightarrow \mathbf{1}.\text{🔒} \qquad \Gamma/r \vdash M : \Box A}{\Gamma \vdash M \mathbin{@} r : A[\eta[r]]}\ \textsc{dra/tm/pra-unmod}$$

Theorem 2. *dra/tm/pra-unmod and dra/tm/unmod are interderivable.*

Proof. Under the hypotheses of dra/tm/pra-unmod, we define $M \mathbin{@} r$ by

$$M \mathbin{@} r \triangleq \mathsf{unmod}(M)[\eta[r]]$$

We now show that this rule is equivalent to dra/tm/unmod. Given $\Gamma \vdash M : \Box A$, we recall that $\epsilon[\Gamma] : \Gamma.\text{🔒}/(!_\Gamma.\text{🔒}) \longrightarrow \Gamma$, so we have

$$\Gamma.\text{🔒}/(!_\Gamma.\text{🔒}) \vdash M[\epsilon[\Gamma]] : (\Box A)[\epsilon[\Gamma]]$$

By naturality, this type is equal to $\Box(A[\epsilon[\Gamma].\text{🔒}])$. Hence, we can define

$$\mathsf{unmod}(M) \triangleq \Gamma.\text{🔒} \vdash M[\epsilon[\Gamma]] \mathbin{@} (!_\Gamma.\text{🔒}) : A[\epsilon[\Gamma].\text{🔒}][\eta[!_\Gamma.\text{🔒}]]$$

The type of this term is equal to A, as $\epsilon[\Gamma].\text{🔒} \circ \eta[!_\Gamma.\text{🔒}]$ is the identity by one of the triangle laws of the adjunction. $\square$

Therefore, the PRA structure allows us to equivalently restate dra/tm/unmod as dra/tm/pra-unmod. It remains to prove that, unlike the former rule, the latter can be made to admit substitution. Given any $\sigma : \Delta \longrightarrow \Gamma$, we may see it as an arrow $r \circ \sigma \longrightarrow r$ in the slice category over $\mathbf{1}.\text{🔒}$. Applying the left adjoint gives $\sigma/\text{🔒} : \Delta/r \circ \sigma \longrightarrow \Gamma/r$. This substitution acts on M to yield $\Delta/r \circ \sigma \vdash M[\sigma/\text{🔒}] : (\Box B)[\sigma/\text{🔒}]$. But $(\Box B)[\sigma/\text{🔒}] = \Box(B[(\sigma/\text{🔒}).\text{🔒}])$, so we have

$$\Delta \vdash M[\sigma/\text{🔒}] \mathbin{@} (r \circ \sigma) : B[(\sigma/\text{🔒}).\text{🔒}][\eta[r \circ \sigma]]$$

This type is equal to $B[\eta[r]][\sigma]$ by the naturality of η. Hence, we can postulate that

$$(M \mathbin{@} r)[\sigma] = M[\sigma/\text{🔒}] \mathbin{@} (r \circ \sigma)$$

In fact, this equation can be derived from $M \mathbin{@} r \triangleq \mathsf{unmod}(M)[\eta[r]]$ by the naturality of η and of $\mathsf{unmod}(-)$.

Some version of Lemma 1 has been shown *en passant* for all prior Fitch-style calculi in the process of proving the admissibility of substitution [4, 9, 16, 25]. In each case the modal elimination rules can be derived by unfolding the components of the parametric adjunction in the general rule dra/tm/pra-unmod. We choose the notation $M \mathbin{@} r$ to emphasize the connection with the application rule.

To recap: We have addressed the issue of admissibility of substitution. In the case of connectives that modify the context—like context extension and (Fitch-style) modalities—we have found that the structure that essentially underlies the admissibility of substitution is that of a parametric right adjoint. We continue by introducing a general type theory, capable of supporting multiple interacting modalities, which requires that each of its modal context operators is a PRA.

3 A MULTIMODAL FITCH-STYLE TYPE THEORY

In this section, we introduce a modal type theory for dependent adjunctions whose left adjoints are also parametric right adjoints, which we call FitchTT. Remarkably, by the reasoning explored in Section 2, this type theory readily scales to not only admit substitution with a Fitch-style modality, but to admit substitution in the presence of an arbitrary collection of modalities and natural transformations between them. The step from a single modality to multiple modalities in the presence of parametric right adjoints is almost mechanical, which is surprising: It was previously unknown how to properly support two distinct Fitch-style modalities in one system.

3.1 Multimode and Multimodal Aspects

While our definition of dependent adjunction involved only a single category, adjunctions in general connect two possibly distinct categories. To obtain the most expressive theory, we therefore allow for multiple categories in FitchTT. We call each such category a *mode*, making FitchTT a *multimode* type theory. Each judgment of FitchTT is annotated by the mode it lives in. We denote modes by m, n, o, and so on.

Accordingly, the modalities of FitchTT are no longer operators on types in a single category, but map types across categories. Each modality $\mu \colon n \longrightarrow m$ induces an operation $\langle \mu \mid - \rangle$ from types at mode n to types at mode m. As we allow many modalities between each pair of modes, FitchTT is a *multimodal* type theory. We denote modalities by μ, ν, ξ, and so on.

Viewing modalities as functors suggests that modes and modalities should form a category: there should be a composite modality $\mu \circ \nu \colon o \longrightarrow m$ for every $\mu \colon n \longrightarrow m$ and $\nu \colon o \longrightarrow n$. To this structure we add one more layer, namely 2-cells *between* modalities. These induce natural transformations: a 2-cell $\alpha : \mu \Rightarrow \nu$ enables the definition of a function $\langle \nu \mid A \rangle \to \langle \mu \mid A \rangle$ for a type A. We denote 2-cells by α, β, γ, and so on.

We follow prior modal type theories in recognizing that the modes, modalities, and 2-cells together constitute a strict 2-category, a *mode theory* [22, 29, 30], for which we usually write $\mathcal{M}$. No rule changes the mode theory: It is a parameter to the type theory.

Notation 3. Given a pair of 2-cells $\alpha : \mu_0 \Rightarrow \nu_0$ and $\beta : \mu_1 \Rightarrow \nu_1$, we write $\beta \bullet \alpha : \mu_1 \circ \mu_0 \Rightarrow \nu_1 \circ \nu_0$ for the *horizontal composition* of these 2-cells.

3.2 The Mode-local Fragment

Each judgment of FitchTT is indexed by a mode. For instance, we indicate that Γ is a well-formed context at mode m by writing $\Gamma\ \mathsf{cx} \mathrel{@} m$. Modes interact with each other only through modal types. In other words, if we do not include any modal rules, each typing derivation remains in a single mode. We call the collection of non-modal rules the *mode-local fragment* of FitchTT. This fragment is given parametrically in the mode m, and consists of the usual rules of MLTT with products, sums, and intensional identity types.

3.3 The Modal Fragment: Formation and Introduction

The modal rules of FitchTT mediate between the different modes of the type theory. They closely follow the DRA calculus described in Section 2 but incorporate slight generalizations to allow for the multimodal structure. We will examine the rules for modal types in FitchTT discursively, but the reader may refer to Appendix A to see the full collection of rules.

The formation and introduction rules are given in Figure 1. For each modality $\mu \colon n \longrightarrow m$, there is both a modal context operator $-.\{\mu\}$ as well as an operator $\langle \mu \mid - \rangle$ on types. Like in DRA, the idea is that $-.\{\mu\}$ is the left adjoint, and $\langle \mu \mid - \rangle$ is its dependent right adjoint. However, a modality may now cross between different modes. Thus, if $\Gamma\ \mathsf{cx} \mathrel{@} m$ is a context at mode m and $\mu \colon n \longrightarrow m$,

$$\frac{\mu\colon n \longrightarrow m \qquad \Gamma\ \mathsf{cx} @ m}{\Gamma.\{\mu\}\ \mathsf{cx} @ n}\ \textsc{fitch/cx/mex} \qquad \frac{\mu\colon n \longrightarrow m \qquad \Gamma.\{\mu\} \vdash A\ \mathsf{type} @ n}{\Gamma \vdash \langle \mu \mid A \rangle\ \mathsf{type} @ m}\ \textsc{fitch/ty/mod}$$

$$\frac{\mu\colon n \longrightarrow m \qquad \Gamma.\{\mu\} \vdash M : A @ n}{\Gamma \vdash \mathsf{mod}_\mu(M) : \langle \mu \mid A \rangle @ m}\ \textsc{fitch/tm/mod}$$

Fig. 1. Introduction and formation for modal types.

then we obtain a context $\Gamma.\{\mu\}\ \mathsf{cx} @ n$ at mode n. This action is *contravariant* by convention: the mode theory $\mathcal{M}$ covariantly specifies the structure of the modalities $\langle \mu \mid - \rangle$, so their left adjoints $-.\{\mu\}$ act with opposite variance.

The introduction rule for modal types, fitch/ty/mod, is a slight variation on dra/ty/mod, which accounts for passing between modes. The same is true for the modal term introduction rule fitch/tm/mod: given M of the appropriate type at mode n, it ensures that $\mathsf{mod}_\mu(M)$ is a term at mode m.

3.4 The Modal Fragment: The Elimination Rule

Unlike the case of introduction, the well-behaved elimination rule of the DRA calculus does not readily adapt to a multimodal setting. We must hence design an elimination rule anew, using the insights we acquired in Section 2.

First, we introduce some notation. Whenever $\mu\colon n \longrightarrow m$, we write $\{\mu\}$ for the context $\mathbf{1}.\{\mu\}\ \mathsf{cx} @ n$. We will also write $\mathbf{m}_\mu \triangleq\ !_\Gamma.\{\mu\}\colon \Gamma.\{\mu\} \longrightarrow \{\mu\} @ n$. Consider the multimodal analogue of the elimination rule of the dependent adjunction, viz.

$$\frac{\mu\colon n \longrightarrow m \qquad \Gamma \vdash M : \langle \mu \mid A \rangle @ m}{\Gamma.\{\mu\} \vdash \mathsf{unmod}_\mu(M) : A @ n}\ \textsc{fitch/tm/unmod-dra}$$

Now suppose that $-.\{\mu\}$ (for $\mu\colon n \longrightarrow m$) is a parametric right adjoint. This means that we are given a modal context operator that maps a $\Gamma\ \mathsf{cx} @ n$ and a substitution $r\colon \Gamma \longrightarrow \{\mu\} @ n$ to a new context $\Gamma/(r : \mu)\ \mathsf{cx} @ m$. The unit for this adjunction gives for each such r a substitution $\eta[r] : \Gamma \rightarrow \Gamma/(r : \mu).\{\mu\} @ n$. Using fitch/tm/unmod-dra, we can derive a rule

$$\frac{r\colon \Gamma \longrightarrow \{\mu\} @ n \qquad \Gamma/(r : \mu) \vdash M : \langle \mu \mid A \rangle @ m}{\Gamma \vdash M @ r : A[\eta[r]] @ n}$$

by setting $M @ r \triangleq \mathsf{unmod}_\mu(M)[\eta[r]]$. Just as in Section 2.2, this new rule is equivalent to fitch/tm/unmod-dra and admits substitution. We take this as the definitive elimination rule for modal types. The basic rules for $-/(- : \mu)$ are given in Figure 2 while the elimination rule for modal types along with its β and η principles are given in Figure 3.

The presence of multiple modalities does not complicate the elimination rule, unlike in other multimodal calculi, e.g., Gratzer et al. [22]. Instead, the interaction of modalities in FitchTT is governed purely by the substitution calculus.

