Directed equality with dinaturality (Updated version)

Andrea Laretto
Tallinn University of Technology, Estonia

Fosco Loregian
Tallinn University of Technology, Estonia

Niccolò Veltri
Tallinn University of Technology, Estonia

Abstract

We show how dinaturality plays a central role in the interpretation of directed type theory where types are interpreted as (1-)categories and directed equality is represented by hom-functors. We present a general elimination principle based on dinaturality for directed equality which very closely resembles the J-rule used in Martin-Löf type theory, and we highlight the syntactic restrictions needed to interpret this rule in the directed case. We argue that the quantifiers of such a directed type theory should be interpreted as ends and coends, which dinaturality allows us to present in adjoint-like correspondences to a weakening operation. We then combine these rules together to give a logical interpretation to (co)end calculus and Yoneda reductions, and we use formal derivations to prove the Fubini rule for quantifier exchange, the adjointness property of Kan extensions via (co)ends, exponential objects of presheaves, and the (co)Yoneda lemma. We show transitivity (composition), congruence (functoriality), and transport (coYoneda) for directed equality by closely following the same approach of Martin-Löf type theory, with the notable exception of symmetry. Our main theorems are formalized in Agda.

2012 ACM Subject Classification Theory of computation → Type theory

Keywords and phrases directed type theory, coend calculus, dinaturality

Digital Object Identifier 10.4230/LIPIcs.CVIT.2016.23

Funding *Fosco Loregian*: Loregian was supported by the Estonian Research Council grant PRG1210.
Niccolò Veltri: Veltri was supported by the Estonian Research Council grant PSG749.

1 Introduction

Equality is one of the most interesting aspects of Martin-Löf type theory: for any $A : \mathsf{Type}$ and $a, b : A$ there is a type of equalities $a =_A b : \mathsf{Type}$, and since this is itself a type, one can talk about the type of equalities between equalities $p =_{a=_A b} q : \mathsf{Type}$ for any $p, q : a = b$. This is the fundamental idea behind homotopy type theory [14, 61], and it allows for types to be interpreted as groupoids [32], where not all equalities are themselves equal, as well as ∞-groupoids [6], where such iterated equalities might never trivialize. In these settings, the inherently *symmetric* nature of equality is what enables types to be given as (∞-)groupoids, where equality is precisely interpreted by morphisms which are always invertible. A natural question follows: can there be a variant of Martin-Löf type theory which enables types to be interpreted as *categories*, where morphisms need not be invertible? Such a system should take the name of *directed type theory* [1, 47, 39, 26], where the directed aspect comes from a non-symmetric interpretation of "equality", which now possesses both a source and a target in the same way that morphisms do in a category.

Directed type theory has been a sought-after goal of recent type-theoretic research [26, 2], with several attempts aimed at pinpointing precisely both its syntactic and semantic aspects. A fundamental aspect of this (1-)category interpretation of type theory is the fact that with each type/category $\mathbb{C}$ there is a naturally associated type $\mathbb{C}^{\mathsf{op}}$, where the objects are the same

42nd Conference on Very Important Topics (CVIT 2016).
Editors: John Q. Open and Joan R. Access; Article No. 23; pp. 23:1–23:26
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

but all directed equalities are reversed. This allows for a *type of directed equalities* $\hom_{\mathbb{C}}(a,b)$ to be given in terms of the functor $\hom_{\mathbb{C}} : \mathbb{C}^{\mathsf{op}} \times \mathbb{C} \to \mathsf{Set}$, which receives a "contravariant" argument $a : \mathbb{C}^{\mathsf{op}}$ and a "covariant" one $b : \mathbb{C}$ and provides a *set* $\hom_{\mathbb{C}}(x,y)$ (i.e., a category with only trivial directed equalities) of directed equalities between any two objects x, y of $\mathbb{C}$. A directed type theory should therefore have some notion of "*polarity*" or "*variance*" which allows variables to appear only in certain positions, as treated in [39, 46, 47, 48].

The introduction rule for symmetric equality is typically given by the term $\mathsf{refl}_a : a = a$. In a directed type theory where types are categories, this should be motivated by the fact that there is a directed equality $\mathsf{id}_a \in \hom_{\mathbb{C}}(a,a)$ (i.e., the identity) for any $a : \mathbb{C}$. However, naïvely stating this typing rule as "$\mathsf{refl}_a : \hom_{\mathbb{C}}(a,a)$" would involve both a contravariant and a covariant occurrence of the variable $a : \mathbb{C}$, and would not be given functorially with respect to the variance of hom. One solution considered by North [47] is to use the maximal subgroupoid $\mathbb{C}^{\mathsf{core}}$ to collapse the two variances, since $(\mathbb{C}^{\mathsf{core}})^{\mathsf{op}} \cong \mathbb{C}^{\mathsf{core}}$. The refl rule can then be expressed as $\mathsf{refl}_a : \hom(\mathsf{i}^{\mathsf{op}}(a), \mathsf{i}(a))$ via the embeddings $\mathsf{i} : \mathbb{C}^{\mathsf{core}} \to \mathbb{C}$ and $\mathsf{i}^{\mathsf{op}} : \mathbb{C}^{\mathsf{core}} \to \mathbb{C}^{\mathsf{op}}$. Another fundamental rule needed to work with equalities is a way to *eliminate* them; this is classically done with the so-called *J-rule* [31],

$$\frac{C : \mathsf{Type}, \quad P : \prod_{a,b:C}(a =_C b \to \mathsf{Type}), \quad t : \prod_{x:C} P(x,x,\mathsf{refl}_x)}{J(t) : \prod_{a,b:C} \prod_{e:a=_C b} P(a,b,e)} \ (J\text{-rule})$$

where the *computation rule* $J(x,x,\mathsf{refl}_x) \equiv t(x)$ holds definitionally. Intuitively, to prove a proposition $P(a,b,e)$ for terms $a, b : C$ with an equality $e : a = b$, it is sufficient to consider the case where a and b are the same term x, and the equality e is exactly refl_x. This fact that "it is enough to consider the case where the equality is refl" bears a striking similarity to the fundamental idea underlying the Yoneda lemma, one of the most central and praised results in category theory [41]. Viewing the Yoneda lemma as a sort of (directed) J-rule is applied in practice by [56, 25] in simplicial type theory, and is investigated in HoTT in [20].

Quantifiers and (co)ends. A central yet unexplored question is how *quantifiers* should be interpreted in the types-as-categories semantics of directed type theory. A well-known rephrasing of the Yoneda lemma (called "ninja" Yoneda lemma [37]) provides inspiration for a possible answer, described in detail in Section 3 and introduced here intuitively. The set of natural transformations appearing in the Yoneda lemma can be characterized in terms of a universal object called the *end* of a functor of a specific type [41, IX.5], [40]. Given a functor $P : \mathbb{C}^{\mathsf{op}} \times \mathbb{C} \to \mathbb{D}$, the *end* of P, denoted as $\int_{x:\mathbb{C}} P(\overline{x}, x)$, is an object of $\mathbb{D}$ with a certain terminal universal property. Notationally, the integral sign of ends binds covariant and contravariant occurrences of variables for the rest of the expression, and we indicate with $\overline{x} : \mathbb{C}^{\mathsf{op}}$ the contravariant occurrences of variables $x : \mathbb{C}$. Ends of certain functors into Set characterize natural transformations: for any two functors $F, G : \mathbb{C} \to \mathbb{D}$ there is an isomorphism $\mathsf{Nat}(F,G) \cong \int_{x:\mathbb{C}} \hom_{\mathbb{D}}(F(\overline{x}), G(x))$ natural in F, G. Note the resemblance between the end of the above functor and the universal quantification expressed in the elementary definition of natural transformation. This allows the Yoneda lemma to be rephrased as an isomorphism (natural in $a : \mathbb{C}$ and $P : \mathbb{C} \to \mathsf{Set}$) between $P(a)$ and a certain set of naturals computed as an end; we show this isomorphism in Figure 1a above its "decategorification"[1], where ends are viewed as a sort of universal quantifier, presheaves as (proof-relevant) predicates, and directed equality is turned back into symmetric equality.

[1] We use double lines in this section to suggest a correspondence between the connectives of both formulas, without giving it here a formal meaning.

(a) $$\frac{P(a) \;\cong\; \int_{x:\mathbb{C}} \;\; \hom_{\mathbb{C}}(a,\overline{x}) \Rightarrow P(x)}{P(a) \Leftrightarrow \forall(x:C).\;\; a =_C x \;\; \Rightarrow P(x)}$$ **(b)** $$\frac{P(a) \;\cong\; \int^{x:\mathbb{C}} \;\; \hom_{\mathbb{C}}(\overline{x},a) \times P(x)}{P(a) \Leftrightarrow \exists(x:C).\;\; x =_C a \;\; \wedge P(x)}$$

Figure 1 Yoneda and coYoneda lemma using (co)ends and their corresponding logical statements.

There is a similar logical correspondence using existential rather than universal quantifiers, shown in Figure 1b: existentials are now interpreted by the dual notion of ends, *coends* [40], denoted as $\int^{x:\mathbb{C}} P(\overline{x}, x)$. Since (co)ends are (co)limits, this result also takes the well-known slogan of "presheaves are colimits of representables" [38] or "coYoneda lemma" [40]. This correspondence with first order logic allows us to reframe these celebrated results of category theory as simple logical equivalences of formulas, which one can validate with a formal system such as sequent calculus or type theory. However, there is currently no formal system for the *directed* case where one can modularly use rules for quantifiers and equality as done in logic, e.g., with suitable introduction/elimination rules specific to directed equality and (co)ends.

(Co)end calculus. There is a formal aspect to the manipulation of ends and coends, outlined in [40], which is common knowledge among theoreticians, that allows non-trivial theorems to be proven using simple formal rules reminiscent of a deductive system. Apart from its applications in pure category theory [40, 60, 16, 33], such "(co)end calculus" has proven to be particularly useful in theoretical computer science, for example in the context of profunctor optics [13, 11] and their string diagrams [57, 10], strong monads and functional programming [3, 4, 30, 63], quantum circuits [27], and logic [52, 54, 23]. Leveraging this abundance of practical applications, [40] hints at the existence of such deductive system, but falls short of precisely pinpointing its rules. This particular application to coend calculus, which as we will see can be interpreted logically as a first-order version of directed type theory, is what motivates our focus on a non-dependent presentation of directed type theory.

Dinaturality. The notion of (co)end, first introduced by Yoneda [67], is intimately connected to dinaturality: dinatural transformations are a generalization of natural transformations which considers families of morphisms between functors with two variances $F : \mathbb{C}^{\mathsf{op}} \times \mathbb{C} \to \mathbb{D}$, and were first introduced by Dubuc and Street [18] as an extension to the extranatural transformations by Eilenberg and Kelly [19]. Famously, however, such generalized natural transformations *do not always compose*: a well-known sufficient condition for the composability of extranaturals is the absence of loops in a suitably associated graph [19], an idea recently revived for dinaturals in [44]. Despite the apparent lack of composition, there are various examples of settings in which all dinaturals compose: the fundamental idea is to single out a class of "definable" dinaturals where composition follows because their formation rules are precisely defined, and given syntactically. This point of view finds its most natural application in the logical setting, where composition corresponds to cut elimination [50, 7, 8, 9, 24]. There is a particularly deep connection between dinaturality and parametricity in programming languages [51, 53, 64], as well as realizable models for System F [5, 21] where all dinaturals compose. An in-depth review on dinaturality and its importance for both computer science and category theory can be found in [58] and [59, Sec. 3].

1.1 Contribution

In this paper we describe how dinaturality allows us to semantically validate an introduction and an elimination rule for directed equality exactly in the style of Martin-Löf type theory, where directed equality is interpreted by hom-functors, types by (1-)categories. Motivated by

the application to (co)end calculus, we consider a (proof-relevant) non-dependent treatment of directed type theory, using (co)presheaves as predicates and dinatural transformations as entailments which are not required to compose. The intuition behind dinatural transformations is that they allow the same variable to appear both covariantly and contravariantly: this is exactly what allows us to resolve the variance problems previously mentioned in the directed refl rule, which we validate using identities in hom-sets. We present a directed equality elimination rule which is syntactically reminiscent of the J-rule used in standard Martin-Löf type theory, and dinaturality is again what permits the same variable x to appear with both variances in the type $P(x, x, \mathsf{refl}_x)$ in (J-rule). This elimination rule is semantically motivated by the connection between dinaturality and naturality, and sheds a light on the syntactic restrictions imposed in a full type theory where equality is now directed rather than symmetric: in short, the syntactic requirement to contract a directed equality in context $\hom_{\mathbb{C}}(x, y)$ for $x : \mathbb{C}^{\mathsf{op}}, y : \mathbb{C}$ is that both x and y must appear only positively (i.e., with the same variance) in the conclusion and only negatively (i.e., with the opposite variance) in the assumptions in context. The rules for directed equality allow us to recover the same type-theoretic definitions about symmetric equality that one expects in standard Martin-Löf type theory, except for symmetry: e.g., transitivity of directed equality (composition in a category), congruences of terms along directed equalities (the action of a functor on morphisms), transport along directed equalities (i.e., the coYoneda lemma). We highlight how the syntactic restrictions imposed by this rule for directed equality elimination do not allow us to obtain that directed equalities are symmetric.
Moreover, we show how dinaturality allows us to more precisely view (co)ends as the "directed quantifiers" of directed type theory, which we present in a correspondence reminiscent of the quantifiers-as-adjoints paradigm of Lawvere [36]. The semantic setting in which we validate our rules is by considering a categorification (both proof-relevant and directed) of non-dependent first-order logic: types are (small) categories (possibly with $-^{\mathsf{op}}$), contexts are lists of categories, terms are functors $\mathbb{C}^{\mathsf{op}} \times \mathbb{C} \to \mathbb{D}$, propositions are functors of type $\mathbb{C}^{\mathsf{op}} \times \mathbb{C} \to \mathsf{Set}$, and entailments are dinatural transformations. We do not provide an account of these rules using categorical semantics precisely because dinaturals do not compose in general; despite this lack of general composition, the rules for directed equality and coends-as-quantifiers can be jointly used to give concise proofs of central theorems in category theory using a distinctly *logical* flavour via a series of isomorphisms: e.g., the (co)Yoneda lemma, Kan extensions computed via (co)ends are adjoint to precomposition, presheaves form a closed category, hom-functors preserve (co)limits, and the Fubini rules; each of these easily follows by modularly using the logical properties of each connective used. Our treatment of (co)ends as quantifiers, combined with a formal view of hom in terms of directed equality rules, is a concrete step towards formally understanding the so-called "(co)end calculus" [40] from a logical perspective. We formalize the main theorems given in this paper about dinaturality using the Agda proof assistant and the `agda-categories` library. Whenever present, the symbol ( ) next to theorems links to the formal proof, for which we report here just the core idea. The full formalization is accessible at `https://github.com/iwilare/dinaturality`.

1.2 Related work

Directed type theory with groupoids. North [47] describes a dependent directed type theory with semantics in the category of (small) categories Cat, but uses groupoidal structure to deal with the problem of variance in both introduction and elimination rules for directed equality. This line of research has been recently expanded in [12] by extending judgements with variance annotations. We focus on non-dependent semantics, and tackle the variance issue precisely with the notion of dinatural transformation; this allows us to characterize directed equality intrinsically, without using any of the groupoidal structure of categories.

Directed type theory, judgemental models. Another approach to modeling directed equality is at the judgemental level. This line of research started with Licata and Harper [39] who introduced a directed type theory with a model in Cat. Since directed equality is treated judgementally, there are no rules governing its behaviour in terms of elimination and introduction principles, although variances are present in the context as we similarly do in our approach. Ahrens et al. [1] similarly identify a type theory with judgmental directed equalities with sound semantics in comprehension bicategories, and extensively compare previous works on both judgemental and propositional directed type theories.

Synthetic logics for category theory. New and Licata [46] give a sound and complete presentation for the internal language of (hyperdoctrines of) certain virtual equipments. These models capture enriched, internal, and fibered categories, and have an intrinsically directed flavour. In these contexts, the type theory can give synthetic proofs of Fubini, Yoneda, and Kan extensions as adjoints. This generality however comes at the cost of a non-standard syntactic structure of the logic, for example when compared to standard Martin-Löf type theory, along with some non-trivial syntactic judgements prescribed by the structure of the models. Directed equality elimination here takes the shape of the (horizontal) identity laws axiomatized in virtual equipments [15], which in Prof is essentially the coYoneda lemma. Their quantifiers are given by the universal properties of tensor and (left/right) internal homs, which in the Prof model are given by certain restricted (co)ends which always come combined with the tensors and internal homs of Set. Our work is similar in spirit in that we provide a formal setting for proving category theoretical theorems using logical methods, but we only focus on the elementary 1-categorical model of categories and do not yet capture enriched and internal settings. However, we treat ends and coends as quantifiers *directly*, with adjoints-like correspondences to weakening functors which only act on the variables of the context and without the need for quantifiers to include (restricted forms of) conjunction and implication. Our rules for directed equality are more direct and reminiscent of standard Martin-Löf type theory, and specifically focus on the semantic justification based on dinaturality. Since we consider less general models, our contexts do not have any linear nor ordered restriction, and the same variable can appear multiple times both in equalities and contexts. This allows us to consider profunctors of many variables and different variances as typically needed in coend calculus.

