Univalent Models of a Non-Associative Composition

Beatrice Szilvasy

August 21, 2026

Abstract

Mixing call-by-name and call-by-value evaluation orders in a single programming language gives rise to a composition which is non-associative. To model such semantics, one may use Munch-Maccagnoni's *duploids*, category-like structures that relax the associativity of composition. We study and formalize duploids in homotopy type theory/univalent foundations. Our work includes a definition of what it means for a duploid to be *univalent*: that isomorphisms of a suitable kind render the objects indistinguishable to the type theory. We show that the existing construction of a duploid from an adjunction fails to be univalent as a "single-sorted" duploid, and provide a novel construction which is. This work is formalized in the UniMath library for the Rocq proof assistant.

Acknowledgements

I am grateful to my supervisor Jon Sterling, and his support, advice and direction that made this project possible. I am equally grateful for the assistance and companionship of numerous friends and colleagues, at St Catharine's College, at Cambridge broadly, and also elsewhere. Discussions with Dominik Joe Pantůček, Guillaume Munch-Maccagnoni, Jacob Bennett-Woolf, Jem Lord, Niels van der Weide, Siddhartha Kasivajhula, Vivian, William Sørensen, and others have helped shape the work in various ways.

Contents

1 Introduction

This dissertation is concerned with the semantics of programming languages in which composition is non-associative. Composition is expressed by the `let` $\cdots$ `in` $\cdots$ construct, and non-associativity means that programs:

$$\texttt{let}\ \Big(\texttt{let}\ e_1\ \texttt{be}\ x\ \texttt{in}\ e_2\Big)\ \texttt{be}\ y\ \texttt{in}\ e_3 \quad \text{and} \tag{1.1}$$

$$\texttt{let}\ e_1\ \texttt{be}\ x\ \texttt{in}\ \Big(\texttt{let}\ e_2\ \texttt{be}\ y\ \texttt{in}\ e_3\Big) \tag{1.2}$$

(where e_3 does not reference x)

may compute differently. One way in which non-associativity arises is from computational effects in the presence of mixed evaluation order.

To illustrate evaluation order, consider the program

$$\texttt{let}\ \big(\texttt{print}(\texttt{"hello"});\ 1\big)\ \texttt{be}\ x\ \texttt{in}\ x + x. \tag{1.3}$$

Here, evaluating $\texttt{print}(v)$ outputs v to a console. There are two canonical evaluations of Program 1.3 depending on the choice of evaluation order for the `let` construct:

- In *call-by-name*, the expression $\texttt{let}\ e_1\ \texttt{be}\ x\ \texttt{in}\ e_2$ evaluates by substituting e_1 for x in e_2. Then, the resulting expression $e_2[x := e_1]$ evaluates. Using call-by-name, Program 1.3 outputs `hellohello` and returns 2.

- In *call-by-value*, $\texttt{let}\ e_1\ \texttt{be}\ x\ \texttt{in}\ e_2$ evaluates by first evaluating e_1 to its value v. Then, v is substituted for x in e_2. Finally, the resulting expression $e_2[x := v]$ evaluates. Using call-by-value, Program 1.3 outputs `hello` and returns 2.

So long as all `let` expressions in a language use the same choice of evaluation order, composition will remain associative. Combining call-by-name and call-by-value in a single language exposes non-associative composition.

To show an example of non-associativity, let us write

$\texttt{let}\ e_1\ \overset{\mathrm{N}}{\texttt{be}}\ x\ \texttt{in}\ e_2$	for call-by-name, and
$\texttt{let}\ e_1\ \overset{\mathrm{V}}{\texttt{be}}\ x\ \texttt{in}\ e_2$	for call-by-value.

Then, consider the program

$$\begin{aligned}&\texttt{let}\ \Big(\texttt{let}\ (\texttt{print("hello"); }1)\ \overset{\mathrm{V}}{\texttt{be}}\ x\ \texttt{in}\\ &\qquad(\texttt{print("world"); }x)\Big)\\ &\overset{\mathrm{N}}{\texttt{be}}\ y\ \texttt{in}\ y+y\end{aligned} \tag{1.4}$$

of the left-associated form (like Program 1.1), and

$$\begin{aligned}&\texttt{let}\ (\texttt{print("hello"); }1)\ \overset{\mathrm{V}}{\texttt{be}}\ x\ \texttt{in}\\ &\Big(\texttt{let}\ (\texttt{print("world"); }x)\ \overset{\mathrm{N}}{\texttt{be}}\ y\ \texttt{in}\ y+y\Big)\end{aligned} \tag{1.5}$$

its reassociation to the right (like Program 1.2). Recall that call-by-name `let` substitutes the entire assigned expression into the body, while call-by-value `let` evaluates it first and then substitutes. We can see that evaluating Program 1.4 will output `helloworldhelloworld`, but Program 1.5 will output `helloworldworld`, and both will return 2. Thus we have two programs that differ only in how they are associated, and yet compute differently due to evaluation order.

It is typical to interpret the semantics of programming languages in *categories*, but this breaks down for modelling non-associative composition. Consider `let` (`let` e_1 `be` x `in` e_2) `be` y `in` e_3 (Program 1.1) and `let` e_1 `be` x `in` (`let` e_2 `be` y `in` e_3) (Program 1.2) to be interpreted in some category $\mathcal{C}$, and assign to each well-typed expression $x : A \vdash e : B$ a morphism $[\![e]\!] : [\![A]\!] \to [\![B]\!]$ in $\mathcal{C}$. Writing composition in diagrammatic order with $⨾$, the question of whether Programs 1.1 and 1.2 compute differently amounts to the putative equation:

$$\Big([\![e_1]\!] ⨾ [\![e_2]\!]\Big) ⨾ [\![e_3]\!] \overset{?}{=} [\![e_1]\!] ⨾ \Big([\![e_2]\!] ⨾ [\![e_3]\!]\Big). \tag{1.6}$$

However, this is trivialized by the associativity of composition in any category $\mathcal{C}$. To model programs where Equation 1.6 should not hold, such as Programs 1.4 and 1.5, we must use a different, more general model. We will use and study Munch-Maccagnoni's *duploids* [19] for modelling non-associativity arising from mixed evaluation order.

For what purpose should we mix evaluation orders? One motivation is that it allows subsuming the theories of both call-by-name and call-by-value into a single paradigm such as *call-by-push-value* [13]. The denotational semantics of call-by-push-value can be modelled "indirectly" by adjunctions [12], generalizing the widespread use of monads for the semantics of call-by-value [17]. Duploids fit into the picture by providing "direct" semantics—as a single category-like structure—for mixed evaluation order. Duploids have a correspondence with adjunctions: every adjunction gives rise to a duploid, and every duploid is equivalent to one that arises this way [19, 20, Ch. I–II].

1.1 Organization and Contributions

The purpose of this dissertation is to develop the theory of duploids using Martin-Löf type theory [16] and from a *univalent point of view* [23]. We[1] formalize some existing and some novel results about duploids within the Rocq [22] library UniMath [10].

The starting point is Munch-Maccagnoni's work on duploids [19, 20, Ch. II], and the existing formalization of univalent categories in UniMath [2, 5, 10].

Our contribution is a formalization of some existing literature on duploids, and some novel contributions specific to univalent foundations. We define what it means for a duploid $\mathcal{D}$ to be *univalent*: that isomorphisms $a \cong b$ in $\mathcal{D}$ (of a suitable kind) render the related objects $a, b : \mathcal{D}$ indistinguishable to the type theory. We formalize two constructions of the duploid arising from an adjunction; the first is Munch-Maccagnoni's own, and the second is a novel construction. We show both to be "univalent," but in different senses. As an overview:

1. In Chapter 2: Background, we start with the context and motivation of duploids in Section 2.1. We recall some preliminary definitions from homotopy type theory/univalent foundations in Section 2.2, and the necessary background on univalent category theory in Section 2.3.

2. In Chapter 3: Unital Magmoids, we introduce *unital magmoids*: categories but without the associativity law. We recall Munch-Maccagnoni's characterisations of polarity, and define what it means for a unital magmoid to be univalent.

3. In Chapter 4: Duploids, we introduce two versions of duploids, *single-sorted duploid* and *split duploids*, and state what it means for each to be univalent.

4. In Chapter 5: Duploids from Adjunctions, we define two versions of *the duploid arising from an adjunction*: the original "oblique duploid" construction of Munch-Maccagnoni [19] which is naturally a split duploid, and a novel "envelope duploid" construction which is a single-sorted duploid. In Section 5.3, we show that these are univalent, in their respective senses, under reasonable conditions.

The formalization diverges from trunk commit `756380729085` and constitutes a change of 64554 lines of code added across 265 files, as counted by Cloc [4]. The source code is online at `https://github.com/eutro/UniMath/tree/39f9a0591d1b`, with the material of this dissertation located in the directory `UniMath/CategoryTheory/Nonassociative/`. The proofs and definitions presented in this text are based on the formalization, but may differ (or be presented differently) for clarity.

Convention 1.1. Technical terms are set in *italics* when introduced, and may be found in the Index. When a theorem or definition has been formalized in UniMath, we shall write the name of its definition in `monospace` font, hyperlinked to the file and line number in the UniMath repository. An example is "**Definition 3.1** (`category`). A *category* … ." Often a theorem or definition is associated with more than one formalized

[1] Me, my notebooks, and my proverbial rubber duck.

definition. In this case one or multiple names may appear next to the statement, with similar or dual names elided with "*etc.*" or "and dual" for brevity: for example "**Definition 3.10** (`is_thunkable`, `is_negative`, *etc.*)."

Convention 1.2. Definitions and theorems may be annotated in the margin with an icon to indicate their novelty.

- means a definition or theorem already present in UniMath.
- means a definition or theorem that is obvious or due to another author, but that has been newly formalized in UniMath as part of this dissertation.
- means novel definitions, statements, or results.

1.2 Related Work

This dissertation is primarily based on Munch-Maccagnoni's "Models of a Non-Associative Composition" [19, 20, Ch. II], which introduces duploids as a mathematical structure. Much of our work is based on formalizing the definitions and proofs of Munch-Maccagnoni in the proof assistant Rocq. The parts that we do not include are the internal language of a duploid, and the *structure theorem* relating the category of duploids to the category of adjunctions. We note that in univalent foundations there is no 1-category of small duploids, for the same reason there is no 1-category of small categories (see Non-example 3.8). The structure theorem for a 2-category of duploids is yet to appear in any literature.