3.5 The Substitution Calculus

The substitution calculus for FitchTT can be divided into the mode-local part—the standard substitution operations of MLTT in each mode—and the part concerning modal operations. For instance,

$$\frac{\mu : n \longrightarrow m \qquad \Gamma\ \mathsf{cx} \mathbin{@} n \qquad r : \Gamma \longrightarrow \{\mu\} \mathbin{@} m}{\Gamma/(r : \mu)\ \mathsf{cx} \mathbin{@} m}\ \textsc{fitch/cx/mres}$$

$$\frac{\mu : n \longrightarrow m \qquad \Gamma\ \mathsf{cx} \mathbin{@} m}{\epsilon[\Gamma] : \Gamma.\{\mu\}/(\mathbf{m}_\mu : \mu) \longrightarrow \Gamma \mathbin{@} m}\ \textsc{fitch/sb/counit}$$

$$\frac{\mu : n \longrightarrow m \qquad \Gamma\ \mathsf{cx} \mathbin{@} n \qquad r : \Gamma \longrightarrow \{\mu\} \mathbin{@} m}{\eta[r] : \Gamma \longrightarrow \Gamma/(r : \mu).\{\mu\} \mathbin{@} m}\ \textsc{fitch/sb/unit}$$

Fig. 2. Selected rules for modal restriction.

$$\frac{\mu : n \longrightarrow m \qquad \Gamma/(r : \mu).\{\mu\} \vdash A\ \mathsf{type} \mathbin{@} n \qquad r : \Gamma \longrightarrow \{\mu\} \mathbin{@} n \qquad \Gamma/(r : \mu) \vdash M : \langle \mu \mid A \rangle \mathbin{@} m}{\Gamma \vdash M \mathbin{@} r : A[\eta[r]] \mathbin{@} n}\ \textsc{fitch/tm/unmod}$$

$$\frac{\mu : n \longrightarrow m \qquad \Gamma/(r : \mu).\{\mu\} \vdash A\ \mathsf{type} \mathbin{@} n \qquad r : \Gamma \longrightarrow \{\mu\} \mathbin{@} n \qquad \Gamma/(r : \mu).\{\mu\} \vdash M : A \mathbin{@} n}{\Gamma \vdash \mathsf{mod}_\mu(M) \mathbin{@} r = M[\eta[r]] : A[\eta[r]] \mathbin{@} n}\ \textsc{fitch/tm/unmod-mod}$$

$$\frac{\mu : n \longrightarrow m \qquad \Gamma.\{\mu\} \vdash A\ \mathsf{type} \mathbin{@} n \qquad \Gamma \vdash M : \langle \mu \mid A \rangle \mathbin{@} m}{\Gamma \vdash M = \mathsf{mod}_\mu(M[\epsilon[\Gamma]] \mathbin{@} \mathbf{m}_\mu) : \langle \mu \mid A \rangle \mathbin{@} m}\ \textsc{fitch/tm/mod-unmod}$$

Fig. 3. The full elimination rule for modal types.

at each mode m there are identities and compositions of substitutions, as well as a unique substitution $!_\Gamma : \Gamma \longrightarrow \mathbf{1} \mathbin{@} m$ for each $\Gamma\ \mathsf{cx} \mathbin{@} m$. Mode-local substitutions are thus standard [27], so we focus on the novel modal ones.

As we mentioned before, the mode theory $\mathcal{M}$ is a strict 2-category. We mirror this fact within the type theory by postulating that the assignment of modal context operators $-.\{\mu\}$ to modalities μ is *2-functorial* in the mode theory. This is established by some of the rules of Figure 4.

Furthermore, each one of these operators $-.\{\mu\}$ is itself a functor between context categories. For each $\mu : n \longrightarrow m$ there is a functorial action on substitutions, which to $\delta : \Gamma \longrightarrow \Delta \mathbin{@} m$ assigns a substitution $\delta.\{\mu\} : \Gamma.\{\mu\} \longrightarrow \Delta.\{\mu\} \mathbin{@} n$. This assignment respects identity and composition. For example, $(\gamma_0 \circ \gamma_1).\{\mu\} = \gamma_0.\{\mu\} \circ \gamma_1.\{\mu\}$. It is also functorial in $\mathcal{M}$, so we have $\gamma.\{\mu \circ \nu\} = \gamma.\{\mu\}.\{\nu\}$. In short, we have a functorial assignment of functors to modalities.

We previously also mentioned that $-.\{\mu\}$ sends a 2-cell $\alpha : \nu \Rightarrow \mu$ to a natural transformation. This is effected by postulating a natural transformation with components $\{\alpha\}_\Gamma : \Gamma.\{\mu\} \longrightarrow \Gamma.\{\nu\} \mathbin{@} n$ at $\Gamma\ \mathsf{cx} \mathbin{@} m$. Notice that the action on 2-cells is also contravariant, so that the substitution $\{\alpha\}_\Gamma$ induces a function $\langle \nu \mid A \rangle \to \langle \mu \mid A \rangle$ in the type theory.

Finally, FitchTT requires that each $-.\{\mu\}$ be a parametric right adjoint. We have already discussed the left adjoint $-/(- : \mu)$ on contexts as well as the unit and counit of the nascent parametric adjunction (Figure 2). It therefore only remains to extend the substitution calculus with the functorial action of $-/(- : \mu)$ (FITCH/SB/MRES) and to ensure that the unit and counit satisfy the triangle equalities (FITCH/SB/FIRST-TRIANGLE-EQ and FITCH/SB/SECOND-TRIANGLE-EQ). These rules are complicated by the fact that $-/(- : \mu)$ is a functor not into the category of contexts and substitutions, but rather into the slice category above $\mathbf{1}.\{\mu\}$. This explains the appearance of δ in the domain of δ/μ in the FITCH/SB/MRES as well as FITCH/SB/UNIT-SLICE-EQ, which states that $\eta[r]$ is a map between two objects in a slice category and therefore part of a commuting triangle.

$$\frac{\Gamma\ \mathsf{cx} \mathbin{@} m}{\Gamma.\{\mathrm{id}_m\} = \Gamma\ \mathsf{cx} \mathbin{@} m}\ \textsc{fitch/cx/id}
\qquad
\frac{\nu : o \longrightarrow n \qquad \mu : n \longrightarrow m \qquad \Gamma\ \mathsf{cx} \mathbin{@} m}{\Gamma.\{\mu \circ \nu\} = \Gamma.\{\mu\}.\{\nu\}\ \mathsf{cx} \mathbin{@} o}\ \textsc{fitch/cx/comp}$$

$$\frac{\mu : n \longrightarrow m \qquad \delta : \Gamma \longrightarrow \Delta \mathbin{@} m}{\delta.\{\mu\} : \Gamma.\{\mu\} \longrightarrow \Delta.\{\mu\} \mathbin{@} n}\ \textsc{fitch/sb/mex}
\qquad
\frac{\mu_0, \mu_1 : n \longrightarrow m \qquad \alpha : \mu_0 \Rightarrow \mu_1 \qquad \Gamma\ \mathsf{cx} \mathbin{@} m}{\{\alpha\}_\Gamma : \Gamma.\{\mu_1\} \longrightarrow \Gamma.\{\mu_0\} \mathbin{@} n}\ \textsc{fitch/sb/coe}$$

$$\frac{\mu : n \longrightarrow m \qquad \delta : \Gamma \longrightarrow \Delta \mathbin{@} n \qquad r : \Delta \longrightarrow \{\mu\} \mathbin{@} m}{\delta/\mu : \Gamma/(r \circ \delta : \mu) \longrightarrow \Delta/(r : \mu) \mathbin{@} m}\ \textsc{fitch/sb/mres}$$

$$\frac{\mu : n \longrightarrow m \qquad \Gamma\ \mathsf{cx} \mathbin{@} n \qquad r : \Gamma \longrightarrow \{\mu\} \mathbin{@} m}{!.\{\mu\} \circ \eta[r] = r : \Gamma \longrightarrow \{\mu\} \mathbin{@} n}\ \textsc{fitch/sb/unit-slice-eq}$$

$$\frac{\Gamma\ \mathsf{cx} \mathbin{@} n \qquad \mu : n \longrightarrow m \qquad r : \Gamma \longrightarrow \{\mu\} \mathbin{@} n}{\epsilon[\Gamma/(r : \mu)] \circ (\eta[r]/\mu) = \mathrm{id} : \Gamma/(r : \mu) \longrightarrow \Gamma/(r : \mu) \mathbin{@} m}\ \textsc{fitch/sb/first-triangle-eq}$$

$$\frac{\Gamma\ \mathsf{cx} \mathbin{@} m \qquad \mu : n \longrightarrow m}{(\epsilon[\Gamma].\{\mu\}) \circ \eta[\mathbf{1}.\{\mu\}] = \mathrm{id} : \Gamma.\{\mu\} \longrightarrow \Gamma.\{\mu\} \mathbin{@} m}\ \textsc{fitch/sb/second-triangle-eq}$$

Fig. 4. Selected rules for multiple modalities and modal substitutions.

3.6 Some Simple Examples

As an example of using the type theory, we show that we can construct type-theoretic equivalences [38, Section 4] that weakly mirror the structure of the mode theory $\mathcal{M}$ within FitchTT. In particular, we show that $\langle \mu \circ \nu \mid A\rangle \simeq \langle \mu \mid \langle \nu \mid A\rangle\rangle$ and $\langle \mathrm{id}_m \mid A\rangle \simeq A$ for appropriate modalities and modes μ, ν, m. Finally, we show that each 2-cell $\alpha : \nu \Rightarrow \mu$ of $\mathcal{M}$ induces a natural transformation $\langle \nu \mid A\rangle \to \langle \mu \mid A\rangle$.

We can straightforwardly construct a function

$$\mathbf{comp}_{\mu,\nu} : \langle \mu \circ \nu \mid A\rangle \to \langle \mu \mid \langle \nu \mid A\rangle\rangle$$

by using fitch/tm/unmod-dra and, crucially, fitch/cx/comp:

$$\mathbf{comp}_{\mu,\nu} \triangleq \lambda(\mathrm{mod}_\mu(\mathrm{mod}_\nu(\mathbf{v}_0[\epsilon] \mathbin{@} \mathbf{m}_{\mu\circ\nu})))$$

This can be shown to be an equivalence. Similary, $\mathbf{triv}(-) : \langle \mathrm{id} \mid A\rangle \simeq A$ follows from fitch/cx/id:

$$\mathbf{triv} \triangleq \lambda(\mathbf{v}_0[\epsilon] \mathbin{@} \mathbf{m}_{\mathrm{id}})$$

To construct a natural transformation $\langle \mu \mid A\rangle \to \langle \nu \mid A\rangle$, we must use the 2-functorial features of the substitution calculus. Specifically, given a 2-cell $\alpha : \mu \Rightarrow \nu$, recall that there is a substitution $\{\alpha\}_\Gamma : \Gamma.\{\nu\} \longrightarrow \Gamma.\{\mu\} \mathbin{@} n$, which serves as the crucial component of the coercion function:

$$\mathbf{coe}[\alpha] : \langle \nu \mid A\rangle \to \langle \mu \mid A\rangle$$
$$\mathbf{coe}[\alpha] = \lambda(\mathrm{mod}_\mu(\mathbf{v}_0[\epsilon \circ \{\alpha\}/\mu] \mathbin{@} \mathbf{m}_\nu \circ \{\alpha\}))$$

Already with definitions like $\mathbf{coe}[\alpha]$, the explicit core calculus syntax can become unwieldy. When working on paper, it is often convenient to repurpose constructs like DRA/TM/UNMOD, which are ill behaved, as derivable rules. These operations cannot be internalized—they are not stable under substitution—and we do not expect them to be directly incorporated into the surface syntax of an implementation of FitchTT. Rather than developing a user-friendly syntax in the present work, however, we opt to treat work rules like DRA/TM/UNMOD as convenient shorthands or 'macros' for common patterns. For instance, we may generalize DRA/TM/UNMOD to bake in a 2-cell directly:

$$\frac{\mu : o \longrightarrow m \qquad \nu : m \longrightarrow n \qquad \xi : o \longrightarrow n \qquad \alpha : \nu \circ \mu \Rightarrow \xi \qquad \Gamma.\{\nu\} \vdash M : \langle \mu \mid A \rangle \mathbin{@} m}{\Gamma.\{\xi\} \vdash \mathsf{unmod}_{\mu,\alpha}(M) \triangleq \mathsf{unmod}_{\mu}(M)[\{\alpha\}_{\Gamma}] : A \mathbin{@} o}$$

With this macro to hand, we can quickly define the modal composition operator $\mathbf{coe}[\alpha](M) \triangleq \mathsf{mod}_{\nu}(\mathsf{unmod}_{\mu,\alpha}(M))$.