Geometric models of directed type theory. Riehl and Shulman [56] introduce a simplicial type theory based on a synthetic description of $(\infty, 1)$-categories. A directed interval type is axiomatized in a style reminiscent of cubical type theory [14], which allows a form of (dependent) Yoneda lemma to be proven using the structure of the identity type. This type theory has been implemented in practice in the Rzk proof assistant [35]. On this line of research, Weaver and Licata [65] present a *bicubical* type theory with a directed interval and investigate a directed analogue of the univalence axiom; the same objective was recently explored in Gratzer et al. [26] with triangulated type theory and modalities. In comparison with the above works, we do not explore the geometrical interpretation of directedness and focus on elementary 1-categorical semantics; moreover, our treatment of directed equality is done intrinsically with elimination rules as in Martin-Löf type theory rather than with synthetic intervals, with semantics directly provided by hom-functors.

Coend calculus, formally. Caccamo and Winskel [17] give a formal system in which one can work with coends and establish non-trivial theorems with a few syntactical rules. The flavour is explicitly that of an axiomatic system, and does not take inspiration from type-theoretic rules: for instance, presheaves are *postulated* to be equivalent under the swapping of quantifiers (Fubini), so this principle is not derived from structural rules as typically done in a logical presentation.

2 Syntax and Semantics

We now present the syntactic judgements of a proof-relevant non-dependent first-order directed type theory, with main semantics in 1-categories. We will consider the following judgements along with their semantics:

- $\boxed{\mathbb{C}\ \mathsf{type}}$ **types** $\mathbb{A}, \mathbb{C}, \mathbb{D}$ are interpreted as small categories, possibly with $-^{\mathsf{op}}$; we axiomatize these by modeling terminal $\top$, product $\mathbb{C} \times \mathbb{D}$ and functor categories $[\mathbb{C}, \mathbb{D}]$. We add a base type C^{op} for each base type C in a fixed signature Σ, and then define an involution $-^{\mathsf{op}}$ on types *by induction*, following the semantics in Cat in the intuitive way. A similar definition of op can be given for terms and predicates, which we detail in Figure 7.
- $\boxed{\Gamma\ \mathsf{ctx}}$ **contexts** Γ, Δ are interpreted as *finite products of categories*;
- $\boxed{\Gamma \vdash F : \mathbb{C}}$ **terms** F, G as *functors* $[\![\Gamma]\!] \to [\![\mathbb{C}]\!]$, which we axiomatize like terms in STLC;
- $\boxed{[\Gamma]\ P\ \mathsf{prop}}$ **predicates** P, Q, R, A, B as *dipresheaves*, i.e., functors $[\![P]\!] : [\![\Gamma]\!]^{\mathsf{op}} \times [\![\Gamma]\!] \to \mathsf{Set}$;
- $\boxed{[\Gamma]\ \Phi\ \mathsf{propctx}}$ **propositional contexts** Φ, Φ' as *pointwise products of dipresheaves*;
- $\boxed{[\Gamma]\ \Phi \vdash \alpha : P}$ **entailments** α, β, γ as *dinatural transformations* $[\![\Phi]\!] \xrightarrow{\bullet\bullet} [\![P]\!]$; we axiomatize composition/cut only with *natural* transformations, not requiring general composition;
- $\boxed{[\Gamma]\ \Phi \vdash \alpha = \beta : P}$ **equality of entailments** as *equality between dinaturals in* Set.

For predicates/dipresheaves, we consider base predicates and the following logical connectives:

- **conjunction**, interpreted as the pointwise product $- \times -$ of dipresheaves in Set;
- **implication**, by postcomposing dipresheaves with the functor $- \Rightarrow - : \mathsf{Set}^{\mathsf{op}} \times \mathsf{Set} \to \mathsf{Set}$;
- **propositional directed equality** is interpreted by hom-*functors* $\hom_{\mathbb{C}} : \mathbb{C}^{\mathsf{op}} \times \mathbb{C} \to \mathsf{Set}$;
- **quantifiers** $\forall, \exists$ as *ends and coends*, logically representing universal and existential quantifiers on predicates/dipresheaves respectively, which we consider in Section 3.

The full set of types, terms, and predicate formation rules is given in Appendix A. We show in Figure 2 only the main rules for entailments, for which we describe the semantics in Section 2.3. The rules for entailments implicitly use the notion of variance for variables, introduced in Remark 12; these are formally captured by the predicates in Figures 5 and 6.

▶ Remark 1. Because of the lack of composition of dinatural transformations, we do not consider a categorical semantics of this syntactic system using standard categorical models, e.g., fibrations [34], since most of them ask for full composition, which cannot be guaranteed in our semantics. We discuss other possible models in Section 4. Hence, our approach is to simply consider the main rules described in Figure 2 (which are equipped with *restricted* rules for composition of entailments) and validate them using categories, dipresheaves, dinatural transformations, and, e.g., introduction and elimination rules for hom-types. We then put these rules into practice by showing (in Section 3.1) how we can prove standard theorems in category theory using a distinctly logical flavour, as well as showing (in Section 2.4) how the rules for directed equality can be used to construct maps and prove properties about directed equality precisely as it can be done in Martin-Löf type theory for the symmetric case. The reader fluent in categorical logic [34, 42] can imagine our logic to capture the behaviour of a specific doctrine $\mathsf{Dinat} : \mathsf{Cat}^{\mathsf{op}} \to \mathsf{PARACAT}$ given by sending a small category $\mathbb{C}$ to the large *paracategory* $[\mathbb{C}^{\mathsf{op}} \times \mathbb{C}, \mathsf{Set}]_{\mathsf{dinat}}$ (a variation of category where the composition operation is partial, see [28, 29]), where such paracategory is defined by having dipresheaves $\mathbb{C}^{\mathsf{op}} \times \mathbb{C} \to \mathsf{Set}$ as objects and *dinatural transformations* as between them as morphisms.

2.1 Dinaturality

The fundamental idea behind the logic is that variables in the term context can appear in predicate symbols both covariantly and contravariantly. This is what allows us to resolve the variance problems in the introduction and elimination rules for directed equality, and is precisely what dinatural transformations capture. We recall basic notions about dinatural transformations, which we abbreviate simply as "dinaturals", and ordinary natural transformations as "naturals".

▶ **Definition 2** (Dipresheaves). *Consider the (strict) comonad* $-^{\diamond} : \mathsf{Cat} \to \mathsf{Cat}$ *defined by* $\mathbb{C} \mapsto \mathbb{C}^{\mathsf{op}} \times \mathbb{C}$*, where the counit is given by projecting and comultiplication by duplicating and swapping. A* dipresheaf *is simply a functor* $\mathbb{C}^{\diamond} \to \mathsf{Set}$ *(i.e. a functor* $\mathbb{C}^{\mathsf{op}} \times \mathbb{C} \to \mathsf{Set}$*).*

We always denote composition diagrammatically, i.e., $f \mathbin{;} g : a \to c$ for $f : a \to b, g : b \to c$.

▶ **Definition 3** (Dinatural transformation [18]). *Given functors* $F, G : \mathbb{C}^{\diamond} \to \mathbb{D}$*, a* dinatural transformation $\alpha : F \xrightarrow{\bullet\bullet} G$ *is a family of arrows* $\alpha_x : F(x,x) \longrightarrow G(x,x)$ *indexed by objects* $x : \mathbb{C}$ *such that for any* $a, b : \mathbb{C}$ *and* $f : a \to b$ *the following equation holds:*
$F(\mathsf{id}_b, f) \mathbin{;} \alpha_b \mathbin{;} G(f, \mathsf{id}_b) = F(f, \mathsf{id}_a) \mathbin{;} \alpha_a \mathbin{;} G(\mathsf{id}_b, f)$.

▶ **Lemma 4** (Dinaturals generalize naturals [18]). *A natural transformation* $\alpha : F \to G$ *for* $F, G : \mathbb{C} \to \mathbb{D}$ *equivalently corresponds with a dinatural* $\alpha : (\pi_2 \mathbin{;} F) \xrightarrow{\bullet\bullet} (\pi_2 \mathbin{;} G) : \mathbb{C}^{\diamond} \to \mathbb{D}$*.*

▶ **Lemma 5** (Naturality to dinaturality). *Naturality in two variables with different variance can be "collapsed" to dinaturality in a single one: given* $F, G : \mathbb{C}^{\mathsf{op}} \times \mathbb{C} \to \mathbb{D}$ *and a natural* $\alpha : F \to G$*, there is a dinatural* $\Delta(\alpha) := \alpha_{xx} : F \xrightarrow{\bullet\bullet} G$*.*

The pointwise composition of two dinatural transformations is not necessarily dinatural (see [44, 22]), but dinaturals always compose with naturals on both the left and right side:

▶ **Lemma 6** (Dinaturals compose with naturals [18]). *Given a dinatural transformation* $\gamma : F \xrightarrow{\bullet\bullet} G$ *and natural transformations* $\alpha : F' \to F, \beta : G \to G'$ *for* $F, F', G, G' : \mathbb{C}^{\mathsf{op}} \times \mathbb{C} \to \mathsf{Set}$*, the map* $\alpha \mathbin{;} \gamma \mathbin{;} \beta : F' \xrightarrow{\bullet\bullet} G'$ *defined by* $(\alpha \mathbin{;} \gamma \mathbin{;} \beta)_x := \alpha_{xx} \mathbin{;} \gamma_x \mathbin{;} \beta_{xx}$ *is dinatural.*

2.2 Notation

We now introduce the concepts of *position, polarity, and variance*, and then describe the notation for variables, predicates and entailments used in the type theory.

▶ **Definition 7** (Positions in a predicate). *The name* position *to indicates a point in which a variable can appear in a predicate, e.g., there are four possible positions* x, y, z, w *for variables to appear in the predicate* $\hom(x, y) \times P(z, f(w))$*.*

▶ **Definition 8** (Polarity of a position). *Positions have a* polarity*, which can either be* positive *or* negative*: intuitively, a position starts out as positive, and flips between being positive and negative precisely in the following cases:*
1. *when it occurs on the left side of* $\hom(x, y)$*, e.g.,* x *is negative in* $\hom(x, c)$*,* $\hom(f(x), y)$*;*
2. *when it occurs on the left of implication* $P \Rightarrow Q$*, e.g.,* x *is negative in* $P(f(x)) \Rightarrow Q(y)$*;*
3. *when it occurs on the left side* $\cdot_1$ *of a base predicate symbol* $P(\cdot_1, \cdot_2) : \mathbb{C}^{\mathsf{op}} \times \mathbb{C} \to \mathsf{Set}$*.*

Moreover, variance can be inverted twice in the intuitive way: for example, x *occurs* positively *in the predicate* $\hom(x, y) \Rightarrow P$ *and* $(\hom(y, x) \Rightarrow P) \Rightarrow P$*.*

Semantically, this flipping of variance corresponds with the presence of the opposite category $\mathbb{C}^{\mathsf{op}}$ on the left side of the functors $\hom_{\mathbb{C}} : \mathbb{C}^{\mathsf{op}} \times \mathbb{C} \to \mathsf{Set}$, and $- \Rightarrow - := \hom_{\mathsf{Set}}$. The fact that $(\mathbb{C}^{\mathsf{op}})^{\mathsf{op}} \equiv \mathbb{C}$ justifies the fact that inverting a negative variable makes it positive again.

▶ **Definition 9** (Variance of a variable). *Variables can occur in multiple positions at the same time: we say that the* variance *of variable x is* positive *iff every position in which it occurs is positive; similarly for negative. A variable is* natural *iff it is either positive or negative. Variance in terms and predicates is captured formally with the judgements in Figures 5 and 6.*

▶ Remark 10 (Notation for variance of variables). We introduce notation to highlight the contravariance of variables in predicates. We indicate with "$\overline{x}$" the *negative* (or "contravariant") occurrences of variables $x : \mathbb{C}$, and simply "x" for the *positive* (or "covariant") ones. We use the same terminology even when $\mathbb{C} := (\mathbb{C}')^{\mathsf{op}}$ is the opposite of some $\mathbb{C}'$.

▶ Remark 11 (Notation for entailments). The type-theoretic notation for entailments $[x : \mathbb{C}, y : \mathbb{D}, ...]\ a : P(\overline{x}, x, \overline{y}, y, ...), b : Q(\overline{x}, x, \overline{y}, y, ...), ... \vdash \alpha[a, b, ...] : R(\overline{x}, x, \overline{y}, y, ...)$ is interpreted semantically as the statement "α is a dinatural from the functor $[\![P]\!] \times [\![Q]\!] \times ...$ to $[\![R]\!] : ([\![\mathbb{C}]\!] \times [\![\mathbb{D}]\!] \times ...)^{\diamond} \to \mathsf{Set}$", where the former functor is interpreted by the pointwise product of the dipresheaves in the list of assumptions $\Phi := P, Q, ...$, which are given names and freely permuted whenever needed. The term context $[x : \mathbb{C}, y : \mathbb{D}, ...]$ indicates the indices of the dinatural, which is to be thought as the context over which the entailments of a fiber live in a fibration [34]. We use square brackets $\alpha[a, b, ...]$ both to indicate assumptions and for functional application in Set, e.g., $\alpha_c[a, b] \in P(c, c)$ whenever $c \in \mathbb{C}$ and $a \in P(c, c), b \in Q(c, c)$. We will often omit in P the (unrestricted) presence of variables coming from a context Γ.

▶ Remark 12 (Variables in entailments). We indicate with $P(\overline{x}, x, \overline{y}, y)$ the fact that the predicate P *can* depend on x, y both negatively and positively; when either polarity is omitted, e.g., in $P(x, \overline{y})$, the predicate must depend *only* on x and $\overline{y}$, i.e., naturally. These restrictions are formally captured using the predicates for variance of Definition 9 in the intuitive way, which we omit their explicit use in the rules for entailments. Naturality for entire contexts is given by $[y : \Gamma]\ P(y)$, i.e., all variables in Γ are used only *positively* in P.

▶ Remark 13 (Notation for inference rules of dinaturals). Following the interpretation of dinaturals as entailments, we use trees of inference rules to indicate that, given certain dinatural(s) as in the rule premise, one obtains a dinatural as in the rule conclusion. Semantically, inference rules are interpreted by functions between sets of dinaturals (natural in all dipresheaves involved). We use double lines to indicate natural isomorphisms of sets of dinaturals, and often omit the name of such isomorphisms in rules, especially in Section 3.1.

2.3 Rules

We now validate and describe the intuition behind each rule for predicates and entailments. Whenever present, the symbol (🐓) links to the Agda formalization of its semantics.

