Models for Polymorphism over Physical Dimensions

Robert Atkey[1], Neil Ghani[2], Fredrik Nordvall Forsberg[2], Timothy Revell[2], and Sam Staton[3]

1 University of Edinburgh
2 University of Strathclyde
3 University of Cambridge

Abstract

We provide a categorical framework for models of a type theory that has special types for physical quantities. The types are indexed by the physical dimensions that they involve. We use fibrations to organize this index structure in the models of the type theory. We develop some informative models of this type theory: firstly, a model based on group actions, which captures invariance under scaling, and secondly, a way of constructing new models using relational parametricity.

1998 ACM Subject Classification F.3.2 Semantics of Programming Languages.

Keywords and phrases Category Theory, Units of Measure, Dimension Types, Type Theory.

1 Introduction

This paper is about semantic models of programs that manipulate physical quantities. Physical quantities are organized into dimensions, such as *Length* or *Time*. A fundamental principle of dimensions is that it is not meaningful to add or compare quantities of different dimensions, but they can be multiplied. To measure a physical quantity we use units, such as metres for length and seconds for time. We understand these units as chosen constant quantities of given dimensions.

Here is a simple polymorphic program that is defined for all dimensions; it takes a quantity x of a given dimension X, and returns its double, which has the same dimension.

$$f := (\Lambda X.\, \lambda x : \mathsf{quantity}(X).\, x + x) : \ \forall X.\, \mathsf{quantity}(X) \to \mathsf{quantity}(X) \tag{1}$$

To illustrate, we can use the polymorphic function f to double a length of 5 metres.

$$f_{\mathit{Length}}\,(5\mathrm{m}) = 10\mathrm{m} : \mathsf{quantity}(\mathit{Length}) \tag{2}$$

There are a few key points that are worth emphasising about examples (1) and (2) above:

- There are two kinds of variable, X and x. The first variable X stands for a dimension whereas x stands for an inhabitant of a type. To emphasise this distinction, we use different abstraction symbols (λ and Λ) for the two kinds of variable.
- The type $\mathsf{quantity}(X)$ depends on a dimension X, and it is inhabited by quantities of that dimension. For example, the standard unit of measurement for length, the metre, is a quantity of that dimension, i.e. a constant $\mathrm{m} : \mathsf{quantity}(\mathit{Length})$.

Several authors have developed programming languages with type systems that support physical quantities [5,7,9,11,15]. Our starting point is the work of Kennedy [9] who developed techniques for reasoning about these kinds of programs. However, we take a different approach to Kennedy by developing a general categorical notion of model for a programming language of this form, and by developing ways of building models.

The main contributions of this paper are as follows.

1. We provide a general notion of a model for a programming language with physical dimension types by introducing the concept of a λD-*model* (Section 3). The basic idea is that for each context of dimension variables, there is a model of the simply typed λ-calculus extended with types of quantities of the dimensions definable in the context ($\mathsf{quantity}(D)$ etc.). Moreover these models of the simply typed λ-calculus are related by substituting for dimension variables, and this also defines a universal property for polymorphic quantification over dimension variables.
2. An important example of a λD-model is built from group actions (Example 9). A difficulty with set-theoretic models of dimension polymorphism is as follows: how does one understand $\mathsf{quantity}(X)$ as a set, if the dimension X is not specified and we have no fixed units of measure for X? We resolve this by interpreting $\mathsf{quantity}(X)$ as the set of magnitudes, i.e. positive real numbers, thought of as quantities of some unspecified unit of measure, but then by equipping $\mathsf{quantity}(X)$ with an action of the scaling group, to explain how to change the units of measure. We can then ask that any function $\mathsf{quantity}(X) \to \mathsf{quantity}(X)$ is invariant under changing that unspecified unit of measure, more precisely, invariant under scaling.
3. We show how the λD-model built from group actions supports a diverse range of parametricity-like theorems, without the need to define a separate relational semantics (Section 4). This results in simple proofs of theorems that would otherwise require more heavy machinery.
4. We explore the relationship between the parametricity-like theorems of the λD-model built from group actions, and a natural notion of a relational model (Section 5). Formally, we show that when interpreting the syntax these two notions coincide.

2 Types with Physical Dimensions

We begin by recalling a simple type theory, which we call λD, indexed by dimensions based on Kennedy's work [9]. Within this type theory we can express programs such as (1) and (2). Since there are two kinds of variables, we have two kinds of contexts.

Dimensions and Dimension Contexts: A dimension context Δ is a finite list of distinct dimension variables. A dimension-expression-in-context $\Delta \vdash D\ \mathsf{Dim}$ is a monomial D in the variables Δ. More precisely, if $\Delta = X_1, \dots, X_n$ and $k_i \in \mathbb{Z}$ then $\Delta \vdash X_1^{k_1} \dots X_n^{k_n}\ \mathsf{Dim}$. We can make the set $\{D \mid \Delta \vdash D\ \mathsf{Dim}\}$ an Abelian group under addition of exponents, and indeed this is the free Abelian group on Δ. This universal property gives a notion of substitution on dimension expressions. For example, $X, Z \vdash (X^2Y^3)[{}^{(XZ^2)}/_{Y}] = X^3Z^6\ \mathsf{Dim}$.

Types: Well-formed types are given by judgements of the form $\Delta \vdash T\ \mathsf{Type}$ where Δ is a dimension context. The judgements are generated by the following rules.

$$\frac{\Delta \vdash D\ \mathsf{Dim}}{\Delta \vdash \mathsf{quantity}(D)\ \mathsf{Type}} \qquad \frac{\Delta, X \vdash T\ \mathsf{Type}}{\Delta \vdash \forall X.T\ \mathsf{Type}} \qquad \frac{\Delta \vdash T\ \mathsf{Type} \quad \Delta \vdash U\ \mathsf{Type}}{\Delta \vdash T \to U\ \mathsf{Type}}$$

$$\frac{}{\Delta \vdash 1\ \mathsf{Type}} \qquad \frac{\Delta \vdash T\ \mathsf{Type} \quad \Delta \vdash U\ \mathsf{Type}}{\Delta \vdash T \times U\ \mathsf{Type}} \qquad \frac{}{\Delta \vdash 0\ \mathsf{Type}} \qquad \frac{\Delta \vdash T\ \mathsf{Type} \quad \Delta \vdash U\ \mathsf{Type}}{\Delta \vdash T + U\ \mathsf{Type}}$$

Notice that we do not have System-F-style polymorphism, but instead dimension polymorphism: types can be parameterised by dimensions, but they cannot be parameterised by types, since we do no have type variables. From the Curry-Howard perspective this is a first-order-logic where the domain of discourse is the theory of Abelian groups and where there is a single atomic predicate, $\mathsf{quantity}$.

Terms and Typing Contexts: Well-formed typing contexts are given by judgements $\Delta \vdash \Gamma\ \mathsf{Ctx}$ where Δ is a dimension context, Γ is of the form $x_1 : T_1, \ldots, x_n : T_n$ and there is a well-formed typing judgement $\Delta \vdash T_i\ \mathsf{Type}$ for every i. Well-formed terms are given by judgements $\Delta; \Gamma \vdash t : T$ where there is a well-formed typing context $\Delta \vdash \Gamma\ \mathsf{Ctx}$ and a well-formed type $\Delta \vdash T\ \mathsf{Type}$. The rules for the type formers 1, $\_ \times \_$, $\_ + \_$ and $\_ \to \_$ are the usual ones from simply typed λ-calculus.