4 SEMANTICS

FitchTT is already given as a generalized algebraic theory and so automatically induces a category of models (algebras and strict homomorphisms). In this section, we aim to restructure that definition in terms of more malleable categorical gadgets. We immediately reap the rewards of this effort by showing how to construct models of FitchTT from adjunctions between presheaf categories, which we use to present various instances of the type theory in Sections 5 and 6. Finally, we relate FitchTT to previous Fitch-style type theories. More specifically, if we equip it with the mode theory generated by a single endomodality, FitchTT is a conservative extension of DRA. We are able to prove this in a syntax-free manner using only the algebraic and categorical structure of the model.

4.1 Natural Models of Type Theory

Each mode of FitchTT includes a completely independent Martin-Löf type theory. There are many equivalent ways of presenting a model of MLTT, but for the purposes of this article we use *natural models* [3, 21], which are a categorical reformulation of *categories with families* [20]. This choice has no essential impact on our results. However, the language of natural models allows for a succinct approach to models of FitchTT, encoding numerous equations as a handful of commuting squares.

Natural models are based around the concept of a representable natural transformation.

Definition 4. A *representable natural transformation over* C is a morphism $u : \dot{U} \longrightarrow U : \mathbf{PSh}(C)$ such that the pullback of u along any morphism $\mathbf{y}(\Gamma) \longrightarrow U$ is representable.

A representable natural transformation of presheaves over a context category is a concise way of encoding the type and term families of CwFs. The category C represents contexts and substitutions. Each $U(\Gamma)$ is the set of types in context $\Gamma : C$. These sets organise into a presheaf under the action of substitution on types. Similarly, the sets $\dot{U}(\Gamma)$ of terms in context Γ also constitute a presheaf under substitution. The morphism u projects a term onto its type, and is thus natural under substitution. Finally, the pullback condition encodes the universal property of context extension.

Remarkably, this basic vocabulary enables a concise encoding of the various requirements for interpreting the connectives of type theory in a highly categorical style. Just as with CwFs, the various connectives may be specified independently on top of the representable natural transformation. Thus, a natural model of type theory in C is given by a particular representable natural transformation $u : \dot{U} \longrightarrow U$ in $\mathbf{PSh}(C)$ equipped with various additional pieces of structure.

Further expository material as well as an in-depth discussion of natural models may be found in the paper by Awodey [3]. In the rest of the section, we will focus on the novel modal types.

4.2 Natural Models and Dependent Adjunctions

Dependent adjunctions can be phrased in the language of natural models [23, Section 7.1]. First, notice that the restriction to one category in the original definition of dependent adjunctions is artificial: the same definition works between any two natural models of type theory. Fix two natural models, $u\colon \dot{U} \longrightarrow U$ in $\mathbf{PSh}(\mathcal{C})$ and $v\colon \dot{V} \longrightarrow V$ in $\mathbf{PSh}(\mathcal{D})$. The left adjoint of the dependent adjunction is a functor $L\colon \mathcal{D} \longrightarrow \mathcal{C}$. However, the dependent right adjoint from u to v has actions on types and terms that may be exactly encoded by a pullback square in $\mathbf{PSh}(\mathcal{D})$:

$$\begin{tikzcd} L^*\dot{U} \arrow[r, "\mathsf{mod}"] \arrow[d, "L^*u"'] \arrow[dr, phantom, "\lrcorner", very near start] & \dot{V} \arrow[d, "v"] \\ L^*U \arrow[r, "\mathsf{Mod}"'] & V \end{tikzcd} \tag{1}$$

The left adjoint L induces a functor $L^* : \mathbf{PSh}(\mathcal{C}) \to \mathbf{PSh}(\mathcal{D})$ by precomposition. Applying this to the family u yields a family of types and terms in contexts of the form $\Gamma.\text{🔒} \triangleq L(\Gamma)$ for $\Gamma \in \mathcal{D}$. The formation rule and introduction rule, which are interpreted by Mod and mod, respectively, map such types and terms to types and terms of the family v. The requirement that mod and Mod be natural transformations encodes that these operations respect substitution.

The universal property of the pullback suffices to interpret the elimination rule. Fix a context $\Gamma : \mathcal{D}$, and consider the representable presheaf $\mathbf{y}(\Gamma) : \mathbf{PSh}(\mathcal{D})$. By the Yoneda lemma, morphisms $M\colon \mathbf{y}(\Gamma) \longrightarrow \dot{V}$ and $A\colon \mathbf{y}(\Gamma) \longrightarrow L^*U$ correspond to a term M over context Γ in $\mathcal{D}$ and a type A over context $L(\Gamma)$ in $\mathcal{C}$. Now, suppose that the following square commutes:

$$\begin{tikzcd} \mathbf{y}(\Gamma) \arrow[dr, dotted, "N"] \arrow[drr, bend left, "M"] \arrow[ddr, bend right, "A"'] & & \\ & L^*\dot{U} \arrow[r, "\mathsf{mod}"] \arrow[d] \arrow[dr, phantom, "\lrcorner", very near start] & \dot{V} \arrow[d] \\ & L^*U \arrow[r, "\mathsf{Mod}"'] & V \end{tikzcd}$$

The outer square encodes that the term M has type $\mathsf{Mod}(A)$ in context Γ.

The universal property of the pullback square yields a morphism $N\colon \mathbf{y}(\Gamma) \longrightarrow L^*\dot{U}$, which, by Yoneda, is precisely a term in context $L(\Gamma)$. The leftmost triangle says that $L(\Gamma) \vdash N : A$, whereas the top triangle ensures that $\Gamma \vdash \mathsf{mod}(N) = M : \mathsf{Mod}(A)$. Thus, N is the "transpose" of M, and we write $\mathsf{unmod}(M) \triangleq N$. In this notation, commutation of the top triangle is the η law, as it encodes that $\mathsf{mod}(\mathsf{unmod}(M)) = M$. The β law follows by the uniqueness of N: if $M = \mathsf{mod}(N')$, then N' fits into the above diagram, and hence $\mathsf{unmod}(M) = N = N'$.

In fact, requiring that the above square is a pullback is not too strong. As limits are computed pointwise in presheaf categories, a square is a pullback square if and only if it satisfies the universal property with respect to all representable presheaves. Therefore, Diagram 1 is a pullback square if and only if the model satisfies the transposition-style elimination rule for the modality. We have already shown that in the presence of a PRA structure on the left adjoint, this transposition elimination rule is equivalent to the full elimination rule for FitchTT modalities. Therefore, we can summarize the full set of requirements of a FitchTT modality in a model as (1) a PRA L between context categories and (2) the existence of Mod and mod fitting into Diagram 1.

We note that this is strongly reminiscent of Voevodsky's notion of universe morphism [39, Section 4].

4.3 Models of FitchTT

The definition of a model of FitchTT assembles mode-local models and modalities into a 2-functor: the zero-dimensional component selects the mode-local model, the one-dimensional action selects the modal context operators, and the two-dimensional action selects appropriate natural transformations. Each modal context operator comes with a dependent right adjoint and is required to be a parametric right adjoint.

Definition 5. A model of FitchTT over the mode theory $\mathcal{M}$ consists of a 2-functor $[\![-]\!] : \mathcal{M}^{\text{coop}} \longrightarrow \mathbf{Cat}$ such that

- For each $m : \mathcal{M}$, there is a natural model $u_m : \dot{U}_m \longrightarrow U_m$ in $\mathbf{PSh}([\![m]\!])$ closed under dependent products, sums, identity types.
- For each $\mu : n \longrightarrow m$, there is a dependent right adjoint from u_n to u_m as explained in Section 4.2 whose left adjoint is given by $[\![\mu]\!]$.
- Finally, each $[\![\mu]\!]$ is a parametric right adjoint.

The category of models of FitchTT with $\mathcal{M}$ has models for objects, and strict morphisms preserving all connectives and operations on-the-nose for morphisms.

As this definition of model is a repackaging of the standard notion of model given by the definition of FitchTT as a generalized algebraic theory, we know that

Example 6. There is an initial model $\mathbb{S}[-]$ of FitchTT given by derivations in the GAT quotiented by definitional equality, which we call the *syntax*. More precisely, $\mathbb{S}[m]$ is the category of contexts and substitutions at mode m, while $\mathbb{S}[\mu]$ and $\mathbb{S}[\alpha]$ respectively become $-.\{\mu\}$ and $\{\alpha\}_{-}$.

4.4 Relationships to Other Fitch-style Type Theories

The initiality of syntax is a powerful tool for relating FitchTT with other type theories. More specifically, if we are able to show that another type theory $\mathbb{T}$ is a model of FitchTT with $\mathcal{M}$, then initiality induces a unique morphism from the syntax of FitchTT to that type theory. This morphism is then a *translation* of FitchTT into $\mathbb{T}$.

For example, we can relate the DRA calculus to FitchTT. First, generate the free mode theory of a single modality: Start with a single mode m and a single morphism $\mu : m \longrightarrow m$ and generate the free (strict) 2-category. Then,

THEOREM 3. FitchTT *with a single endomodality μ is a conservative extension of* DRA.

PROOF. By definition, a model of DRA is a model of FitchTT if and only if the functor $-.$🔒 is a parametric right adjoint. Thus, every model of FitchTT is a model of DRA. Moreover, every morphism of FitchTT models is *a fortiori* a morphism of DRA models (as the latter is a weaker theory than the former).

Consider the free algebra of DRA. While not explicitly stated as such, $-.$🔒 is a parametric right adjoint. The left adjoint to this parametric adjoint is given by "deleting up to a 🔒" so that explicitly we send $\Gamma.$🔒$.\Delta$ to Γ (with 🔒 $\notin \Delta$). The definition of non-standard definition of substitution given by Birkedal et al. [9], $\Gamma \longrightarrow \Delta$ are *silently* given this type, so $-.$🔒 is a PRA with the identity substitution serving as both unit and counit. Showing that this overloading is coherent and that substitution can be given an action on the syntactic model proven by induction on substitutions [9, Lemma 10]. Therefore, the syntax of DRA is a model of FitchTT with $-.\{\mu\} \triangleq -.$🔒. This induces a unique morphism F from the syntax $\mathbb{S}[-]$ of FitchTT into the syntax of DRA.