▶ **Theorem 14** (Dinatural semantics). *Each rule presented in Figure 2 is validated using the semantics in categories, functors, dipresheaves, dinatural transformations. Sequents are interpreted by functions between sets of dinaturals; these are isomorphisms when double-lines appear. Moreover, every function is* natural *in all the dipresheaves that appear in the rule.*

We unpack this theorem by validating and describing the intuition behind each rule.
Products. (🐓) Dinaturals validate the interpretation of conjunction in (prod) via the pointwise product of dipresheaves in Set; the bottom sequent indicates the product of sets of dinaturals. Moreover, (wk) states that dinaturals always compose on the left with projections.
Op-types. The entailments $[x : \mathbb{C}]\ F(\overline{x}) \vdash \alpha\ : G(\overline{x}, x)$ and $[x : \mathbb{C}^{\mathsf{op}}]\ F(x) \vdash \alpha' : G(x, \overline{x})$ clearly denote the same dinatural in the semantics, for $F : \mathbb{C}^{\mathsf{op}} \to \mathbb{D}, G : \mathbb{C}^{\mathsf{op}} \times \mathbb{C} \to \mathbb{D}$. We capture this idea with the rule (op), stating precisely that positive positions of type $\mathbb{C}^{\mathsf{op}}$ are the same as negative positions of type $\mathbb{C}$, and viceversa.

$$\boxed{[\Gamma]\ \Phi \vdash \alpha : P} \qquad \frac{}{[\Gamma]\ \Phi, a : P, \Phi' \vdash a : P}\ (\mathsf{var}) \qquad \frac{[\Gamma]\ \Phi \vdash \alpha : P}{[\Gamma]\ A, \Phi \vdash \mathsf{wk}(\alpha) : P}\ (\mathsf{wk}) \qquad \frac{}{[\Gamma]\ \Phi \vdash ! : \top}\ (\top)$$

$$\frac{\Gamma \vdash F : \mathbb{C} \qquad [x : \mathbb{C}, \Gamma]\ \Phi(\overline{x}, x) \vdash \alpha : Q(\overline{x}, x)}{[\Gamma]\ \Phi(F^{\mathsf{op}}(\overline{x}), F(x)) \vdash F^{*}(\alpha) : Q(F^{\mathsf{op}}(\overline{x}), F(x))}\ (\mathsf{idx})$$

$$\frac{[\Gamma]\ \Phi \vdash P \times Q}{[\Gamma]\ \Phi \vdash P, \qquad [\Gamma]\ \Phi \vdash Q}\ (\mathsf{prod}) \qquad \frac{[x : \Gamma]\ A(\overline{x}, x), \Phi(\overline{x}, x) \vdash B(\overline{x}, x)}{[x : \Gamma] \qquad \Phi(\overline{x}, x) \vdash A^{\mathsf{op}}(x, \overline{x}) \Rightarrow B(\overline{x}, x)}\ (\mathsf{exp})$$

$$\frac{\begin{array}{r} [a : \Delta^{\mathsf{op}}, b : \Delta, x : \Gamma]\ \Phi(\overline{x}, x, a, b) \vdash \alpha \quad : P(a, b) \\ [z : \Delta, x : \Gamma]\ k : P(\overline{z}, z), \Phi(\overline{x}, x, \overline{z}, z) \vdash \gamma[k] : Q(\overline{z}, z) \end{array}}{[z : \Delta, x : \Gamma]\ \Phi(\overline{x}, x, \overline{z}, z) \vdash \gamma[\alpha] : Q(\overline{z}, z)}\ (\mathsf{cut\text{-}din})$$

$$\frac{\begin{array}{r} [z : \Delta, x : \Gamma]\ \Phi(\overline{x}, x, \overline{z}, z) \vdash \gamma \quad : P(\overline{z}, z) \\ [a : \Delta^{\mathsf{op}}, b : \Delta, x : \Gamma]\ k : P(a, b), \Phi(\overline{x}, x, \overline{a}, \overline{b}) \vdash \alpha[k] : Q(a, b) \end{array}}{[z : \Delta, x : \Gamma]\ \Phi(\overline{x}, x, \overline{z}, z) \vdash \alpha[\gamma] : Q(\overline{z}, z)}\ (\mathsf{cut\text{-}nat})$$

$$\frac{}{[x : \mathbb{C}, \Gamma]\ \Phi \vdash \mathsf{refl}_{\mathbb{C}} : \hom_{\mathbb{C}}(\overline{x}, x)}\ (\mathsf{refl}) \qquad \frac{[z : \mathbb{C}, \Gamma] \qquad \Phi(\overline{z}, z) \vdash h : P(\overline{z}, z)}{[a : \mathbb{C}^{\mathsf{op}}, b : \mathbb{C}, \Gamma]\ e : \hom_{\mathbb{C}}(a, b), \Phi(\overline{b}, \overline{a}) \vdash J(h)[e] : P(a, b)}\ (J)$$

$$\boxed{[\Gamma]\ \Phi \vdash \alpha = \beta : P} \qquad \frac{}{[z : \mathbb{C}, \Gamma]\ k : \Phi(\overline{z}, z) \vdash J(h)[\mathsf{refl}_{\mathbb{C}}] = h : P(\overline{z}, z)}\ (J\text{-}\mathsf{comp})$$

$$\frac{[z : \mathbb{C}, \Gamma]\ \Phi(\overline{z}, z) \vdash \alpha[\mathsf{refl}_{\mathbb{C}}] = \beta[\mathsf{refl}_{\mathbb{C}}] : P(\overline{z}, z)}{[a : \mathbb{C}^{\mathsf{op}}, b : \mathbb{C}, \Gamma]\ e : \hom_{\mathbb{C}}(a, b), \Phi(\overline{b}, \overline{a}) \vdash \alpha[e] = \beta[e] : P(a, b)}\ (J\text{-}\mathsf{eq})$$

Figure 2 Main rules for entailments of first-order dinatural directed type theory.

Polarized exponentials. Contrary to naturals and presheaves [38], dinaturals can be curried directly via the (exp) rule by currying each component of α in Set. This construction is similarly given in [24, 5] and called *twisted exponential*. The intuition is that dipresheaves move between the two sides of the turnstile by inverting the polarity of all their positions.
Reindexing with functors as terms. Following the doctrinal presentation of logic (see [34, 55] for standard accounts), dinaturals can be "reindexed" by functors via the rule (idx), i.e., variables in entailments can be substituted with concrete functors, viewed as terms.
Cut naturals-dinaturals. We present two restricted cut rules ($\mathsf{cut\text{-}din}$), ($\mathsf{cut\text{-}nat}$) between naturals and dinaturals, corresponding to Lemma 6. Associativity is captured in Figure 8. The occurrences $\overline{a}, \overline{b}$ in Φ in ($\mathsf{cut\text{-}nat}$) are needed to make α natural in a, b when the domain is *just* P, i.e., by using (exp) to move Φ and invert $\overline{a}, \overline{b}$. Note that P must not depend on Γ.
Directed equality introduction. The rule (refl) states reflexivity of directed equality, and is validated semantically by $\alpha_x(h) := \mathsf{id}_x$. Dinaturality holds by $\forall f : a \to b, f ; \mathsf{id}_b = \mathsf{id}_a ; f$.
Directed equality elimination. The rule (J) is the directed version of the J rule in Martin-Löf type theory. Note the syntactic restrictions: P cannot have contravariant occurrences $\overline{a}, \overline{b}$, and Φ cannot have covariant occurrences a, b. A computation rule (J-comp) holds, using ($\mathsf{cut\text{-}nat}$) on $J(h)$ to compose with (refl). Semantically, $J(h)$ is defined by $J(h)_{abx}[e, k] := ([\![\Phi]\!](\mathsf{id}_b, e, \mathsf{id}_x, \mathsf{id}_x) ; h_{bx} ; [\![P]\!](e, \mathsf{id}_b, \mathsf{id}_x, \mathsf{id}_x))[k]$. The computation rule clearly holds when $a = b = z$ and $e = \mathsf{id}_z$, without using dinaturality.

The operational meaning behind (J) is the following: having identified two covariant positions $a : \mathbb{C}^{\mathsf{op}}$ and $b : \mathbb{C}$ in the predicate P, if there is a directed equality $\hom_{\mathbb{C}}(a, b)$ in context then it is enough to prove that P holds "on the diagonal", where the two positions have been identified with the same dinatural variable $z : \mathbb{C}$; moreover, a, b can be identified in the context Φ *only if they appear contravariantly*, i.e., as $\overline{a}$ and $\overline{b}$. One can use (op) to equivalently state (J) to have $a, b : \mathbb{C}$ to have the same type: the formulation above with $a : \mathbb{C}^{\mathsf{op}}, b : \mathbb{C}$ emphasizes how the two variables play different asymmetric roles.

Dependent hom **elimination.** A *dependent* version of directed J, rule (J-eq), is needed in Section 2.4 to prove equational properties of maps definable with (J); this is done by allowing $\hom(a, b)$ to be contracted *inside equality judgements*. Intuitively, given $\alpha[e]$ and $\beta[e]$ with an equality in context $e : \hom_{\mathbb{C}}(a, b)$ which can be contracted using (J), α and β are equal everywhere as soon as they are equal on $e = \mathsf{refl}_{\mathbb{C},z}$ for all $z : \mathbb{C}$. This is validated in the model, crucially, using the fact that α, β are *dinatural*, as in the Yoneda lemma [38, 4.2].

▶ Remark 15 (Exponentials for naturals). The following derivation using (exp) elucidates why the exponential object in the category of presheaves and *naturals* is non-trivial, and is not the pointwise hom in Set; by directly applying (exp) for (co)presheaves $F, G, H : \mathbb{C} \to \mathsf{Set}$,

$$\frac{[x : \mathbb{C}] \;\; F(x) \times G(x) \vdash H(x)}{[x : \mathbb{C}] \qquad G(x) \vdash F^{\mathsf{op}}(\overline{x}) \Rightarrow H(x)} \;(\mathsf{exp})$$

but the bottom family of arrows is *dinatural* in x, since it appears both co- and contravariantly. We show in Theorem 32 how (exp) and the rules for directed equality can be used to give a logical proof that the usual definition of exponential for presheaves [38, 6.3.20] is correct.

▶ Remark 16 (Failure of symmetry for directed equality). The restrictions in (J) illustrate why one *cannot* derive that directed equality is symmetric, i.e., obtain a general map $[a : \mathbb{C}^{\mathsf{op}}, b : \mathbb{C}] \; e : \hom(a, b) \vdash \mathsf{sym} : \hom(\overline{b}, \overline{a})$. The equality $e : \hom(a, b)$ cannot be contracted because $\overline{a}$ appears in the conclusion negatively (similarly with $\overline{b}$), whereas (J) requires that the conclusion only has *positive* occurrences of the variables being contracted.

As in the symmetric case, (J) and hom can be characterized via an isomorphism [34, 3.2.3]:

▶ **Theorem 17** (Directed J as isomorphism). *Rule* (J) *is an isomorphism, and the inverse map is given by* $J^{-1}(h) := h[\mathsf{refl}_{\mathbb{C}}]$ *using* (cut-nat) *and* (refl)*. We refer to the whole isomorphism as the rule* (hom)*. Moreover,* $J^{-1} \,;\, J = \mathsf{id}$ *is logically equivalent to* (J-eq)*.*

Proof. The computation rule states precisely that $J ; J^{-1} = \mathsf{id}$. To show $J^{-1} ; J = \mathsf{id}$, we instantiate (J-eq) with $\alpha := J(\beta[\mathsf{refl}_{\mathbb{C}}])$ and use (J-comp) in the hypothesis, i.e., $J(\beta[\mathsf{refl}_{\mathbb{C}}])[\mathsf{refl}_{\mathbb{C}}] = \beta[\mathsf{refl}_{\mathbb{C}}]$, to obtain $J(\beta[\mathsf{refl}_{\mathbb{C}}]) = \beta$ as desired. We show that $J^{-1} ; J = \mathsf{id}$ implies (J-eq): the hypothesis in (J-eq) is exactly $J^{-1}(\alpha) = J^{-1}(\beta)$, hence $\alpha = \beta$ by applying J on both sides. ◀

▶ **Theorem 18** ($\hom \Rightarrow \mathsf{refl}$). *Rules* (refl) *and* (J) *are logically equivalent to* (hom).

Proof. Clearly (J) is the top-to-bottom direction of (hom). The rule (refl) follows from J^{-1} in Theorem 17 by picking $P := \hom$ and using the projection (var) as the bottom side map h. Semantically, the map obtained must coincide with refl: $J^{-1}(h) := h[\mathsf{refl}_{\mathbb{C}}]$, and since we picked h to be the projection π_1, we have that $h = \pi_1[\mathsf{refl}_{\mathbb{C}}, k] = \mathsf{refl}_{\mathbb{C}}$ as desired. ◀

▶ Remark 19 (Groupoidal case). When $\mathbb{C} \cong \mathbb{C}^{\mathsf{op}}$ is a groupoid, (hom) simply becomes the characterization of symmetric equality as left adjoint to reindexing on diagonals [34, 3.2.4].

Moreover, the non-compositionality of dinaturals is an intrinsic property of *directed* proof-relevant type theory, since in the groupoidal case they all compose (in the proof-irrelevant case, where Set is replaced by the preorder $\mathbf{I} := \{0 \to 1\}$, dinaturals compose trivially since there is no hexagon to check):

▶ **Theorem 20** (Dinaturals in groupoids). *Given a groupoid* $\mathbb{C} \cong \mathbb{C}^{\mathsf{op}}$ *and any* $\mathbb{D}$*, all dinaturals* $\alpha : F \xrightarrow{\bullet\bullet} G, \beta : G \xrightarrow{\bullet\bullet} H$ *for* $F, G, H : \mathbb{C}^{\diamond} \to \mathbb{D}$ *compose.*

2.4 Examples for directed equality

We show how the rules for directed equality can be used to obtain the same terms definable with symmetric equality in Martin-Löf type theory, and proving properties about them follows precisely the steps of the usual proofs [61, 31]. All examples in this section satisfy the constraints for (cut-nat) to be applied: this kind of composition essentially corresponds to vertical composition in Prof as a virtual equipment [46, 15]. We start by showing transitivity of directed equality, i.e., the fact that every category is equipped with composition maps.

▶ **Example 21** (Composition in a category). The following derivation constructs the *composition map for* $\mathbb{C}$, which is natural in $a : \mathbb{C}^{\mathsf{op}}, c : \mathbb{C}$ and dinatural in $b : \mathbb{C}$:

$$\dfrac{\dfrac{}{[z:\mathbb{C}, c:\mathbb{C}] \quad \hom(\overline{z}, c) \vdash \mathsf{id} : \hom(\overline{z}, c)}\,(\mathsf{var})}{[a:\mathbb{C}^{\mathsf{op}}, b:\mathbb{C}, c:\mathbb{C}] \ f : \hom(a,b),\ g:\hom(\overline{b}, c) \vdash J(\mathsf{id}) : \hom(a,c)}\,(J)$$

We contracted the first equality $f : \hom(a, b)$. Rule (J) can be applied since a, b appear only negatively in context (a does not appear) and positively in the conclusion ($\overline{b}$ does not). We now prove that $\mathsf{comp}[f, g] := J(\mathsf{id})$, denoted as "$f \,;\, g$", is unital on identities (i.e., $\mathsf{refl}_{\mathbb{C}}$) and associative. Since we chose to contract f, the computation rule ensures unitality on the left:

$$\dfrac{}{[z:\mathbb{C}, c:\mathbb{C}] \ g:\hom(\overline{z}, c) \vdash \mathsf{refl}_z \,;\, g = g : \hom(\overline{z}, c)}\,(J\text{-}\mathsf{comp})$$

essentially because $J^{-1}(J(\mathsf{id}) {=} \mathsf{comp}) {=} \mathsf{id}$. On the other hand, to show that composition is right-unital we must use directed equality induction for equalities (J-eq), where now it is enough to just consider the case in which $a = z = w$ and $f = \mathsf{refl}_w$,

$$\dfrac{\dfrac{}{[w:\mathbb{C}] \ \top \vdash \mathsf{refl}_w \,;\, \mathsf{refl}_w = \mathsf{refl}_w : \hom(\overline{w}, w)}\,(J\text{-}\mathsf{comp})}{[a:\mathbb{C}^{\mathsf{op}}, z:\mathbb{C}] \ f:\hom(a,z) \vdash f \,;\, \mathsf{refl}_z = f : \hom(a,z)}\,(J\text{-}\mathsf{eq})$$

which follows by the computation rule for comp since refl_w is on the left. Similarly, to show associativity we just need to consider the case $a = b = z$ and $f = \mathsf{refl}_z$,

$$\dfrac{\dfrac{}{[z:\mathbb{C}, c:\mathbb{C}, d:\mathbb{C}] \quad g:\hom(\overline{z}, c), h:\hom(\overline{c}, d) \vdash \mathsf{refl}_z \,;\, (g \,;\, h) = (\mathsf{refl}_z \,;\, g) \,;\, h : \hom(\overline{z}, d)}\,(J\text{-}\mathsf{comp})}{[a:\mathbb{C}, b:\mathbb{C}, c:\mathbb{C}, d:\mathbb{C}] \ f:\hom(\overline{a}, b), g:\hom(\overline{b}, c), h:\hom(\overline{c}, d) \vdash f \,;\, (g \,;\, h) = (f \,;\, g) \,;\, h : \hom(\overline{a}, d)}\,(J\text{-}\mathsf{eq})$$

where the top sequent equals $g \,;\, h$ by the computation rules for $\mathsf{comp} := J(\mathsf{id})$.

▶ **Example 22** (Functorial action on morphisms). For any functor $\mathbb{C} \vdash F : \mathbb{D}$, the functorial action on morphisms of F corresponds with the fact that any term/functor F respects directed equality, i.e., directed equality is a congruence:

$$\dfrac{\dfrac{}{[z:\mathbb{C}] \ \top \vdash F^*(\mathsf{refl}_{\mathbb{C}}) : \hom_{\mathbb{D}}(F^{\mathsf{op}}(\overline{z}), F(z))}\,(\mathsf{idx}){+}(\mathsf{refl})}{[x:\mathbb{C}, y:\mathbb{C}] \ \hom_{\mathbb{C}}(\overline{x}, y) \vdash J(F^*(\mathsf{refl}_{\mathbb{C}})) : \hom_{\mathbb{D}}(F^{\mathsf{op}}(\overline{x}), F(y))}\,(J)$$

and thus we define $\mathsf{map}_F[f] := J(F^*(\mathsf{refl}_{\mathbb{C}}))$, using (idx) with F in the top sequent.