The rest of Munch-Maccagnoni's PhD thesis [20] explores non-associative composition and polarization more broadly, relating it to programs, with first-class continuations/continuation-passing-style translations, and to proofs, with the involutive negation in classical logic. Other work on duploids includes that of Mangel et al. [15] about linear classical logic, which also includes some general theory about duploids. We note that our "correct notion of isomorphism" in a unital magmoid (Definition 3.17) differs from that of Mangel et al. [15], but they do coincide for preduploids (see Remark 3.21 and Corollary 4.14).

Mixed evaluation order is related to Levy's call-by-push-value [13]. Especially pertinent are Levy's "Contextual isomorphisms." Levy defines *contextual isomorphism* of types $\theta : A \cong B$ **syntactically** with respect to an equivalence relation $\sim$ on typed terms. Given a typing judgement with a type hole $\Gamma[-] \vdash C[-]$, such a contextual isomorphism bijectively (and naturally) maps $\sim$-classes of typed terms $\Gamma[A] \vdash t : C[A]$ to $\sim$-classes of typed terms $\Gamma[B] \vdash \theta(t) : C[B]$. Levy shows for call-by-push-value that when $\sim$ includes computation rules, contextual isomorphisms correspond to thunkable (Führmann [8]) isomorphisms for value (positive) types, and linear (Munch-Maccagnoni [19]) isomorphisms for computation (negative) types. This result is similar to our definition of univalence for duploids, and our proofs of univalence in Section 5.3.

For the "univalent" part, this dissertation is based on the study of category theory in univalent foundations initiated by Ahrens et al. in "Univalent categories and the

Rezk completion" [2]. Our work is also based on *op. cit.* in the literal sense that we directly use its Rocq formalization, which is today part of UniMath [10]. There are other works that formulate and formalize some areas of category theory "univalently." Examples include bicategories [1], monoidal categories [26] and monads [25].

Another work is *The Univalence Principle* by Ahrens et al. [3], deriving algorithmically what it means for a large class of mathematical structures to be "univalent." Ahrens et al. mention duploids, but beware that they use a different definition to the one in this dissertation (see Remark 4.9). Still, our definitions of "univalent duploid" (Definitions 4.15 and 4.21) are the same as derived by their procedure.

2 Background

2.1 Semantics and Evaluation Order

It is a well-established tradition in computer science to consider programs as terms in variants of the lambda calculus. In the study of their denotational semantics, we assign to each term a mathematical object that captures its computational behaviour. For the *pure* simply typed lambda calculus, evaluation order does not matter, and it is the *cartesian closed categories* that model their semantics [11]. However, many languages include *(computational) effects* such as input/output, mutable state, or non-termination, so their models must account for evaluation order. Models include monads for call-by-value [17, 18], comonads for call-by-name [9, 12], adjunctions for call-by-push-value [12], and duploids for a mixed ("polarized") evaluation order [19, 20, Ch. I]. In this section we cover this context.

A simply typed lambda calculus term $\Gamma \vdash e : A$ may be interpreted in a cartesian closed category as a morphism $[\![e]\!] : [\![\Gamma]\!] \to [\![A]\!]$ by recursion on the typing judgement [11]. This interpretation is sound and complete with respect to program equivalence. Two terms $\Gamma \vdash e_1, e_2 : A$ are $\beta\eta$-equivalent if and only if their interpretations $[\![e_1]\!], [\![e_2]\!] : [\![\Gamma]\!] \to [\![A]\!]$ are equal in all cartesian closed categories.

Moggi pointed out that the pure lambda calculus and its $\beta\eta$-equivalence are overly simplistic for program equivalence in the presence of effects, and suggested the use of monads to model effectful call-by-value lambda calculi instead [17, 18]. Given a suitable monad $T : \mathcal{C} \to \mathcal{C}$ on a category $\mathcal{C}$, a compuational lambda term $\Gamma \vdash e : A$ can be interpreted as a morphism $[\![e]\!] : [\![\Gamma]\!] \to T\,[\![A]\!]$ in $\mathcal{C}$. Dually, one may use comonads to model call-by-name lambda calculi [12, 19]. Given a suitable comonad $U : \mathcal{D} \to \mathcal{D}$ on a category $\mathcal{D}$, we can interpret effectful call-by-name lambda terms $\Gamma \vdash e : A$ as morphisms $[\![e]\!] : U\,[\![\Gamma]\!] \to [\![A]\!]$.

Moggi's interpretation into monads involves a sort of "compilation" of lambda terms into a monadic style, inserting monad operations—units and Kleisli extensions, often called "return" and "bind"—that were not present in the source. In the parlance of Führmann [8], monads constitute *indirect semantics* for the computational lambda calculus, in contrast to *direct semantics* such as cartesian closed categories for the pure lambda calculus. In direct semantics, the syntax is more closely related to the constructs readily available, whereas indirect semantics require an extra transformation step. Führmann [8] provides direct semantics for the computational lambda calculus, which we call *thunk-force categories* [19]. These interpret terms $x : A \vdash e : B$ as morphisms $[\![e]\!] : [\![A]\!] \to [\![B]\!]$ in a single category of computations. All monads give rise to a thunk-force category by its *Kleisli category*, and all thunk-force categories are equivalent to one that arises this way. The call-by-name dual of a thunk-force category is a *runnable monad* [19].

The models mentioned so far presuppose a fixed evaluation order. However, as argued by Levy [13], the combination of call-by-name and call-by-value into a single subsuming paradigm, *call-by-push-value*, obviates the need to reason about both separately. Call-by-push-value has indirect semantics in adjunctions $\varphi_{p,n} : \mathcal{N}[\![Lp, n]\!] \simeq \mathcal{P}[\![p, Rn]\!]$ [12]. The syntactically-distinct value types V and computation types $\underline{C}$ are interpreted in $\mathcal{P}$ and $\mathcal{N}$ respectively. Then, call-by-push-value terms $x : V \vdash e : \underline{C}$, which produce computations from values, are interpreted as "oblique" morphisms $[\![e]\!] : [\![V]\!] \to R\,[\![\underline{C}]\!]$ in $\mathcal{P}$, or equivalently (by φ), $[\![e]\!] : L\,[\![V]\!] \to [\![\underline{C}]\!]$ in $\mathcal{N}$. These semantics are "subsuming": the translation of call-by-name and call-by-value into call-by-push-value, when interpreted in the adjunction semantics, gives rise to the corresponding comonad and monad semantics of the source [12].

Munch-Maccagnoni introduces direct semantics for mixed evaluation order in the form of *duploids* [19]. The goal is still to interpret terms $x : A \vdash e : B$ as morphisms $[\![e]\!] : [\![A]\!] \to [\![B]\!]$ in a single category-like structure. However, due to the non-associativity of composition with mixed call-by-name and call-by-value, duploids cannot require associativity like categories do. Instead, duploids are instances of *unital magmoids*: categories but without the associativity law.

Duploids are closely related to adjunctions, in that each adjunction $L \dashv R$ gives rise to a duploid $\mathcal{D}$, and all duploids are equivalent to one that arises this way [19]. Moreover, this duploid $\mathcal{D}$ contains as subcategories—that is, as unital sub-magmoids of $\mathcal{D}$ which are associative—the Kleisli and co-Kleisli categories associated to the adjunction. In this way, duploids subsume both monads for call-by-value and comonads for call-by-name.

Although the background and context of this work lies in the semantics of programming languages, they will play little role beyond this section. Instead, we now focus our attention entirely on the mathematical structures themselves.

2.2 Preliminary Type Theory

In the following sections, the theorems, definitions, and proofs are presented in ordinary mathematical English, but they are also assigned meaning as types and terms in the language of homotopy type theory. In order to interpret informal mathematical prose precisely in type theory, we adopt the following conventions in this text.

As in the UniMath [10] formalization, we work in Martin-Löf type theory [16] with intensional identity types and no W-types. Unlike the UniMath formalization, we do not assume in this text the existence of universes unless stated. We freely assume function extensionality.

Notation 2.1. Write: $\prod_{x:B} E[x]$ for dependent products, $\sum_{x:B} E[x]$ for dependent sums, $A \times B$ for binary products, $A \amalg B$ for binary sums, and $A \to B$ for non-dependent function types. Given $t_1, t_2 : A$, write $t_1 =_A t_2$ or just $t_1 = t_2$ for the identification type. Judgemental equality is written for types as$A \equiv B$ type or just $A \equiv B$, and for terms as $t_1 \equiv t_2 : A$ or just $t_1 \equiv t_2$.

Notation 2.2. Where a type contains a binary operator such as $\to$, $=$, or $\simeq$, we may use the infix notation $a \xrightarrow{f} b$ to mean $f\colon a \to b$. Sharing expressions in such infix notation, as in $t_1 \overset{p}{=} t_2 \overset{q}{=} t_3$, denotes the independent judgements $p : t_1 = t_2$ and $q : t_2 = t_3$. It may additionally denote the appropriate composition of p and q of the type $t_1 = t_3$.

Notation 2.3. Given a type A, write $\mathtt{iscontr}(A)$ for the proposition that A is contractible, and $\|A\|$ for the propositional truncation of A. For sum types, write $\mathtt{iscontr}\left(\sum_{x:B} E[x]\right)$ as $\exists! x : B.\, E[x]$, then $\left\|\sum_{x:B} E[x]\right\|$ as $\exists x : B.\, E[x]$, and finally $\|A \amalg B\|$ as $A \vee B$.

Convention 2.4. As suggested by Notation 2.3, and unlike the *HoTT Book* [23], we use propositionally truncated interpretations of "there exists" and "or". We use "there is" and "either … or" for the constructive variants. The terms "merely" and "chosen" may also be used respectively for emphasis.

Finally, we briefly recall the univalence axiom and its prerequisites.

Definition 2.5. A *universe* $(\mathcal{U}, \mathsf{El})$ is a type $\mathcal{U}$ of *codes*, along with a decoding $\mathsf{El}(A)$ type for all $A : \mathcal{U}$. We shall abuse notation and use $\mathcal{U}$ to mean the universe $(\mathcal{U}, \mathsf{El})$.