$$\frac{\Delta \vdash \Gamma, \Gamma'\ \mathsf{Ctx} \quad \Delta \vdash T\ \mathsf{Type}}{\Delta; \Gamma, x : T, \Gamma' \vdash x : T} \qquad \frac{\Delta; \Gamma, x : T \vdash t : U}{\Delta; \Gamma \vdash \lambda x.t : T \to U} \qquad \frac{\Delta; \Gamma \vdash t : T \to U \quad \Delta; \Gamma \vdash u : T}{\Delta; \Gamma \vdash t\,u : U}$$

$$\frac{\Delta \vdash \Gamma\ \mathsf{Ctx}}{\Delta; \Gamma \vdash () : 1} \qquad \frac{\Delta; \Gamma \vdash t_1 : T_1 \qquad \Delta; \Gamma \vdash t_2 : T_2}{\Delta; \Gamma \vdash (t_1, t_2) : T_1 \times T_2} \qquad \frac{\Delta; \Gamma \vdash t : T_1 \times T_2}{\Delta; \Gamma \vdash \mathsf{pr}_i(t) : T_i}$$

$$\frac{\Delta; \Gamma \vdash t : 0 \quad \Delta \vdash T\ \mathsf{Type}}{\Delta; \Gamma \vdash \mathsf{case}\, t : T} \qquad \frac{\Delta; \Gamma \vdash t : T_i}{\Delta; \Gamma \vdash \mathsf{inj}_i\, t : T_1 + T_2}$$

$$\frac{\Delta; \Gamma \vdash t : T_1 + T_2 \quad \big(\Delta; \Gamma, x_i : T_i \vdash u_i : U\big)_{i \in \{1,2\}}}{\Delta; \Gamma \vdash \mathsf{case}\, t\, \mathsf{of}\, \{\mathsf{inj}_1\, x_1 \mapsto u_1; \mathsf{inj}_2\, x_2 \mapsto u_2\} : U}$$

In addition, we have the introduction and elimination rules for quantification over a unit variable.

$$\frac{\Delta, X; \Gamma \vdash t : T}{\Delta; \Gamma \vdash \Lambda X.t : \forall u.T} \qquad \frac{\Delta \vdash D\ \mathsf{Dim} \quad \Delta; \Gamma \vdash t : \forall X.T}{\Delta; \Gamma \vdash t_D : T[D/X]}$$

We use bool as an abbreviation for $1 + 1$. We may work with some dimension constants and term constants by judging terms in a context $(\Delta_{dim}; \Gamma_{ops})$. For instance, we could consider $\Delta_{dim} = (\mathit{Length}, \mathit{Time})$ and

$$\begin{aligned}
\Gamma_{ops} = (&\mathrm{m} : \mathsf{quantity}(\mathit{Length}),\ \mathrm{s} : \mathsf{quantity}(\mathit{Time}),\\
&+ : \forall X.\mathsf{quantity}(X) \times \mathsf{quantity}(X) \to \mathsf{quantity}(X),\\
&\times : \forall X_1.\forall X_2.\mathsf{quantity}(X_1) \times \mathsf{quantity}(X_2) \to \mathsf{quantity}(X_1 \cdot X_2),\ 1 : \mathsf{quantity}(1),\\
&\mathsf{inv} : \forall X.\mathsf{quantity}(X) \to \mathsf{quantity}(X^{-1}),\\
&< : \forall X.\mathsf{quantity}(X) \times \mathsf{quantity}(X) \to \mathsf{bool}).
\end{aligned}$$

One could also define a type of signed/zero quantities $\mathsf{real}(X) \coloneqq \mathsf{quantity}(X) + 1 + \mathsf{quantity}(X)$, and then extend the language with further arithmetic term constants such as signed addition $+ : \forall X.\mathsf{real}(X) \times \mathsf{real}(X) \to \mathsf{real}(X)$.

This is an idealized language, designed to demonstrate polymorphism over dimension types. As such it is missing many important features for a general purpose language, such as recursive types and terms.

3 Categorical Semantics of Dimension Types

Next up we give a general categorical semantics for the λD type theory. Central to this is the notion of a $\lambda\forall$-fibration.

▶ **Definition 1.** A $\lambda\forall$-*fibration* is a bicartesian closed fibration with simple products.

It is well-known that $\lambda\forall$-fibrations give a categorical model of the fragment of first-order logic without existential quantifiers. Nevertheless, we briefly introduce the basic notions now, since they are central to our development. We refer to [8] for the full details.

A fibration $p : \mathcal{E} \to \mathcal{B}$ is a functor between categories satisfying certain conditions. These conditions (along with the structure in Definition 2) allow us to model the λD type theory. The basic idea is that dimension contexts will be interpreted as objects $\mathcal{B}$. Then for each $B \in \mathcal{B}$ we consider the full subcategory $\mathcal{E}_B$ of $\mathcal{E}$, with objects $E \in \mathcal{E}$ for which $p(E) = B$. The objects of $\mathcal{E}_B$ will be used to interpret types in dimension context B, and the morphisms in $\mathcal{E}_B$ will be used to interpret terms. We can substitute dimension expressions for dimension variables, and this substitution will be interpreted using morphisms in $\mathcal{B}$. Since p is a fibration, for each morphism $f : B \to B'$ in $\mathcal{B}$ there is a canonical associated reindexing functor $f^* : \mathcal{E}_{B'} \to \mathcal{E}_B$, which we will use to describe substitution for dimension variables in types and terms.

A fibration is said to be bicartesian closed if $\mathcal{E}_B$ is a Cartesian closed category with coproducts for all B, and each reindexing functor $f^* : \mathcal{E}_{B'} \to \mathcal{E}_B$ preserves products, exponentials and coproducts. This bicartesian closed structure is needed to interpret the product, function and coproduct types.

Concatenation of dimension contexts will be interpreted using products in the category $\mathcal{B}$. The reindexing functors $\pi^* : \mathcal{E}_B \to \mathcal{E}_{B\times B'}$ for the product projections $\pi : B \times B' \to B$ correspond to context-weakening. A fibration $p : \mathcal{E} \to \mathcal{B}$ is said to have simple products if $\mathcal{B}$ has products and the reindexing functors for the product projections have right adjoints $\forall : E_{B\times B'} \to E_B$ that are compatible with reindexing ('Beck-Chevalley'). A fibration is said to have products if this condition holds for all morphisms in the base, not just projections. These right adjoints are needed to interpret universal quantification of dimension variables in types.

▶ **Definition 2.** A λD-model (p, G, Q) is a $\lambda\forall$-fibration $p : \mathcal{E} \to \mathcal{B}$, an Abelian group object G in $\mathcal{B}$, and an object Q in the fibre $\mathcal{E}_G$.

Recall that an Abelian group object in a category $\mathcal{B}$ with products is given by an object G together with maps $e : 1 \to G$, $m : G \times G \to G$ and $i : G \to G$ satisfying the laws of Abelian groups. This group structure is needed to interpret dimension expressions: for each vector of n integers we have a morphism $G^n \to G$.

An equivalent way to define Abelian group objects if $\mathcal{B}$ has chosen products is as follows. Recall that the Lawvere theory for Abelian groups is the category L_{Ab} whose objects are natural numbers, and where a morphism $m \to n$ is an $m \times n$ matrix of integers. Composition of morphisms is given by matrix multiplication, and categorical products are given by arithmetic addition of natural numbers. An Abelian group object in $\mathcal{B}$ is an object G of $\mathcal{B}$ together with a strictly-product-preserving functor $F : \mathsf{L}_{\mathsf{Ab}} \to \mathcal{B}$ such that $F(1) = G$.

We remark that the Abelian group G in a λD-model is analogous to the generic object in a model of System F.

In order to ascertain the value of Definition 2, we now do three things: i) we show that a λD-model in fact does provide categorical models of dimension types, ii) we give examples of λD-models, and iii) we prove theorems that show the viability of reasoning at this level of abstraction.

3.1 Modelling Dimension Types

To show that λD-models provide a categorical semantics for dimension types, we must show how to interpret the syntax given in Section 2 in any given λD-model. We will use the $\lambda\forall$-fibration to separate the indexing information (the dimensions) from the indexed information (the types and terms). This means, that the base category of the fibration will be used to interpret dimension contexts, and types and terms will be interpreted as objects in the fibres above the dimension contexts in which they are defined. Cartesian closure of the fibres will allow us to inductively interpret types built from 1, $\times$ and $\rightarrow$, and we will take the standard approach in categorical logic to interpret quantification of dimensions - by using right adjoints. Finally, since dimension expressions for a dimension context, are defined as elements of the free Abelian group on that dimension context, we will use the Abelian group object structure to interpret such expressions. Formally, we interpret the syntax as follows.