The computation rule gives that F maps identities to identities:

$$\dfrac{}{[z:\mathbb{C}] \ \top \vdash \mathsf{map}_F[\mathsf{refl}_{\mathbb{C}}] = F^*(\mathsf{refl}_{\mathbb{C}}) : \hom_{\mathbb{D}}(F^{\mathsf{op}}(\overline{x}), F(x))}\,(J\text{-}\mathsf{comp})$$

Functoriality holds, since both top sides equal $\mathsf{map}_F[g]$ using computation rules:

$$\dfrac{\dfrac{}{[z:\mathbb{C}, c:\mathbb{C}] \quad g:\hom(\overline{z}, c) \vdash \mathsf{map}_F[\mathsf{refl}_z \,;\, g] = \mathsf{refl}_{F(z)} \,;\, \mathsf{map}_F[g] : \hom(\overline{z}, d)}\,(J\text{-}\mathsf{comp})}{[a:\mathbb{C}, b:\mathbb{C}, c:\mathbb{C}] \ f:\hom(\overline{a}, b), g:\hom(\overline{b}, c) \vdash \mathsf{map}_F[f \,;\, g] = \mathsf{map}_F[f] \,;\, \mathsf{map}_F[g] : \hom(\overline{a}, d)}\,(J\text{-}\mathsf{eq})$$

▶ **Example 23** (Transport)**.** Transporting points of predicates along directed equalities [61, 2.3.1] is the functorial action of copresheaves $P : \mathbb{C} \to \mathsf{Set}$, i.e., $[x : \mathbb{C}]\ P\ \mathsf{prop}$, for x only positive:

$$\dfrac{\dfrac{}{[z : \mathbb{C}]\ P(z) \vdash P(z)}\ (\mathsf{var})}{[a : \mathbb{C}^{\mathsf{op}}, b : \mathbb{C}]\ \ \hom(a, b), P(\overline{a}) \vdash P(b)}\ (J) \qquad \dfrac{}{[z : \mathbb{C}]\ k : P(z) \vdash \mathsf{subst}[\mathsf{refl}_{\mathbb{C}}, k] = k : P(z)}\ (J\text{-}\mathsf{comp})$$

The computation rule simply states that transporting a point of $P(a)$ along the identity morphism with $\mathsf{subst}[f, k] := J(\mathsf{id})$ is the same as giving the point itself, i.e., $\mathsf{subst}[\mathsf{refl}_{\mathbb{C}}] = \mathsf{id}$.

▶ **Example 24** (Internal dinaturality for entailments)**.** For any $[x : \mathbb{C}]\ P(\overline{x}, x) \vdash \alpha : Q(\overline{x}, x)$, an internal version of (di)naturality for entailments, as in Theorem 3, holds via (J-comp):

$$\dfrac{\dfrac{}{\begin{array}{r}[z : \mathbb{C}]\ k : P(\overline{z}, z) \vdash \mathsf{subst}_Q[(\mathsf{refl}, \mathsf{refl}), [\alpha[\mathsf{subst}_P[(\mathsf{refl}, \mathsf{refl}), k]]]] \\ = \mathsf{subst}_Q[(\mathsf{refl}, \mathsf{refl}), [\alpha[\mathsf{subst}_P[(\mathsf{refl}, \mathsf{refl}), k]]]] : Q(\overline{z}, z)\end{array}}\ (J\text{-}\mathsf{comp})}{\begin{array}{r}[a : \mathbb{C}^{\mathsf{op}}, b : \mathbb{C}]\ f : \hom_{\mathbb{C}}(a, b), k : P(\overline{b}, \overline{a}) \vdash \mathsf{subst}_Q[(\mathsf{refl}, f), [\alpha[\mathsf{subst}_P[(f, \mathsf{refl}), k]]]] \\ = \mathsf{subst}_Q[(f, \mathsf{refl}), [\alpha[\mathsf{subst}_P[(\mathsf{refl}, f), k]]]] : Q(a, b)\end{array}}\ (J\text{-}\mathsf{eq})$$

3 (Co)ends as quantifiers

In this section we describe how dinaturality allows us to give an interpretation of ends and coends as the "directed quantifiers" for the (1-)category interpretation of directed type theory. We provide rules for ends and coends which are reminiscent of the quantifiers-as-adjoints paradigm by Lawvere [36]; we show in Theorem 27 that ends and coends can be captured as "right and left adjoint" operations to a common weakening operation which only operates on the context of entailments [34, 1.9.1]: this adjointness relation should be only interpreted suggestively, since (co)ends are functorial operations for naturals but in general not dinaturals [40, 1.1.7]. Despite this, the rules for (co)ends can be combined with the ones given for directed equality (hom) to provide concise proofs of theorems in category theory using sequences of natural isomorphisms, following an approach similar to that described as "(co)end calculus" [40] but emphasizing the role of (co)ends as quantifiers and hom as directed equality. This approach has the advantage that several properties of quantifiers, e.g., that they can be exchanged and permuted whenever possible, follow automatically from certain structural properties of contexts, which we omit here since they are standard. For example, in first order logic the formulas $\forall x.\forall y.P \Leftrightarrow \forall y.\forall x.P \Leftrightarrow \forall(x, y).P$ are logically equivalent for any predicate $P(x, y)$: this is indeed also verified for ends (and coends with existentials) and takes the name of "Fubini rule" [41, IX.8], [40, 1.3.1], which we prove in Theorem 34. More details on (co)ends and their calculus can be found in [41, IX.5-6], [40, Ch. 1].

▶ **Definition 25** ((Co)wedges for P [40, 1.1.4])**.** *Given $P : \mathbb{C}^{\mathsf{op}} \times \mathbb{C} \to \mathbb{D}$, a* wedge *for P is a pair object/dinatural $(X : \mathbb{D}, \alpha : K_X \xrightarrow{\bullet\bullet} P)$, where K_X is the constant functor in X. A* wedge morphism *$(X, \alpha) \to (Y, \alpha')$ is an $f : X \to Y$ of $\mathbb{D}$ such that $\forall c : \mathbb{C}, \alpha_c = f \,;\, \alpha'_c$. A* cowedge *is a wedge in $\mathbb{D}^{\mathsf{op}}$, denoting the categories of (co)wedges as $\mathsf{Wedge}(P)$, $\mathsf{Cowedge}(P)$.*

▶ **Definition 26** ((Co)ends [40, 1.1.6])**.** *Given a functor $P : \mathbb{C}^{\mathsf{op}} \times \mathbb{C} \to \mathbb{D}$, the* end *of P is defined to be the terminal object of $\mathsf{Wedge}(P)$, whose object in $\mathbb{D}$ is denoted as $\int_{x:\mathbb{C}} P(\overline{x}, x)$. Dually, the* coend *of P is the initial object of $\mathsf{Cowedge}(P)$, denoted similarly as $\int^{x:\mathbb{D}} P(\overline{x}, x)$. The integral symbol acts as a binder, in the sense that "$\int_{c:\mathbb{C}} P(c, c)$" and "$\int_{x:\mathbb{C}} P(x, x)$" are ($\alpha$-)equivalent; moreover, P can depend on many parameters, e.g., if $P : (\mathbb{A}^{\mathsf{op}} \times \mathbb{A}) \times (\mathbb{B}^{\mathsf{op}} \times \mathbb{B}) \to \mathbb{D}$ then $\int_{b:\mathbb{B}} P(\overline{a}, a, \overline{b}, b) : \mathbb{A}^{\mathsf{op}} \times \mathbb{A} \to \mathbb{D}$. (Co)ends exist when $\mathbb{D}$ is (co)complete [40].*

▶ **Theorem 27** (Ends and coends as quantifiers). *The following rules are validated by isomorphisms of sets of dinaturals,* natural *in* $P : \mathbb{C}^\diamond \to \mathsf{Set}$, $Q : \mathbb{A}^\diamond \times \mathbb{C}^\diamond \to \mathsf{Set}$*, where* $\mathbb{C} := [\![\Gamma]\!]$*:*

$$\frac{[a : \mathbb{A}, \Gamma]\ \Phi \vdash Q(\overline{a}, a)}{[\Gamma]\ \Phi \vdash \int_{a:\mathbb{A}} Q(\overline{a}, a)}\ (\mathsf{end}) \qquad \frac{[\Gamma]\ \left(\int^{a:\mathbb{A}} Q(\overline{a}, a)\right), \Phi \vdash P}{[a : \mathbb{A}, \Gamma]\ Q(\overline{a}, a), \Phi \vdash P}\ (\mathsf{coend{+}frob.})$$

Theorem 27 expresses an adjoint-like (up to the non-composition of dinaturals) correspondence $\int^{\mathbb{A}[\mathbb{C}]} \dashv \pi^*_{\mathbb{A}[\mathbb{C}]} \dashv \int_{\mathbb{A}[\mathbb{C}]}$ between the weakening functor $\pi^*_{\mathbb{A}[\mathbb{C}]} : [\mathbb{C}^\diamond, \mathsf{Set}] \to [\mathbb{A}^\diamond \times \mathbb{C}^\diamond, \mathsf{Set}]$ and the functors $\int^{\mathbb{A}[\mathbb{C}]}, \int_{\mathbb{A}[\mathbb{C}]} : [\mathbb{A}^\diamond \times \mathbb{C}^\diamond, \mathsf{Set}] \to [\mathbb{C}^\diamond, \mathsf{Set}]$ sending dipresheaves to their (co)end in A.

We leave the weakening operations of the above rules implicit for the rest of the paper. Quantifiers in categorical logic typically have to satisfy additional requirements in order to faithfully model logical operations: the Beck-Chevalley condition [34, 1.9.4] states that "quantifiers commute with substitution", and the Frobenius condition [34, 1.9.12] logically corresponds to having an additional context Φ in rules for colimit-like connectives [34, 3.4.4]. The rule given in Theorem 27 is already stated with such additional context.

▶ **Theorem 28** (Beck-Chevalley and Frobenius condition for (co)ends). *(Co)ends satisfy a* Beck-Chevalley condition*, in the sense that for all* $F : \mathbb{C}^\diamond \to \mathbb{D}$ *there is a strict isomorphism* $\int_{\mathbb{A}[\mathbb{D}]} ; F^* \cong (\mathsf{id}_{\mathbb{A}^\diamond} \times F)^* ; \int_{\mathbb{A}[\mathbb{C}]}$ *in the (large) functor category* $[[\mathbb{A}^\diamond \times \mathbb{D}^\diamond, \mathsf{Set}], [\mathbb{D}^\diamond, \mathsf{Set}]]$*, where* $\int_{\mathbb{A}[\mathbb{C}]}, \int^{\mathbb{A}[\mathbb{C}]} : [\mathbb{A}^\diamond \times \mathbb{C}^\diamond, \mathsf{Set}] \to [\mathbb{C}^\diamond, \mathsf{Set}]$ *are the functors sending dipresheaves to their (co)end in* $\mathbb{A}$ *and* $F^* : [\mathbb{D}^\diamond, \mathsf{Set}] \to [\mathbb{C}^\diamond, \mathsf{Set}]$ *is precomposition with* $F^\diamond$*.*

Moreover, a Frobenius condition *for coends is satisfied, in the sense that there is an isomorphism* $\int^{\mathbb{A}[\mathbb{C}]}(\pi^*_{\mathbb{A}[\mathbb{C}]}(P) \wedge \Gamma) \cong \pi^*_{\mathbb{A}[\mathbb{C}]}(P) \wedge \int^{\mathbb{A}[\mathbb{C}]}(\Gamma)$, natural *in* $\Gamma : \mathbb{A}^\diamond \times \mathbb{C}^\diamond \to \mathsf{Set}$, $P : \mathbb{C}^\diamond \to \mathsf{Set}$*, where* $- \wedge - : [\mathbb{C}, \mathsf{Set}] \times [\mathbb{C}, \mathsf{Set}] \to [\mathbb{C}, \mathsf{Set}]$ *for any* $\mathbb{C}$ *is the product of (di)presheaves.*

Proof. Beck-Chevalley is immediate. For Frobenius, our logical rules can be used to apply exactly the argument given in [34, 1.9.12(i)], detailed in Appendix C. ◀

3.1 Coend calculus via dinaturality

We show how the rules for directed equality and (co)ends can be used to give concise proofs with a distinctly logical flavour to several central theorems of category theory. The technique we use mirrors the way coend calculus is applied in practical settings (e.g., [11, 30, 57]) via a "Yoneda-like" series of *natural* isomorphisms of sets: to prove that two objects $A, B : \mathbb{C}$ are isomorphic, one can assume to have a generic object Φ and then apply a series of isomorphisms of sets *natural* in Φ to establish that $\mathbb{C}(\Phi, A) \cong \mathbb{C}(\Phi, B)$, from which $A \cong B$ follows the fully faithfulness of the Yoneda embedding [11, 38]. The same technique can be used to show that *functors* are naturally isomorphic, as well as adjunctions, e.g., Theorems 32 and 33. Our proofs follow a different approach to that taken in [40, 17], since we use sets of *dinaturals* and explicitly view $\hom(a, b)$ in terms of directed equality and (co)ends via their characterization in terms of contextual operations, rather than applying arbitrary semantic rules. We now show our main examples, with additional derivations in Appendix B.

▶ Remark 29 (Internalizing Yoneda). The above technique is not captured as part of the type theory, and is just a semantic result in the *meta*-theory; our logic makes this easier since it constructs the required natural isos. This could be internalized in the type theory by adding a universe Set and a univalence-like statement $\hom_{\mathsf{Set}}(A, B) \cong A \Rightarrow B$ (as in [2, 26]), which allows for implication to be represented as a directed equality and then contracted with (J); then, using a similar argument as in Theorem 24, one can interalize naturality and the standard proof of the Yoneda lemma, e.g., [38]. We show internal naturality in Theorem 42.

▶ Remark 30 (Naturality of rules). () All rules given in previous sections are *natural* in each of the dipresheaves involved. In the following series of examples no proof ever involves a "dinatural isomorphism", since Yoneda cannot be applied between sets which are not hom-sets; natural isomorphisms between sets of *di*naturals are only used as intermediate steps. This technique allows us to sidestep the issue of the compositionality of dinaturals, and motivates our approach omitting the underlying equational theory.

▶ **Example 31** ((co)Yoneda lemma)**.** For any presheaf $P : \mathbb{C} \to \mathsf{Set}$, and a presheaf $\Phi : \mathbb{C} \to \mathsf{Set}$ acting as generic context, the following derivations capture the Yoneda lemma [40, Thm. 1] (using the characterization of naturals as an end) and coYoneda lemma [41, III.7, Theorem 1] (i.e., presheaves are isomorphic to a weighted colimit of representables), respectively.

$$
\begin{array}{cl}
[a{:}\mathbb{C}]\ \Phi(a) \vdash \int_{x:\mathbb{C}} \hom^{\mathsf{op}}_{\mathbb{C}}(a,\overline{x}) \Rightarrow P(x) & \\
\hline\hline
[a{:}\mathbb{C}, x{:}\mathbb{C}]\ \Phi(a) \vdash \hom^{\mathsf{op}}_{\mathbb{C}}(a,\overline{x}) \Rightarrow P(x) & (\mathsf{end}) \\
\hline\hline
[a{:}\mathbb{C}, x{:}\mathbb{C}]\ \ \hom_{\mathbb{C}}(\overline{a},x) \times \Phi(a) \vdash P(x) & (\mathsf{exp}) \\
\hline\hline
[z:\mathbb{C}]\ \Phi(z) \vdash P(z) & (\mathrm{hom})
\end{array}
\qquad
\begin{array}{cl}
[a{:}\mathbb{C}]\ \int^{x:\mathbb{C}} \hom_{\mathbb{C}}(\overline{x},a) \times P(x) \vdash \Phi(a) & \\
\hline\hline
[a{:}\mathbb{C}, x{:}\mathbb{C}]\ \ \hom_{\mathbb{C}}(\overline{a},x) \times P(a) \vdash \Phi(x) & (\mathsf{coend}) \\
\hline\hline
[z:\mathbb{C}]\ P(z) \vdash \Phi(z) & (\mathrm{hom})
\end{array}
$$

▶ **Example 32** (Presheaves are cartesian closed)**.** For any $A, B, \Phi : \mathbb{C} \to \mathsf{Set}$, the following derivation expresses that the internal hom in the category of presheaves and naturals defined by $(A \Rightarrow B)(x) := \mathsf{Nat}(\hom(x,-) \times A, B)$ is indeed the correct one, shown here via the isomorphism of sets of the tensor/hom adjunction:

$$
\begin{array}{cl}
[x:\mathbb{C}]\ \Phi(x) \vdash (A \Rightarrow B)(x) := \mathsf{Nat}(\hom_{\mathbb{C}}(x,-) \times A, B) & \\
\qquad = \int_{y:\mathbb{C}} \hom^{\mathsf{op}}_{\mathbb{C}}(x,\overline{y}) \times A^{\mathsf{op}}(\overline{y}) \Rightarrow B(y) & \\
\hline\hline
[x:\mathbb{C}, y:\mathbb{C}]\ \Phi(x) \vdash \hom^{\mathsf{op}}_{\mathbb{C}}(x,\overline{y}) \times A^{\mathsf{op}}(\overline{y}) \Rightarrow B(y) & (\mathsf{end}) \\
\hline\hline
[x:\mathbb{C}, y:\mathbb{C}]\ A(y) \times \hom_{\mathbb{C}}(\overline{x},y) \times \Phi(x) \vdash B(y) & (\mathsf{exp}) \\
\hline\hline
[y:\mathbb{C}]\ A(y) \times \left(\int^{x:\mathbb{C}} \hom_{\mathbb{C}}(\overline{x},y) \times \Phi(x)\right) \vdash B(y) & (\mathsf{coend{+}frob.}) \\
\hline\hline
[y:\mathbb{C}]\ A(y) \times \Phi(y) \vdash B(y) & (\mathsf{coYoneda})
\end{array}
$$

Note that (hom) cannot be used in this derivation since y appears positively in context in $A(y)$, whereas it should be negative to identify it with x. Instead, we apply the rule (coYoneda) given in Example 31 to show that copresheaf $\Phi : C \to \mathsf{Set}$ *as a whole* is isomorphic to a certain functor $C \to \mathsf{Set}$ computed as a coend, independently of the point on which it is evaluated (in this case y). The above derivation is a simple application of our rules via dinaturality, but it is unclear how it can be captured using the proarrow equipment approach of [46, 66] as an abstract property of Prof, due to the repetition of the variables $y, \overline{y}$.