Definition 2.6. For a universe $(\mathcal{U}, \mathsf{El})$, we say that a type A is *$\mathcal{U}$-small* if there is an element $\mathsf{code}(A) : \mathcal{U}$ such that $\mathsf{El}(\mathsf{code}(A)) \equiv A$ type. We shall typically omit El and code, so that $A : \mathcal{U}$ implies A type, and vice-versa when A is $\mathcal{U}$-small. We may also leave the universe implicit or unnamed and simply say that a type is "small."

Definition 2.7. A function $f\colon A \to B$ is an *equivalence*, written $\mathtt{isweq}(f)$,[1] when for all elements $y : B$, there exists a unique $x : A$ with identification $f(x) =_B y$. Thus,

$$\mathtt{isweq}(f) \overset{\text{def}}{\equiv} \prod_{y:B} \exists! x : A.\, f\ x =_B y$$

is the proposition of f being an equivalence. Write $A \simeq B$ for the type of equivalences,

$$A \simeq B \overset{\text{def}}{\equiv} \sum_{f\colon A \to B} \mathtt{isweq}(f).$$

Every equivalence $f : A \simeq B$ has a two-sided inverse $f^{-1}\colon B \to A$ with

$$\prod_{x:A} f^{-1}(f(x)) = x \quad \text{and} \quad \prod_{y:B} f(f^{-1}(y)) = y.$$

Moreover, every function with a two-sided inverse may be upgraded to an equivalence.[2]

[1] The name `isweq`, like most others in this section, comes from Voevodsky's original Foundations library [24]. It may be considered short for "is a weak equivalence," although there is nothing weak about it.

[2] The type of "f has a two-sided inverse" is not as well-behaved as the type "f is an equivalence." Notably, the former fails to be a proposition for types that are not sets. This situation is similar to

Example 2.8. The identity function $\mathsf{id}_A : A \to A$, defined $\mathsf{id}_A\ x \overset{\text{def}}{\equiv} x$, is an equivalence.

Definition 2.9. In every universe $\mathcal{U}$, there is a canonical map

$$\begin{aligned}\texttt{id\_to\_weq} : \prod_{A,B:\mathcal{U}} (A =_{\mathcal{U}} B) \to (A \simeq B), \quad \text{given by} \\ \texttt{id\_to\_weq}\ A\ A\ \mathsf{refl}_A \overset{\text{def}}{\equiv} \mathsf{id}_A.\end{aligned} \tag{2.1}$$

A *univalent universe* $\mathcal{U}$ is one where $\texttt{id\_to\_weq}\ A\ B$ is an equivalence for all $\mathcal{U}$-small types $A, B : \mathcal{U}$. For a given universe $\mathcal{U}$, the *univalence axiom* states that $\mathcal{U}$ is univalent.

Remark 2.10. The univalence axiom for a universe $\mathcal{U}$ provides that all type families $E[A]$ indexed by $A : \mathcal{U}$ are congruent with equivalences: that each $A \simeq B$ lifts to an equivalence $E[A] \simeq E[B]$. However, this principle may also be proven—without univalence—for many type families, such as $\texttt{iscontr}(A)$. We shall therefore not actually need the univalence axiom for the main results of this paper.

2.3 Univalent Categories

Beyond the foundation of mathematics in which we work, the content of this dissertation is about non-associative generalizations of categories. Namely, Munch-Maccagnoni's duploids [19]. Before we can talk about univalent duploids, we should understand categories in the context of univalent foundations.

The authoritative paper on categories in univalent foundations is by Ahrens et al. [2], whose formalization in Rocq is today part of UniMath. An expanded version of that paper is included as Chapter 9 of the *HoTT Book* [23]. The development has since diverged from the paper: we use the names "`category`" and "`univalent_category`" for what used to be "precategory" and "category" respectively.

It is well-known that in a category $\mathcal{C}$, the correct notion of "sameness" between two objects $a, b : \mathcal{C}$ is *isomorphism* $a \cong b$, and not necessarily the foundations' notion of "equality" $a = b$. In set-based foundations, equality is overly restrictive: there may be objects $a \cong b$ that are not equal,[3] and there may be isomorphisms $a \cong a$ that are not the identity.[4] In univalent foundations, the type of identifications $a =_{\mathsf{ob}\,\mathcal{C}} b$ may very well correspond to isomorphisms $a \cong b$. A *univalent category* is one where they do.

When the univalence axiom holds, many naturally-arising categories are univalent. For example, for a univalent universe $\mathcal{U}$, the category $\mathbf{Set}_{\mathcal{U}}$ (or simply $\mathbf{Set}$) of $\mathcal{U}$-small sets $A, B : \texttt{hSet}_{\mathcal{U}}$ and functions $A \to B$ is univalent (`HSET_univalent_category`). As an extension, it was shown by Coquand et al. [7] that the univalence axiom extends to a "large class of algebraic structures." This result implies that categories of set-based algebraic structures such as monoids and rings are univalent too.

Every category $\mathcal{C}$ is weakly equivalent to a univalent category $\mathfrak{R}\mathcal{C}$, in the sense that

adjoint equivalences in category theory.

[3] Say, the sets $\{0, 1\}$ and $\{1, 2\}$.

[4] Say, the set $\mathbb{B}$ under boolean negation.

there is a fully-faithful and essentially surjective functor $\mathfrak{r}_{\mathcal{C}} \colon \mathcal{C} \to \mathfrak{R}\mathcal{C}$. We call the category $\mathfrak{R}\mathcal{C}$ the *Rezk completion* of $\mathcal{C}$. One way to construct the Rezk completion is as the image of the Yoneda embedding $\mathsf{y} \colon \mathcal{C} \to \mathbf{Fn}[\mathcal{C}^{\mathsf{op}}, \mathbf{Set}_{\mathcal{U}}]$ [2].

An informative use-case of univalence is in relation to weak equivalences between categories. It is a theorem, without the axiom of choice, that for a univalent category $\mathcal{C}$ and a category $\mathcal{D}$, a fully-faithful and essentially surjective functor $F \colon \mathcal{C} \to \mathcal{D}$ is an adjoint equivalence. The analogous statement in set-based foundations amounts to the axiom of choice. More generally, objects determined by a universal property in a univalent category $\mathcal{C}$ become literally unique, as do left- and right- adjoints of a fixed functor $F \colon \mathcal{C} \to \mathcal{D}$ to another category $\mathcal{D}$. In other words, universal properties become mere properties and, when they hold, contractible. Thus, univalence has the practical benefit of being able to talk about *the* object (functor, category, *etc.*) satisfying a universal property, with the representative being canonical and choice-free.

The univalence of a category is sensitive to how the category is constructed. A relevant example is the *Kleisli category* of a monad $T \colon \mathcal{C} \to \mathcal{C}$ on a category $\mathcal{C}$. One way to construct the Kleisli category of T is as the category of Kleisli arrows $\mathbf{Kleisli}_{\mathsf{arr}}(\mathcal{C}, T)$. The objects are those of $\mathcal{C}$, and for objects $a, b : \mathcal{C}$, the morphisms are the *Kleisli arrows* $a \to Tb$ in $\mathcal{C}$. Even when $\mathcal{C}$ is univalent, this construction of the Kleisli category is **not** generally univalent: only certain $\mathbf{Kleisli}_{\mathsf{arr}}(\mathcal{C}, T)$-isomorphisms $f \colon a \to Tb$ and $f^{-1} \colon b \to Ta$ actually correspond to those $a \cong b$ in $\mathcal{C}$. Under certain conditions (namely, Definition 5.45), it is the "thunkable" isomorphisms (Definition 3.10) that do [14].

Example 2.11 (Levy [14])**.** Consider the state monad $\mathrm{State}_{\mathbb{N}}\, a \overset{\text{def}}{\equiv} (\mathbb{N} \Rightarrow a \times \mathbb{N})$ on $\mathbf{Set}$. The Kleisli arrows $a \to \mathrm{State}_{\mathbb{N}}\, b$ correspond to set-functions $a \times \mathbb{N} \to b \times \mathbb{N}$. From this, it is easy to see that there is an isomorphism $\mathtt{unit} \cong \mathtt{bool}$ in $\mathbf{Kleisli}_{\mathsf{arr}}(\mathbf{Set}, \mathrm{State}_{\mathbb{N}})$: this is simply an isomorphism $\mathtt{unit} \times \mathbb{N} \cong \mathtt{bool} \times \mathbb{N}$ in $\mathbf{Set}$. However, the type `unit` is certainly not identified with `bool`!

There is a univalent construction of the Kleisli category, however, as the subcategory of free T-algebras in the Eilenberg-Moore category of T (`univalent_kleisli_cat`), $\mathbf{Kleisli}_{\mathsf{alg}}(\mathcal{C}, T)$. This amounts to a change in the **objects** to incorporate the effect of T. Which category out of $\mathbf{Kleisli}_{\mathsf{alg}}(\mathcal{C}, T)$ or $\mathbf{Kleisli}_{\mathsf{arr}}(\mathcal{C}, T)$ to use depends on the point of view on which isomorphisms should cause objects to be identified. In $\mathbf{Kleisli}_{\mathsf{alg}}(\mathcal{C}, T)$, isomorphisms may make use of the effects of T, but in $\mathbf{Kleisli}_{\mathsf{arr}}(\mathcal{C}, T)$ they must be "pure" isomorphisms. It is the non-univalent category $\mathbf{Kleisli}_{\mathsf{arr}}(\mathcal{C}, T)$ that appears as the positive subcategory of each Duploid arising from an Adjunction.

3 Unital Magmoids

In this chapter we study *unital magmoids*, a generalization of categories that drops the associativity of composition. We shall build on unital magmoids to study duploids in subsequent chapters. In Section 3.1, we review some necessary concepts of category theory, and how they generalize to unital magmoids. In Section 3.2, we introduce Munch-Maccagnoni's characterization of *polarity* as the failure of associativity [19, 20]. In Section 3.3, we shall define what a "univalent" unital magmoid is, as a generalization of univalent categories.