- Dimension contexts $\Delta = X_1, \ldots, X_n$ are interpreted as the product of the Abelian group object $[\![\Delta]\!] = G^n$ in $\mathcal{B}$.
- Dimension expressions $\Delta \vdash D\ \mathsf{Dim}$ are interpreted as morphisms $G^n \to G$ in the base $\mathcal{B}$, by using the structure of the Abelian group object G. For example, $[\![X, Y \vdash X \cdot Y^{-1}]\!] = G \times G \xrightarrow{\mathrm{id}_G \times i} G \times G \xrightarrow{m} G$. We assume that there is a given interpretation $[\![d]\!] : 1 \to G$ for every primitive dimension constant $d \in \Delta_{dim}$.
- Well-formed types $\Delta \vdash T\ \mathsf{Type}$ are interpreted as objects $[\![T]\!]$ in the fibre above $[\![\Delta]\!]$, defined by induction on the structure of T. We interpret 1, $\times$ and $\rightarrow$ using the Cartesian closed structure of the fibres, and quantification of a dimension variable $[\![\Delta \vdash \forall X.T]\!]$ is defined by right adjoint to reindexing along the projection $\pi : [\![\Delta \vdash \Gamma, X]\!] \to [\![\Delta \vdash \Gamma]\!]$. Quantities $(\Delta \vdash \mathsf{quantity}(D))$ are interpreted by reindexing the object Q along the interpretation of D, i.e. $[\![\Delta \vdash \mathsf{quantity}(D)]\!] = [\![\Delta \vdash D\ \mathsf{Dim}]\!]^*(Q)$.
- Well-formed typing contexts $\Delta \vdash \Gamma\ \mathsf{ctxt}$ are interpreted as products in the fibre above $[\![\Delta]\!]$, i.e. $[\![\Delta \vdash x_1 : T_1, \ldots, x_n : T_n]\!] = [\![\Delta \vdash T_1]\!] \times \ldots \times [\![\Delta \vdash T_n]\!]$.
- Well-formed terms $\Delta, \Gamma \vdash t : T$ are interpreted as morphisms $[\![t]\!] : [\![\Gamma]\!] \to [\![T]\!]$ in the fibre above $[\![\Delta]\!]$. We assume that there is an interpretation $\eta_{ops} : 1 \to [\![\Gamma_{ops}]\!]$ of all the primitive operations.

In this paper we have only considered universal quantification of units but existential quantification can be given just as easily. Existential quantification is interpreted as the left adjoint to reindexing along a projection. Properties of existential quantification can be proven by dualising the relevant proofs of properties about universal quantification.

3.2 First Examples of λD-Models

We now give some examples of λD-models. We begin by noting that in Kennedy's paper [9], a simpler approach is taken to the semantics of dimensions, the dimensions are simply thrown away in a *dimension-erasure semantics.* From the categorical perspective, this means the calculus is stripped of its fibred structure leaving only a simply typed λ-calculus, which Kennedy models, as to be expected, within a CCC. In particular, he chooses the CCC of complete partial orders which he needs for recursion. Nevertheless, Kennedy's model can be viewed as a λD-model.

▶ **Example 3.** (Dimension-Erasure Models) Let $\mathcal{C}$ be a bicartesian closed category. Then the functor $\mathcal{C} \to 1$ is a $\lambda\forall$-fibration. The unique object of 1 is a trivial Abelian group object. By taking $\mathcal{C}$ to be the category of complete partial orders and continuous functions, and by choosing the flat pointed cpo $\mathbb{Q}_\perp$ to interpret $\mathsf{quantity}$ we obtain a model corresponding to Kennedy's dimension-erasure model. This model supports a plethora of primitive operations, including all the standard arithmetical ones. However, the model also contains many functions which are not dimensionally invariant, i.e. they do not scale appropriately under change of units — Kennedy uses relational parametricity [13] to remove these unwanted elements; we will come back to his relational model in Section 5.

▶ **Example 4.** (Syntactical Models) We can construct a $\lambda\forall$-fibration $\mathcal{C}\ell(\lambda D)$ from the syntax in a standard way. The base category $\mathcal{B}$ is the Lawvere theory of Abelian groups L_{Ab}. The fibre $\mathcal{C}\ell(\lambda D)_n$ over n is the category whose objects are types with n dimension variables, and whose morphisms are terms in context, modulo a standard notion of conversion. The object 1 in L_{Ab} is an Abelian group object, and $(\mathcal{C}\ell(\lambda D) \to \mathsf{L}_{\mathsf{Ab}}, 1, (X \vdash \mathsf{quantity}(X)\ \mathsf{Type}))$ is a λD-model.

▶ **Example 5.** (The Dimension-Indexed Families Model) Let $\mathsf{Fam}(\mathsf{Set})$ be the category whose objects are pairs $(I, \{X_i\}_{i\in I})$ of a set I and an I-indexed family of sets $\{X_i\}_{i\in I}$. A morphism $(I, \{X_i\}_{i\in I}) \to (J, \{Y_j\}_{j\in J})$ is a pair $(f, \{\phi_i\}_{i\in I})$ where f is a function $f : I \to J$ and ϕ_i is a function $\phi_i : X_i \to Y_{f(i)}$ for all $i \in I$. It is well known that the forgetful functor $(I, \{X_i\}_{i\in I}) \mapsto I : \mathsf{Fam}(\mathsf{Set}) \to \mathsf{Set}$, taking a family to its index set, is a $\lambda\forall$-fibration (see e.g. Jacobs [8, Lemma 1.9.5]). For any given set B of fundamental dimensions (e.g. *Length*, *Time*, *Mass* etc.), let G be the free Abelian group on B. Suppose that we also have a set Q_d of quantities for each dimension $d \in G$ (for instance, we can choose $Q_d = \mathbb{R}^+ \times \{\overline{d}\}$ where $\overline{d}$ is a unit of measure for the dimension d, e.g. $\overline{\mathit{Length}} = \mathrm{m}$, $\overline{\mathit{Time}} = \mathrm{s}$, $\overline{d \cdot d'} = \overline{d} \cdot \overline{d'}$ etc). We then have a λD-model with $\mathsf{quantity}$ interpreted as $(G, \{Q_d\}_{d\in G})$.

In this model, a dimension expression $X_1, \ldots, X_n \vdash D\ \mathsf{Dim}$ is interpreted as a function $G^n \to G$ using the free Abelian group structure on G: for each valuation of the dimension variables as physical dimensions, we have an interpretation of the expression as a physical dimension. A type with a free dimension variable $X \vdash T\ \mathsf{Type}$ is interpreted as a family of sets, indexed by the dimensions in G. Similarly a term with a free dimension variable is interpreted as a family of functions, one for each dimension in G. This model does support many primitive operations, but it does not support dimension invariant polymorphism. For instance, the model supports adding a term $\mathsf{eq} : \forall X_1.\forall X_2.\mathsf{bool}$ which tests whether two dimensions are the same, which is clearly not invariant under change of representation.

Related examples include the relations fibration $\mathsf{Rel} \to \mathsf{Set}$ and the subobject fibration $\mathsf{Sub}(\mathsf{Set}) \to \mathsf{Set}$. This example can also be generalised to the fibration $\mathsf{Fam}(\mathcal{C}) \to \mathsf{Set}$, which is a $\lambda\forall$-fibration if $\mathcal{C}$ is bicartesian closed.

A Source of Fibrations with Simple Products

We next look at a particular class of λD models, where the fibres in the $\lambda\forall$-fibration are functors. We prove a general theorem for such fibrations, and instantiate it to construct several examples. We first introduce some notation. Let $\mathcal{S}$ be a category (typically $\mathcal{S} = \mathsf{Set}$), and consider the category $\mathsf{Cat}/\!/\mathcal{S}$. The objects are pairs $(\mathcal{C}, P : \mathcal{C} \to \mathcal{S})$, where $\mathcal{C}$ is a small category and $P : \mathcal{C} \to \mathcal{S}$ a functor. Morphisms $(F, \phi) : (\mathcal{C}, P) \to (\mathcal{D}, Q)$ are pairs of a functor $F : \mathcal{C} \to \mathcal{D}$ and a natural transformation $\phi : P \to Q \circ F$. The obvious projection functor $(\mathcal{C}, P) \mapsto \mathcal{C} : \mathsf{Cat}/\!/\mathcal{S} \to \mathsf{Cat}$ is a fibration. The fibre over a small category $\mathcal{C}$ is the category $\mathcal{S}^{\mathcal{C}}$ of functors $[\mathcal{C} \to \mathcal{S}]$ and natural transformations between them. Reindexing is given by precomposition of functors.

▶ **Theorem 6.** *If $\mathcal{S}$ has all small limits then the fibration $\mathsf{Cat}/\!/\mathcal{S} \to \mathsf{Cat}$ has simple products.*

This result appears to be fairly well-known folklore (see e.g. Lawvere [10, end of §3], Melliès and Zeilberger [12]), but since it is important in what follows we sketch a proof.