▶ **Example 33** (Pointwise formula for right Kan extensions)**.** Using our rules, we give a logical proof that the functor $\mathsf{Ran}_F : [\mathbb{C}, \mathsf{Set}] \to [\mathbb{D}, \mathsf{Set}]$ sending (co)presheaves to their Kan extensions along $F : \mathbb{C} \to \mathbb{D}$ computed via ends [40, 2.3.6] is right adjoint to precomposition $(F\ ;\ -) : [\mathbb{D}, \mathsf{Set}] \to [\mathbb{C}, \mathsf{Set}]$. Note the similarity between this derivation and the argument given in [55, 5.6.6] to compute adjoints in a general doctrine. For any $P : \mathbb{C} \to \mathsf{Set}$, a functor/term $F : \mathbb{C} \to \mathbb{D}$ and a generic $\Phi : \mathbb{D} \to \mathsf{Set}$:

$$
\begin{array}{cl}
[y:\mathbb{D}]\ \Phi(y) \vdash (\mathsf{Ran}_F P)(y) := \int_{x:\mathbb{C}} \hom^{\mathsf{op}}_{\mathbb{D}}(y, F^{\mathsf{op}}(\overline{x})) \Rightarrow P(x) & \\
\hline\hline
[x:\mathbb{C}, y:\mathbb{D}]\ \Phi(y) \vdash \hom^{\mathsf{op}}_{\mathbb{D}}(y, F^{\mathsf{op}}(\overline{x})) \Rightarrow P(x) & (\mathsf{end}) \\
\hline\hline
[x:\mathbb{C}, y:\mathbb{D}]\ \ \hom_{\mathbb{D}}(\overline{y}, F(x)) \times \Phi(y) \vdash P(x) & (\mathsf{exp}) \\
\hline\hline
[x:\mathbb{C}]\ \int^{y:\mathbb{D}} \hom_{\mathbb{D}}(\overline{y}, F(x)) \times \Phi(y) \vdash P(x) & (\mathsf{coend}) \\
\hline\hline
[y:\mathbb{C}]\ \Phi(F(x)) \vdash P(x) & (\mathsf{coYoneda})
\end{array}
$$

▶ **Example 34** (Fubini rule for ends)**.** For convenience we only show the case for ends. For any dipresheaf $\Phi : \top^{\mathsf{op}} \times \top \to \mathsf{Set}$ (a dipresheaf in the empty context, i.e., simply an object $\Phi : \mathsf{Set}$) and $P : (\mathbb{C}^{\mathsf{op}} \times \mathbb{C}) \times (\mathbb{D}^{\mathsf{op}} \times \mathbb{D}) \to \mathsf{Set}$ the following are all equivalent,

$$
\begin{array}{ll}
[\,]\ \Phi \vdash \int_{x:\mathbb{C}} \int_{y:\mathbb{D}} P(\overline{x}, x, \overline{y}, y) & \\
\hline\hline
[x : \mathbb{C}]\ \Phi \vdash \int_{y:\mathbb{D}} P(\overline{x}, x, \overline{y}, y) & (\mathsf{end}) \\
\hline\hline
[x : \mathbb{C}, y : \mathbb{D}]\ \Phi \vdash P(\overline{x}, x, \overline{y}, y) & (\mathsf{end}) \\
\hline\hline
[y : \mathbb{D}, x : \mathbb{C}]\ \Phi \vdash P(\overline{x}, x, \overline{y}, y) & (\mathsf{structural\ property})
\end{array}
\qquad
\begin{array}{ll}
\dots & \\
\hline\hline
[p : \mathbb{C} \times \mathbb{D}]\ \Phi \vdash P(\overline{p}, p) & (\mathsf{structural\ property}) \\
\hline\hline
[y : \mathbb{D}]\ \Phi \vdash \int_{x:\mathbb{C}} P(\overline{x}, x, \overline{y}, y) & (\mathsf{end}) \\
\hline\hline
[\,]\ \Phi \vdash \int_{y:\mathbb{D}} \int_{x:\mathbb{C}} P(\overline{x}, x, \overline{y}, y) & (\mathsf{end}) \\
\hline\hline
[\,]\ \Phi \vdash \int_{p:\mathbb{C}\times\mathbb{D}} P(\overline{x}, x, \overline{y}, y) & (\mathsf{end})
\end{array}
$$

using the fact that certain structural properties of contexts hold by cartesianness of Cat.

4 Conclusions and future work

In this paper we showed how dinaturality plays a crucial role in the semantics of a directed type theory where types are interpreted as (1-)categories and directed equality as hom-functors, which we then applied to give a distinctly logical interpretation to (co)end calculus by viewing it in terms of directed equality.

Our treatment of directed equality is a first step towards a formal understanding of the role played by directedness and variance, and their relationship with the syntax of standard Martin-Löf type theory. The most important aspect left out from our work is a precise treatment of the compositionality of dinatural transformations, a famously difficult problem [58] which would, as we argued, give insight on a satisfactory syntactic treatment of (dependent) directed type theory with 1-categories. Strong dinaturals [45, 49] provide a hint in this direction but lack in expressivity, e.g., they are not closed in general [62]. Following Theorem 19, this non-compositionality seems to be an intrinsic characteristic of interpreting dinaturals in the directed proof-relevant setting, i.e., non-groupoidal categories.

Organizing the syntax into a suitable initial object in a category of models requires a more detailed (and possibly more general) notion of model that can axiomatize the behaviour of variables in dinaturals and naturals (e.g., as in [58]): one possible approach could be to abstractly consider two classes of maps (dinaturals, naturals) and requiring such maps to interact as in (cut-nat), (cut-din) and (assoc-nat-din-nat). This would be particularly important towards a fully *dependent* treatment of dinatural directed type theory, in order to investigate how polarity of variables is influenced by their appearance inside types.

All of our results can be specialized in the category of posets Pos rather than Cat, where dinaturals compose trivially and our work provides a "logic of posets", captured via a bona fide doctrine, at the cost of trivializing (co)ends with (co)products. This case could be axiomatized in the style of the doctrinal approach [34, 43], by introducing a notion of *directed doctrine* that captures the roles played by variance, the $-^{\mathsf{op}}$ involution, and (di)naturality.

Aside from adding a Set-like universe to the type theory as mentioned in Remark 29, there are useful examples of (higher) coend calculus which have not yet been interpreted in terms of directed equality: for instance, one should be able to express that composition maps exist *for all* categories $\mathbb{C} : \mathsf{Cat}$, where this quantification can be expressed via a suitable pseudo-end in Cat [40, 7.1]; similarly, the category of elements of a functor, reminiscent of a Σ-type, can be given as the pseudo-coend $\mathsf{El}(F) \cong \int^{c:\mathbb{C}} c/\mathbb{C} \times F(c)$, where $c/\mathbb{C}$ is the coslice category and $F(c)$ is seen as a discrete category [40, 4.2.2]. These examples could be captured by considering the category of small categories Cat as a suitable universe of types [31]. We leave investigating the relation between dinaturality and geometric models of ω-categories in the spirit of [56, 26, 65] for future work.

References

1 Ahrens, B., North, P.R., van der Weide, N.: Bicategorical type theory: semantics and syntax. Mathematical Structures in Computer Science pp. 1–45 (Oct 2023). https://doi.org/10.1017/S0960129523000312, https://www.cambridge.org/core/journals/mathematical-structures-in-computer-science/article/bicategorical-type-theory-semantics-and-syntax/725F2E17B25094145F9D037D9465A534

2 Altenkirch, T., Neumann, J.: Synthetic 1-Categories in Directed Type Theory (Oct 2024). https://doi.org/10.48550/arXiv.2410.19520

3 Asada, K.: Arrows are strong monads. In: Proceedings of the third ACM SIGPLAN workshop on Mathematically structured functional programming. pp. 33–42. MSFP '10, Association for Computing Machinery, New York, NY, USA (Sep 2010). https://doi.org/10.1145/1863597.1863607

4 Asada, K., Hasuo, I.: Categorifying Computations into Components via Arrows as Profunctors. Electronic Notes in Theoretical Computer Science **264**(2), 25–45 (Aug 2010). https://doi.org/10.1016/j.entcs.2010.07.012, https://www.sciencedirect.com/science/article/pii/S157106611000071X

5 Bainbridge, E.S., Freyd, P.J., Scedrov, A., Scott, P.J.: Functorial polymorphism. Theoretical Computer Science **70**(1), 35–64 (Jan 1990). https://doi.org/10.1016/0304-3975(90)90151-7, https://www.sciencedirect.com/science/article/pii/0304397590901517

6 van den Berg, B., Garner, R.: Types are weak ω-groupoids. Proceedings of the London Mathematical Society **102**(2), 370–394 (Oct 2010). https://doi.org/10.1112/plms/pdq026

7 Blute, R.F., Scott, P.J.: Linear Läuchli semantics. Annals of Pure and Applied Logic **77**(2), 101–142 (Jan 1996). https://doi.org/10.1016/0168-0072(95)00017-8

8 Blute, R.F., Scott, P.J.: The Shuffle Hopf Algebra and Noncommutative Full Completeness. Journal of Symbolic Logic **63**(4), 1413–1436 (1998). https://doi.org/10.2307/2586659

9 Blute, R.: Linear logic, coherence and dinaturality. Theoretical Computer Science **115**(1), 3–41 (Jul 1993). https://doi.org/10.1016/0304-3975(93)90053-V, https://www.sciencedirect.com/science/article/pii/030439759390053V

10 Boisseau, G.: String Diagrams for Optics. In: Ariola, Z.M. (ed.) 5th International Conference on Formal Structures for Computation and Deduction (FSCD 2020). Leibniz International Proceedings in Informatics (LIPIcs), vol. 167, pp. 17:1–17:18. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, Dagstuhl, Germany (2020). https://doi.org/10.4230/LIPIcs.FSCD.2020.17, https://drops.dagstuhl.de/opus/volltexte/2020/12339

11 Boisseau, G., Gibbons, J.: What you needa know about Yoneda: profunctor optics and the Yoneda lemma (functional pearl). Proceedings of the ACM on Programming Languages **2**(ICFP), 84:1–84:27 (Jul 2018). https://doi.org/10.1145/3236779

12 Chu, F., Mangel, E., North, P.R.: A directed type theory for 1-categories. In: 30th International Conference on Types for Proofs and Programs TYPES 2024–Abstracts. p. 205 (2024), https://types2024.itu.dk/abstracts.pdf#page=215

13 Clarke, B., Elkins, D., Gibbons, J., Loregian, F., Milewski, B., Pillmore, E., Román, M.: Profunctor optics, a categorical update (2022). https://doi.org/10.48550/arxiv.2001.07488

14 Cohen, C., Coquand, T., Huber, S., Mörtberg, A.: Cubical Type Theory: A Constructive Interpretation of the Univalence Axiom. In: Uustalu, T. (ed.) 21st International Conference on Types for Proofs and Programs (TYPES 2015). Leibniz International Proceedings in Informatics (LIPIcs), vol. 69, pp. 5:1–5:34. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany (2015). https://doi.org/10.4230/LIPIcs.TYPES.2015.5

15 Cruttwell, G., Shulman, M.: A unified framework for generalized multicategories. Theory Appl. Categ. **24**, 580–655 (2010). https://doi.org/10.48550/arxiv.0907.2460

16 Curien, P.L.: Operads, Clones, and Distributive Laws, pp. 25–49 (Feb 2012). https://doi.org/10.1142/9789814365123_0002

17 Cáccamo, M., Winskel, G.: A Higher-Order Calculus for Categories. In: Boulton, R.J., Jackson, P.B. (eds.) Theorem Proving in Higher Order Logics. pp. 136–153. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg (2001). https://doi.org/10.1007/3-540-44755-5_11

18 Dubuc, E., Street, R.: Dinatural transformations. In: MacLane, S., Applegate, H., Barr, M., Day, B., Dubuc, E., Phreilambud, Pultr, A., Street, R., Tierney, M., Swierczkowski, S. (eds.) Reports of the Midwest Category Seminar IV. pp. 126–137. Lecture Notes in Mathematics, Springer, Berlin, Heidelberg (1970). https://doi.org/10.1007/BFb0060443

19 Eilenberg, S., Kelly, G.M.: A generalization of the functorial calculus. Journal of Algebra **3**(3), 366–375 (May 1966). https://doi.org/10.1016/0021-8693(66)90006-8, https://www.sciencedirect.com/science/article/pii/0021869366900068

20 Escardó, M.: Using yoneda rather than j to present the identity type. https://www.cs.bham.ac.uk/~mhe/yoneda/yoneda.html (2015), updated 2015, 2016, 2017

21 Freyd, P.J., Robinson, E.P., Rosolini, G.: Dinaturality for free. In: Pitts, A.M., Fourman, M.P., Johnstone, P.T. (eds.) Applications of Categories in Computer Science: Proceedings of the London Mathematical Society Symposium, Durham 1991, pp. 107–118. London Mathematical Society Lecture Note Series, Cambridge University Press, Cambridge (1992). https://doi.org/10.1017/CBO9780511525902.007, https://www.cambridge.org/core/books/applications-of-categories-in-computer-science/dinaturality-for-free/E6E7F60E4A647D401B901BB936104359

22 Freyd, P.J., Robinson, E.P., Rosolini, G.: Functorial parametricity. In: Proceedings of the Seventh Annual Symposium on Logic in Computer Science (LICS '92), Santa Cruz, California, USA, June 22-25, 1992. pp. 444–452. IEEE Computer Society (1992). https://doi.org/10.1109/LICS.1992.185555

23 Galal, Z.: A Profunctorial Scott Semantics. In: Ariola, Z.M. (ed.) 5th International Conference on Formal Structures for Computation and Deduction (FSCD 2020). Leibniz International Proceedings in Informatics (LIPIcs), vol. 167, pp. 16:1–16:18. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, Dagstuhl, Germany (2020). https://doi.org/10.4230/LIPIcs.FSCD.2020.16, https://drops.dagstuhl.de/opus/volltexte/2020/12338

24 Girard, J.Y., Scedrov, A., Scott, P.J.: Normal Forms and Cut-Free Proofs as Natural Transformations. In: Moschovakis, Y.N. (ed.) Logic from Computer Science. pp. 217–241. Springer, New York, NY (1992). https://doi.org/10.1007/978-1-4612-2822-6_8

25 Gratzer, D., Weinberger, J., Buchholtz, U.: The yoneda embedding in simplicial type theory, https://arxiv.org/abs/2501.13229

26 Gratzer, D., Weinberger, J., Buchholtz, U.: Directed univalence in simplicial homotopy type theory (2024). https://doi.org/10.48550/arXiv.2407.09146, arXiv.2407.09146

27 Hefford, J., Comfort, C.: Coend Optics for Quantum Combs. Electronic Proceedings in Theoretical Computer Science **380**, 63–76 (Aug 2023). https://doi.org/10.4204/EPTCS.380.4

28 Hermida, C., Mateus, P.: Paracategories i: internal paracategories and saturated partial algebras **309**(1–3), 125–156. https://doi.org/10.1016/s0304-3975(03)00135-x

29 Hermida, C., Mateus, P.: Paracategories ii: adjunctions, fibrations and examples from probabilistic automata theory **311**(1–3), 71–103. https://doi.org/10.1016/s0304-3975(03)00317-7

30 Hinze, R.: Kan Extensions for Program Optimisation Or: Art and Dan Explain an Old Trick. In: Gibbons, J., Nogueira, P. (eds.) Mathematics of Program Construction. pp. 324–362. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg (2012). https://doi.org/10.1007/978-3-642-31113-0_16

31 Hofmann, M.: Syntax and Semantics of Dependent Types. In: Pitts, A.M., Dybjer, P., Pitts, A.M., Dybjer, P. (eds.) Semantics and Logics of Computation, pp. 79–130. Publications of the Newton Institute, Cambridge University Press, Cambridge (1997). https://doi.org/10.1017/CBO9780511526619.004, https://www.cambridge.org/core/books/semantics-and-logics-of-computation/syntax-and-semantics-of-dependent-types/119C8085C6A1A0CD7F24928EF866748F