3.1 Definitions

Definition 3.1 (`category` [2])**.** A *category* $\mathcal{C}$ consists of the following:

1. A type of *objects*, written $\mathsf{ob}\,\mathcal{C}$ or just $\mathcal{C}$.
2. For each pair of objects $a, b : \mathcal{C}$, a type of *morphisms* from a to b, also known as a *hom-type*, written $\mathcal{C}[\![a, b]\!]$ or $a \to b$.
3. For each object $a : \mathcal{C}$, a distinguished *identity* morphism $\mathsf{id}_a : a \to a$.
4. A *composition* operator $\fatsemi_{\mathcal{C}}$ or simply $\fatsemi$ assigning to each pair of morphisms $f : a \to b$ and $g : b \to c$ a composite $f \fatsemi g : a \to c$ such that:
   a) Composition is *unital*: for $a \xrightarrow{f} b$ we have $\mathsf{id}_a \fatsemi f = f$ and $f \fatsemi \mathsf{id}_b = f$.
   b) Composition is *associative*: for $a \xrightarrow{f} b \xrightarrow{g} c \xrightarrow{h} d$ we have $f \fatsemi (g \fatsemi h) = (f \fatsemi g) \fatsemi h$.
5. $\mathcal{C}$ *has hom-sets*: each hom-type $\mathcal{C}[\![a, b]\!]$ is a set.

Most of Definition 3.1 will be familiar to a mathematician who has come across categories. Indeed, one obtains a typical set-theoretic definition by substituting "class" for "type" in the definition of objects and morphisms. The hom-set law (5) is orthogonal to matters of size, it simply makes identitification $f = g$ of morphisms $f, g : a \to b$ be a proposition. We shall therefore find it convenient to use the language of "equality" for identifications between morphisms.

Definition 3.2 (`unital_magmoid`)**.** A *unital magmoid* or *non-associative category* [15] is a category but without the associativity condition. Explicitly, a unital magmoid $\mathcal{M}$ consists of the same data as a category—objects $\mathsf{ob}\,\mathcal{M}$, hom-types $\mathcal{M}[\![a, b]\!]$, identities id_a, and composition $f \fatsemi g$—such that composition is unital and all hom-types are sets.

We import some definitions of category theory for unital magmoids directly, as much as they are well-defined. We spell out some of these explicitly here.

Definition 3.3 (is_z_isomorphism, z_iso)**.** A morphism $f\colon a \to b$ in a unital magmoid or category *has an inverse* if there is a chosen morphism $f^{-1}\colon b \to a$ with

$$f \mathbin{⨾} f^{-1} = \mathsf{id}_a \quad \text{and} \quad f^{-1} \mathbin{⨾} f = \mathsf{id}_b. \tag{3.1}$$

In this case, we say that f is an *isomorphism*, written $f : a \cong b$.

Remark 3.4 (isaprop_is_z_isomorphism). Inverses are unique in a category. That is, if we have a morphism $f\colon a \to b$ in a category and two morphisms $p, q\colon b \to a$ satisfying Equation 3.1, then we have $p = q$. The proof depends on associativity:

$$p = p \mathbin{⨾} (f \mathbin{⨾} q) = (p \mathbin{⨾} f) \mathbin{⨾} q = q.$$

This result makes the type "f has an inverse" be a proposition in a category. In a unital magmoid this is no longer true in general, but we still use "isomorphism" to mean a morphism with a specified inverse.

Definition 3.5 (is_univalent [2])**.** Let $\mathcal{C}$ be a category. There is a canonical map

$$\texttt{id\_to\_iso} : \prod_{a,b:\mathsf{ob}\,\mathcal{C}} (a =_{\mathsf{ob}\,\mathcal{C}} b) \to (a \cong b), \qquad \text{given by}$$
$$\texttt{id\_to\_iso}\ a\ a\ \mathsf{refl}_a \overset{\text{def}}{\equiv} \mathsf{id}_a.$$

The category $\mathcal{C}$ is *univalent* if $\texttt{id\_to\_iso}\ a\ b$ is an equivalence for all $a, b : \mathcal{C}$.

We do not generalize the definition of "univalent category" to unital magmoids directly. Instead, we shall (in Definition 3.17) use a stronger notion of isomorphism for the univalence condition of unital magmoids, which simplifies to Definition 3.5 when the unital magmoid is a category.

As is typical in category theory, we are not only concerned with unital magmoids, but also with maps between unital magmoids, and maps betwen those maps. We import the concepts of *functors* and *natural transformations* directly, but also provide some necessary generalizations.

Definition 3.6 (functor)**.** A *functor* (or *categorical functor* for emphasis) F between categories or unital magmoids $\mathcal{N}$ and $\mathcal{M}$, written $F : \mathcal{N} \to \mathcal{M}$, consists of:

1. A mapping on objects $F_0\colon \mathsf{ob}\,\mathcal{N} \to \mathsf{ob}\,\mathcal{M}$.
2. For all objects $a, b : \mathsf{ob}\,\mathcal{N}$, a mapping on morphisms $F_1 : \mathcal{N}[\![a, b]\!] \to \mathcal{N}[\![F_0\,a, F_0\,b]\!]$.
3. F preserves:
    a) Identities:

$$F_1\left(a \xrightarrow{\mathsf{id}_a} a\right) = F_0\ a \xrightarrow{\mathsf{id}_{F_0 a}} F_0\ a \qquad \text{for all } a : \mathcal{N}.$$

b) Composition:

$$F_1\left(a \xrightarrow{f} b \xrightarrow{g} c\right) = F_0\, a \xrightarrow{F_1 f} F_0\, b \xrightarrow{F_1 g} F_0\, c \qquad \text{for all } a \xrightarrow{f} b \xrightarrow{g} c \text{ in } \mathcal{N}.$$

Both F_0 and F_1 may simply be written F.

Definition 3.7 (rxfunctor)**.** A *reflexive graph functor* F between unital magmoids or categories $\mathcal{N}$ and $\mathcal{M}$, written $F\colon \mathcal{N} \dashrightarrow \mathcal{M}$, consists of the same data as a categorical functor, but need only preserve identities.

Non-example 3.8. Functors between categories or unital magmoids have identities and composition. We shall write these $\mathsf{Id}_{\mathcal{M}}\colon \mathcal{M} \to \mathcal{M}$, and $GF\colon \mathcal{M}_1 \to \mathcal{M}_3$ (in the opposite order to $⨾$) for $F\colon \mathcal{M}_1 \to \mathcal{M}_2$ and $G\colon \mathcal{M}_2 \to \mathcal{M}_1$. Despite satisfying the unitality and associativity laws of a category, $\mathcal{U}$-small categories and functors between them (for some universe $\mathcal{U}$) do **not** themselves form a category $\mathbf{Cat}_{\mathcal{U}}$. This is because the type of functors does not in general form a set.[1] The situation would be similar for unital magmoids and derived structures like duploids. One may restrict to *strict categories* $\mathbf{Cat}_{\mathtt{hSet}_{\mathcal{U}}}$, categories $\mathcal{C}$ whose objects $\mathsf{ob}\,\mathcal{C}$ form a set, but this privileges isomorphisms of categories over adjoint equivalence. One may wish to consider $\mathbf{Cat}_{\mathcal{U}}$ as a 2-category instead [1].

Definition 3.9 (nat_trans)**.** A *natural transformation* α between categorical or reflexive graph functors $F, G\colon \mathcal{N} \dashrightarrow \mathcal{M}$, written $\alpha\colon F \Rightarrow G$, consists of:

1. For each object $a : \mathcal{N}$, a morphism $\alpha_a\colon Fa \to Ga$.
2. *Naturality*: for all objects $a, b : \mathcal{N}$ and morphisms $f\colon a \to b$, the equation $\alpha_a ⨾ Gf = Ff ⨾ \alpha_b$ holds. That is, the *naturality square*

$$\begin{array}{ccc} Fa & \xrightarrow{Ff} & Fb \\ {\scriptstyle \alpha_a}\downarrow & & \downarrow{\scriptstyle \alpha_b} \\ Ga & \xrightarrow{Gf} & Gb \end{array} \qquad \text{in } \mathcal{M} \text{ commutes.} \tag{3.2}$$

3.2 Polarities

Many theorems about categories which fail to generalize to unital magmoids can be salvaged by hypothesizing associativity of composites involving certain objects or morphisms. In this section, we define the concept of linearity, thunkability, and polarization, or rather their characterizations in unital magmoids due to Munch-Maccagnoni [19]. We also coin the term *intermediate* for a less-studied notion similar to linearity and thunkability. We find use for it to define univalence of unital magmoids in Section 3.3.

[1] Consider the two natural isomorphisms from $\mathbf{1} \xrightarrow{\mathtt{bool}} \mathbf{Set}$ to itself, when $\mathbf{Set}$ is univalent.

Definition 3.10 (`is_thunkable`, `is_negative`, *etc.* [19])**.** A path of morphisms

$$a_0 \xrightarrow{f_1} a_1 \xrightarrow{f_2} a_2 \to \cdots \to a_{n-1} \xrightarrow{f_n} a_n \tag{3.3}$$

in a unital magmoid *associates* when all ways of parenthesizing their composite $f_1 \fatsemi f_2 \fatsemi \cdots \fatsemi f_n$ ($= \mathsf{id}_{a_0}$ if empty) are equal. In the simplest non-trivial case, a triple of morphisms $a \xrightarrow{f} b \xrightarrow{g} c \xrightarrow{h} d$ associates if and only if we have $f \fatsemi (g \fatsemi h) = (f \fatsemi g) \fatsemi h$.

We say that:

- f is *thunkable* if all triples (f, g, h) associate.
- g is *intermediate* if all triples (f, g, h) associate.
- h is *linear* if all triples (f, g, h) associate.
- b is (*semantically*) *negative* if all morphisms into b are thunkable. Equivalently, if all morphisms out of b are intermediate.
- c is (*semantically*) *positive* when all morphisms out of c are linear. Equivalently, if all morphisms into c are intermediate.

These are depicted pictorially in Figure 3.1.

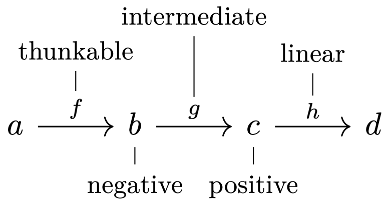

Figure 3.1 The triple (f, g, h) associates if any of the labelled properties holds.

Notation 3.11. When a path of morphisms as in Diagram 3.3 is known to associate, we may write its composite without parentheses as $f_1 \fatsemi f_2 \fatsemi \cdots \fatsemi f_n$. Following Notation 2.2, we may use the diagram $a_0 \xrightarrow{f_1} a_1 \xrightarrow{f_2} a_2 \to \cdots \to a_{n-1} \xrightarrow{f_n} a_n$ itself for the composite.

Remark 3.12. Evidently, thunkability and linearity are formally dual, as are negativity and positivity, while intermediateness is self-dual.