Conversely, there is a unique morphism G from the syntax of DRA into $\mathbb{S}[-]$. But by our previous observation, F is also a morphism of DRA models, so $F \circ G$ is a morphism of DRA models from the syntax of DRA to itself and therefore must be the identity. Hence, G faithfully embeds DRA into FitchTT. □

Consequently, the addition of the $-/(- : \mu)$ operator on contexts does not change the strength of the type theory in the case of a single endomodality.

This technique extends to other Fitch-style type theories. For example, consider the mode theory $\mathcal{M}_\Box$ consisting again of a single mode m and endomodality $\mu\colon m \longrightarrow m$, but force $\mu \circ \mu = \mu$ and include a 2-cell $\mu \Rightarrow \mathrm{id}$ such that $\mathcal{M}_\Box$ is the walking idempotent comonad. The exact same technique can be used with the type theory $\mathsf{MLTT}_{🔒}$ [25] to prove that

Theorem 4. *FitchTT with the mode theory $\mathcal{M}_\Box$ can be embedded into* $\mathsf{MLTT}_{🔒}$.

This again relies on the fact $-.🔒$ is a parametric right adjoint, which was once more a lemma of the metatheory [26, Lemma 1.2.11]. However, FitchTT with $\mathcal{M}_\Box$ is *not* a conservative extension of $\mathsf{MLTT}_{🔒}$, for the latter proves some nonstandard theorems of modal logic, e.g., $(A \to \Box B) \to \Box(A \to B)$. In particular, FitchTT instantiated with $\mathcal{M}_\Box$ is not a model for $\mathsf{MLTT}_{🔒}$, so only one of the two morphisms of models used in the proof of Theorem 3 can be recovered in this case.

4.5 Presheaf Models

We now give a theorem for constructing the most important class of non-syntactic models of FitchTT, viz. presheaf categories with adjunctions between them. First, we recall from Hofmann [27] that any presheaf category $\mathbf{PSh}(C)$ supports a model of Martin-Löf type theory. Accordingly, we only need to show how various well-known adjunctions that are induced between presheaf categories can be used to model FitchTT.

To begin, we recall some standard facts about presheaves. Any functor $f : C \longrightarrow \mathcal{D}$ induces an adjoint triple $f_! \dashv f^* \dashv f_*$. The middle functor $f^*\colon \mathbf{PSh}(\mathcal{D}) \longrightarrow \mathbf{PSh}(C)$ is defined by precomposition, i.e., $f^*(X)(c) \triangleq X(f(c))$. The other two functors $f_!$ and f_* are given by Kan extension [2, Section 9.6].

We will show that the two left adjoints $f_!$ and f^* may be used to interpret the context operator $-.\{\mu\}$. We have shown in previous work that their corresponding right adjoints f^* and f_* extend to dependent right adjoints [23, Section 7]. Thus, to satisfy Definition 5 it remains to show that the left adjoints themselves are PRAs. This is trivial for f^*, as every right adjoint is a PRA. However, $f_!$ is not always a PRA, but it is whenever f itself is a PRA between C and $\mathcal{D}$:

Lemma 5. *If $f : C \longrightarrow \mathcal{D}$ is a PRA, then so is $f_! : \mathbf{PSh}(C) \longrightarrow \mathbf{PSh}(\mathcal{D})$.*

When putting these together into a model of FitchTT a coherence problem arises. The definition requires $[\![\mu]\!] \circ [\![\nu]\!] = [\![\mu \circ \nu]\!]$, but in general we only have $f_! \circ g_! \cong (f \circ g)_!$. This means that a putative model in which $-.\{\mu\}$ and $-.\{\nu\}$ are interpreted by $f_!$ and $g_!$ will not satisfy fitch/cx/comp.

This strictness mismatch is addressed by a strictification theorem for MTT [24] adapted to FitchTT. This strictification theorem is considerably simpler than strictification of substitution, and it essentially follows from the standard categorical result replacing a pseudofunctor by a strict 2-functor up to 2-equivalence.

These considerations lead us to the following theorem, which states that well-behaved adjunctions of the form $f_! \dashv f^*$ and $f^* \dashv f_*$ are models of FitchTT.

Theorem 6. *Fix a pseudofunctor $F\colon \mathcal{M}^{\mathrm{coop}} \longrightarrow \mathbf{Cat}$ such that $F(m) = \mathbf{PSh}(C_m)$ for each $m : \mathcal{M}$, and for each $\mu\colon n \longrightarrow m$ the functor $F(\mu)$ satisfies one of the following two conditions:*

(1) *$F(\mu) = f_!$ for a PRA $f : C_m \longrightarrow C_n$.*
(2) *$F(\mu) = f^*$ for an arbitrary functor $f : C_n \longrightarrow C_m$.*

Then there exists a model of FitchTT *with mode theory $\mathcal{M}$ where each mode m is modelled by $F(m) = \mathbf{PSh}(C_m)$ and each modality μ by the dependent right adjoint of $F(\mu)$.*

5 PARAMETRIC TYPE THEORY AND FITCHTT

As we saw in Section 2.2, a simple source of parametric right adjoints is the cartesian product: given a closed type $\vdash \mathfrak{A}$ type, the context extension operator $-.\mathfrak{A}$ is a parametric right adjoint, and we have get types in the form of the function type former $\mathfrak{A} \to (-)$. In this section we examine how this picture generalizes to *substructural* function types from a fixed object. Concretely, we examine Bernardy et al.'s *parametric type theory* [7], which relies on affine variables supporting weakening and exchange but not contraction. We find that their *parametricity types* can be seen as modal types in FitchTT. Although completely capturing all aspects of parametric type theory requires more than modal types, FitchTT neatly resolves the issues of substitution that arise from the new variables.

5.1 Parametric Type Theory

Bernardy et al.'s parametric type theory extends Martin-Löf type theory with new primitives that make parametricity theorems internally provable. As an example, it becomes possible to show *within the type theory* that any polymorphic function $(A : \mathsf{U}) \to A \to A$ is identified with the polymorphic identity function. This is possible in part due the introduction of a form of substructural variable, variously called a *color* or *dimension* variable. These variables are *affine*: they support weakening and exchange but not contraction.

In this theory we may extend a context Γ by a dimension variable $i : \mathbb{I}$. Given a context of the form $(\Gamma, i : \mathbb{I})$, we may think of the assumptions in Γ as being *separated* from i. In particular, we cannot use one dimension variable to instantiate two: There is no "diagonal" substitution from $\Gamma, i : \mathbb{I} \longrightarrow \Gamma, j : \mathbb{I}, k : \mathbb{I}$, as this would invalidate the separation of j from k. Finally, in any context, we have a dimension constant $\Gamma \vdash 0 : \mathbb{I}$.

A family $\Gamma, i : \mathbb{I} \vdash A$ type is to be thought of as a predicate on its "endpoint" $A[0/i]$. Likewise, an element $\Gamma, i : \mathbb{I} \vdash M : A$ is a witness that its endpoint $M[0/i]$ satisfies the predicate A. Dimension quantification is internalized by *parametricity types*, whose elements are abstracted terms with a fixed endpoint. The formation and introduction rules for these types are given as follows[2]:

$$\frac{\Gamma, i : \mathbb{I} \vdash A \text{ type} \qquad \Gamma \vdash M : A[0/i]}{\Gamma \vdash \mathsf{Pred}(i.A, M) \text{ type}} \qquad \frac{\Gamma, i : \mathbb{I} \vdash M : A}{\Gamma \vdash \lambda i.\, M : \mathsf{Pred}(i.A, M[0/i])}$$

The idea is that an element of $\mathsf{Pred}(i.A, M)$ is a witness that M belongs to the predicate represented by $i.A$. This intuition that types over $\mathbb{I}$ correspond to predicates is justified by an equivalence $\mathsf{Pred}(i.\mathsf{U}, A) \simeq (A \to \mathsf{U})$, the inverse map of which is effected by an additional *colored type pair* connective [7, Theorem 3.1]. The fact that predicates may be represented by affine functions from $\mathbb{I}$ then implies that all constructions on types have an action on predicates, a form of parametricity.

The application rule given for parametricity types in Bernardy et al. [7] enforces the "no-diagonal" restriction by assuming a fresh variable in the conclusion,

$$\frac{\Gamma \vdash P : \mathsf{Pred}(i.A, M)}{\Gamma, i : \mathbb{I} \vdash P \mathbin{@} i : A} \qquad \frac{\Gamma \vdash P : \mathsf{Pred}(i.A, M)}{\Gamma \vdash P \mathbin{@} 0 = M : A[0/i]}$$

One is thus prevented from writing $(P \mathbin{@} i) \mathbin{@} i$. As we have seen with the corresponding rule DRA/TM/UNMOD, this creates a theory where substitution is not admissible.

[2]In Bernardy, Coquand, and Moulin's notation, the type $\mathsf{Pred}(i.A, M)$ is written $A \ni_i M$.

Cavallo and Harper [16] introduce a cubical parametric type theory with a *dimension restriction* operator, following Cheney's approach to nominal type theory [17],

$$\frac{\Gamma \vdash r : \mathbb{I} \qquad \Gamma/(r : \mathbb{I}) \vdash P : \mathsf{Pred}(i.A, M)}{\Gamma \vdash P \mathbin{@} r : A}$$

When r is a variable the restriction $\Gamma/(r : \mathbb{I})$ removes r and terms succeeding it from the context. When it is a constant, the restriction is the identity i $\Gamma/(0 : \mathbb{I}) \triangleq \Gamma$. The admissibility of substitution then relies on the existence of a functorial action by restriction: given $\sigma \colon \Gamma \longrightarrow \Delta$ and $\Delta \vdash r : \mathbb{I}$, there is some $\sigma/\mathbb{I} \colon \Gamma/(r[\sigma] : \mathbb{I}) \longrightarrow \Delta/(r : \mathbb{I})$ computed by induction on σ.

5.2 Recovering Parametric Type Theory

We now show that the judgmental structure of parametric type theory—viz. dimension variables and a parametricity type internalizing them—can be recovered as an instance of FitchTT. On its own, this instance is insufficient to reconstruct, for example, the proof that all functions $(A : U) \to A \to A$ are equal to the identity. It does, however, provide the basis on which the necessary additional structure can be specified, resolving the technical issues around substitution and affine dimension variables. The remainder may be found in Cavallo [15, Chapter 11].