32 Hofmann, M., Streicher, T.: The groupoid interpretation of type theory. In: Sambin, G., Smith, J.M. (eds.) Twenty-five years of constructive type theory (Venice, 1995), Oxford Logic Guides, vol. 36, pp. 83–111. Oxford Univ. Press, New York, New York (Oct 1998). https://doi.org/10.1093/oso/9780198501275.003.0008

33 Hyland, J.M.E.: Elements of a theory of algebraic theories **546**, 132–144. https://doi.org/10.1016/j.tcs.2014.03.005, models of Interaction: Essays in Honour of Glynn Winskel

34 Jacobs, B.P.F.: Categorical Logic and Type Theory, Studies in Logic and the Foundations of Mathematics, vol. 141. North-Holland (1999)

35 Kudasov, N., Riehl, E., Weinberger, J.: Formalizing the ∞-categorical yoneda lemma. In: Proceedings of the 13th ACM SIGPLAN International Conference on Certified Programs and Proofs. pp. 274–290. CPP 2024, Association for Computing Machinery, New York, NY, USA (2024). https://doi.org/10.1145/3636501.3636945

36 Lawvere, F.W.: Adjointness in Foundations. Dialectica **23**(3/4), 281–296 (1969), `https://www.jstor.org/stable/42969800`

37 Leinster, T.: Coend computation. MathOverflow (2010), `https://mathoverflow.net/q/20451`, MO question (v2010-04-06)

38 Leinster, T.: Basic Category Theory. Cambridge Studies in Advanced Mathematics, Cambridge University Press, Cambridge (2014). https://doi.org/10.1017/CBO9781107360068, `https://www.cambridge.org/core/books/basic-category-theory/A72533879BBC7BD956CC415777B7DA99`

39 Licata, D.R., Harper, R.: 2-Dimensional Directed Type Theory. Electronic Notes in Theoretical Computer Science **276**, 263–289 (Sep 2011). https://doi.org/10.1016/j.entcs.2011.09.026, `https://www.sciencedirect.com/science/article/pii/S1571066111001174`

40 Loregian, F.: (Co)end Calculus. London Mathematical Society Lecture Note Series, Cambridge University Press, Cambridge (2021). https://doi.org/10.1017/9781108778657, `https://www.cambridge.org/core/books/coend-calculus/C662E90767358B336F17B606D19D8C43`

41 Mac Lane, S.: Categories for the Working Mathematician, Graduate Texts in Mathematics, vol. 5. Springer-Verlag New York, 2nd edn. (1998). https://doi.org/10.1007/978-1-4757-4721-8

42 Maietti, M.E., Rosolini, G.: Quotient Completion for the Foundation of Constructive Mathematics. Logica Universalis **7**(3), 371–402 (Sep 2013). https://doi.org/10.1007/s11787-013-0080-2

43 Maietti, M.E., Rosolini, G.: Unifying Exact Completions. Applied Categorical Structures **23**(1), 43–52 (Feb 2015). https://doi.org/10.1007/s10485-013-9360-5

44 McCusker, G., Santamaria, A.: Composing dinatural transformations: Towards a calculus of substitution. Journal of Pure and Applied Algebra **225**(10), 106689 (Oct 2021). https://doi.org/10.1016/j.jpaa.2021.106689

45 Neumann, J.: Paranatural Category Theory. Tech. rep. (Jul 2023). https://doi.org/10.48550/arXiv.2307.09289, arXiv:2307.09289 [math] type: article

46 New, M.S., Licata, D.R.: A Formal Logic for Formal Category Theory. In: Kupferman, O., Sobocinski, P. (eds.) Foundations of Software Science and Computation Structures. pp. 113–134. Lecture Notes in Computer Science, Springer Nature Switzerland, Cham (2023). https://doi.org/10.1007/978-3-031-30829-1_6

47 North, P.R.: Towards a Directed Homotopy Type Theory. Electronic Notes in Theoretical Computer Science **347**, 223–239 (Nov 2019). https://doi.org/10.1016/j.entcs.2019.09.012, `https://www.sciencedirect.com/science/article/pii/S1571066119301288`

48 Nuyts, A.: Towards a Directed Homotopy Type Theory based on 4 Kinds of Variance. Master's thesis, KU Leuven (2015)

49 Paré, R., Román, L.: Dinatural numbers. Journal of Pure and Applied Algebra **128**(1), 33–92 (Jun 1998). https://doi.org/10.1016/S0022-4049(97)00036-4

50 Petrić, Z.: G-dinaturality. Annals of Pure and Applied Logic **122**(1), 131–173 (Aug 2003). https://doi.org/10.1016/S0168-0072(03)00003-4, https://www.sciencedirect.com/science/article/pii/S0168007203000034

51 Pistone, P.: On Dinaturality, Typability and beta-eta-Stable Models. In: Miller, D. (ed.) 2nd International Conference on Formal Structures for Computation and Deduction (FSCD 2017). Leibniz International Proceedings in Informatics (LIPIcs), vol. 84, pp. 29:1–29:17. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany (2017). https://doi.org/10.4230/LIPIcs.FSCD.2017.29

52 Pistone, P.: Proof nets, coends and the yoneda isomorphism. In: Ehrhard, T., Fernández, M., de Paiva, V., de Falco, L.T. (eds.) Proceedings Joint International Workshop on Linearity & Trends in Linear Logic and Applications, Linearity-TLLA@FLoC 2018, Oxford, UK, 7-8 July 2018. EPTCS, vol. 292, pp. 148–167 (2018). https://doi.org/10.4204/EPTCS.292.9

53 Pistone, P.: On completeness and parametricity in the realizability semantics of System F. Logical Methods in Computer Science **Volume 15, Issue 4** (Oct 2019). https://doi.org/10.23638/LMCS-15(4:6)2019, https://lmcs.episciences.org/5878

54 Pistone, P., Tranchini, L.: The Yoneda Reduction of Polymorphic Types. In: Baier, C., Goubault-Larrecq, J. (eds.) 29th EACSL Annual Conference on Computer Science Logic (CSL 2021). Leibniz International Proceedings in Informatics (LIPIcs), vol. 183, pp. 35:1–35:22. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, Dagstuhl, Germany (2021). https://doi.org/10.4230/LIPIcs.CSL.2021.35, https://drops.dagstuhl.de/opus/volltexte/2021/13469

55 Pitts, A.M.: Categorical logic. In: Abramsky, S., Gabbay, D.M., Maibaum, T.S.E. (eds.) Handbook of Logic in Computer Science: Volume 5: Logic and Algebraic Methods, pp. 39–123. Oxford University Press (May 1995). https://doi.org/10.1093/oso/9780198537816.003.0002

56 Riehl, E., Shulman, M.: A type theory for synthetic ∞-categories (May 2017). https://doi.org/10.48550/arXiv.1705.07442

57 Román, M.: Open Diagrams via Coend Calculus. In: Electronic Proceedings in Theoretical Computer Science. vol. 333, pp. 65–78 (Feb 2020). https://doi.org/10.4204/EPTCS.333.5

58 Santamaria, A.: Towards a Godement calculus for dinatural transformations. Ph.D. thesis, University of Bath (2019), https://ethos.bl.uk/OrderDetails.do?uin=uk.bl.ethos.787523

59 Scott, P.J.: Some aspects of categories in computer science. Handbook of Algebra, vol. 2, pp. 3–77. North-Holland (Jan 2000). https://doi.org/10.1016/S1570-7954(00)80027-3

60 Street, R.: Conspectus of variable categories. Journal of Pure and Applied Algebra **21**(3), 307–338 (Jun 1981). https://doi.org/10.1016/0022-4049(81)90021-9, https://www.sciencedirect.com/science/article/pii/0022404981900219

61 Univalent Foundations Program, T.: Homotopy Type Theory: Univalent Foundations of Mathematics. https://homotopytypetheory.org/book, Institute for Advanced Study (2013)

62 Uustalu, T.: A note on strong dinaturality, initial algebras and uniform parameterized fixpoint operators. In: Santocanale, L. (ed.) 7th Workshop on Fixed Points in Computer Science, FICS 2010, Brno, Czech Republic, August 21-22, 2010. pp. 77–82. Laboratoire d'Informatique Fondamentale de Marseille (2010), https://hal.archives-ouvertes.fr/hal-00512377/document#page=78

63 Uustalu, T., Veltri, N., Zeilberger, N.: Eilenberg-Kelly Reloaded. Electronic Notes in Theoretical Computer Science **352**, 233–256 (Oct 2020). https://doi.org/10.1016/j.entcs.2020.09.012, https://www.sciencedirect.com/science/article/pii/S1571066120300633

64 Voigtländer, J.: Free Theorems Simply, via Dinaturality. In: Hofstedt, P., Abreu, S., John, U., Kuchen, H., Seipel, D. (eds.) Declarative Programming and Knowledge Management. pp. 247–267. Springer International Publishing, Cham (2020). https://doi.org/10.1007/978-3-030-46714-2_16

65 Weaver, M.Z., Licata, D.R.: A Constructive Model of Directed Univalence in Bicubical Sets. In: Proceedings of the 35th Annual ACM/IEEE Symposium on Logic in Computer Science.

pp. 915–928. LICS ’20, Association for Computing Machinery, New York, NY, USA (Jul 2020). https://doi.org/10.1145/3373718.3394794

66 Wood, R.J.: Abstract proarrows I. Cahiers de topologie et géometrie différentielle categoriques **23**(3), 279–290 (1982)

67 Yoneda, N.: On Ext and exact sequences. J. Fac. Sci. Univ. Tokyo Sect. I **8**, 507–576 (1960) (1960)

A Complete rules for first-order dinatural directed type theory

The complete rules for dinatural directed type theory are listed in Figures 3 to 6.

The rules to formally capture the positive and negative polarity of variables is given in Figure 6 for predicates and Figure 5.

In light of the convention for natural variables described in Remark 12, we do not explicitly show the use of the above positive/negative variable constraints in the rules for entailments.

The rules for types, terms and predicates are assumed to depend on an underlying signature $\Sigma_B, \Sigma_T, \Sigma_P$, which can be used to extend the type theory with additional symbols. We omit the straightforward details for this extension and its corresponding interpretation.

$$\boxed{A\ \mathsf{type}} \quad \frac{C \in \Sigma_B}{C\ \mathsf{type}} \quad \frac{C \in \Sigma_B}{C^{\mathsf{op}}\ \mathsf{type}} \quad \frac{\mathbb{C}\ \mathsf{type} \quad \mathbb{D}\ \mathsf{type}}{\mathbb{C} \times \mathbb{D}\ \mathsf{type}} \quad \frac{\mathbb{C}\ \mathsf{type} \quad \mathbb{D}\ \mathsf{type}}{[\mathbb{C}, \mathbb{D}]\ \mathsf{type}} \quad \frac{}{\top\ \mathsf{type}}$$

$$\boxed{\Gamma\ \mathsf{ctx}} \quad \frac{}{[\,]\ \mathsf{ctx}} \quad \frac{\Gamma\ \mathsf{ctx} \quad \mathbb{C}\ \mathsf{type}}{\Gamma, \mathbb{C}\ \mathsf{ctx}} \quad \boxed{\Gamma \ni x : \mathbb{C}} \quad \frac{}{\Gamma, x : \mathbb{C} \ni x : \mathbb{C}} \quad \frac{\Gamma \ni x : \mathbb{C}}{\Gamma, y : \mathbb{D} \ni x : \mathbb{C}}$$

$$\boxed{\Gamma \vdash t : \mathbb{C}} \quad \frac{\Gamma \ni x : \mathbb{C}}{\Gamma \vdash x : \mathbb{C}} \quad \frac{f \in \Sigma_F \quad \Gamma \vdash t : \mathsf{dom}(f)}{\Gamma \vdash f(t) : \mathsf{cod}(f)}$$

$$\frac{}{\Gamma \vdash {!} : \top} \quad \frac{\Gamma \vdash s : \mathbb{C} \quad \Gamma \vdash t : \mathbb{D}}{\Gamma \vdash \langle s, t \rangle : \mathbb{C} \times \mathbb{D}} \quad \frac{\Gamma \vdash p : \mathbb{C} \times \mathbb{D}}{\Gamma \vdash \pi_1(p) : \mathbb{C}} \quad \frac{\Gamma \vdash p : \mathbb{C} \times \mathbb{D}}{\Gamma \vdash \pi_2(p) : \mathbb{D}}$$

$$\frac{\Gamma \vdash s : [\mathbb{C}, \mathbb{D}] \quad \Gamma \vdash t : \mathbb{C}}{\Gamma \vdash s \cdot t : \mathbb{D}} \quad \frac{\Gamma, x : \mathbb{C} \vdash t(x) : \mathbb{D}}{\Gamma \vdash \lambda x.t(x) : [\mathbb{C}, \mathbb{D}]}$$

$$\boxed{\Gamma \vdash t = t' : \mathbb{C}} \quad \frac{\Gamma, x : \mathbb{C} \vdash f(x) : \mathbb{D} \quad \Gamma \vdash t : \mathbb{C}}{\Gamma \vdash (\lambda x.f(x)) \cdot t = f[x \mapsto t] : \mathbb{D}} \quad \frac{\Gamma, x : \mathbb{C} \vdash f(x) : \mathbb{D}}{\Gamma, x : \mathbb{C} \vdash (\lambda x.f(x)) \cdot x = f(x) : \mathbb{D}}$$

$$\frac{\Gamma \vdash p : \mathbb{C} \times \mathbb{D}}{\Gamma \vdash \langle \pi_1(p), \pi_2(p) \rangle = p : \mathbb{C} \times \mathbb{D}} \quad \frac{\Gamma \vdash t : \top}{\Gamma \vdash t = {!} : \top} \quad \frac{\Gamma \vdash s : \mathbb{C} \quad \Gamma \vdash t : \mathbb{D}}{\Gamma \vdash \pi_1(\langle s, t \rangle) = s : \mathbb{C}} \quad \frac{\Gamma \vdash s : \mathbb{C} \quad \Gamma \vdash t : \mathbb{D}}{\Gamma \vdash \pi_2(\langle s, t \rangle) = t : \mathbb{D}}$$

Figure 3 Syntax of first-order directed type theory – types (categories) and terms (functors).

▶ **Theorem 35** ($\mathsf{op}, \mathsf{op}_t, \mathsf{op}_\varphi, \mathsf{op}_\Phi$ are involutions)**.** *All the operations given in Figure 4 are involutions, i.e.,* $(A^{\mathsf{op}})^{\mathsf{op}} \equiv A$ *for any type* A*,* $\mathsf{op}_t(\mathsf{op}_t(t)) \equiv t$ *for any term* t*, etc.*

Proof. Straightforward by induction on derivations. ◀

▶ **Theorem 36** (op inverts polarity)**.** *The judgement* $\Gamma \ni x : \mathbb{A}\ \mathsf{pos\ in}\ t$ *is derivable if and only if the judgement* $\Gamma \ni x : \mathbb{A}\ \mathsf{neg\ in}\ \mathsf{op}_t(t)$ *is derivable; similarly for negative variables.*

Proof. Straightforward by induction on the derivation for predicates. ◀

$$\boxed{[\Gamma]\ \varphi\ \mathsf{prop}} \qquad \frac{[\Gamma]\ \varphi\ \mathsf{prop} \quad [\Gamma]\ \psi\ \mathsf{prop}}{[\Gamma]\ \varphi \times \psi\ \mathsf{prop}} \qquad \frac{[\Gamma]\ \varphi\ \mathsf{prop} \quad [\Gamma]\ \psi\ \mathsf{prop}}{[\Gamma]\ \varphi \Rightarrow \psi\ \mathsf{prop}}$$

$$\frac{}{[\Gamma]\ \top\ \mathsf{prop}} \qquad \frac{\Gamma^{\mathsf{op}}, \Gamma \vdash s : \mathbb{C}^{\mathsf{op}} \quad \Gamma^{\mathsf{op}}, \Gamma \vdash t : \mathbb{C}}{[\Gamma]\ \hom_{\mathbb{C}}(s,t)\ \mathsf{prop}} \qquad \frac{P \in \Sigma_P \quad \Gamma^{\mathsf{op}}, \Gamma \vdash s : \mathsf{neg}(P)^{\mathsf{op}} \quad \Gamma^{\mathsf{op}}, \Gamma \vdash t : \mathsf{pos}(P)}{[\Gamma]\ P(s \mid t)\ \mathsf{prop}}$$

$$\frac{[\Gamma, x : \mathbb{C}]\ \varphi(x)\ \mathsf{prop}}{[\Gamma]\ \int^{x:\mathbb{C}} \varphi(x)\ \mathsf{prop}} \qquad \frac{[\Gamma, x : \mathbb{C}]\ \varphi(x)\ \mathsf{prop}}{[\Gamma]\ \int_{x:\mathbb{C}} \varphi(x)\ \mathsf{prop}}$$

Figure 4 Syntax of directed first-order logic – predicates (dipresheaves).

$$\boxed{\Gamma \ni x : \mathbb{A}\ \mathsf{unused\ in}\ t : \mathbb{C}} \qquad \frac{x \neq y}{\Gamma \ni y : \mathbb{C}\ \mathsf{unused\ in}\ x : \mathbb{C}} \qquad \frac{\Gamma \ni x : \mathbb{A}\ \mathsf{unused\ in}\ t : \mathrm{dom}(f)}{\Gamma \ni x : \mathbb{A}\ \mathsf{unused\ in}\ f(t) : \mathrm{cod}(f)}$$

$$\frac{}{\Gamma \ni x : \mathbb{A}\ \mathsf{unused\ in}\ ! : \top} \qquad \frac{\Gamma \ni x : \mathbb{A}\ \mathsf{unused\ in}\ s : \mathbb{C} \quad \Gamma \ni x : \mathbb{A}\ \mathsf{unused\ in}\ t : \mathbb{D}}{\Gamma \ni x : \mathbb{A}\ \mathsf{unused\ in}\ \langle s,t \rangle : \mathbb{C} \times \mathbb{D}}$$

$$\frac{\Gamma \ni x : \mathbb{A}\ \mathsf{unused\ in}\ p : \mathbb{C} \times \mathbb{D}}{\Gamma \ni x : \mathbb{A}\ \mathsf{unused\ in}\ \pi_1(p) : \mathbb{C}} \qquad \frac{\Gamma \ni x : \mathbb{A}\ \mathsf{unused\ in}\ p : \mathbb{C} \times \mathbb{D}}{\Gamma \ni x : \mathbb{A}\ \mathsf{unused\ in}\ \pi_2(p) : \mathbb{D}}$$

$$\frac{\Gamma \ni x : \mathbb{A}\ \mathsf{unused\ in}\ s : [\mathbb{C}, \mathbb{D}] \quad \Gamma \ni x : \mathbb{A}\ \mathsf{unused\ in}\ t : \mathbb{C}}{\Gamma \ni x : \mathbb{A}\ \mathsf{unused\ in}\ s \cdot t : \mathbb{D}} \qquad \frac{\Gamma, x : \mathbb{C} \vdash t(x) : \mathbb{D}}{\Gamma \ni x : \mathbb{A}\ \mathsf{unused\ in}\ \lambda x.t(x) : [\mathbb{C}, \mathbb{D}]}$$

Figure 5 Syntax of first-order directed type theory – predicates for syntactically unused variables in terms.