Definition 3.13 (`linear_category`, `negative_category`, *etc.* [19])**.** Let $\mathcal{M}$ be a unital magmoid. The following are subcategories of $\mathcal{M}$—that is, unital sub-magmoids of $\mathcal{M}$ which are associative.

- The *linear subcategory* $\mathcal{M}_l$ of linear morphisms.
- The *thunkable subcategory* $\mathcal{M}_t$ of thunkable morphisms.

- The *intermediate subcategory* $\mathcal{M}_i$ of intermediate morphisms.
- The (*semantically*) *negative subcategory* $\mathcal{M}^\ominus$ of semantically negative objects.
- The (*semantically*) *positive subcategory* $\mathcal{M}^\oplus$ of semantically positive objects.
- Combinations of the above, notably:
  - The *linear-and-thunkable subcategory* $\mathcal{M}_{lt}$ of linear and thunkable morphisms.
  - The *linear-and-thunkable-and-intermediate subcategory* $\mathcal{M}_{lti}$.
  - The *negative-linear subcategory* $\mathcal{M}_l^\ominus$.
  - The *positive-thunkable subcategory* $\mathcal{M}_t^\oplus$.

Remark 3.14. By the property of polarization, all morphisms of $\mathcal{M}^\ominus$ and $\mathcal{M}_l^\ominus$ are thunkable and intermediate, and all morphisms of $\mathcal{M}^\oplus$ and $\mathcal{M}_t^\oplus$ are linear and intermediate.

Notation 3.15. We write the sub- and super-scripts from Definition 3.13 onto the binary operators of morphisms ($\to$) or isomorphisms ($\cong$) to denote that the morphism or isomorphism (and its inverse) resides in the corresponding subcategory of the unital magmoid.

Convention 3.16. Phrasing such as "linear-and-thunkable isomorphism" has the meaning suggested by Notation 3.15. That is, a linear-and-thunkable isomorphism $f : a \cong_{lt} b$ has that both $f : a \to b$ and its inverse $f^{-1} : b \to a$ are linear as well as thunkable.

3.3 Univalence

In this section we define what it means for a unital magmoid to be *univalent*. Definition 3.17 defines a sort of "internal" univalence condition that describes identifications of objects within the unital magmoid. We also state an "external" type of univalence that describes identifications between unital magmoids. That identifications between unital magmoids correspond to isomorphisms of unital magmoids, and isomorphisms between univalent unital magmoids correspond to weak equivalences.

★ **Definition 3.17** (`is_unital_magmoid_univalent`)**.** A unital magmoid $\mathcal{M}$ is *univalent* if the linear-and-thunkable-and-intermediate subcategory $\mathcal{M}_{lti}$ is univalent.

Remark 3.18. Since all paths in a category $\mathcal{C}$ associate, $\mathcal{C}_{lti}$ is isomorphic to $\mathcal{C}$. Thus Definition 3.17 corresponds to Definition 3.5 in categories (see Theorem 3.25 below).

Remark 3.19. The idea of Definition 3.17 is to capture which isomorphisms are sufficiently well-behaved to correspond to identifications. To justify it, observe that `id_to_iso` $a\ b\ p : a \cong b$ shares all properties with identities id_a, by induction on $p : a = b$. In particular, it must be linear, thunkable, and intermediate; so we require these in Definition 3.17. Theorem 3.20 justifies this to be sufficient.

★ **Theorem 3.20.** *In a unital magmoid, linear-and-thunkable-and-intermediate isomorphisms $a \cong_{lti} b$ correspond to the indiscernibilities $a \asymp b$ of Ahrens et al. [3].* □

Remark 3.21. Linear-and-thunkable isomorphisms are **not** always intermediate in a unital magmoid.[2] However, they are in the presence of negative or positive objects (see Theorem 4.13).

Definition 3.22 (`fully_faithful`)**.** A functor $F\colon \mathcal{N} \to \mathcal{M}$ is *fully-faithful* if its action on morphisms $F_1 : \mathcal{N}[\![a, b]\!] \to \mathcal{M}[\![Fa, Fb]\!]$ is an equivalence for all $a, b : \mathcal{N}$.

Definition 3.23 (`catiso`)**.** A functor $F\colon \mathcal{N} \to \mathcal{M}$ between unital magmoids or categories is an *isomorphism of unital magmoids* or *of categories*, written $F : \mathcal{N} \cong \mathcal{M}$, if it is fully faithful, and $F_0 \colon \mathsf{ob}\, \mathcal{N} \to \mathsf{ob}\, \mathcal{M}$ is an equivalence (of types).

Remark 3.24 (`weq_catiso_precategory_data_path'`). When unital magmoids or categories $\mathcal{N}$ and $\mathcal{M}$ are $\mathcal{U}$-small for a **univalent** universe $\mathcal{U}$, isomorphisms $\mathcal{N} \cong \mathcal{M}$ are equivalent to identifications $\mathcal{N} = \mathcal{M}$, regardless of univalence of $\mathcal{N}$ and $\mathcal{M}$. In line with Remark 2.10, however, we derive the following directly.

Theorem 3.25 (`weq_is_univalent_over_catiso`)**.** *If $F : \mathcal{C} \cong \mathcal{D}$ is an isomorphism of categories, then $\mathcal{C}$ is univalent if and only if $\mathcal{D}$ is.* □

Definition 3.26 (`is_lti_essentially_surjective`)**.** Let $F\colon \mathcal{N} \to \mathcal{M}$ be a functor between unital magmoids. For $b : \mathcal{M}$, the type $\mathsf{wfib}_F(b) \stackrel{\text{def}}{\equiv} \sum_{a:\mathcal{N}} Fa \cong_{lti} b$ is called the (*weak*) *fibre* of F over b. The functor F is *essentially surjective* when all its fibres are inhabited, and *split essentially surjective* when all its fibres have a chosen element.

Definition 3.27. A categorical functor $F\colon \mathcal{N} \to \mathcal{M}$ between unital magmoids is a *weak equivalence* if it is fully faithful and essentially surjective, and a *strong equivalence* if it is fully faithful and split essentially surjective.

Theorem 3.28 (`is_catiso_from_weak_um_equiv`)**.** *All weak equivalences between* ***univalent*** *unital magmoids are isomorphisms of unital magmoids.* □

[2]Consider the free unital magmoid generated by morphisms $a \xrightarrow{f} b \xrightarrow{g} c$ and a linear-and-thunkable isomorphism $p : b \cong_{lt} d$, formalized as `lini`.

4 Duploids

In this chapter we introduce *duploids*: unital magmoids equipped with *polarity shifts*. In Section 4.1, we introduce two variants of duploid: a more recent definition that treats polarities as a property [6, 15], and one closer to the original that treats polarities as structure [19]. We coin the terms *single-sorted duploid* and *split duploid* for the two. The purpose of considering both in this dissertation is that they have different univalence conditions, and thus different constructions of the Duploid from an Adjunction. In Section 4.2 we define what it means for a duploid to be univalent, and show that it makes polarity shifts a property rather than structure.

4.1 Definitions and Properties

Definition 4.1 (`preduploid` [19, 6, 15])**.** A *preduploid* (or *single-sorted preduploid*) is a unital magmoid such that all objects are semantically negative or positive.

Definition 4.2 (`duploid` [19, 6, 15])**.** A *duploid* (or *single-sorted duploid*) is a preduploid $\mathcal{D}$ equipped with *polarity shifts*:

1. *Negative shifts* (`has_negative_shifts`). For every object $a : \mathcal{D}$,
   a) its *upshift*, a semantically negative object written $\Uparrow a : \mathcal{D}$,
   b) a linear morphism $\texttt{force}_a : \Uparrow a \to_l a$,
   c) a linear-and-thunkable inverse for $\texttt{force}_a$ called $\texttt{delay}_a : a \to_{lt} \Uparrow a$.
2. *Positive shifts* (`has_positive_shifts`). Dually, for every object $a : \mathcal{D}$,
   a) its *downshift*, a semantically positive object written $\Downarrow a : \mathcal{D}$,
   b) a thunkable morphism $\texttt{wrap}_a : a \to \Downarrow a$,
   c) a linear-and-thunkable inverse for $\texttt{wrap}_a$ called $\texttt{unwrap}_a : \Downarrow a \to_{lt} a$.

Remark 4.3 (`upshift_'_to_negative`, `force_'`, `force_negative_linear`, *etc.* [19]). Polarity shifts extend to reflexive graph functors $\Uparrow : \mathcal{D} \dashrightarrow \mathcal{D}^{\ominus}$ and $\Downarrow : \mathcal{D} \dashrightarrow \mathcal{D}^{\oplus}$, as

$$\Uparrow\left(a \xrightarrow{f} b\right) \overset{\text{def}}{\equiv} \Uparrow a \xrightarrow{\texttt{force}_a}_l a \xrightarrow{f} b \xrightarrow{\texttt{delay}_b}_{lt} \Uparrow b \quad \text{and}$$

$$\Downarrow\left(a \xrightarrow{g} b\right) \overset{\text{def}}{\equiv} \Downarrow a \xrightarrow{\texttt{unwrap}_a}_{lt} a \xrightarrow{g} b \xrightarrow{\texttt{wrap}_b}_t \Downarrow b.$$

These may be restricted and co-restricted to **categorical** functors $\Uparrow : \mathcal{D}_l \to \mathcal{D}_l^{\ominus}$ and $\Downarrow : \mathcal{D}_t \to \mathcal{D}_t^{\oplus}$ that take part in various adjunctions [19], which we will not detail here.

Definition 4.4 (split_preduploid)**.** A *split preduploid* is a unital magmoid $\mathcal{D}$ equipped with a chosen *polarity mapping* $\omega\colon \mathsf{ob}\,\mathcal{D} \to \{\ominus, \oplus\}$. We say that an object a is *syntactically negative* if $\omega(a) = \ominus$ and *syntactically positive* if $\omega(a) = \oplus$, and require that every **syntactically** negative/positive object is also **semantically** negative/positive.

Definition 4.5 (split_duploid)**.** Let $\mathcal{D}$ be a single-sorted duploid with a polarity mapping that makes it a split preduploid. Then $\mathcal{D}$ is a *split duploid* when all $\Uparrow a / \Downarrow a$ are syntactically negative/positive.