To cast the kernel of parametric type theory as an instance of FitchTT, we first decompose $\mathsf{Pred}(i.A, M)$ into a combination of an identity type and an affine function type, $(i : \mathbb{I}) \multimap A$, similar to $\mathsf{Pred}(i.A, M)$ but with no fixed endpoint:

$$\mathsf{Pred}(i.A, M) \triangleq (p : (i : \mathbb{I}) \multimap A) \times \mathsf{Id}_{A[0/i]}(p \mathbin{@} 0, M)$$

This encoding will not satisfy the definitional η-principle enjoyed by primitive parametricity types, but it suffices for proving parametricity theorems.[3] The $(i : \mathbb{I}) \multimap -$ connective is then specified by the following rules:

$$\textsc{ptt/ty/aff-form}\ \frac{\Gamma, i : \mathbb{I} \vdash A\ \mathsf{type}}{\Gamma \vdash (i : \mathbb{I}) \multimap A\ \mathsf{type}} \qquad \textsc{ptt/tm/aff-intro}\ \frac{\Gamma, i : \mathbb{I} \vdash M : A}{\Gamma \vdash \lambda i.\, M : (i : \mathbb{I}) \multimap A} \qquad \textsc{ptt/tm/aff-elim}\ \frac{\Gamma \vdash r : \mathbb{I} \qquad \Gamma/(r : \mathbb{I}) \vdash P : (i : \mathbb{I}) \multimap A}{\Gamma \vdash P \mathbin{@} r : A}$$

We formulate $(i : \mathbb{I}) \multimap A$ as a modal type in FitchTT specialized with the mode theory $\mathcal{M}_{\mathsf{aff}}$, for which see Figure 5. We use a single mode m with a modality $\mu \colon m \longrightarrow m$, intending to replace context extension by $\mathbb{I}$ with $-.\{\mu\}$. The choice of 2-cells and equations corresponds to the structural rules supported by dimension variables: we have 2-cells for weakening (w) and exchange (e) but none for contraction. Note that the presentation in the mode theory is the "reverse" of what one might expect, because we axiomatize the behavior of affine *functions*. For instance, weakening is a 2-cell $w : \mathsf{id} \Rightarrow \mu$ and not $w : \mu \Rightarrow \mathsf{id}$. We add a "face map" $f : \mu \Rightarrow \mathsf{id}$ to induce the dimension constant 0, then obtained as $0 \triangleq \{f\}_1 \circ !_\Gamma : \Gamma \longrightarrow \{\mu\} \mathbin{@} m$. Equations specify the interactions among the structural rules and face map. For example, the equation $f \circ w = \mathsf{id}$ states that weakening and then substituting 0 for the new variable has no effect, while the Yang-Baxter equation on 2-cells $\mu \circ \mu \circ \mu \Rightarrow \mu \circ \mu \circ \mu$ relates two patterns of overlapping exchanges. This theory gives $(m \longrightarrow m)^{\mathsf{op}}$ the structure of a symmetric monoidal category with terminal object id [32, Proposition 3.1].[4] Indeed, it is the free such category generated by an object μ with face map $f : \mu \Rightarrow \mathsf{id}$.

[3] We conjecture that we could have our cake and eat it too by generalizing FitchTT's modal types to match Riehl and Shulman's *extension types* [35], allowing them to require a definitional equality on some boundary. We could then encode $\mathsf{Pred}(i.A, M)$ directly instead of going through $(i : \mathbb{I}) \multimap A$.

[4] This structure could be disrupted by further additions to the mode theory, for example of a second modality not commuting with μ.

Modes and modalities:

$$m \qquad\qquad \overset{\text{DIMENSION}}{\mu\colon m \longrightarrow m}$$

2-cells:

$$\overset{\text{WEAKENING}}{w : \mathsf{id} \Rightarrow \mu} \qquad \overset{\text{EXCHANGE}}{e : \mu \circ \mu \Rightarrow \mu \circ \mu} \qquad \overset{\text{FACE}}{f : \mu \Rightarrow \mathsf{id}}$$

Equations:

$$e \circ (\mathsf{id} \bullet w) = w \bullet \mathsf{id} : \mu \circ \mu \Rightarrow \mu \qquad\qquad e \circ e = \mathsf{id} : \mu \Rightarrow \mu$$

$$(e \bullet \mathsf{id}) \circ (\mathsf{id} \bullet e) \circ (e \bullet \mathsf{id}) = (\mathsf{id} \bullet e) \circ (e \bullet \mathsf{id}) \circ (\mathsf{id} \bullet e) : \mu \circ \mu \circ \mu \Rightarrow \mu \circ \mu \circ \mu$$

$$f \circ w = \mathsf{id} : \mathsf{id} \Rightarrow \mathsf{id} \qquad\qquad (\mathsf{id} \bullet f) \circ e = f \bullet \mathsf{id} : \mathsf{id} \to \mathsf{id}$$

Fig. 5. $\mathcal{M}_{\mathsf{aff}}$: A mode theory for affine functions.

The affine function type is now given by $\mathbb{I} \multimap A \triangleq \langle \mu \mid A \rangle$ (here we switch to a variable-free presentation). With this definition, the operations and equations for the affine line $\mathbb{I} \multimap A$ can be recovered directly from the rules of modal types in FitchTT, with the modal restriction operation induced by the PRA structure on $-.\{\mu\}$ playing the role of dimension restriction. More precisely, PTT/TY/AFF-FORM becomes FITCH/TY/MOD, PTT/TM/AFF-INTRO becomes FITCH/TM/MOD and PTT/TM/AFF-ELIM becomes FITCH/TM/UNMOD.

Note that the correspondence between Cavallo and Harper's $\Gamma/(r : \mathbb{I})$ and the modal restriction operation of FitchTT is not exact. In Parametric FitchTT, we can only show that Γ is a retract of $\Gamma/(0 : \mu)$, which is weaker than the equation $\Gamma/(0 : \mathbb{I}) = \Gamma$ of Cavallo and Harper [16]. The lack of this equation is not, however, an obstacle to the practice of internal parametricity.

5.3 Models

We can straightforwardly obtain a presheaf model for this instance of FitchTT by way of Theorem 6. In fact, the model we construct will interpret not only dimension variables but also additional axioms necessary to obtain internal parametricity theorems. In Bernardy et al. [7], these theorems rely on the fact that the canonical map $\mathsf{Pred}(-,-) : \mathsf{Pred}(\_.\mathsf{U}, A) \to (A \to \mathsf{U})$ is an equivalence. In fact, it is sufficient for many purposes to work with a "weak inverse" that cancels it only up to equivalence rather than equality. By asking only for a weak inverse, we are able to give a simpler model than that constructed in Bernardy et al. [7]. In particular, contexts may be taken to be presheaves over the following category, rather than the "refined presheaves" over the same used there.[5]

Definition 7. Define $\mathbf{pI}$ to be the category whose objects are finite sets and whose morphisms $S \to T$ are functions $g : T \to S + \mathbf{1}$ that, when restricted to the preimage of S, are injective.

We think of a set S as standing for a context of $|S|$ dimension variables; a function g as above is a substitution assigning to each variable in T either a variable in S or the dimension constant.

The empty set $\emptyset$ is a zero object of $\mathbf{pI}$: It is both initial and terminal. There is a functor $F\colon \mathbf{pI} \longrightarrow \mathbf{pI}$ that takes a set S to $S + \mathbf{1}$; by extending this to a functor on the presheaf category $F_!\colon \mathbf{PSh}(\mathbf{pI}) \longrightarrow \mathbf{PSh}(\mathbf{pI})$ we can interpret extension by a dimension variable as $[\![\Gamma.\{\mu\}]\!] \triangleq F_![\![\Gamma]\!]$.

[5]The refined presheaf model *is* a model of this instance of FitchTT but not as a consequence of Theorem 6.

To apply Theorem 6, it remains to show that (1) the 2-cells exist and their equations are satisfied and (2) F is a PRA, so that $F_!$ is as well. The first is a routine calculation: the 2-cells for weakening, exchange, and the face map are induced via $(-)_!$ by natural transformations $w : \mathrm{id} \Rightarrow F$, $e : F \circ F \Rightarrow F \circ F$, and $f : F \Rightarrow \mathrm{id}$ in $\mathbf{pI}$ respectively. The second follows by defining a left adjoint $G\colon \mathbf{pI}/F(\emptyset) \longrightarrow \mathbf{pI}$ as follows:

$$G(s : S \to F(\emptyset)) \triangleq \begin{cases} S \setminus s(\star) & \text{if } s(\star) \in S \\ S & \text{if } s(\star) \in \mathbf{1}_{\mathbf{Set}} \end{cases}$$

Note that s above is a set-theoretic function $\mathbf{1}_{\mathbf{Set}} \longrightarrow S + \mathbf{1}_{\mathbf{Set}}$.

Now we can apply Theorem 6 together with the results of Bernardy et al. [7] and Cavallo and Harper [16] to obtain:

Theorem 7. *There is a model of Parametric* FitchTT *in* $\mathbf{PSh}(\mathbf{pI})$*, which interprets* $-.\{\mu\}$ *by* $F_!$*. Moreover, in this model there is a weak inverse to the canonical map* $\mathsf{Pred}(-,-) : \mathsf{Pred}(\_.\mathsf{U}, A) \to (A \to \mathsf{U})$.

Summarizing, this model ensures that one may soundly postulate a weak inverse to the canonical projection $\mathsf{Pred}(\_.\mathsf{U}, A) \to (A \to \mathsf{U})$ in FitchTT and, with this in hand, reproduce examples from Bernardy et al. [7] in Parametric FitchTT.

6 GUARDED TYPE THEORY AND FITCHTT

One of the motivations for modal type theories is to obtain a syntax for *guarded recursion* [11, 33]. In this section we show not only that FitchTT can be a flexible guarded type theory, but that the extra structure of parametric right adjoints gives rise to a rationalization of the *tick variables* introduced in **Clocked Type Theory (**CloTT**)** [4].

Guarded type theories support *guarded recursive definitions*. This is achieved by using modalities that explicitly control productivity, such as the *later modality* ($\triangleright$). Intuitively, $\triangleright A$ classifies data that can only be accessed after "one step of computation" has taken place. This fine control serves a similar purpose to the syntactic productivity checks used in coinductive definitions. In dependent guarded type theory, both recursive types and functions follow from a single principle, viz. *Löb induction*, an axiom of type $(\triangleright A \to A) \to A$ [10]. For instance, we can define the type of *guarded streams* $\mathsf{gStr}_A \cong A \times \triangleright \mathsf{gStr}_A$ by using Löb induction on the universe.

The $\triangleright$ modality and Löb induction comprise a useful framework for guarded definitions. However, the functions definable in this setting are *causal*, in that they proceed in lockstep with time. For example, the guarded type gStr_A does not admit a function $\mathsf{tail}_A : \mathsf{gStr}_A \to \mathsf{gStr}_A$: we can always project out the tail of a guarded stream, but it will have type $\triangleright \mathsf{gStr}_A$ instead, and we can only access that in the next step. The need to obtain fully defined, total objects (i.e., perform a definition by *coinduction*) dictates the introduction of a second modality, the *always* modality $\square$. Intuitively, $\square A$ classifies fully defined coinductive data (i.e., global sections). The usual type of streams is given by $\mathsf{Str}_A \triangleq \square \mathsf{gStr}_A$. Moreover, we expect an equivalence $\square A \simeq \square \triangleright A$.

This combination of modalities has been explored previously [19], but a simple syntax that combines them had proved elusive until recently [22, Section 9]. In the meantime a number of papers focussed on generalizing $\triangleright$ to a system of *ticks and clocks* [1, 4, 12, 31]. These systems are flexible, but have complicated semantics [31]. However, CloTT [4] presents an enticing syntax for guarded recursion, where the $\triangleright$ operator behaves almost like a function. However, these approaches are far from a parsimonious setting of two interacting modalities.

Here we show that instantiating FitchTT with a mode theory for guarded recursion gives rise to another practicable guarded type theory. Moreover, we observe that the extra structure of parametric right adjoints is precisely what is required to account for tick variables and the functional

Modalities:

$$\ell : m \longrightarrow m \qquad\qquad b : m \longrightarrow m$$

2-cells :

$$b \leq \mathrm{id} \qquad\qquad b \circ b = b \qquad\qquad \mathrm{id} \leq \ell \qquad\qquad b \circ \ell \leq b$$

Fig. 6. A mode theory for guarded recursion.

presentation of $\triangleright$. In fact, the *tick constant* introduced by Bahr et al. [4] emerges naturally from the 2-cell inducing the equivalence $\Box \triangleright A \simeq \Box A$. Hence, we obtain the first purely algebraic presentation of CloTT (though limited to a single clock) and give a semantics that is simpler than that of Mannaa et al. [31]. To focus on the purely modal aspects of guarded type theories, we will set aside considerations of Löb induction. We mention that it cannot be recovered through modal machinery in any known framework, so must be added axiomatically and justified externally. Moreover, the specialized modifications in CloTT to ensure normalization in the presence of Löb induction can be applied to this instantiation of FitchTT.