B (Co)end calculus, other derivations

We report here additional examples of derivations for (co)end calculus using our rules.

▶ **Example 37** ($\Rightarrow$ resp. limits)**.** Ends are limits [40], and functors $- \Rightarrow - : \mathsf{Set}^{\mathsf{op}} \times \mathsf{Set} \to \mathsf{Set}$ preserve them (ends/limits in $\mathsf{Set}^{\mathsf{op}}$, i.e., coends/colimits in Set). For $\Phi, Q : \mathsf{Set}$, $P : \mathbb{C}^{\diamond} \to \mathsf{Set}$:

$$\begin{array}{c} [\,]\ \Phi \vdash Q \Rightarrow \int_{x:\mathbb{C}} P(\overline{x}, x) \\ \hline\hline [\,]\ Q, \Phi \vdash \int_{x:\mathbb{C}} P(\overline{x}, x) \\ \hline\hline [x : \mathbb{C}]\ Q, \Phi \vdash P(\overline{x}, x) \\ \hline\hline [x : \mathbb{C}]\ \Phi \vdash Q \Rightarrow P(\overline{x}, x) \\ \hline\hline [\,]\ \Phi \vdash \int_{x:\mathbb{C}} (Q \Rightarrow P(\overline{x}, x)) \end{array} \begin{array}{l} \\ (\mathsf{exp}) \\ (\mathsf{end}) \\ (\mathsf{exp}) \\ (\mathsf{end}) \end{array} \qquad \begin{array}{c} [\,]\ \Phi \vdash (\int^{x:\mathbb{C}} P(\overline{x}, x)) \Rightarrow Q \\ \hline\hline [\,]\ (\int^{x:\mathbb{C}} P(\overline{x}, x)), \Phi \vdash Q \\ \hline\hline [x : \mathbb{C}]\ P(\overline{x}, x), \Phi \vdash Q \\ \hline\hline [x : \mathbb{C}]\ \Phi \vdash P^{\mathsf{op}}(x, \overline{x}) \Rightarrow Q \\ \hline\hline [\,]\ \Phi \vdash \int_{x:\mathbb{C}} P^{\mathsf{op}}(x, \overline{x}) \Rightarrow Q \end{array} \begin{array}{l} \\ (\mathsf{exp}) \\ (\mathsf{coend{+}frob.}) \\ (\mathsf{exp}) \\ (\mathsf{end}) \end{array}$$

▶ **Example 38** (Pointwise fomula for left Kan extensions)**.** Dually to Theorem 33, we give a logical proof that the functor $\mathsf{Lan}_F : [\mathbb{C}, \mathsf{Set}] \to [\mathbb{D}, \mathsf{Set}]$ sending (co)presheaves to their left Kan extensions along $F : \mathbb{C} \to \mathbb{D}$ computed via coends [40, 2.3.6] is left adjoint to precomposition $(F\,;-) : [\mathbb{D}, \mathsf{Set}] \to [\mathbb{C}, \mathsf{Set}]$. For any $P : \mathbb{C} \to \mathsf{Set}$, a functor/term $F : \mathbb{C} \to \mathbb{D}$

$$\boxed{\Gamma \ni x : \mathbb{A}\ \mathsf{pos\ in}\ \varphi}$$

$$\frac{\Gamma \ni x : \mathbb{A}\ \mathsf{pos\ in}\ \varphi \qquad \Gamma \ni x : \mathbb{A}\ \mathsf{pos\ in}\ \psi}{\Gamma \ni x : \mathbb{A}\ \mathsf{pos\ in}\ \varphi \wedge \psi} \qquad \frac{\Gamma \ni x : \mathbb{A}\ \mathsf{neg\ in}\ \varphi \qquad \Gamma \ni x : \mathbb{A}\ \mathsf{pos\ in}\ \psi}{\Gamma \ni x : \mathbb{A}\ \mathsf{pos\ in}\ \varphi \Rightarrow \psi}$$

$$\frac{}{\Gamma \ni x : \mathbb{A}\ \mathsf{pos\ in}\ \top} \qquad \frac{\Gamma, y : \mathbb{D} \ni x : \mathbb{C}\ \mathsf{pos\ in}\ \varphi}{\Gamma \ni x : \mathbb{A}\ \mathsf{pos\ in}\ \int^{x:\mathbb{C}} \varphi(x)} \qquad \frac{\Gamma, y : \mathbb{D} \ni x : \mathbb{C}\ \mathsf{pos\ in}\ \varphi}{\Gamma \ni x : \mathbb{A}\ \mathsf{pos\ in}\ \int_{x:\mathbb{C}} \varphi(x)}$$

$$\frac{\Gamma^{\mathsf{op}}, \Gamma \ni x : \mathbb{A}\ \mathsf{unused\ in}\ s : \mathbb{C}^{\mathsf{op}} \qquad \Gamma^{\mathsf{op}}, \Gamma \ni \overline{x} : \mathbb{A}^{\mathsf{op}}\ \mathsf{unused\ in}\ t : \mathbb{C}}{\Gamma \ni x : \mathbb{A}\ \mathsf{pos\ in}\ \hom_{\mathbb{C}}(s,t)}$$

$$\frac{\Gamma^{\mathsf{op}}, \Gamma \ni x : \mathbb{A}\ \mathsf{unused\ in}\ s : \mathsf{neg}(P)^{\mathsf{op}} \qquad \Gamma^{\mathsf{op}}, \Gamma \ni \overline{x} : \mathbb{A}^{\mathsf{op}}\ \mathsf{unused\ in}\ t : \mathsf{pos}(P)}{\Gamma \ni x : \mathbb{A}\ \mathsf{pos\ in}\ P(s \mid t)}$$

$$\boxed{\Gamma \ni x : \mathbb{A}\ \mathsf{neg\ in}\ \varphi}$$

$$\frac{\Gamma \ni x : \mathbb{A}\ \mathsf{neg\ in}\ \varphi \qquad \Gamma \ni x : \mathbb{A}\ \mathsf{neg\ in}\ \psi}{\Gamma \ni x : \mathbb{A}\ \mathsf{neg\ in}\ \varphi \wedge \psi} \qquad \frac{\Gamma \ni x : \mathbb{A}\ \mathsf{pos\ in}\ \varphi \qquad \Gamma \ni x : \mathbb{A}\ \mathsf{neg\ in}\ \psi}{\Gamma \ni x : \mathbb{A}\ \mathsf{neg\ in}\ \varphi \Rightarrow \psi}$$

$$\frac{}{\Gamma \ni x : \mathbb{A}\ \mathsf{neg\ in}\ \top} \qquad \frac{\Gamma, y : \mathbb{D} \ni x : \mathbb{C}\ \mathsf{neg\ in}\ \varphi}{\Gamma \ni x : \mathbb{A}\ \mathsf{neg\ in}\ \int^{x:\mathbb{C}} \varphi(x)} \qquad \frac{\Gamma, y : \mathbb{D} \ni x : \mathbb{C}\ \mathsf{neg\ in}\ \varphi}{\Gamma \ni x : \mathbb{A}\ \mathsf{neg\ in}\ \int_{x:\mathbb{C}} \varphi(x)}$$

$$\frac{\Gamma^{\mathsf{op}}, \Gamma \ni \overline{x} : \mathbb{A}\ \mathsf{unused\ in}\ s : \mathbb{C}^{\mathsf{op}} \qquad \Gamma^{\mathsf{op}}, \Gamma \ni x : \mathbb{A}^{\mathsf{op}}\ \mathsf{unused\ in}\ t : \mathbb{C}}{\Gamma \ni x : \mathbb{A}\ \mathsf{neg\ in}\ \hom_{\mathbb{C}}(s,t)}$$

$$\frac{\Gamma^{\mathsf{op}}, \Gamma \ni \overline{x} : \mathbb{A}\ \mathsf{unused\ in}\ s : \mathsf{neg}(P)^{\mathsf{op}} \qquad \Gamma^{\mathsf{op}}, \Gamma \ni x : \mathbb{A}^{\mathsf{op}}\ \mathsf{unused\ in}\ t : \mathsf{pos}(P)}{\Gamma \ni x : \mathbb{A}\ \mathsf{neg\ in}\ P(s \mid t)}$$

$$\boxed{\Gamma \ni x : \mathbb{A}\ \mathsf{pos\ in}\ \Phi} \qquad \frac{}{\Gamma \ni x : \mathbb{A}\ \mathsf{pos\ in}\ [\,]} \qquad \frac{\Gamma \ni x : \mathbb{A}\ \mathsf{pos\ in}\ \Phi \qquad \Gamma \ni x : \mathbb{A}\ \mathsf{pos\ in}\ \varphi}{\Gamma \ni x : \mathbb{A}\ \mathsf{pos\ in}\ \Phi, \varphi}$$

$$\boxed{\Gamma \ni x : \mathbb{A}\ \mathsf{neg\ in}\ \Phi} \qquad \frac{}{\Gamma \ni x : \mathbb{A}\ \mathsf{neg\ in}\ [\,]} \qquad \frac{\Gamma \ni x : \mathbb{A}\ \mathsf{neg\ in}\ \Phi \qquad \Gamma \ni x : \mathbb{A}\ \mathsf{neg\ in}\ \varphi}{\Gamma \ni x : \mathbb{A}\ \mathsf{neg\ in}\ \Phi, \varphi}$$

Figure 6 Syntax of first-order directed type theory – mutually defined syntactic conditions for positive/negative variables in predicates and propositional contexts.

and a generic $\mathit{\Gamma} : \mathbb{D} \to \mathsf{Set}$:

$$\begin{array}{rl}
[y : \mathbb{D}] \quad (\mathsf{Lan}_F P)(x) := \int^{x:\mathbb{C}} \hom_{\mathbb{C}}(F^{\mathsf{op}}(\overline{x}), y) \times P(x) \vdash \mathit{\Gamma}(y) & \\
\hline\hline
[x : \mathbb{C}, y : \mathbb{D}]\ \hom_{\mathbb{C}}(F^{\mathsf{op}}(\overline{x}), y) \times P(x) \vdash \mathit{\Gamma}(y) & (\mathsf{coend}) \\
\hline\hline
[x : \mathbb{C}, y : \mathbb{D}]\ P(x) \vdash \hom^{\mathsf{op}}_{\mathbb{C}}(F(x), \overline{y}) \Rightarrow \mathit{\Gamma}(y) & (\mathsf{exp}) \\
\hline\hline
[x : \mathbb{C}]\ P(x) \vdash \int_{y:\mathbb{D}} \hom^{\mathsf{op}}_{\mathbb{D}}(F(x), \overline{y}) \Rightarrow \mathit{\Gamma}(y) & (\mathsf{end}) \\
\hline\hline
[x : \mathbb{C}]\ P(x) \vdash \mathit{\Gamma}(F(x)) & (\mathsf{Yoneda})
\end{array}$$

▶ **Example 39** (Right rifts in profunctors). We give a logical proof that composition (on both sides) in Prof has a right adjoint [40, 5.2.5 and Exercise 5.2]. This makes Prof a bicategory where *right extensions* and *right liftings* exist. For simplicity we only treat precomposition, although postcomposition is completely analogous. For any composable

$$
\begin{aligned}
&-^{\mathsf{op}} : \{-\ \mathsf{type}\} \to \{-\ \mathsf{type}\} \\
&(\mathbb{C}^{\mathsf{op}})^{\mathsf{op}} := \mathbb{C} \\
&(\mathbb{C})^{\mathsf{op}} := \mathbb{C}^{\mathsf{op}} \\
&(A \times B)^{\mathsf{op}} := A^{\mathsf{op}} \times B^{\mathsf{op}} \\
&([\mathbb{C}, \mathbb{D}])^{\mathsf{op}} := [A^{\mathsf{op}}, B^{\mathsf{op}}] \\
&(\top)^{\mathsf{op}} := \top
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{op}_t : \{\Gamma \vdash - : \mathbb{C}\} \to \{\Gamma^{\mathsf{op}} \vdash - : \mathbb{C}^{\mathsf{op}}\} \\
&\mathsf{op}_t(x) := x \\
&\mathsf{op}_t(f(t)) := f(\mathsf{op}_t(t)) \\
&\mathsf{op}_t(\langle s, t\rangle) := \langle \mathsf{op}_t(s), \mathsf{op}_t(t)\rangle \\
&\mathsf{op}_t(\pi_1(p)) := \pi_1(\mathsf{op}_t(p)) \\
&\mathsf{op}_t(\pi_2(p)) := \pi_2(\mathsf{op}_t(p)) \\
&\mathsf{op}_t(s \cdot t) := \mathsf{op}_t(s) \cdot \mathsf{op}_t(t) \\
&\mathsf{op}_t(\lambda x.t(x)) := \lambda x.\mathsf{op}_t(t(x))
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{op}_\varphi : \{[\Gamma]\ -\ \mathsf{prop}\} \to \{[\Gamma]\ -\ \mathsf{prop}\} \\
&\mathsf{op}_\varphi(\top) := \top \\
&\mathsf{op}_\varphi(\hom_{\mathbb{C}}(s,t)) := \hom_{\mathbb{C}^{\mathsf{op}}}(\mathsf{op}_t(t), \mathsf{op}_t(s)) \\
&\mathsf{op}_\varphi(P(s \mid t)) := P(\mathsf{op}_t(t) \mid \mathsf{op}_t(s)) \\
&\mathsf{op}_\varphi\left(\int^{x:\mathbb{C}} \varphi(x)\right) := \int^{x:\mathbb{C}^{\mathsf{op}}} \mathsf{op}_\varphi(\varphi(x)) \\
&\mathsf{op}_\varphi\left(\int_{x:\mathbb{C}} \varphi(x)\right) := \int_{x:\mathbb{C}^{\mathsf{op}}} \mathsf{op}_\varphi(\varphi(x))
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{op}_\Phi : \{[\Gamma]\ -\ \mathsf{propctx}\} \to \{[\Gamma]\ -\ \mathsf{propctx}\} \\
&\mathsf{op}_\Phi([\,]) := [\,] \\
&\mathsf{op}_\Phi(\Phi, \varphi) := \mathsf{op}_\Phi(\Phi), \mathsf{op}_\varphi(\varphi)
\end{aligned}
$$

$$
\frac{[x : \mathbb{C}, \Gamma]\ \Phi(\overline{x}, x) \vdash \alpha : P(\overline{x}, x)}{[x : \mathbb{C}^{\mathsf{op}}, \Gamma]\ \Phi(x, \overline{x}) \vdash \alpha : P(x, \overline{x})}\ (\mathsf{op})
$$

Figure 7 Syntax of first-order directed type theory – op-operations, defined by induction on derivations. The last rule for entailments is similarly admissible by induction.