Remark 4.6 (polarity_mapping_to_has_polarities, polarity_mapping_from_LEM). Every split preduploid is evidently a single-sorted preduploid. Conversely, the law of excluded middle allows defining a polarity mapping for any **preduploid**. For example,

$$\omega(a) \overset{\text{def}}{\equiv} \begin{cases} \ominus & \text{if } a \text{ is semantically negative,} \\ \oplus & \text{otherwise.} \end{cases} \tag{4.1}$$

The law of excluded middle does not help for defining a polarity mapping for **duploids**, as ω in Equation 4.1 will not respect the polarity shifts in general.

Example 4.7. A category $\mathcal{C}$ can be viewed as a duploid with polarity shifts $\langle \Uparrow a, \mathtt{force}_a \rangle \overset{\text{def}}{\equiv} \langle a, \mathsf{id}_a \rangle$ and $\langle \Downarrow a, \mathtt{wrap}_a \rangle \overset{\text{def}}{\equiv} \langle a, \mathsf{id}_a \rangle$. If an object $a : \mathcal{C}$ exists, then $\mathcal{C}$ has no definable shift-respecting polarity mapping.

Definition 4.8 (chosen_negative_category, and dual)**.** Let $\mathcal{D}$ be a split preduploid with polarity mapping $\omega\colon \mathsf{ob}\,\mathcal{D} \to \{\ominus, \oplus\}$. Extending Definition 3.13, there are the following subcategories of $\mathcal{D}$.

- The *syntactically negative subcategory* $\mathcal{D}^{\ominus\omega}$ of syntactically negative objects.
- The *syntactically positive subcategory* $\mathcal{D}^{\oplus\omega}$ of syntactically positive objects.
- Combinations with those of Definition 3.13, notably:
  - The *syntactically negative-linear subcategory* $\mathcal{D}_l^{\ominus\omega}$.
  - The *syntactically positive-thunkable subcategory* $\mathcal{D}_t^{\oplus\omega}$.

Remark 4.9. Definition 4.1 makes the polarities in a preduploid be a mere property on unital magmoids (recall Convention 2.4), rather than structure as Munch-Maccagnoni [19] originally defined it. The original "two-sorted" duploids of *op. cit.* are precisely the split duploids $\mathcal{D}$ where $\langle \Uparrow a, \mathtt{force}_a \rangle =_{\{a\}^{\Leftarrow}_{\mathcal{D}_l^{\ominus\omega}}} \langle a, \mathsf{id}_a \rangle$ for syntactically negative $a : \mathcal{D}$, and dually $\langle \Downarrow b, \mathtt{wrap}_b \rangle =_{\{b\}^{\Rightarrow}_{\mathcal{D}_l^{\ominus\omega}}} \langle b, \mathsf{id}_b \rangle$ for syntactically positive $b : \mathcal{D}$.

We note the following characterizations of polarity in a duploid.

Theorem 4.10 (is_thunkable_iff_delay_wrap, and dual [19])**.** *In a duploid, a morphism $f\colon a \to b$ is thunkable if and only if the triple*

$$a \xrightarrow{f} b \xrightarrow{\mathtt{delay}_b}_{lt} \Uparrow b \xrightarrow{\mathtt{wrap}_{\Uparrow b}}_{ti} \Uparrow\Downarrow b \quad \textit{associates.}$$

Dually, $f\colon a \to b$ is linear if and only if the triple

$$\Uparrow\Downarrow a \xrightarrow{\mathtt{force}_{\Downarrow a}}_{li} \Downarrow a \xrightarrow{\mathtt{unwrap}_a}_{lt} a \xrightarrow{f} b \quad \textit{associates.}$$

Proof. See Proposition 13 of Munch-Maccagnoni, 2014 [19]. □

Theorem 4.11 (`is_positive_of_lt_iso`)**.** *Let $p : a \cong_{lt} b$ be a linear-and-thunkable isomorphism in a unital magmoid. The object a is positive if and only if b is positive, and a is negative if and only if b is negative.* □

Corollary 4.12 (`lt_iso_upshift_of_negative`, and dual)**.** *For an object $a : \mathcal{D}$ in a duploid $\mathcal{D}$, the following are equivalent:*

1. *a is negative.*
2. *$\mathtt{force}_a : \Uparrow a \to_l a$ is thunkable.*
3. *$\mathtt{force}_a$ is a linear-and-thunkable isomorphism $\Uparrow a \cong_{lt} a$, with inverse $\mathtt{delay}_a$.*

Dually, for $a : \mathcal{D}$ again, the following are equivalent.

A. *a is positive.*

B. *$\mathtt{wrap}_a : a \to_t \Downarrow a$ is linear.*

C. *$\mathtt{wrap}_a$ is a linear-and-thunkable isomorphism $a \cong_{lt} \Downarrow a$ with inverse $\mathtt{unwrap}_a$.*

4.2 Univalence

Since single-sorted preduploids are just unital magmoids with an extra property, their univalence condition should be the same. However, we can simplify the notion of isomorphism used. We also find that univalence of preduploids has a useful characterization in terms of the polarized subcategories.

Theorem 4.13 (`lt_iso_is_intermediate_from_polarized`)**.** *If an object a in a unital magmoid is negative or positive, then all linear-and-thunkable isomorphisms $p : a \cong_{lt} b$ are intermediate.*

Proof. The objects a and b share a polarity by Theorem 4.11. Intermediateness of p is immediate. □

Corollary 4.14 (`preduploid_lt_iso_is_intermediate`)**.** *Every linear-and-thunkable isomorphism $p : a \cong_{lt} b$ in a preduploid is intermediate.*

Definition 4.15 (`is_duploid_univalent`)**.** A duploid or preduploid $\mathcal{D}$ is *univalent* (*as a single-sorted (pre)duploid*) if its linear-and-thunkable subcategory $\mathcal{D}_{lt}$ is univalent.

Remark 4.16 (`unital_magmoid_univalent_iff_duploid_univalent`). Definition 4.15 coincides with univalence of $\mathcal{D}$ as a unital magmoid by Corollary 4.14.

We use a specialization of Rijke's *fundamental theorem of identity types* [21] for the following proof.[1]

Definition 4.17 (edges_from, edges_to)**.** For an object $a : \mathcal{C}$ in a category $\mathcal{C}$, the type

$$\{a\}_{\mathcal{C}}^{\rightrightarrows} \overset{\text{def}}{\equiv} \sum_{b:\mathcal{C}} a \cong b \qquad \text{is called the } \textit{outward fan} \text{ of } a\text{, and}$$

$$\{a\}_{\mathcal{C}}^{\leftleftarrows} \overset{\text{def}}{\equiv} \sum_{b:\mathcal{C}} b \cong a \qquad \text{is called the } \textit{inward fan} \text{ of } a.$$

Theorem 4.18 (is_rxgraph_univalent_from_isaprop_edges_from [21, Ch. 11])**.** *Let $\mathcal{C}$ be a category. The following are equivalent.*

1. *The category $\mathcal{C}$ is univalent.*
2. *All outward (dually, inward) fans of $\mathcal{C}$ are propositions.*
3. *All outward (dually, inward) fans $\{a\}_{\mathcal{C}}^{\rightrightarrows}$ are contractible with centre $\langle a, \mathsf{id}_a \rangle : \{a\}_{\mathcal{C}}^{\rightrightarrows}$.*

Theorem 4.19 (is_duploid_univalent_from_positive_thunkable_and_negative_-linear_categories)**.** *A preduploid $\mathcal{D}$ is univalent if and only if the subcategories $\mathcal{D}_t^{\oplus}$ and $\mathcal{D}_l^{\ominus}$ are univalent.*

Proof. The "only if" direction follows from fully-faithfulness of the inclusion functors $\mathcal{D}_t^{\oplus} \hookrightarrow \mathcal{D}_{lt} \hookleftarrow \mathcal{D}_l^{\ominus}$. For the "if" direction, we apply Theorem 4.18 and show that every fan $\{a\}_{\mathcal{D}}^{\rightrightarrows}$ is a proposition. Using the preduploid property, if $a : \mathcal{D}$ is negative, the fan

$$\{a\}_{\mathcal{D}}^{\rightrightarrows} \equiv \sum_{b:\mathcal{D}} a \cong_{lt} b \qquad \text{is equivalent to the fan}$$

$$\{a\}_{\mathcal{D}_l^{\ominus}}^{\rightrightarrows} \equiv \sum_{b:\mathcal{D}_l^{\ominus}} a \cong_l^{\ominus} b \qquad \text{by Theorem 4.11.}$$

The latter is a proposition by assumption. Dually, $\{a\}_{\mathcal{D}}^{\rightrightarrows} \simeq \{a\}_{\mathcal{D}_t^{\oplus}}^{\rightrightarrows}$ if a is positive. □

For split (pre)duploids, Definition 4.15 is incorrect in general, due to the fact that objects may be both negative and positive **semantically**, but not **syntactically**. Instead, we use a syntactic variant of Theorem 4.19 as the definition of univalence for split duploids.

Theorem 4.20 (neg_is_duploid_univalent_if_split)**.** *Let $\mathcal{D}$ be a split duploid. If there is an object $a : \mathcal{D}$ which is both semantically negative and semantically positive, then the underlying single-sorted duploid of $\mathcal{D}$ is not univalent.*

Proof. Univalence applied to $\Uparrow a \cong_{lt} \Downarrow a$ would create $\ominus = \omega(\Uparrow a) = \omega(\Downarrow a) = \oplus$, which is a contradiction. □

[1] The formalization and Rijke both make a more general statement about "reflexive graphs" and "identity systems" respectively, but we do not develop the language for those statements in this dissertation.

Definition 4.21 (`is_split_duploid_univalent`)**.** A split duploid or preduploid $\mathcal{D}$ is *univalent* (*as a split duploid*) if both its syntactically negative-linear subcategory $\mathcal{D}_l^{\ominus\omega}$ and and its syntactically positive-thunkable subcategory $\mathcal{D}_t^{\oplus\omega}$ are univalent.

Remark 4.22. Definition 4.21 coincides with that derived for two-sorted duploids by Ahrens et al. [3, Example 13.5].

Univalence makes certain choices unique when they might not be in set-theoretic foundations. We do not yet have a general principle of uniqueness for universal properties in preduploid, but we can prove it directly for the polarity shifts.