6.1 Guarded Type Theory in FitchTT

In Figure 6 we present a mode theory for guarded recursion in FitchTT. The mode theory is similar to that used with MTT in Gratzer et al. [22, Section 9], but it only uses one mode to facilitate comparison with CloTT. Note also that it is only poset-enriched: There is at most one 2-cell between any pair of modalities.

Instantiating FitchTT with this mode theory yields a modal type theory with modalities $\triangleright A \triangleq \langle \ell \mid A \rangle$, and $\Box A \triangleq \langle b \mid A \rangle$. When used with the (in)equations of the mode theory, the combinators of Section 3.6 induce standard operations. The most important is the "cancellation" of $\triangleright$ by $\Box$:

$$\mathsf{now} \triangleq \mathbf{comp}^{-1}_{b,\ell}(-) : \Box \triangleright A \to \Box A$$

The standard model of guarded recursion in $\mathbf{PSh}(\omega)$ [11] is also a model of FitchTT with this mode theory.

Theorem 8. FitchTT *with the guarded mode theory is soundly modelled by* $\mathbf{PSh}(\omega)$*, where the modality* b *is interpreted by the global sections comonad, and* ℓ *by the* $\triangleright$ *endofunctor.*

As both $\triangleright$ and $\Box$ have left adjoints given by precomposition [23, Section 9.2], the result follows from Theorem 6(2).

6.2 Tick Variables

Clocked type theory alters the context structure of MLTT to introduce *tick variables.* A tick variable provides the capability to discard a $\triangleright$ modality. We begin by considering a simplified clocked type theory, the *Ticked Type Theory (TTT)* of Mannaa et al. [31]. TTT extends MLTT with the following rules:

$$\textsc{ctt/later-form}\ \frac{\Gamma.\checkmark \vdash A\ \mathsf{type}}{\Gamma \vdash \triangleright A\ \mathsf{type}} \qquad \textsc{ctt/later-intro}\ \frac{\Gamma.\checkmark \vdash M : A}{\Gamma \vdash \lambda(M) : \triangleright A} \qquad \textsc{ctt/later-elim}\ \frac{|\Gamma_2| = k \qquad \Gamma_1 \vdash M : \triangleright A}{\Gamma_1.\checkmark.\Gamma_2 \vdash M(\boldsymbol{\alpha}_k) : A[\uparrow^{\Gamma_2}]}$$

The first two rules insinuate that $\triangleright$ is a dependent right adjoint to a tick. The elimination rule ctt/later-elim allows us to eliminate a $\triangleright$ by consuming a tick. We write $\boldsymbol{\alpha}_k$ to refer to the tick variable at the kth position in the context. This rule weakens the context by some additional assumptions Γ_2, which may contain additional tick variables. Consequently, CTT/LATER-ELIM enforces an *affine* discipline on tick variables.

We can embed TTT into guarded FitchTT. First, we interpret $\Gamma.\checkmark$ as $\Gamma.\{\ell\}$. CTT/LATER-FORM and CTT/LATER-INTRO are just FITCH/TY/MOD and FITCH/TM/MOD, respectively. The elimination rule is less immediate: CTT/LATER-ELIM is not exactly FITCH/TM/UNMOD, but it is very similar to the elimination rule DRA/TM/UNMOD* of the DRA calculus. We may thus obtain it as FITCH/TM/UNMOD followed by weakening:

$$M(\boldsymbol{\alpha}_k) \triangleq \mathsf{unmod}_\ell(M)[\uparrow^{\Gamma_2}]$$

There is one important qualitative difference with DRA: the weakening Γ_2 may also include tick variables, while in DRA the rest of the context may not include further locks. Thus in defining $\uparrow^{\Gamma_2}$ we may have to use the substitution $\Gamma.\{\ell\} \longrightarrow \Gamma$ induced by the inequality $1 \leq \ell$ to eliminate ticks.

We have therefore established that

THEOREM 9. *Ticked Type Theory can be embedded in* FitchTT.

6.3 Tick Constants

As mentioned previously, the combination of $\triangleright$ and Löb induction is not sufficiently expressive. We thus need some way of obtaining totalized, coinductive objects. Rather than introducing a second modality such as $\square$, the clocked type theory CloTT parameterizes $\triangleright$ by a *clock symbol* κ. Clock symbols may be quantified over with *clock quantification*, denoted $\forall\kappa.A$. Intuitively, each clock represents a distinct stream of time, and $\triangleright^\kappa$ only affects the clock κ. The clock quantifier is then used to "cancel a $\triangleright$," much like $\square$ does:

$$\forall\kappa.\, \triangleright^\kappa A \simeq \forall\kappa.A \tag{2}$$

The pivotal insight behind CloTT is this: Clocks allow us to recast a semantic check ("this context is constant in time") into a syntactic check ("this context does not mention a clock"). This check is performed in the rule for the *tick constant*, which in turn induces Equation (2):

$$\textsc{ctt/tm/now}\quad \frac{\Delta, \kappa; \Gamma \vdash M : \triangleright^\kappa A \qquad \kappa \notin \Gamma \qquad \kappa' \in \Delta}{\Delta; \Gamma \vdash M(\diamond)[\kappa'/\kappa] : A[\mathsf{id}.\diamond][\kappa'/\kappa]}$$

The syntactic check $\kappa \notin \Gamma$ ensures that nothing in Γ depends upon the clock κ. Hence, it is safe to eliminate $\triangleright^\kappa$, as the ticking of κ will not interfere with the term M. While this rule is sound, it is difficult to implement. Notice that κ does not appear at all in the conclusion of the rule. Accordingly, it is difficult to see how one might write down an algorithmic version of it: We would in fact need to conjure κ, M and A from just $M[\kappa'/\kappa]$ and $A[\kappa'/\kappa]$.

The same result can be achieved in guarded FitchTT in a more direct manner. Just as the $\triangleright$ modality replaces syntactic productivity checks, the $\square$ modality can be used to supplant syntactic constancy checks. In particular, a context of the form $\Gamma.\{b\}$ is "semantically constant." A term depending on $\Gamma.\{b\}$ cannot depend on any temporal aspects of data in Γ, as the $-.\{b\}$ operator prohibits access to anything that may change over time.

Moreover, the unique 2-cell $\alpha : b \circ \ell \Rightarrow b$ induces a substitution $\{\alpha\}_\Gamma : \Gamma.\{b\} \longrightarrow \Gamma.\{b\}.\{\ell\}\ @\ m$, which allows us to absorb any occurrences of ℓ following a b. This substitution and term now replace $\diamond$ and Equation (2) respectively. Using this encoding of $\diamond$ we obtain a "rationalization" of CTT/TM/NOW:

$$\frac{\Gamma.\{b\} \vdash M : \triangleright A\ @\ m}{\Gamma.\{b\} \vdash M(\diamond) \triangleq \mathsf{unmod}_{\ell,\alpha}(M) : A[\{\alpha\}_\Gamma]\ @\ m}$$

The encoding reconstructs a "single-clock" variant of CloTT. It is rich enough to allow definition by coinduction inside guarded type theory while also retaining the convenient functional syntax of CloTT. Moreover, the ingredients used to simulate CTT/TM/NOW do not suffer from the same issues as the original rule in CloTT, so that an algorithmic version of this syntax now seems achievable.

Using the primitives of FitchTT, we have shown that the more convenient syntax of (single-clock) CloTT can be systematically elaborated into semantically well-understood and well-behaved modal combinators. While prior work in guarded recursion was often centered around different sets of modal combinators, there was no translation procedure showing that the flexible and convenient "tick" syntax introduced by CloTT could be reconstituted in this form. By encoding a single-clock variant of CloTT in FitchTT, we show that one may have the best of both worlds: convenient syntax, and a simple set of modal operations.

This elaboration also provides a model in the standard semantics of guarded recursion and avoids the need for more complex clock categories. Finally, we note that non-dependent variants of (single-clock) CloTT have proven useful for modelling reactive programming [5, 6]; these calculi can also be encoded in Guarded FitchTT.

7 RELATED WORK

As it was designed to be a unifying Fitch-style modal type theory [18], FitchTT is closely related to many prior modal type theories.

The Fitch-style approach to modal types begins with the simply-typed system of Clouston [18], which was quickly adapted to the dependent type theory DRA [9]. The other two dependent systems in existence, namely $\mathsf{MLTT}_{🔒}$ [25] and CloTT [4], have already been discussed at length. FitchTT serves as either a rationalization or a generalization of each of these type theories: the PRA structure and the induced "functional" syntax given in this article is entirely novel.

Other Fitch-style type theories, which were crafted for more specialized applications, have a weaker relationship with FitchTT. For example, RaTT [5, 6] can be encoded in FitchTT, but this encoding would fail to capture many restrictions placed on modalities in order to ensure domain-specific theorems about RaTT (e.g., freedom from space leaks). We believe that, while FitchTT does not directly capture these restrictions, it can be manually adapted to give a dependent generalization of RaTT. As with Löb induction in guarded type theory, it would be necessary to extend FitchTT with specific constants.

By recognizing the central rôle of PRAs, the relationship between nominal type theory [17] and Fitch-style type theories that is suggested in Birkedal et al. [9] can be made more precise and extended to include parametric type theories [7, 16]. In particular, the discussion in Section 5 adapts *mutatis mutandis* to show that nominal type theory can be encoded in FitchTT.

Recently, MTT [22] also attempted to generalize DRA to support multiple modes and modalities, but without recognizing the PRA structure. Instead, it adopted a "pattern-matching" modal elimination rule, which is strictly weaker than DRA/TM/UNMOD.[6] In addition to being weaker, this pattern-matching eliminator introduces a prohibitive overhead in certain crucial examples. For instance, there is no way to elaborate the systematic "λ-notation" used in Section 5 to treat the modality $\mathbb{I} \multimap A$ as a function. As a result, terms that are simple to write down in FitchTT must be elaborated into complex manipulations of 2-cells in MTT. For instance, consider the following term of Parametric FitchTT:

$$\lambda i.\, \lambda j.\, \lambda k.\, \lambda l.\, M @ l @ k$$

[6] Note that the pattern-matching elimination rule of MTT can be expressed in FitchTT, so MTT can be embedded in it.

Attempting to replicate it in MTT yields

$$\begin{aligned}&\text{let } \mathsf{mod}_\mu(x_0) \leftarrow M \text{ in let } \mathsf{mod}_\mu(x_1) \leftarrow M \text{ in}\\&\text{let } \mathsf{mod}_\mu(x_2) \leftarrow M \text{ in let } \mathsf{mod}_\mu(x_3) \leftarrow M \text{ in}\\&\quad \mathsf{mod}_\mu(\mathsf{mod}_\mu(x_3^{(e \bullet \mu \bullet \mu)\circ(\mu \bullet \mu \bullet e)\circ(\mu \bullet e \bullet \mu)\circ(\mu \bullet \mu \bullet w \bullet w)}))\end{aligned}$$

Moreover, it is unclear that such a translation can be done systematically. As the MTT elimination rule is weaker than the corresponding FitchTT rule, it is necessary to make some 'non-local' modifications to a term. Worse, there is no MTT term corresponding to the FitchTT term $f : \langle \mu \mid A \rangle.\{\mu\} \vdash \mathsf{unmod}_\mu(f) : A \mathbin{@} m$. Thus, any potential elaboration algorithm must impose further restrictions on the input to ensure that such terms can be dealt with by a candidate non-local transformation.