$$
\boxed{[\Gamma]\ \Phi \vdash \alpha = \beta : P} \quad \cdots
$$

$$
\frac{\begin{array}{ll}
[a : \Delta^{\mathsf{op}}, b : \Delta, x : \Gamma] & \Phi(\overline{x}, x, a, b) \vdash \alpha \quad : P(a, b) \\
[z : \Delta, x : \Gamma] & k : P(\overline{z}, z), \Phi(\overline{x}, x, \overline{z}, z) \vdash \gamma[k] : Q(\overline{z}, z) \\
[a : \Delta^{\mathsf{op}}, b : \Delta, x : \Gamma] & k : Q(a, b), \Phi(\overline{x}, x, \overline{a}, \overline{b}) \vdash \beta[k] : R(a, b)
\end{array}}{[z : \Delta, x : \Gamma] \quad \Phi(\overline{x}, x, \overline{z}, z) \vdash (\beta[\gamma])[\alpha] = \beta[\gamma[\alpha]] : R(\overline{z}, z)}\ (\mathsf{assoc\text{-}nat\text{-}din\text{-}nat})
$$

Figure 8 Syntax of first-order directed type theory – Associativity for natural-dinatural-natural cuts in the equational theory, using ($\mathsf{cut\text{-}din}$) and ($\mathsf{cut\text{-}nat}$).

profunctors $P : \mathbb{C}^{\mathsf{op}} \times \mathbb{A} \to \mathsf{Set}$, $Q : \mathbb{A}^{\mathsf{op}} \times \mathbb{D} \to \mathsf{Set}$ and a generic $\mathit{\Gamma} : \mathbb{C}^{\mathsf{op}} \times \mathbb{D} \to \mathsf{Set}$:

$$
\cfrac{\cfrac{\cfrac{\cfrac{\begin{array}{l}[x : \mathbb{C}^{\mathsf{op}}, z : \mathbb{D}] \quad (P\,;-)(Q)(x,z) := \\ \qquad \int^{y:\mathbb{A}} P(x,y) \times Q(\overline{y}, z) \vdash \mathit{\Gamma}(x,z)\end{array}}{[x : \mathbb{C}^{\mathsf{op}}, y : \mathbb{A}, z : \mathbb{D}]\ P(x,y) \times Q(\overline{y}, z) \vdash \mathit{\Gamma}(x,z)}\ (\mathsf{coend})}{[x : \mathbb{C}^{\mathsf{op}}, y : \mathbb{A}, z : \mathbb{D}]\ Q(\overline{y}, z) \vdash P^{\mathsf{op}}(\overline{x}, \overline{y}) \Rightarrow \mathit{\Gamma}(x,z)}\ (\mathsf{exp})}{[y : \mathbb{A}, z : \mathbb{D}]\ Q(\overline{y}, z) \vdash \int_{x:\mathbb{C}} P^{\mathsf{op}}(\overline{x}, \overline{y}) \Rightarrow \mathit{\Gamma}(x,z)}\ (\mathsf{end})}{\begin{array}{l}[y : \mathbb{A}^{\mathsf{op}}, z : \mathbb{D}]\ Q(y, z) \vdash \int_{x:\mathbb{C}} P^{\mathsf{op}}(\overline{x}, y) \Rightarrow \mathit{\Gamma}(x,z) \\ \qquad := \mathsf{Rift}_P(\mathit{\Gamma})(y,z)\end{array}}\ (\mathsf{op})
$$

where the last (end) can be applied since $x : \mathbb{C}$ does not appear on the left.

▶ **Example 40** (Composition of profunctors is associative). Using our approach relying on

$$\boxed{[\Gamma]\ \Phi \vdash \alpha : P} \quad \frac{[a:\mathbb{C},\Gamma]\ \Phi \vdash \alpha : Q(\overline{a},a)}{[\Gamma]\ \Phi \vdash \mathsf{end}(\alpha) : \int_{a:\mathbb{C}} Q(\overline{a},a)}\ (\mathsf{end}) \quad \frac{[\Gamma]\ \Phi \vdash \alpha : \int_{a:\mathbb{C}} Q(\overline{a},a)}{[a:\mathbb{C},\Gamma]\ \Phi \vdash \mathsf{end}^{-1}(\alpha) : Q(\overline{a},a)}\ (\mathsf{end})$$

$$\boxed{[\Gamma]\ \Phi \vdash \alpha = \beta : P} \quad \frac{[a:\mathbb{C},\Gamma]\ \Phi \vdash \alpha : Q(\overline{a},a)}{[a:\mathbb{C},\Gamma]\ \Phi \vdash \mathsf{end}^{-1}(\mathsf{end}(\alpha)) = \alpha : Q(\overline{a},a)} \quad \frac{[\Gamma]\ \Phi \vdash \alpha : \int_{a:\mathbb{C}} Q(\overline{a},a)}{[\Gamma]\ \Phi \vdash \mathsf{end}(\mathsf{end}^{-1}(\alpha)) = \alpha : \int_{a:\mathbb{C}} Q(\overline{a},a)}$$

Figure 9 Syntax of first-order directed type theory – Explicitly showing a bidirectional rule, e.g., for ends.

contextual operations we easily show that composition of profunctors, defined via a coend [40], is associative and essentially follows from associativity of products. For composable profunctors $P : \mathbb{A}^{\mathsf{op}} \times \mathbb{B} \to \mathsf{Set}, Q : \mathbb{B}^{\mathsf{op}} \times \mathbb{C} \to \mathsf{Set}, R : \mathbb{C}^{\mathsf{op}} \times \mathbb{D} \to \mathsf{Set}$ and a generic $\mathit{\Gamma} : \mathbb{A}^{\mathsf{op}} \times \mathbb{D} \to \mathsf{Set}$:

$$\begin{array}{ll}
[a:\mathbb{A}, d:\mathbb{D}]\ \int^{b:\mathbb{B}} P(\overline{a},b) \times \left(\int^{c:\mathbb{C}} Q(\overline{b},c) \times R(\overline{c},d)\right) \vdash \mathit{\Gamma}(\overline{a},d) & \\
\hline\hline & (\mathsf{coend}) \\
[a:\mathbb{A}, b:\mathbb{B}, d:\mathbb{D}]\ P(\overline{a},b) \times \left(\int^{c:\mathbb{C}} Q(\overline{b},c) \times R(\overline{c},d)\right) \vdash \mathit{\Gamma}(\overline{a},d) & \\
\hline\hline & (\mathsf{coend{+}frob.}) \\
[a:\mathbb{A}, b:\mathbb{B}, c:\mathbb{C}, d:\mathbb{D}]\ P(\overline{a},b) \times (Q(\overline{b},c) \times R(\overline{c},d)) \vdash \mathit{\Gamma}(\overline{a},d) & \\
\hline\hline & (\mathsf{structural\ property}) \\
[a:\mathbb{A}, b:\mathbb{B}, c:\mathbb{C}, d:\mathbb{D}]\ (P(\overline{a},b) \times Q(\overline{b},c)) \times R(\overline{c},d) \vdash \mathit{\Gamma}(\overline{a},d) & \\
\hline\hline & (\mathsf{coend{+}frob.}) \\
[a:\mathbb{A}, c:\mathbb{C}, d:\mathbb{D}]\ \left(\int^{b:\mathbb{B}} P(\overline{a},b) \times Q(\overline{b},c)\right) \times R(\overline{c},d) \vdash \mathit{\Gamma}(\overline{a},d) & \\
\hline\hline & (\mathsf{coend}) \\
[a:\mathbb{A}, d:\mathbb{D}]\ \int^{c:\mathbb{C}} \left(\int^{b:\mathbb{B}} P(\overline{a},b) \times Q(\overline{b},c)\right) \times R(\overline{c},d) \vdash \mathit{\Gamma}(\overline{a},d) &
\end{array}$$

▶ **Example 41** (Dinaturals as an end)**.** The set of dinaturals $\mathsf{Dinat}(P,Q) := \{P \xrightarrow{\bullet\bullet} Q\}$ between dipresheaves $P, Q : \mathbb{C}^{\mathsf{op}} \times \mathbb{C} \to \mathsf{Set}$ can be characterized in terms of the following end [18, Thm. 1], $\mathsf{Dinat}(P,Q) \cong \int_{x:\mathbb{C}} P^{\mathsf{op}}(x,\overline{x}) \Rightarrow Q(\overline{x},x)$.

We give a simple derivation by characterizing all points of the end:

$$\begin{array}{rll}
\mathsf{Dinat}(P,Q) := & [x:\mathbb{C}]\ P(\overline{x},x) \vdash Q(\overline{x},x) & \\
\hline\hline & & (\mathsf{exp}) \\
& [x:\mathbb{C}]\ \top \vdash P^{\mathsf{op}}(x,\overline{x}) \Rightarrow Q(\overline{x},x) & \\
\hline\hline & & (\mathsf{end}) \\
& [\,]\ \top \vdash \int_{x:\mathbb{C}} P^{\mathsf{op}}(x,\overline{x}) \Rightarrow Q(\overline{x},x) &
\end{array}$$

Since dinaturals generalize naturals, a similar derivation justifies the well-known description of natural transformations as ends shown in Section 1 for $F, G : \mathbb{C} \to \mathsf{Set}$,

$$\mathsf{Nat}(F,G) \cong \int_{x:\mathbb{C}} F^{\mathsf{op}}(\overline{x}) \Rightarrow G(x).$$

▶ **Example 42** (Internal naturality for natural transformations)**.** We show that naturality for natural transformations, expressed as ends [40], holds internally by directed equality elimination. Given terms $\mathbb{C} \vdash F, G : \mathbb{D}$, we use the counit of Theorem 27 to extract the family of hom-sets. We first explicitly show the rules used to construct the two sides of a naturality square:

$$\dfrac{\dfrac{\dfrac{[a:\mathbb{C}^{\mathsf{op}}, b:\mathbb{C}]\; f:\hom_{\mathbb{C}}(a,b), \eta : \int_{x:\mathbb{C}} \hom_{\mathbb{D}} F(\overline{x}), G(x) \vdash \eta : \int_{x:\mathbb{C}} \hom(F(\overline{x}), G(x))}{[a:\mathbb{C}^{\mathsf{op}}, b:\mathbb{C}, x:\mathbb{C}]\; f:\hom_{\mathbb{C}}(a,b), \eta : \ldots \vdash \mathsf{end}^{-1}(\eta) : \hom(F(\overline{x}), G(x))}\;(\mathsf{end}^{-1})}{[a:\mathbb{C}^{\mathsf{op}}, b:\mathbb{C}]\; f:\hom_{\mathbb{C}}(a,b), \eta : \ldots \vdash \Delta^*(\mathsf{end}^{-1}(\eta)) : \hom(F(a), G(\overline{a}))}\;(\mathsf{idx})}{[a:\mathbb{C}^{\mathsf{op}}, b:\mathbb{C}]\; f:\hom_{\mathbb{C}}(a,b), \eta : \ldots \vdash \mathsf{comp}[\Delta^*(\mathsf{end}^{-1}(\eta)), \mathsf{cong}_G[f]] : \hom(F(a), G(b))}\;(\mathsf{cut\text{-}nat})$$

where Δ^* is the reindexing functor which collapses a, x to a single variable a, and (cut-nat) is used to apply comp on cong for G. This composition can be done since both cong and comp have the correct naturality shape that allows for (cut-nat) to be applied.

The other derivation is obtained similarly:

$$\dfrac{\dfrac{\dfrac{[a:\mathbb{C}^{\mathsf{op}}, b:\mathbb{C}]\; f:\hom_{\mathbb{C}}(a,b), \eta : \int_{x:\mathbb{C}} \hom_{\mathbb{D}} F(\overline{x}), G(x) \vdash \eta : \int_{x:\mathbb{C}} \hom(F(\overline{x}), G(x))}{[a:\mathbb{C}^{\mathsf{op}}, b:\mathbb{C}, x:\mathbb{C}]\; f:\hom_{\mathbb{C}}(a,b), \eta : \ldots \vdash \mathsf{end}^{-1}(\eta) : \hom(F(\overline{x}), G(x))}\;(\mathsf{end}^{-1})}{[a:\mathbb{C}^{\mathsf{op}}, b:\mathbb{C}]\; f:\hom_{\mathbb{C}}(a,b), \eta : \ldots \vdash \Delta^*(\mathsf{end}^{-1}(\eta)) : \hom(F(\overline{b}), G(b))}\;(\mathsf{idx})}{[a:\mathbb{C}^{\mathsf{op}}, b:\mathbb{C}]\; f:\hom_{\mathbb{C}}(a,b), \eta : \ldots \vdash \mathsf{comp}[\mathsf{cong}_F[f], \Delta^*(\mathsf{end}^{-1}(\eta))] : \hom(F(a), G(b))}\;(\mathsf{cut\text{-}nat})$$

We show that the two maps constructed, corresponding to the two sides of a naturality square, are equal using directed equality elimination; let $K := \Delta^*(\mathsf{end}^{-1}(\eta))$:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{[z:\mathbb{C}]\ldots \vdash K = K : \hom(F(\overline{z}), G(z))}}{[z:\mathbb{C}]\ldots \vdash \mathsf{comp}[\mathsf{refl}_z, K] = \mathsf{comp}[K, \mathsf{refl}_z] : \hom(F(\overline{z}), G(z))}\;(J\text{-}\mathsf{comp})}{[z:\mathbb{C}]\ldots \vdash \mathsf{comp}[\mathsf{cong}_F[\mathsf{refl}_z], K] = \mathsf{comp}[K, \mathsf{cong}_G[\mathsf{refl}_z]] : \hom(F(\overline{z}), G(z))}\;(J\text{-}\mathsf{comp})}{[a:\mathbb{C}^{\mathsf{op}}, b:\mathbb{C}]\; f:\hom_{\mathbb{C}}(a,b), \ldots \vdash \mathsf{comp}[\mathsf{cong}_F[f], K] = \mathsf{comp}[K, \mathsf{cong}_G[f]] : \hom(F(a), G(b))}\;(J\text{-}\mathsf{eq})$$

where the equations used follow by the computation rules for cong and left and right unitality of comp. Note that (J-eq) can be used since a, b appear precisely with the correct types that allow for (J) to be applied to contract the equality.

This naturality can then be used to prove a suitable internal Yoneda lemma for the hom of categories by following the standard argument, e.g., given in [38].

C Frobenius condition for coends

▶ **Theorem 43** (Frobenius condition for coends). *For any* $\Gamma : \mathbb{A}^\diamond \times \mathbb{C}^\diamond \to \mathsf{Set}$ *and a generic* $K : \mathbb{C}^\diamond \to \mathsf{Set}$*, the following series of derivations establishes the Frobenius condition given in Theorem 28, which we prove by following exactly the argument given in [34, 1.9.12(i)] in the case of fibrations with exponentials. Note that we use the same Yoneda technique described in Remark 30.*

$$
\cfrac{\cfrac{\cfrac{\cfrac{[y:\mathbb{C}]\ \int^{x:\mathbb{A}[y:\mathbb{C}]}(P(\overline{y},y)\times\Gamma(\overline{x},x,\overline{y},y))\vdash K(\overline{y},y)}{[x:\mathbb{A},y:\mathbb{C}]\ P(\overline{y},y),\Gamma(\overline{x},x,\overline{y},y)\vdash K(\overline{y},y)}\ (\mathsf{coend})}{[x:\mathbb{A},y:\mathbb{C}]\ \Gamma(\overline{x},x,\overline{y},y)\vdash P^{\mathsf{op}}(y,\overline{y})\Rightarrow K(\overline{y},y)}\ (\mathsf{exp})}{[y:\mathbb{C}]\ \int^{x:\mathbb{A}[y:\mathbb{C}]}\Gamma(\overline{x},x,\overline{y},y)\vdash P^{\mathsf{op}}(y,\overline{y})\Rightarrow K(\overline{y},y)}\ (\mathsf{coend})}{[y:\mathbb{C}]\ P(\overline{y},y),\int^{x:\mathbb{A}[y:\mathbb{C}]}\Gamma(\overline{x},x,\overline{y},y)\vdash K(\overline{y},y)}\ (\mathsf{exp})
$$

▶ **Theorem 44** ((coend) ⇒ (coend+frob.)). *Following a similar argument to the one given in Theorem 43 using the existence of exponentials, the rule* (coend+frob.) *can be directly justified using* (coend)*, as follows:*

$$
\cfrac{\cfrac{\cfrac{[x:\mathbb{C}]\ \left(\int^{a:\mathbb{A}}Q(\overline{a},a,\overline{x},x)\right),\Gamma(\overline{x},x)\vdash P(\overline{x},x)}{[x:\mathbb{C}]\ \int^{a:\mathbb{A}}Q(\overline{a},a,\overline{x},x)\vdash\Gamma^{\mathsf{op}}(x,\overline{x})\Rightarrow P(\overline{x},x)}\ (\mathsf{exp})}{[x:\mathbb{C},y:\mathbb{C}]\ Q(\overline{a},a,\overline{x},x)\vdash\Gamma^{\mathsf{op}}(x,\overline{x})\Rightarrow P(\overline{x},x)}\ (\mathsf{coend})}{[x:\mathbb{C}]\ Q(\overline{a},a,\overline{x},x),\Gamma(\overline{x},x)\vdash P(\overline{x},x)}\ (\mathsf{exp})
$$