Proof Sketch. For any functor $F : \mathcal{C} \to \mathcal{D}$, the reindexing functor $F^* : \mathcal{S}^{\mathcal{D}} \to \mathcal{S}^{\mathcal{C}}$ has a right adjoint $F_* : \mathcal{S}^{\mathcal{C}} \to \mathcal{S}^{\mathcal{D}}$, known as the 'right Kan extension along F', which always exists when $\mathcal{S}$ has limits. For simple products, we are only interested in a right adjoint to weakening, i.e. in the functor $\forall_{\mathcal{C}} : \mathcal{S}^{\mathcal{C}\times\mathcal{D}} \to \mathcal{S}^{\mathcal{C}}$ which is the right Kan extension along along the projection functor $\pi_{\mathcal{C}} : \mathcal{C} \times \mathcal{D} \to \mathcal{C}$. Expanding the definitions, we see that $\forall_{\mathcal{C}}(P) : \mathcal{C} \to \mathcal{S}$ is a point-wise limit:

$$(\forall_{\mathcal{C}}\, P)(c) = \lim_{d\in\mathcal{D}} P(c, d) \quad . \tag{3}$$

The Beck-Chevalley condition requires that the canonical map $F^*\forall_{\mathcal{C}'} \to \forall_{\mathcal{C}}(F \times \mathrm{id}_{\mathcal{D}})^*$ is a natural isomorphism for all functors $F : \mathcal{C} \to \mathcal{C}'$. Indeed, for any $P : \mathcal{C} \times \mathcal{D} \to \mathcal{S}$, $c \in \mathcal{C}$:

$$\begin{aligned}(F^*(\forall_{\mathcal{C}'}\, P))(c) = (\forall_{\mathcal{C}}\, P)(F(c)) &\cong \lim_{d\in\mathcal{D}}(F(c), d) = \lim_{d\in\mathcal{D}}(((F \times \mathrm{id}_{\mathcal{D}})^*(P))(c, d)) \\ &\cong (\forall_{\mathcal{C}}((F \times \mathrm{id}_{\mathcal{D}})^*(P))(c) \quad .\end{aligned}$$ ◀

A Source of Models by Change of Base

In general, a useful way of building fibrations is by changing the base. If $p : \mathcal{E} \to \mathcal{B}$ be a fibration, and $F : \mathcal{A} \to \mathcal{B}$ is a functor, then the pullback of p along F in Cat, denoted $F^*p : F^*\mathcal{E} \to \mathcal{A}$, is again a fibration. The same is true of λD-models.

▶ **Theorem 7.** *Let $p : \mathcal{E} \to \mathcal{B}$ be a fibration, and let $F : \mathcal{A} \to \mathcal{B}$ be a functor.*

(i) If p has simple products and F preserves products, then $F^*p : F^*\mathcal{E} \to \mathcal{A}$ has simple products.
(ii) If p is bicartesian closed then $F^*p : F^*\mathcal{E} \to \mathcal{A}$ is bicartesian closed.
(iii) If G is an Abelian group object in $\mathcal{A}$ and $(p, F(G), Q)$ is a λD-model, then $(F^*p, G, (G, Q))$ is also a λD-model.

Proof. For item (i): for any $A \in \mathcal{A}$, reindexing along a projection $\pi_A : A \times A' \to A$ in $\mathcal{A}$ is by construction reindexing along $F(\pi_A)$ in $\mathcal{B}$, which (as F preserves finite products) is the same as reindexing along a projection $\pi_{FA} : FA \times FA' \to FA$, which has a right adjoint and satisfies the Beck-Chevalley condition, since p has simple products.

For item (ii): F^*p is a bicartesian closed fibration since each fibre $(F^*\mathcal{E})_A$ is by construction of the form $\mathcal{E}_{FA}$ and hence bicartesian closed, and reindexing by f in $\mathcal{A}$ is by construction defined to be reindexing by Ff in $\mathcal{B}$, which preserves the structure.

Item (iii) is an immediate corollary. ◀

For a simple illustration of the change of base result, notice that the dimension-erasure fibration $\mathcal{C} \to 1$ arises from pulling back the families fibration $\mathsf{Fam}(\mathcal{C}) \to \mathsf{Set}$ along the unique product-preserving functor $1 \to \mathsf{Set}$.

▶ **Example 8.** (Models over the Lawvere theory L_{Ab}) Let $(p : \mathcal{E} \to \mathcal{B}, G, Q)$ be a λD-model. Recall that the Abelian group object G in $\mathcal{B}$ gives rise to a unique product-preserving functor $F : \mathsf{L}_{\mathsf{Ab}} \to \mathcal{B}$ such that $F(1) = G$. By Theorem 7, we have a λD-model $(F^*p, 1, (1, Q))$.

▶ **Example 9.** (A Model Built from Group Actions) Let G be a group. Recall that a G-set consists of a set A together with a group action, i.e. a function $\cdot_A : G \times A \to A$ such that $e \cdot_A a = a$ and $(gh) \cdot_A a = g \cdot_A (h \cdot_A a)$. The category $\mathsf{Grp}/\!/\mathsf{Set}$ has as objects pairs (G, A) where G is a group and A is a G-set. A morphism $(G, A) \to (H, B)$ in $\mathsf{Grp}/\!/\mathsf{Set}$ is given by a group homomorphism $\phi : G \to H$ and a function $f : A \to B$ such that for any $g \in G$ and $a \in A$ we have $f(g \cdot_A a) = (\phi g) \cdot_B (fa)$. Let Grp be the category of groups and homomorphisms. We call the forgetful functor $p : \mathsf{Grp}/\!/\mathsf{Set} \to \mathsf{Grp}$ the $\mathsf{Grp}/\!/\mathsf{Set}$ fibration.

▶ **Proposition 10.** *Let G be an Abelian group, and let Q be a G-set. Then $(p : \mathsf{Grp}/\!/\mathsf{Set} \to \mathsf{Grp}, G, Q)$ is a λD-model.*

Proof. For any group H, the fibre above H is the category of H-sets and equivariant functions. This is isomorphic to the the functor category Set^H, where we consider the group H as a category with one object $\star$ and a morphism for each element of H. Indeed, there is a product-preserving, full and faithful functor $\mathsf{Grp} \to \mathsf{Cat}$, taking a group to the corresponding one-object category. The fibration $\mathsf{Grp}/\!/\mathsf{Set} \to \mathsf{Grp}$ is thus the pullback of the fibration $\mathsf{Cat}/\!/\mathsf{Set} \to \mathsf{Cat}$ along this embedding $\mathsf{Grp} \to \mathsf{Cat}$. Thus, by Theorem 6 and Theorem 7(i), $\mathsf{Grp}/\!/\mathsf{Set} \to \mathsf{Grp}$ has simple products.

Each fibre is bicartesian closed. The products, coproducts and function spaces are inherited from Set. For the function space, let A and B be G-sets; then the set of functions $(A \to B)$ is also a G-set, with the action given by $(g \cdot_{(A \to B)} f)(x) \coloneqq g \cdot_B (f(g^{-1} \cdot_A x))$. It follows that reindexing preserves the bicartesian closed structure. (This is not the case more generally for $\mathsf{Cat}/\!/\mathsf{Set} \to \mathsf{Cat}$, so Theorem 7(ii) does not apply.) Finally, an Abelian group object in Grp is the same thing as an Abelian group. Hence (p, G, Q) is a λD-model. ◀

In the $\mathsf{Grp}/\!/\mathsf{Set}$ fibration, the Abelian group G can be thought of as a group of scaling factors, and the G-set Q is a set of quantities together with a scaling action. For instance, let $Q = G = (\mathbb{R}^+, \times, 1)$, the positive reals. We model a type with a free dimension variable $X \vdash T$ Type as a G-set. A term with a free dimension variable is interpreted as a function that is invariant under G. We explore this model in more detail in Section 4.

The $\mathsf{Grp}/\!/\mathsf{Set}$ model cannot support dimension constants because there is only one group homomorphism $1 \to G$. It does support several term constants, which we discuss after Theorem 13.

More generally, instead of having sets and group actions, we also have λD-models built from actions of groupoids.