8 CONCLUSIONS AND FUTURE WORK

In this article, we have introduced the notion of *parametric right adjoints* as a desirable universal property for context-modifying operations in type theory. We have shown that this extra property is essential for obtaining workable calculi based around dependent right adjoints. Through this observation we have generalized DRA to FitchTT, which supports multiple modes and modalities. Finally, we have shown that FitchTT can be instantiated to recover existing type theories for parametricity and guarded recursion. In the latter case, we provide a conceptual explanation and well-behaved syntax for ticks and the tick constant. In the future, we plan to develop these applications further.

Normalization and decidability of type-checking in FitchTT also offer interesting avenues for future work, and would possibly aid with implementing single-clock CloTT.

APPENDIX

A COMPLETE DEFINITION OF FITCHTT

We include the new rules of FitchTT. We have elided rules for dependent products, dependent sums, (intensional) identity types, because these are unchanged from MLTT.

A.1 Contexts, Types, and Terms

$$\boxed{\Gamma \ \mathsf{cx} \mathbin{@} m}$$

$$\frac{}{\mathbf{1}\ \mathsf{cx} \mathbin{@} m} \qquad \frac{\Gamma\ \mathsf{cx} \mathbin{@} m}{\Gamma.A\ \mathsf{cx} \mathbin{@} m} \qquad \frac{\Gamma\ \mathsf{cx} \mathbin{@} m \qquad \mu : n \longrightarrow m}{\Gamma.\{\mu\}\ \mathsf{cx} \mathbin{@} n}$$

$$\frac{\Gamma\ \mathsf{cx} \mathbin{@} n \qquad \mu : n \longrightarrow m \qquad r : \Gamma \longrightarrow \{\mu\} \mathbin{@} m}{\Gamma/(r : \mu)\ \mathsf{cx} \mathbin{@} m}$$

$$\frac{\Gamma\ \mathsf{cx} \mathbin{@} m \qquad \mu : n \longrightarrow m \qquad \nu : o \longrightarrow n}{\Gamma.\{\mu\}.\{\nu\} = \Gamma.\{\mu \circ \nu\}\ \mathsf{cx} \mathbin{@} o} \qquad \frac{\Gamma\ \mathsf{cx} \mathbin{@} m}{\Gamma.\{1\} = \Gamma\ \mathsf{cx} \mathbin{@} m}$$

$$\boxed{\Gamma \vdash A\ \mathsf{type} \mathbin{@} m \qquad \Gamma \vdash M : A \mathbin{@} m}$$

$$\frac{\Gamma.\{\mu\} \vdash A\ \mathsf{type} \mathbin{@} n \qquad \mu : n \longrightarrow m}{\Gamma \vdash \langle \mu \mid A \rangle\ \mathsf{type} \mathbin{@} m} \qquad \frac{\Gamma.\{\mu\} \vdash M : A \mathbin{@} n}{\Gamma \vdash \mathsf{mod}_\mu(M) : \langle \mu \mid A \rangle \mathbin{@} m}$$

$$\frac{\Gamma/(r:\mu) \vdash M : \langle \mu \mid A \rangle \mathbin{@} m \qquad r\colon \Gamma \longrightarrow \{\mu\} \mathbin{@} n}{\Gamma \vdash M \mathbin{@} r : A[\eta[r]] \mathbin{@} n}$$

$$\frac{\Gamma/(r:\mu).\{\mu\} \vdash M : A \mathbin{@} n \qquad r\colon \Gamma \longrightarrow \{\mu\} \mathbin{@} n}{\Gamma \vdash \mathsf{mod}_\mu(M) \mathbin{@} r = M[\eta[r]] : A[\eta[r]] \mathbin{@} n} \qquad \frac{\Gamma \vdash M : \langle \mu \mid A \rangle \mathbin{@} m}{\Gamma \vdash M = \mathsf{mod}_\mu(M[\epsilon] \mathbin{@} \mathbf{1}.\{\mu\}) : \langle \mu \mid A \rangle \mathbin{@} m}$$

A.2 The Substitution Calculus

Given its complexity, we have separated the rules for the substitution calculus into several distinct blocks. First, we have rules covering the formation of substitutions.

$$\frac{\delta\colon \Gamma \longrightarrow \Delta \mathbin{@} n \qquad \mu\colon n \longrightarrow m}{\delta.\{\mu\}\colon \Gamma.\{\mu\} \longrightarrow \Delta.\{\mu\} \mathbin{@} m} \qquad \frac{\delta\colon \Gamma \longrightarrow \Delta \mathbin{@} n \qquad \mu\colon n \longrightarrow m \qquad r\colon \Delta \longrightarrow \{\mu\} \mathbin{@} m}{\delta/\mu\colon \Gamma/(r \circ \delta : \mu) \longrightarrow \Delta/(r:\mu) \mathbin{@} m}$$

$$\frac{\Gamma\ \mathsf{cx} \mathbin{@} m \qquad \mu\colon n \longrightarrow m}{\epsilon[\Gamma]\colon \Gamma.\{\mu\}/\mu \longrightarrow \Gamma \mathbin{@} m} \qquad \frac{\mu\colon n \longrightarrow m \qquad \Gamma\ \mathsf{cx} \mathbin{@} n \qquad r\colon \Gamma \longrightarrow \{\mu\} \mathbin{@} m}{\eta[r]\colon \Gamma \longrightarrow \Gamma/(r:\mu).\{\mu\} \mathbin{@} n}$$

$$\frac{\Gamma\ \mathsf{cx} \mathbin{@} m \qquad \mu, \nu\colon n \longrightarrow m \qquad \alpha : \nu \Rightarrow \mu}{\{\alpha\}_\Gamma\colon \Gamma.\{\mu\} \longrightarrow \Gamma.\{\nu\} \mathbin{@} n}$$

The operations sending $-.\{\mu\}$ and $-/\mu$ assemble into functors. We therefore require the following equations guaranteeing functoriality:

$$\frac{\mu\colon n \longrightarrow m \qquad \Gamma\ \mathsf{cx} \mathbin{@} m}{\mathsf{id}.\{\mu\} = \mathsf{id}\colon \Gamma.\{\mu\} \longrightarrow \Gamma.\{\mu\} \mathbin{@} n}$$

$$\frac{\Gamma, \Delta, \Xi\ \mathsf{cx} \mathbin{@} m \qquad \mu\colon n \longrightarrow m \qquad \delta\colon \Gamma \longrightarrow \Delta \mathbin{@} m \qquad \xi\colon \Delta \longrightarrow \Xi \mathbin{@} m}{(\xi \circ \delta).\{\mu\} = \xi.\{\mu\} \circ \delta.\{\mu\}\colon \Gamma.\{\mu\} \longrightarrow \Xi.\{\mu\} \mathbin{@} n}$$

$$\frac{\Gamma, \Delta, \Xi\ \mathsf{cx} \mathbin{@} n \qquad \mu\colon n \longrightarrow m \qquad r\colon \Xi \longrightarrow \{\mu\} \mathbin{@} n \qquad \delta\colon \Gamma \longrightarrow \Delta \mathbin{@} n \qquad \xi\colon \Delta \longrightarrow \Xi \mathbin{@} n}{(\xi \circ \delta)/\mu = \xi/\mu \circ \delta/\mu\colon \Gamma/(r \circ \xi \circ \delta : \mu) \longrightarrow \Xi/(r:\mu) \mathbin{@} m}$$

$$\frac{\mu\colon n \longrightarrow m \qquad \Gamma\ \mathsf{cx} \mathbin{@} n \qquad r\colon \Gamma \longrightarrow \{\mu\} \mathbin{@} n}{\mathsf{id}/\mu = \mathsf{id}\colon \Gamma/(r:\mu) \longrightarrow \Gamma/(r:\mu) \mathbin{@} m}$$

We impose further equations on $-.\{\mu\}$ to ensure that not only is each $-.\{\mu\}$ a functor, but the entire collection of $-.\{-\}$ is a 2-functor:

$$\frac{\Gamma, \Delta\ \mathsf{cx} \mathbin{@} m \qquad \mu\colon n \longrightarrow m \qquad \mu\colon o \longrightarrow n \qquad \delta\colon \Gamma \longrightarrow \Delta \mathbin{@} m}{\delta.\{\nu \circ \mu\} = \delta.\{\nu\}.\{\mu\}\colon \Gamma.\{\nu \circ \mu\} \longrightarrow \Delta.\{\nu \circ \mu\} \mathbin{@} o}$$

$$\frac{\Gamma, \Delta\ \mathsf{cx} \mathbin{@} m \qquad \delta\colon \Gamma \longrightarrow \Delta \mathbin{@} m}{\delta.\{1\} = \delta\colon \Gamma \longrightarrow \Delta \mathbin{@} m}$$

$$\frac{\Gamma\ \mathsf{cx} \mathbin{@} m \qquad \mu_0, \mu_1, \mu_2\colon n \longrightarrow m \qquad \alpha_0 : \mu_0 \Rightarrow \mu_1 \qquad \alpha_1 : \mu_1 \Rightarrow \mu_2}{\{\alpha_1 \circ \alpha_0\}_\Gamma = \{\alpha_0\}_\Gamma \circ \{\alpha_1\}_\Gamma\colon \Gamma.\{\mu_2\} \longrightarrow \Gamma.\{\mu_0\} \mathbin{@} n}$$

$$\frac{\Gamma\ \mathsf{cx} \ @\ m \qquad \nu_0, \nu_1 : o \longrightarrow n \qquad \mu_0, \mu_1 : n \longrightarrow m \qquad \beta : \nu_0 \Rightarrow \nu_1 \qquad \alpha : \mu_0 \Rightarrow \mu_1}{\{\alpha \bullet \beta\}_\Gamma = \{\alpha\}_\Gamma.\{\nu_1\} \circ \{\beta\}_{\Gamma.\{\mu_0\}} : \Gamma.\{\mu_0 \circ \nu_0\} \longrightarrow \Gamma.\{\mu_1 \circ \nu_1\} \ @\ o}$$

$$\frac{\Gamma\ \mathsf{cx} \ @\ m \qquad \mu : n \longrightarrow m}{\mathsf{id} = \{1_\mu\}_\Gamma : \Gamma.\{\mu\} \longrightarrow \Gamma.\{\mu\} \ @\ n}$$

$$\frac{\Gamma, \Delta\ \mathsf{cx} \ @\ m \qquad \mu, \nu : n \longrightarrow m \qquad \delta : \Gamma \longrightarrow \Delta \ @\ m \qquad \alpha : \nu \Rightarrow \mu}{\{\alpha\}_\Gamma \circ (\delta.\{\mu\}) = (\delta.\{\nu\}) \circ \{\alpha\}_\Delta : \Gamma.\{\mu\} \longrightarrow \Delta.\{\nu\} \ @\ n}$$

The final set of equations ensure that $-/(- : \mu)$ and $-.\{\mu\}$ encode a parametric adjunction. In particular, we impose the two triangle inequalities on the unit and counit of the adjunction.