Theorem 4.23 (`isaprop_has_polarity_shifts`)**.** *For a univalent preduploid $\mathcal{D}$, the type "$\mathcal{D}$ is a duploid" is a proposition.*

Proof. Suppose there are two negative shifts $\langle \Uparrow, \mathtt{force} \rangle$ and $\langle \Uparrow', \mathtt{force}' \rangle$ for $\mathcal{D}$. Then for all $a : \mathcal{D}$, there is a linear isomorphism $I \stackrel{\text{def}}{\equiv} \mathtt{force}_a \mathbin{⨾} \mathtt{delay}'_a : \Uparrow a \simeq_l \Uparrow' a$, which is also thunkable by the polarities of $\Uparrow a$ and $\Uparrow' a$. Moreover, $\mathtt{force}_a : \Uparrow a \to a$ transported across I, as $I^{-1} \mathbin{⨾} \mathtt{force}_a : \Uparrow' a \to a$, is equal to $\mathtt{force}'_a : \Uparrow' a \to a$. It follows from univalence of $\mathcal{D}$ and Theorem 4.18 that $\langle \Uparrow' a, I \rangle =_{\{\Uparrow a\}^{\rightrightarrows}_{\mathcal{D}_{lt}}} \langle \Uparrow a, \mathsf{id}_{\Uparrow a} \rangle$, and hence $\langle \Uparrow, \mathtt{force} \rangle = \langle \Uparrow', \mathtt{force}' \rangle$. The case is analogous for positive shifts. □

Remark 4.24 (`duploid_eq_weq_duploid_equivalence`). It follows from Theorem 4.23 that univalent single-sorted duploids carry no extra structure compared to unital magmoids. Thus, isomorphisms of (the underlying unital magmoids of) duploids satisfy the same external univalence property described in Remark 3.24.

5 Duploids from Adjunctions

Duploids are related to adjunctions: every adjunction gives rise to a duploid, and every duploid is equivalent to one that arises from an adjunction. In Section 5.1 we construct the split duploid arising from an adjunction, and in Section 5.2 we construct the single-sorted one. We also prove the single-sorted version of the theorem, and show the two constructions to be equivalent as unital magmoids. In Section 5.3 we show the univalence of both constructions.

Recall that an adjunction $L \dashv R : \mathcal{N} \to \mathcal{P}$ may be viewed as an isomorphism of sets:

$$\mathcal{N}[\![Lp, n]\!] \overset{\varphi_{p,n}}{\simeq} \mathcal{P}[\![p, Rn]\!] \qquad \text{natural in } p : \mathcal{P}^{\mathsf{op}} \text{ and } n : \mathcal{N}.$$

Following Levy [12], we call the morphisms on either side of $\varphi_{p,n}$ the *oblique morphisms*, written $\mathcal{O}_{L \dashv R}[\![p, n]\!]$ or just $\mathcal{O}[\![p, n]\!]$. Rather than making an arbitrary choice of the side, however, we use the following symmetric definition:

Definition 5.1 (`oblique_mor`)**.** Let $\varphi_{p,n} : \mathcal{N}[\![Lp, n]\!] \simeq \mathcal{P}[\![p, Rn]\!]$ be the natural isomorphism witnessing an adjunction $L \dashv R$. An *oblique morphism* $f : \mathcal{O}[\![p, n]\!]$ consists of a *negative mate* $f^\flat : \mathcal{N}[\![Lp, n]\!]$ and a *positive mate* $f^\sharp : \mathcal{P}[\![p, Rn]\!]$, satisfying $\varphi_{p,n}(f^\flat) = f^\sharp$.

In this chapter, we describe two duploids whose morphisms shall be given by oblique morphisms. But what should the objects be? A first approach may be to use objects from either $\mathcal{N}$ or $\mathcal{P}$.

Definition 5.2 (`oblique_ob`)**.** An *oblique object* $a : \mathcal{O}[L \dashv R]$ is either $\iota^\ominus_\mathcal{O}\, n$ for $n : \mathcal{N}$ or $\iota^\oplus_\mathcal{O}\, p$ for $p : \mathcal{P}$. For an oblique object a, define the *negative* and *positive projections*, $\pi^\ominus_\mathcal{O}\, a : \mathcal{N}$ and $\pi^\oplus_\mathcal{O}\, a : \mathcal{P}$ in the obvious way, as

$$\pi^\ominus_\mathcal{O}(\iota^\ominus_\mathcal{O}\, n) \overset{\text{def}}{\equiv} n \qquad\qquad \pi^\oplus_\mathcal{O}(\iota^\ominus_\mathcal{O}\, n) \overset{\text{def}}{\equiv} Rn$$
$$\pi^\ominus_\mathcal{O}(\iota^\oplus_\mathcal{O}\, p) \overset{\text{def}}{\equiv} Lp \qquad\qquad \pi^\oplus_\mathcal{O}(\iota^\oplus_\mathcal{O}\, p) \overset{\text{def}}{\equiv} p.$$

The morphisms between oblique objects a and b are defined as $\mathcal{O}[\![\pi^\oplus_\mathcal{O}\, a, \pi^\ominus_\mathcal{O}\, b]\!]$. This is the start of the definition of the *oblique duploid* (Section 5.1).

In light of Theorem 4.20, this approach has an unfortunate deficiency: it creates extra structure by making a polarity mapping $\omega(\iota^\varepsilon_\mathcal{O}(x)) \overset{\text{def}}{\equiv} \varepsilon : \{\ominus, \oplus\}$ definable.[1] For

[1]The "extra structure" becomes apparent if the type $\mathcal{N} \amalg \mathcal{P}$ is expressed as $\sum_{\varepsilon : \{\ominus,\oplus\}} O(\varepsilon)$ for $O(\ominus) \overset{\text{def}}{\equiv}$ $\mathsf{ob}\,\mathcal{N}$ and $O(\oplus) \overset{\text{def}}{\equiv} \mathsf{ob}\,\mathcal{P}$. The polarity mapping ω is then simply the first projection. From this perspective, the polarity is literally an extra data field!

an alternative definition without a definable polarity mapping, we shall need to use different objects.

Let us consider what the objects really need. Reasoning backwards, to define morphisms $a \to b$ in terms of $\mathcal{O}[\![p, n]\!]$, we shall need to associate some $p : \mathcal{P}$ and some $n : \mathcal{N}$ to all objects a, b. Next, in order to uniformly define identities, it stands to reason that the $p : \mathcal{P}$ and $n : \mathcal{N}$ associated to an object a ought to be related by a morphism $\mathcal{O}[\![p, n]\!]$ that will become id_a. We call the data $p : \mathcal{P}$, $n : \mathcal{N}$ and $f : \mathcal{O}[\![p, n]\!]$ an *envelope pre-object*; f will later need an extra property to behave like an identity.

Definition 5.3. An *envelope pre-object* a in $L \dashv R : \mathcal{N} \to \mathcal{P}$ is a triple of an object $\pi^{\ominus}_{\mathcal{E}}\, a : \mathcal{N}$, an object $\pi^{\oplus}_{\mathcal{E}}\, a : \mathcal{P}$, and a morphism $\mathsf{cross}_{\mathcal{E}}\, a : \mathcal{O}[\![\pi^{\oplus}\, a, \pi^{\ominus}\, a]\!]$. We may write an envelope pre-object a as simply the diagram of its oblique morphism:

$$\pi^{\oplus}_{\mathcal{E}}\, a \xrightarrow{\mathsf{cross}_{\mathcal{E}}\, a} \pi^{\ominus}_{\mathcal{E}}\, a.$$

The morphisms between two envelope pre-objects a and b, are given by $\mathcal{O}[\![\pi^{\oplus}_{\mathcal{E}}\, a, \pi^{\ominus}_{\mathcal{E}}\, b]\!]$.

Notation 5.4. The operator cross should be read with a higher precedence than $\flat$ and $\sharp$. That is, the following all mean the same thing:

$$(\mathsf{cross}\, a)^{\flat} \qquad \mathsf{cross}(a)^{\flat} \qquad \mathsf{cross}\, a^{\flat}.$$

Definition 5.5 (oblique_identity)**.** To tie envelope pre-objects back to the oblique objects, we may associate to each oblique object $a : \mathcal{O}[L \dashv R]$ an envelope pre-object $\hat{a}$ as

$$\left(\pi^{\oplus}_{\mathcal{E}}\, \hat{a} \xrightarrow{\mathsf{cross}_{\mathcal{E}}\, \hat{a}} \pi^{\ominus}_{\mathcal{E}}\, \hat{a}\right) \stackrel{\text{def}}{\equiv} \left(\pi^{\oplus}_{\mathcal{O}}\, a \xrightarrow{\mathsf{cross}_{\mathcal{O}}\, a} \pi^{\ominus}_{\mathcal{O}}\, a\right),$$

where $\mathsf{cross}_{\mathcal{O}}\, a : \mathcal{O}_{L \dashv R}[\![\pi^{\oplus}_{\mathcal{O}}\, a, \pi^{\ominus}_{\mathcal{O}}\, a]\!]$ is defined as

$$\mathsf{cross}_{\mathcal{O}}(\iota^{\ominus}_{\mathcal{O}}\, n)^{\flat} \stackrel{\text{def}}{\equiv} LRn \xrightarrow{\epsilon_n} n \qquad \mathsf{cross}_{\mathcal{O}}(\iota^{\oplus}_{\mathcal{O}}\, p)^{\flat} \stackrel{\text{def}}{\equiv} Rn \xrightarrow{\mathsf{id}_{Rn}} Rn$$

$$\mathsf{cross}_{\mathcal{O}}(\iota^{\ominus}_{\mathcal{O}}\, n)^{\sharp} \stackrel{\text{def}}{\equiv} Lp \xrightarrow{\mathsf{id}_{Lp}} Lp \qquad \mathsf{cross}_{\mathcal{O}}(\iota^{\oplus}_{\mathcal{O}}\, p)^{\sharp} \stackrel{\text{def}}{\equiv} p \xrightarrow{\eta_p} RLp.$$

Turning the definition of $\hat{a}$ around, notice that oblique objects are precisely the envelope pre-objects b for which either $\mathsf{cross}_{\mathcal{E}}\, b^{\flat}$ or $\mathsf{cross}_{\mathcal{E}}\, b^{\sharp}$ is the identity. Thus, to lose the polarity mapping structure, we could propositionally truncate away the aforementioned condition. Of course, internal to the type theory, we cannot ask an arbitrary morphism whether it is equal to the identity: $\mathsf{cross}_{\mathcal{E}}\, b^{\flat} =_{\mathcal{N}[\![L\, \pi^{\oplus}_{\mathcal{E}}\, b, \pi^{\ominus}\, b]\!]} \mathsf{id}_{L\, \pi^{\oplus}_{\mathcal{E}}\, b}$ is ill-typed. Instead, we ask for $\mathsf{cross}_{\mathcal{E}}\, b^{\flat}$ or $\mathsf{cross}_{\mathcal{E}}\, b^{\sharp}$ to be an *isomorphism*—an inverse for either is called a *polarization choice*. This is the inspiration behind the envelope duploid (Section 5.2).