▶ **Example 11.** (A Model Built from Groupoid Actions) Recall that a *groupoid* is a small category $\mathcal{C}$ where every morphism is an isomorphism, and that a functor $\mathcal{C} \to \mathsf{Set}$ is called a groupoid action (or presheaf). The category $\mathsf{Gpd}/\!/\mathsf{Set}$ has as objects pairs $(\mathcal{A}, \phi)$ where $\mathcal{A}$ is a groupoid and $\phi : \mathcal{A} \to \mathsf{Set}$ is a functor. A morphism $(\mathcal{A}, \phi) \to (\mathcal{B}, \psi)$ in $\mathsf{Gpd}/\!/\mathsf{Set}$ is given by a functor $F : \mathcal{A} \to \mathcal{B}$ and a natural transformation $\eta : \phi \to \psi \circ F$ between functors $\mathcal{A} \to \mathsf{Set}$. Let Gpd be the category of groupoids and functors. Then the forgetful functor $\mathsf{Gpd}/\!/\mathsf{Set} \to \mathsf{Gpd}$, which we call the $\mathsf{Gpd}/\!/\mathsf{Set}$ fibration, is a $\lambda\forall$-model, and the proof of this is very similar to Example 9.

On theme with this subsection, the $\mathsf{Gpd}/\!/\mathsf{Set}$ fibration is related to the other fibrations by change of base:

- The families fibration $\mathsf{Fam(Set)} \to \mathsf{Set}$ from Example 5 arises from pulling back the groupoid action fibration $\mathsf{Gpd}/\!/\mathsf{Set} \to \mathsf{Gpd}$ along the discrete-groupoid-functor $\mathsf{Set} \to \mathsf{Gpd}$.
- The group action fibration $\mathsf{Grp}/\!/\mathsf{Set} \to \mathsf{Grp}$ from Example 9 arises from pulling back the groupoid action fibration $\mathsf{Gpd}/\!/\mathsf{Set} \to \mathsf{Gpd}$ along the functor $\mathsf{Grp} \to \mathsf{Gpd}$ that regards each group as a groupoid with one object.

We now discuss λD-models in $\mathsf{Gpd}/\!/\mathsf{Set}$. Let $f : G \to H$ be a homomorphism of Abelian groups. This induces a groupoid whose objects are the elements of H, and where the hom-sets are $\mathrm{mor}(h, h') = \{g \in G \mid f(g) \cdot_H h = h'\}$. The group operation in G provides composition of morphisms. This groupoid can be given the structure of an Abelian group object in Gpd, and, moreover, every Abelian group in Gpd arises in this way [3].

We have already seen that the $\mathsf{Gpd}/\!/\mathsf{Set}$ fibration subsumes the families and group actions fibrations. It also subsumes them as λD-models. To recover group actions (Example 9), let G be an Abelian group of scale factors. The Abelian group object induced by the unique homomorphism $G \to 1$ is a one-object groupoid, and hence we build the λD-models of group actions. To recover the families example (Example 5), fix a set of dimension constants and let H be the free Abelian group on that set. The unique homomorphism $1 \to H$ induces the discrete groupoid whose objects are H, and hence we build the λD-models of families of sets.

4 Group Actions and Dimension Types

In this section we will look in greater detail at the λD-model given by the $\mathsf{Grp}/\!/\mathsf{Set}$ fibration. It turns out that many interesting theorems can be proven in this model, and so to aid us in this task we first concretely spell out the reindexing and simple product structure.

Given a group G, we write $\mathcal{G}$ (with a different font) for the corresponding one-element category, which has morphisms given by elements of G and composition given by group multiplication. Suppose that $\phi : \mathcal{G} \to \mathsf{Set}$ is a G-set (considered as a functor). We write $|\phi| := \phi(\star)$ for the underlying carrier set of ϕ. Reindexing along $\pi : \mathcal{G} \times \mathcal{H} \to \mathcal{G}$ yields the $G \times H$-set given by $\phi \circ \pi$. In other words, $\pi^*\phi$ is a $G \times H$-set with the same underlying carrier $|\pi^*\phi| = |\phi|$ as the G-set ϕ, and action given by $(g, h) \cdot_{\pi^*\phi} x = g \cdot_\phi x$.

Now suppose that $\psi : \mathcal{G} \times \mathcal{H} \to \mathsf{Set}$ is a $G \times H$-set. According to Theorem 6 (equation (3)), the underlying set of $\forall_\pi \psi$ is given by $|\forall_\pi \psi| = \lim_{y \in \mathcal{H}} \psi(\star, y)$. By the universal property of limits,

$$\lim_{y \in \mathcal{H}} \psi(\star, y) \cong \mathsf{Set}\big(1, \lim_{y \in \mathcal{H}} \psi(\star, y)\big) \cong [\mathcal{H}, \mathsf{Set}]\big(K1, \psi(\star, \_)\big) \ ,$$

hence $|\forall_\pi \psi| = \{y \in |\psi| \mid \forall h \in H \ . \ (e_A, h) \cdot_\psi y = y\}$, and the action is given by $g \cdot_{\forall_\pi \psi} x = (g, e_H) \cdot_\psi x$. Notice that to give the group action of $\forall_\pi \psi$, we had to make a particular choice of an element in H, namely the identity element e_H. However, any element of H would have given the same result, since for all $y \in |\forall_\pi \psi|$,

$$(g, h) \cdot_\psi y = ((g, e_H)(e_G, h)) \cdot_\psi y = (g, e_H) \cdot_\psi ((e_G, h) \cdot_\psi y) = (g, e_H) \cdot_\psi y.$$

Many of the properties of dimension types that Kennedy proves using parametricity can be shown to hold in the $\mathsf{Grp}/\!/\mathsf{Set}$-fibration, without having to define a separate relational semantics and this is the content of Theorems 13 - 18. Before we formally state and prove these we introduce a substitution lemma, which holds in any model.

▶ **Lemma 12.** *(Substitution Lemma) Suppose that* $\Delta, X \vdash T$ Type *and that* $\Delta \vdash D$ Dim *denotes a dimension expression. Then* $[\![T[D/X]]\!] \cong (\mathrm{id}_{[\![\Delta]\!]}, [\![D]\!])^* [\![T]\!]$.

Proof. By induction on the structure of T. ◀

Explicitly, Lemma 12 says that the semantics of substituting a dimension expression for a dimension variable is given by reindexing along the identity paired with the dimension expression. Since reindexing is given by precomposition we have that

$$(\mathrm{id}_{[\![\Delta]\!]}, [\![D]\!])^* [\![T]\!] \cong [\![T]\!](\mathrm{id}_{[\![\Delta]\!]}, [\![D]\!]) \ ,$$

i.e., substitution of the n^{th} unit variable is given by precomposition at the n^{th} component.

For the rest of this section, we will use semantic brackets $[\![\_]\!]$ to refer only to the $\mathsf{Grp}/\!/\mathsf{Set}$ interpretation.

▶ **Theorem 13.** *Suppose that* $X_1, \ldots, X_n, X \vdash S, T$ Type*. Then*

$$|[\![\forall X.S \to T]\!]| \cong [\mathcal{G}, \mathsf{Set}]([\![S]\!](\underbrace{\star, \ldots, \star}_{n-times}, \_), [\![T]\!](\underbrace{\star, \ldots, \star}_{n-times}, \_))$$

Proof. By the Kan extension formula and Yoneda. ◀

This Theorem says that in the Grp//Set model a universally quantified variable over an arrow type can be considered as a natural transformation between the domain and codomain of the arrow type, with the first n components fixed. In other words, it is interpreted as the set of functions that are equivariant in the last argument.

In particular, if $X_1 \ldots X_n \vdash S, T$ Type then the type $(\forall \vec{X}. S \to T)$ is interpreted as the set of all homomorphisms $[\mathcal{G}^n, \mathsf{Set}]([\![S]\!], [\![T]\!])$. We use this fact to conclude that the group actions model supports several term constants. For any $q \in Q$, we can accommodate a term constant $q : \mathsf{quantity}(1)$, which is interpreted by $[\![q]\!] = q$. When $Q = G$, we can also accommodate a term constant for multiplication

$$\times : \forall X.\forall Y. \mathsf{quantity}(X) \times \mathsf{quantity}(Y) \to \mathsf{quantity}(X \cdot Y)$$

which is interpreted as the group operation. When $Q = G = (\mathbb{R}^+, \times, 1)$, the positive reals, we also have addition, $+ : \forall X. \mathsf{quantity}(X) \times \mathsf{quantity}(X) \to \mathsf{quantity}(X)$, which is equivariant since $q(r + s) = qr + qs$.