$$\frac{\mu : n \longrightarrow m \qquad \Gamma\ \mathsf{cx} \ @\ n \qquad r : \Gamma \longrightarrow \{\mu\} \ @\ m}{!.\{\mu\} \circ \eta[r] = r : \Gamma \longrightarrow \{\mu\} \ @\ n}$$

$$\frac{\Gamma, \Delta\ \mathsf{cx} \ @\ n \qquad \mu : n \longrightarrow m \qquad \delta : \Gamma \longrightarrow \Delta \ @\ n \qquad r : \Gamma \longrightarrow \{\mu\} \ @\ n}{\eta[r] \circ \delta = \delta/\mu.\{\mu\} \circ \eta[r \circ \delta] : \Gamma \longrightarrow \Delta/(r : \mu).\{\mu\} \ @\ m}$$

$$\frac{\Gamma, \Delta\ \mathsf{cx} \ @\ m \qquad \mu : n \longrightarrow m \qquad \delta : \Gamma \longrightarrow \Delta \ @\ m}{\delta \circ \epsilon[\Gamma] = \epsilon[\Delta] \circ \delta.\{\mu\}/\mu : \Gamma.\{\mu\}/\mu \longrightarrow \Delta \ @\ m}$$

$$\frac{\Gamma\ \mathsf{cx} \ @\ n \qquad \mu : n \longrightarrow m \qquad r : \Gamma \longrightarrow \{\mu\} \ @\ n}{\epsilon[\Gamma/(r : \mu)] \circ \eta[r]/\mu = \mathsf{id} : \Gamma/(r : \mu) \longrightarrow \Gamma/(r : \mu) \ @\ m}$$

$$\frac{\Gamma\ \mathsf{cx} \ @\ m \qquad \mu : n \longrightarrow m}{\epsilon[\Gamma].\{\mu\} \circ \eta[\mathbf{1}.\{\mu\}] = \mathsf{id} : \Gamma.\{\mu\} \longrightarrow \Gamma.\{\mu\} \ @\ m}$$

REFERENCES

[1] Robert Atkey and Conor McBride. 2013. Productive coprogramming with guarded recursion. In *Proceedings of the 18th ACM SIGPLAN International Conference on Functional Programming (ICFP'13)*. Association for Computing Machinery, 197–208. https://doi.org/10.1145/2500365.2500597

[2] Steve Awodey. 2010. *Category Theory*. Oxford University Press.

[3] Steve Awodey. 2018. Natural models of homotopy type theory. *Math. Struct. Comput. Sci.* 28, 2 (2018), 241–286. https://doi.org/10.1017/S0960129516000268 arXiv:1406.3219.

[4] Patrick Bahr, Hans Bugge Grathwohl, and Rasmus Ejlers Møgelberg. 2017. The clocks are ticking: No more delays! In *Proceedings of the 32nd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS'17)*. IEEE. https://doi.org/10.1109/LICS.2017.8005097

[5] Patrick Bahr, Christian Uldal Graulund, and Rasmus Ejlers Møgelberg. 2019. Simply RaTT: A fitch-style modal calculus for reactive programming without space leaks. *Proc. ACM Program. Lang.* 3, Article 109 (2019), 109:1–109:27 pages. Issue ICFP. https://doi.org/10.1145/3341713

[6] Patrick Bahr, Christian Uldal Graulund, and Rasmus Ejlers Møgelberg. 2021. Diamonds are not forever: Liveness in reactive programming with guarded recursion. *Proc. ACM Program. Lang.* 5, Article 2 (2021), 28 pages. https://doi.org/10.1145/3434283

[7] Jean-Philippe Bernardy, Thierry Coquand, and Guilhem Moulin. 2015. A presheaf model of parametric type theory. *Electr. Not. Theor. Comput. Sci.* 319 (2015), 67–82. https://doi.org/10.1016/j.entcs.2015.12.006

[8] G. M. Bierman and V. C. V. de Paiva. 2000. On an intuitionistic modal logic. *Stud. Logic.* 65, 3 (2000). https://doi.org/10.1023/A:1005291931660

[9] Lars Birkedal, Ranald Clouston, Bassel Mannaa, Rasmus Ejlers Møgelberg, Andrew M. Pitts, and Bas Spitters. 2020. Modal dependent type theory and dependent right adjoints. *Math. Struct. Comput. Sci.* 30, 2 (2020), 118–138. https://doi.org/10.1017/S0960129519000197 arXiv:1804.05236.

[10] Lars Birkedal and Rasmus Ejlers Møgelberg. 2013. Intensional type theory with guarded recursive types qua fixed points on universes. In *Proceedings of Logic in Computer Science*.

[11] Lars Birkedal, Rasmus Møgelberg, Jan Schwinghammer, and Kristian Støvring. 2012. First steps in synthetic guarded domain theory: Step-indexing in the topos of trees. *Logic. Methods Comput. Sci.* 8, 4 (2012). https://doi.org/10.2168/LMCS-8(4:1)2012

[12] Aleš Bizjak, Hans Bugge Grathwohl, Ranald Clouston, Rasmus E. Møgelberg, and Lars Birkedal. 2016. Guarded Dependent Type Theory with Coinductive Types. In *Foundations of Software Science and Computation Structures*, Bart Jacobs and Christof Löding (Eds.). Springer, Berlin, 20–35.

[13] Aurelio Carboni and Peter Johnstone. 1995. Connected limits, familial representability and Artin glueing. *Math. Struct. Comput. Sci.* 5, 4 (1995), 441–459. https://doi.org/10.1017/S0960129500001183

[14] John Cartmell. 1978. *Generalised Algebraic Theories and Contextual Categories*. Ph.D. Dissertation. University of Oxford.

[15] Evan Cavallo. 2021. *Higher Inductive Types and Internal Parametricity for Cubical Type Theory*. Ph.D. Dissertation. Carnegie Mellon University.

[16] Evan Cavallo and Robert Harper. 2020. Internal parametricity for cubical type theory. In *Proceedings of the 28th EACSL Annual Conference on Computer Science Logic (CSL'20)*, Maribel Fernández and Anca Muscholl (Eds.), Leibniz International Proceedings in Informatics (LIPIcs), Vol. 152. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 13:1–13:17. https://doi.org/10.4230/LIPIcs.CSL.2020.13

[17] James Cheney. 2012. A dependent nominal type theory. *Log. Methods Comput. Sci.* 8, 1 (2012). https://doi.org/10.2168/LMCS-8(1:8)2012

[18] Ranald Clouston. 2018. Fitch-Style Modal Lambda Calculi. In *Foundations of Software Science and Computation Structures*, Christel Baier and Ugo Dal Lago (Eds.). Springer International Publishing, 258–275.

[19] Ranald Clouston, Aleš Bizjak, Hans Bugge Grathwohl, and Lars Birkedal. 2015. Programming and reasoning with guarded recursion for coinductive types. In *Foundations of Software Science and Computation Structures*, Andrew Pitts (Ed.). Springer, Berlin, 407–421.

[20] Peter Dybjer. 1996. Internal type theory. In *Types for Proofs and Programs*, Stefano Berardi and Mario Coppo (Eds.). Springer, Berlin, 120–134. https://doi.org/10.1007/3-540-61780-9_66

[21] Marcelo Fiore. 2012. Discrete generalised polynomial functors. Retrieved from https://www.cl.cam.ac.uk/~mpf23/talks/ICALP2012.pdf.

[22] Daniel Gratzer, G. A. Kavvos, Andreas Nuyts, and Lars Birkedal. 2020. Multimodal dependent type theory. In *Proceedings of the 35th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS'20)*. ACM, 492–506. https://doi.org/10.1145/3373718.3394736

[23] Daniel Gratzer, G. A. Kavvos, Andreas Nuyts, and Lars Birkedal. 2021. Multimodal dependent type theory. *Logical Methods in Computer Science* 17, 3 (2021). https://doi.org/10.46298/lmcs-17%283%3A11%292021

[24] Daniel Gratzer, G. A. Kavvos, Andreas Nuyts, and Lars Birkedal. 2020. Type theory à la mode. Retrieved from https://jozefg.github.io/papers/type-theory-a-la-mode.pdf.

[25] Daniel Gratzer, Jonathan Sterling, and Lars Birkedal. 2019. Implementing a modal dependent type theory. *Proc. ACM Program. Lang.* 3 (2019). https://doi.org/10.1145/3341711

[26] Daniel Gratzer, Jonathan Sterling, and Lars Birkedal. 2019. Normalization-by-evaluation for modal dependent type theory. Retrieved from https://jozefg.github.io/papers/2019-implementing-modal-dependent-type-theory-tech-report.pdf.

[27] Martin Hofmann. 1997. Syntax and semantics of dependent types. In *Semantics and Logics of Computation*, Andrew M. Pitts and P. Dybjer (Eds.). Cambridge University Press, 79–130. https://doi.org/10.1017/CBO9780511526619.004

[28] Ambrus Kaposi, András Kovács, and Thorsten Altenkirch. 2019. Constructing quotient inductive-inductive types. *Proc. ACM Program. Lang.* 3, Article 2 (January 2019), 24 pages. https://doi.org/10.1145/3290315

[29] Daniel R. Licata and Michael Shulman. 2016. Adjoint logic with a 2-category of modes. In *Logical Foundations of Computer Science*, Sergei Artemov and Anil Nerode (Eds.). Springer International Publishing, 219–235. https://doi.org/10.1007/978-3-319-27683-0_16

[30] Daniel R. Licata, Michael Shulman, and Mitchell Riley. 2017. A fibrational framework for substructural and modal logics. In *Proceedings of the 2nd International Conference on Formal Structures for Computation and Deduction (FSCD 2017)*, Dale Miller (Ed.), Leibniz International Proceedings in Informatics (LIPIcs), Vol. 84. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 25:1–25:22. https://doi.org/10.4230/LIPIcs.FSCD.2017.25

[31] Bassel Mannaa, Rasmus Ejlers Møgelberg, and Niccolò Veltri. 2020. Ticking clocks as dependent right adjoints: Denotational semantics for clocked type theory. *Logic. Methods Comput. Sci.* 16, 4 (December 2020).

[32] Luca Mauri. 2017. Algebraic theories in monoidal categories. arXiv:math.LO/1705.09202. Retrieved from https://arxiv.org/abs/1705.09202.

[33] H. Nakano. 2000. A modality for recursion. In *Proceedings of the 15th Annual IEEE Symposium on Logic in Computer Science*. IEEE Computer Society, 255–266.

[34] Frank Pfenning and Rowan Davies. 2001. A judgmental reconstruction of modal logic. *Math. Struct. Comput. Sci.* 11, 4 (2001), 511–540. https://doi.org/10.1017/S0960129501003322

[35] Emily Riehl and Michael Shulman. A type theory for synthetic ∞-categories.

[36] Michael Shulman. 2018. Brouwer's fixed-point theorem in real-cohesive homotopy type theory. *Math. Struct. Comput. Sci.* 28, 6 (2018), 856–941. https://doi.org/10.1017/S0960129517000147

[37] Thomas Streicher. 2020. Fibred categories à la Jean Bénabou. arXiv:1801.02927. Retrieved from https://arxiv.org/abs/1801.02927.

[38] The Univalent Foundations Program. 2013. *Homotopy Type Theory: Univalent Foundations of Mathematics*. Institute for Advanced Study.

[39] Vladimir Voevodsky. 2015. A C-system defined by a universe category. *Theory Appl. Categor.* 30 (2015), 1181–1214.

[40] Mark Weber. 2007. Familial 2-functors and parametric right adjoints. *Theory Appl. Categor.* 18 (2007), 665–732.

Received April 2021; revised September 2021; accepted January 2022