▶ **Theorem 14.** *Suppose that* $\Delta, X \vdash T$ Type*. Then* $|[\![\forall X.\mathsf{quantity}(X) \to T]\!]| \cong |[\![T[1/X]]\!]|$.

Proof. By Theorem 13, Lemma 12 and Yoneda. ◀

We now prove some theorems about the Grp//Set fibration that are parametricity results in Kennedy's original paper. The proofs here involve applications of Lemma 12, Theorem 13 and Theorem 14. First, we take a look at the interplay between scaling factors and polymorphic functions.

▶ **Theorem 15.** *(Scaling Factors) Suppose* $\Delta_{dim}; \Gamma_{ops} \vdash t : \forall X.\mathsf{quantity}(X) \to \mathsf{quantity}(X^n)$*, where* $n \in \mathbb{N}$*. Then for all* $g \in G$ *and* $x \in |[\![\mathsf{quantity}(X)]\!]|$*, we have* $[\![t]\!](g \cdot x) = g^n \cdot [\![t]\!]x$.

Proof. We know from Theorem 13 that $[\![t]\!] \in [\mathcal{G}, \mathsf{Set}]([\![\mathsf{quantity}(X)]\!], [\![\mathsf{quantity}(X^n)]\!])$. In other words, $[\![t]\!](g \cdot x) = g^n \cdot [\![t]\!]x$ for all $x \in |[\![\mathsf{quantity}(X)]\!]|$, as required. ◀

This theorem tells us that polymorphic functions are *invariant under scaling*. Intuitively we see that scaling factors must be changed in an appropriately polymorphic way. If we apply Theorem 14 to the type $\forall X.\mathsf{quantity}(X) \to \mathsf{quantity}(X^n)$, we see that

$$|[\![\forall X.\mathsf{quantity}(X) \to \mathsf{quantity}(X^n)]\!]| \cong |[\![\mathsf{quantity}(1^n)]\!]| \cong |[\![\mathsf{quantity}(1)]\!]| \cong Q,$$

Putting $Q = G$, we conclude that all the terms of type $\forall X.\mathsf{quantity}(X) \to \mathsf{quantity}(X^n)$ are of the form $\Lambda X. \lambda q : \mathsf{quantity}(X).\ r \times q^n$ for $r \in G$.

▶ **Theorem 16.** *There is no ground term* $\vdash t : \forall X.\mathsf{quantity}(X^2) \to \mathsf{quantity}(X)$*., i.e., we cannot write a polymorphic square root function.*

Proof. To see this we exhibit a model where the existence of such a term is impossible. Consider the λD-model $(p : \mathsf{Grp}/\!/\mathsf{Set} \to \mathsf{Grp}, \mathbb{Z}_2, \mathbb{Z}_2)$ where the Abelian group $\mathbb{Z}_2 = (\{-1, 1\}, \cdot, 1)$

is used to interpret both dimensions and units. Theorem 13 says that the interpretation of the type $\forall X.\mathsf{quantity}(X^2) \to \mathsf{quantity}(X)$ is given by

$$|[\![\forall X.\mathsf{quantity}(X^2) \to \mathsf{quantity}(X)]\!]| \cong [\mathbb{Z}_2, \mathsf{Set}]([\![\mathsf{quantity}(X^2)]\!], [\![\mathsf{quantity}(X)]\!])$$

i.e. any element f of $|[\![\forall X.\mathsf{quantity}(X^2) \to \mathsf{quantity}(X)]\!]|$, satisfies for all $g, x \in \mathbb{Z}_2$

$$f(g^2 \cdot x) = g \cdot (fx) \tag{$*$}$$

If f exists, then either $f(-1) = -1$ or $f(-1) = 1$, but both lead to contradictions. To this end suppose that $f(-1) = -1$, then by $(*)$ we have $f((-1)^2 \cdot -1) = (-1) \cdot f(-1)$, which is a contradiction since the left-hand side is equal to -1 and the right-hand side is equal to 1. A similar argument shows that $f(-1) = 1$ is also not possible, and hence there exists no such f. ◀

This result can be extended to also include terms t using primitive operations, i.e. $\Gamma_{ops} \vdash t : \forall X.\mathsf{quantity}(X^2) \to \mathsf{quantity}(X)$, as long as these operations can be interpreted in the model in question. For example, the result holds in the presence of multiplication

$$\times : \forall X.\forall Y.\, \mathsf{quantity}(X) \times \mathsf{quantity}(Y) \to \mathsf{quantity}(X \cdot Y).$$

Note however that this model does not support a polymorphic zero constant $0 : \forall X.\, \mathsf{quantity}(X)$, as such a primitive would of course gives rise to a trivial counterexample to the theorem.

Next, we can prove a theorem that relates a dimensionally invariant function to a dimensionless one. This is a simplified version of the *Buckingham Pi Theorem* of dimensional analysis [4] (or for a more modern introduction see Sonin [14]).

▶ **Theorem 17.** *We have a bijection*

$$|[\![\forall X.\mathsf{quantity}(X) \times \mathsf{quantity}(X) \to \mathsf{quantity}(1)]\!]| \cong |[\![\mathsf{quantity}(1) \to \mathsf{quantity}(1)]\!]|$$

Proof. This is a consequence of Theorem 14, after currying. ◀

We finish this section with another uninhabitedness result, this time about a higher order type.

▶ **Theorem 18.** *There is no term*

$$\vdash t : \forall X_1.\forall X_2.(\mathsf{quantity}(X_1) \to \mathsf{quantity}(X_2)) \to \mathsf{quantity}(X_1 \cdot X_2) \ \ .$$

Proof. Choose G and Q to be $\mathbb{Z}_2$. Interpreting the type of t, we have

$$\begin{aligned}&[\![\forall X_1.\forall X_2.(\mathsf{quantity}(X_1) \to \mathsf{quantity}(X_2)) \to \mathsf{quantity}(X_1 \cdot X_2)]\!]\\ &= \{t \in (\mathbb{Z}_2 {\to} \mathbb{Z}_2) {\to} \mathbb{Z}_2 \mid \forall g_1, g_2 \in \mathbb{Z}_2,\ f : \mathbb{Z}_2 {\to} \mathbb{Z}_2.\ (g_1 g_2) {\cdot} (t(f)) = t(\lambda q \in \mathbb{Z}_2.\, g_2 {\cdot} (f(g_1^{-1} {\cdot} q)))\}\end{aligned}$$

Hence for any $t \in [\![\forall X_1.\forall X_2.(\mathsf{quantity}(X_1) \to \mathsf{quantity}(X_2)) \to \mathsf{quantity}(X_1 \cdot X_2)]\!]$, instantiating $f = \mathrm{id}_Q$ we get that $(g_1 g_2) \cdot (t(\mathrm{id}_Q)) = t(\lambda q \in \mathbb{Z}_2.\, g_2 \cdot (g_1^{-1} \cdot q))$ for all g_1 and g_2, but this is not possible. If $g_1 = 1$ and $g_2 = -1$, then the equation reduces to $-1 \cdot t(\mathrm{id}_Q) = t(\mathrm{id}_Q)$, which is a contradiction since $t(\mathrm{id}_Q) \in \mathbb{Z}_2 = \{-1, 1\}$. ◀

Again, the result can be extended to terms

$$\Gamma_{ops} \vdash t : \forall X_1.\forall X_2.(\mathsf{quantity}(X_1) \to \mathsf{quantity}(X_2)) \to \mathsf{quantity}(X_1 \cdot X_2)$$

as long as all primitive operations in Γ_{ops} can be interpreted in the model.

5 Relational Models

In Section 4 we pointed out that many of the results that we proved in the $\mathsf{Grp}/\!/\mathsf{Set}$ λD-model are results that Kennedy [9] proves using parametricity. It is curious how the parametricity-style proofs in the $\mathsf{Grp}/\!/\mathsf{Set}$ λD-model are simple and slick and do not require a separate relational semantics. One cannot help but wonder, is the $\mathsf{Grp}/\!/\mathsf{Set}$ λD-model really as good as having full-blown parametricity at ones finger tips?

To answer this question we look at a general method of attaching a (fibrational) logic to a λD-model to give a notion of a *relational* λD-model. This allows us to reconstruct Kennedy's relational parametricity in our setting (Example 21), as well as talking about a relational version of the $\mathsf{Grp}/\!/\mathsf{Set}$ λD-model (Example 22).

To begin this section, we first recall a theorem about the composition of fibred structure.

▶ **Theorem 19.** *Suppose that $p : \mathcal{A} \to \mathcal{B}$ and $q : \mathcal{B} \to \mathcal{C}$ are fibrations and let $u : \mathcal{A} \to \mathcal{C}$ denote the composite $q \circ p$ (hence u is also a fibration). Suppose further that q has simple products. For any projection map $\pi_{q(B)} : q(B) \times Y \to q(B)$ in $\mathcal{C}$, denote the Cartesian morphism in $\mathcal{B}$ above it by $\pi^{\S}_{q(B)} : \pi^* B \to B$. Then u has simple products that are preserved by p if and only if for any projection map $\pi : q(B) \times Y \to q(B)$ in $\mathcal{C}$, the functor $(\pi^{\S}_{q(B)})^* : \mathcal{A}_B \to \mathcal{A}_{\pi^* B}$ has right adjoints for all $B \in \mathcal{B}$, satisfying the Beck-Chevalley condition.*

$$\begin{tikzcd} \mathcal{A} \arrow[r, "p"] \arrow[dr, "u"'] & \mathcal{B} \arrow[d, "q"] \\ & \mathcal{C} \end{tikzcd}$$

Proof. This theorem is proven by using the factorisation and lifting properties of the 2-category *Fib* as outlined by Hermida [6]. Though the proof is not too difficult, it does require 2-categorical technology, which we do not introduce here. Hence, we leave the proof as an exercise for the 2-category-savvy reader. ◀

We now put this theorem to use. Given a λD-model $q : \mathcal{A} \to \mathcal{L}$ and a logic $p : \mathcal{E} \to \mathcal{B}$, there is a natural way to glue them together to provide a relational semantics.

$$\begin{tikzcd} F^*(\mathcal{E}) \arrow[r] \arrow[d, "F^*p"'] \arrow[dr, phantom, "\lrcorner", very near start] & \mathcal{E} \arrow[d, "p"] \\ \mathcal{A} \arrow[r, "F"'] \arrow[d, "q"'] & \mathcal{B} \\ \mathcal{L} & \end{tikzcd}$$

▶ **Theorem 20.** *Let $(q : \mathcal{A} \to \mathcal{L}, G, Q_0)$ be a λD-model, $F : \mathcal{A} \to \mathcal{B}$ a product preserving functor and $p : \mathcal{E} \to \mathcal{B}$ a bicartesian closed fibration with products. Consider the pullback of p along F, and let Q_R denote an object in the fibre $\mathcal{E}_{F(Q_0)}$. Then $(q \circ F^*p : F^*\mathcal{E} \to \mathcal{L}, G, (Q_0, Q_R))$ is a λD-model.*

Proof. Clearly G is an Abelian group object in $\mathcal{L}$, and (Q_0, Q_R) is in the fibre $(F^*\mathcal{E})_G$. To check that $F^*p \circ u$ is a bicartesian closed fibration is a simple exercise. Finally, since p has all products, so does F^*p. Hence, $F^*p \circ u$ has simple products by Theorem 19. ◀

Next, we look at an example that uses Theorem 20 to generate Kennedy's original relationally parametric model of dimension types [9] from essentially the dimension-erasure model back in Example 3.

▶ **Example 21.** Let G be an Abelian group. Then using the notation from Theorem 20, let $\mathcal{L}$ be the Lawvere theory of Abelian groups L_{Ab}, $\mathcal{A}$ be the category $\mathsf{L}_{\mathsf{Ab}} \times \mathsf{Set}$, $q : \mathsf{L}_{\mathsf{Ab}} \times \mathsf{Set} \to \mathsf{L}_{\mathsf{Ab}}$ be the fibration given by the first projection, and $p : \mathsf{Sub}(\mathsf{Set}) \to \mathsf{Set}$ be the subset fibration. Define $F : \mathsf{L}_{\mathsf{Ab}} \times \mathsf{Set} \to \mathsf{Set}$ to be the product preserving functor defined on objects

$(n, X) \in \mathsf{L}_{\mathsf{Ab}} \times \mathsf{Set}$ by $F(n, X) = G^n \times X \times X$, and on morphisms $(f, g) : (n, X) \to (m, Y)$ by $F(f, g) = (G^f, g, g)$. Finally, we let $Q_0 = G$, and $Q_R = \{(g, g_1, g_2) \mid gg_1 = g_2\} \subseteq G \times G \times G$.

In this model, each type $\Delta \vdash T$ is interpreted as a triple $(|\Delta|, [\![T]\!]_o, [\![T]\!]_r) \in \mathsf{L}_{\mathsf{Ab}} \times \mathsf{Set} \times \mathsf{Sub}(\mathsf{Set})$, where $[\![T]\!]_r \subseteq G^n \times [\![T]\!]_o \times [\![T]\!]_o$. Spelling this out explicitly, we have the following interpretations, which are equivalent to Kennedy's original relationally parametric model for dimension types:

$$
\begin{array}{ll}
[\![\Delta \vdash \mathsf{quantity}(D)]\!] & = (|\Delta|, G, \{(g, g_1, g_2) \mid ([\![D]\!]g)g_1 = g_2\}) \\
[\![\Delta \vdash T \times U]\!] & = (|\Delta|, [\![T]\!]_o \times [\![U]\!]_o, \\
& \qquad \{(g, (t_1, u_1), (t_2, u_2)) \mid (g, t_1, t_2) \in [\![T]\!]_r, (g, u_1, u_2) \in [\![U]\!]_r\} \\
[\![\Delta \vdash T + U]\!] & = (|\Delta|, [\![T]\!]_o + [\![U]\!]_o, \\
& \qquad \{(g, \mathsf{inj}_1\, t, \mathsf{inj}_1\, t') \mid (g, t, t') \in [\![T]\!]_r\} \cup \{(g, \mathsf{inj}_2\, u, \mathsf{inj}_2\, u') \mid (g, u, u') \in [\![U]\!]_r\} \\
[\![\Delta \vdash T \to U]\!] & = (|\Delta|, [\![T]\!]_o \to [\![U]\!]_o, \\
& \qquad \{(g, f_1, f_2) \mid \forall t_1, t_2.\ (g, t_1, t_2) \in [\![T]\!]_r \implies (g, f_1 t_1, f_2 t_2) \in [\![U]\!]_r\} \\
[\![\Delta \vdash \forall X.\, T]\!] & = (|\Delta|, [\![T]\!]_o, \{(g, t_1, t_2) \mid \forall g' \in G.\ ((g, g'), t_1, t_2) \in [\![T]\!]_r\})
\end{array}
$$

Note that, in the interpretation types $\forall X.\, T$, the "carrier" (i.e., the second component) is exactly the carrier of the interpretation of T.

We can also apply Theorem 20 to obtain a natural relational model for the $\mathsf{Grp}/\!/\mathsf{Set}$ λD-model (Example 9).

▶ **Example 22.** As before, let G be an Abelian group and $\mathcal{L}$ be the Lawvere theory of Abelian groups L_{Ab}. Let $q : \mathcal{A} \to \mathsf{L}_{\mathsf{Ab}}$ be the pullback of the fibration $\mathsf{Grp}/\!/\mathsf{Set} \to \mathsf{Grp}$ along the unique product-preserving functor $M : \mathsf{L}_{\mathsf{Ab}} \to \mathsf{Grp}$ with $M(1) = G$, as in Example 8, so that the objects of $\mathcal{A}$ are triples (n, X, ϕ) with (X, ϕ) a G^n-set. Let $p : \mathsf{Sub}(\mathsf{Set}) \to \mathsf{Set}$ be the subset fibration. Define $F : \mathcal{A} \to \mathsf{Set}$ to be the product preserving functor defined on objects by $F(n, X, \phi) = G^n \times X \times X$ and on morphisms $(f, \alpha) : (n, X, \phi) \to (m, Y, \psi)$ by $F(f, \alpha) = (\alpha, f, f)$. Finally, we let $Q_0 = (G, \phi)$, where ϕ denotes group multiplication, and $Q_R = \{(g, g_1, g_2) \mid gg_1 = g_2\} \subseteq G \times G \times G$.

Then each type $\Delta \vdash T$ is again interpreted as a triple $(|\Delta|, [\![T]\!]_o, [\![T]\!]_r) \in \mathsf{L}_{\mathsf{Ab}} \times \mathsf{Sub}(\mathsf{Set})$, with $[\![T]\!]_r \subseteq G^n \times [\![T]\!]_o \times [\![T]\!]_o$. The only difference between the interpretation of types in this example and Example 21 is the second component of the interpretation of dimension quantification:

$$
\begin{aligned}
[\![\Delta \vdash \forall X.\, T]\!]_r = (|\Delta|, &\{t \in |[\![T]\!]_r| \mid \forall g \in G.\ ((e_{G^{|\Delta|}}, g), t, t) \in [\![T]\!]_r\}, \\
&\{(g, t_1, t_2) \mid \forall g' \in G.\ ((g, g'), t_1, t_2) \in [\![T]\!]_r\})
\end{aligned}
$$

This interpretation, in contrast to the interpretation in Example 21, has "cut-down" the carrier of the interpretation of $\forall$-types to only include the "parametric" elements. As a consequence, this interpretation satisfies an analogue of the *Identity Extension* lemma from relationally parametric models of System F [13].

▶ **Proposition 23.** *For all type interpretations $(|\Delta|, [\![T]\!]_o, [\![T]\!]_r)$, we have:*

$$\forall x_1, x_2 \in [\![T]\!]_o.\ (e, x_1, x_2) \in [\![T]\!]_r \Leftrightarrow x_1 = x_2$$

Compare this to the identity extension property for System F models, which states that if we instantiate the relational interpretation of a type with the equality relation for all of its free variables, then the resulting relation is the equality relation. In the current setting, equality relations for the free variables are replaced by the unit element of the groups $G^{|\Delta|}$. Indeed, this model is equivalent to the restriction to one-dimensional scalings of the reflexive graph model for System Fω with geometric symmetries presented by Atkey [1].

We end this discussion of relational models by showing the relationships between the models in Examples 21 and 22 and the $\mathsf{Grp}//\mathsf{Set}$ model we considered in detail in Section 4. By construction, the carriers of the interpretations of each type in the model in Example 22 and the $\mathsf{Grp}//\mathsf{Set}$ model are identical. Moreover, the relational interpretation in Example 22 and the group action in the $\mathsf{Grp}//\mathsf{Set}$ model are related as follows.

▶ **Theorem 24.** *For all types $\Delta \vdash T$ Type, if the interpretation of T in the model of Example 22 is $(|\Delta|, A, P \subseteq G^{|\Delta|} \times A \times A)$ and the $\mathsf{Grp}//\mathsf{Set}$ model interpretation is (G^n, A, ψ), then $(g, a_1, a_2) \in P \Leftrightarrow g \cdot_\psi a_1 = a_2$.*

Proof. By induction on the derivation of $\Delta \vdash T$ Type. ◀

Using Theorem 24, we can see that we could have used the relationally parametric model to derive the results in Section 4. There is literally no difference between the two models for the purposes of interpreting the types of our calculus.

We can also relate the relationally parametric model from Example 22 to the dimension-erasure semantics in Example 3. By constructing a logical relation between the two models, we can show:

▶ **Theorem 25.** *For any closed term $\vdash t : \mathsf{bool}$, the interpretation of t in the dimension-erasure model of Example 3 is equal to the interpretation of t in the relationally parametric model of Example 22.*

By the compositionality of both interpretations, this theorem means that if we can show that two open terms s and t are equal in the model of Example 22 (and equivalently, the $\mathsf{Grp}//\mathsf{Set}$ model), then they will be contextually equivalent for the dimension erasure model.

It remains to discuss the relationship between Kennedy's original relational model (Example 21), and the relational model in Example 22 that satisfies the identity extension property. As noted above, the difference between these interpretations lies in the semantics of the $\forall$-type. Kennedy's model does not restrict the carrier of the interpretation to just the "parametric" elements, *i.e.*, the elements that preserve all relations. Therefore, the interpretations of types that contain nested $\forall$s are not directly comparable. We might expect that we could observe a difference between the two models when proving statements about terms whose types contain negatively nested forall types. However, Kennedy's original work does not present any results involving terms with such types, and we have not found any natural examples. This is in contrast with the situation with relationally parametric models of System F, where the proof that final coalgebras can be represented crucially relies on the restriction of the interpretation of quantified types to the parametric elements [2].

Therefore, our $\mathsf{Grp}//\mathsf{Set}$ model and the equivalent relational model in Example 22 practically coincides with Kennedy's original model, but offer the advantage of not requiring a separate relational semantics to prove important theorems. This in many cases makes proofs of these theorems clearer. Additionally, the $\mathsf{Grp}//\mathsf{Set}$ model offers an interpretation that directly links the semantics to symmetry.

6 Concluding Remarks

To conclude, we have studied a typed λ-calculus with polymorphism over physical dimensions, which we called λD (Section 2) and we have developed a model theory for the calculus. Under the Curry-Howard correspondence, the λD-calculus is a fragment of first-order logic where the domain of discourse is an unspecified Abelian group, and so our notion of model (Definition 2) is based on the standard fibrational techniques in categorical logic.

One particular model turned out to be particularly straightforward and yet informative — the model based on group actions (Example 9). Of course, automorphisms and group actions play a key role in the classical model theory of first order logic, but in this paper we have shown that these techniques are also useful on the other side of the Curry-Howard correspondence. Many arguments about the λD-calculus, including type isomorphisms and definability arguments, can be made in this model (Section 4).

Parametricity is most often studied using relational techniques, and in this paper we have developed a method for building relational λD-models (Theorem 20). Using this method we were able to reconstruct two particular relational models: a relational model due to Kennedy (Example 21, [9]) and a restriction of a relational model due to Atkey (Example 22, [1]). Although the group-actions model is different in style, we showed (formally) that it is actually closely related to the two relational models (Theorems 24 and 25).

Acknowledgements

We are grateful to participants in a discussion on the categories mailing list about the Grp//Set fibration.

References

1. Robert Atkey. From parametricity to conservation laws, via Noether's theorem. In *Proceedings of the 41st ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL 2014)*, pages 491–502, 2014.
2. Lars Birkedal and Rasmus E Møgelberg. Categorical models for Abadi and Plotkin's logic for parametricity. *Mathematical Structures in Computer Science*, 15(04):709–772, 2005.
3. Ronald Brown and Christopher B Spencer. G-groupoids, crossed modules and the fundamental groupoid of a topological group. *Proc. Indag. Math.*, 79(4):296–302, 1976.
4. Edgar Buckingham. On physically similar systems; illustrations of the use of dimensional equations. *Physical Review*, 4(4):345–376, 1914.
5. Martin Erwig and Margaret Burnett. Adding apples and oranges. In *Practical Aspects of Declarative Languages*, pages 173–191. Springer, 2002.
6. Claudio Hermida. Some properties of Fib as a fibred 2-category. *Journal of Pure and Applied Algebra*, 134(1):83–109, 1999.
7. Ronald T. House. A proposal for an extended form of type checking of expressions. *The Computer Journal*, 26(4):366–374, 1983.
8. Bart Jacobs. *Categorical logic and type theory*, volume 141. Elsevier, 1999.
9. Andrew J. Kennedy. Relational parametricity and units of measure. In *Proceedings of the 24th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, POPL '97, pages 442–455, New York, NY, USA, 1997. ACM.
10. F. William Lawvere. Adjointness in foundations. *Dialectica*, 23(3-4):281–296, 1969.
11. R Männer. Strong typing and physical units. *ACM Sigplan Notices*, 21(3):11–20, 1986.
12. Paul-André Melliès and Noam Zeilberger. Functors are type refinement systems. In *Proceedings of the 42nd ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL 2015)*, pages 3–16, 2015.
13. John Reynolds. Types, abstraction and parametric polymorphism. In *Information Processing*, 1983.
14. Ain A Sonin. The physical basis of dimensional analysis. *Department of Mechanical Engineering, MIT, Cambridge, MA*, 2001.
15. Mitchell Wand and Patrick O'Keefe. Automatic dimensional inference. In *Computational Logic – Essays in Honor of Alan Robinson*, pages 479–483, 1991.