Notation 5.6. We shall freely omit the subscripts $\mathcal{E}$ from $\iota^{\oplus}_{\mathcal{E}}$, $\iota^{\ominus}_{\mathcal{E}}$, $\pi^{\oplus}_{\mathcal{E}}$, $\pi^{\ominus}_{\mathcal{E}}$, and $\mathsf{cross}_{\mathcal{E}}$, and likewise for the $\mathcal{O}$ variants.

★ **Definition 5.7** (envelope_polarization_choice)**.** Let a be an envelope pre-object in $L \dashv R$. A *polarization choice* for a is a (two-sided) inverse for either mate of

$\mathsf{cross}\, a : \mathcal{O}_{L \dashv R}[\![\pi^{\oplus}\, a, \pi^{\ominus}\, a]\!]$. That is, either

$$\text{an inverse} \quad R(\pi^{\ominus}\, a) \xrightarrow{(\mathsf{cross}\, a^{\sharp})^{-1}} \pi^{\oplus}\, a \quad \text{for } \mathsf{cross}\, a^{\sharp}, \text{ or}$$
$$\text{an inverse} \quad \pi^{\ominus}\, a \xrightarrow{(\mathsf{cross}\, a^{\flat})^{-1}} L(\pi^{\oplus}\, a) \quad \text{for } \mathsf{cross}\, a^{\flat}.$$

The former is a *negative polarization choice*, the latter is a *positive polarization choice*.

Definition 5.8 (envelope_ob)**.** An *envelope object* $a : \mathcal{E}[L \dashv R]$ is an envelope pre-object in $L \dashv R$ such that there exists a polarization choice for a. That is, $\mathsf{cross}\, a^{\sharp}$ or $\mathsf{cross}\, a^{\flat}$ is an isomorphism.

Remark 5.9. Beware Convention 2.4: the polarization choice for each envelope object is under propositional truncation. Importantly, it does not partake in identifications of envelope objects. However, it may not be obtained constructively unless the choice is irrelevant.

TODO: does it make sense to define the composition for both at the same time as well?

5.1 The Oblique Duploid

The *oblique duploid* is the name I give to the construction of the duploid from an adjunction described by Munch-Maccagnoni [19]. In this section, we shall see how the oblique duploid construction gives rise to a distinctly split duploid, that is therefore unable to be univalent as a single-sorted duploid in general.

Definition 5.10 (oblique_mor')**.** Let $L \dashv R : \mathcal{N} \to \mathcal{P}$ be an adjunction. The *oblique duploid*-to-be $\mathcal{O}[L \dashv R]$ over $L \dashv R$ has objects and morphisms as follows.

- The objects $\mathsf{ob}\, \mathcal{O}[L \dashv R] \overset{\text{def}}{\equiv} \mathcal{N} \amalg \mathcal{P}$ are the oblique objects (Definition 5.2), which are written respectively $\iota^{\ominus}_{\mathcal{O}}\, n$ for $n : \mathcal{N}$, and $\iota^{\oplus}_{\mathcal{O}}\, p$ for $p : \mathcal{P}$.
- The morphisms $\mathcal{O}[L \dashv R][\![a, b]\!]$ are given by the oblique morphisms $\mathcal{O}_{L \dashv R}[\![\pi^{\oplus}\, a, \pi^{\ominus}\, b]\!]$.
- Identities $\mathsf{id}_a : \mathcal{O}[L \dashv R][\![a, a]\!]$ are given by $\mathsf{cross}\, a : \mathcal{O}[\![\pi^{\oplus}\, a, \pi^{\ominus}\, a]\!]$ (Definition 5.5).

How should we compose the morphisms of the oblique duploid-to-be? Given morphisms $f : \mathcal{O}[\![\pi^{\oplus}\, a, \pi^{\ominus}\, b]\!]$ and $g : \mathcal{O}[\![\pi^{\oplus}\, b, \pi^{\ominus}\, c]\!]$, we must find an oblique morphism $f \mathbin{⨟} g : \mathcal{O}[\![\pi^{\oplus}\, a, \pi^{\ominus}\, c]\!]$. Composition needs to be unital, so it must satisfy $\mathsf{cross}\, b \mathbin{⨟} g = g$

and $f \mathbin{⨾} \mathsf{cross}\, b = f$. Let us therefore depict f, g, and $\mathsf{cross}\, b$ on a diagram:

$$\begin{tikzcd}
L(\pi^\oplus a) \arrow[dr, "f^\flat"] & \\
L(\pi^\oplus b) \arrow[r, dashed, "\mathsf{cross}\, b^\flat"] \arrow[dr, "g^\flat"'] & \pi^\ominus b \\
& \pi^\ominus c
\end{tikzcd}
\qquad
\begin{tikzcd}
\pi^\oplus a \arrow[dr, "f^\sharp"] & \\
\pi^\oplus b \arrow[r, dashed, "\mathsf{cross}\, b^\sharp"] \arrow[dr, "g^\sharp"'] & R(\pi^\ominus b) \\
& R(\pi^\ominus c)
\end{tikzcd}$$

We need to find a composite in either category: either $(f \mathbin{⨾} g)^\flat : \mathcal{N}[\![L(\pi^\oplus a), \pi^\ominus c]\!]$ or $(f \mathbin{⨾} g)^\sharp : \mathcal{P}[\![\pi^\oplus a, R(\pi^\ominus c)]\!]$. If we had a morphism in the opposite direction to $\mathsf{cross}\, b$ in either category, so one $h : \mathcal{N}[\![\pi^\ominus b, L(\pi^\oplus b)]\!]$ or $k : \mathcal{P}[\![R(\pi^\ominus b), \pi^\oplus b]\!]$, then we could simply compose it between f and g, as $f^\flat \mathbin{⨾} h \mathbin{⨾} g^\flat$ or $f^\sharp \mathbin{⨾} k \mathbin{⨾} g^\sharp$. We do have such a morphism: recall from Definition 5.5 that either $\mathsf{cross}_{\mathcal{O}}\, b^\flat$ or $\mathsf{cross}_{\mathcal{O}}\, b^\sharp$ is the identity, and thus an isomorphism.

Definition 5.11 (`oblique_compose`)**.** The oblique duploid-to-be $\mathcal{O}[L \dashv R]$ has composition defined as follows. Let $f\colon a \to b$ and $g\colon b \to c$ be morphisms in $\mathcal{O}[L \dashv R]$. Define $f \mathbin{⨾} g\colon a \to c$ by discrimination on b, as either

$$\begin{aligned}
(f \mathbin{⨾} g)^\flat &\overset{\text{def}}{\equiv} L(\pi^\oplus a) \xrightarrow{Lf^\sharp} LR(\pi^\ominus b) \xrightarrow{L(\mathsf{cross}_{\mathcal{O}}\, b^\sharp)^{-1}} L(\pi^\oplus b) \xrightarrow{g^\flat} \pi^\ominus c && \text{and} \\
(f \mathbin{⨾} g)^\sharp &\overset{\text{def}}{\equiv} \pi^\oplus a \xrightarrow{f^\sharp} R(\pi^\ominus b) \xrightarrow{(\mathsf{cross}_{\mathcal{O}}\, b^\sharp)^{-1}} \pi^\oplus b \xrightarrow{g^\sharp} R(\pi^\ominus c) && \text{if } b \equiv \iota^\ominus n, \text{ or} \\
(f \mathbin{⨾} g)^\flat &\overset{\text{def}}{\equiv} L(\pi^\oplus a) \xrightarrow{f^\flat} \pi^\ominus b \xrightarrow{(\mathsf{cross}_{\mathcal{O}}\, b^\flat)^{-1}} L(\pi^\oplus b) \xrightarrow{g^\flat} \pi^\ominus c && \text{and} \\
(f \mathbin{⨾} g)^\sharp &\overset{\text{def}}{\equiv} \pi^\oplus a \xrightarrow{f^\sharp} R(\pi^\ominus b) \xrightarrow{R(\mathsf{cross}_{\mathcal{O}}\, b^\flat)^{-1}} RL(\pi^\oplus b) \xrightarrow{Rg^\flat} R(\pi^\ominus c) && \text{if } b \equiv \iota^\oplus p.
\end{aligned}$$

Remark 5.12. Definition 5.11 "forgets" that the mentioned isomorphisms $\mathsf{cross}_{\mathcal{O}}\, b^\sharp$ and $\mathsf{cross}_{\mathcal{O}}\, b^\flat$ are the identity, in anticipation of the generalization in Section 5.2. If we remember this fact, we can instead define, for $a \xrightarrow{f} b \xrightarrow{g} c$ in $\mathcal{O}[L \dashv R]$:

$$\begin{aligned}
(f \mathbin{⨾} g)^\sharp &\overset{\text{def}}{\equiv} \pi^\oplus a \xrightarrow{f^\sharp} Rn \xrightarrow{g^\sharp} R(\pi^\ominus c) && \text{if } b \equiv \iota^\ominus n \\
(f \mathbin{⨾} g)^\flat &\overset{\text{def}}{\equiv} L(\pi^\oplus a) \xrightarrow{f^\flat} Lp \xrightarrow{g^\flat} \pi^\ominus c && \text{if } b \equiv \iota^\oplus p.
\end{aligned}$$

This is the actual definition of `oblique_compose` used in the formalization, and by the original construction of Munch-Maccagnoni [19].

Lemma 5.13 (`oblique_left_id`, and dual)**.** *The oblique duploid-to-be $\mathcal{O}[L \dashv R]$, with composition as in Definition 5.11, is unital.* □

Lemma 5.14 (`oblique_negative_is_negative`, and dual)**.** *In the oblique duploid-to-be $\mathcal{O}[L \dashv R]$, the objects $\iota^\ominus n$ for $n : \mathcal{N}$ are semantically negative, and those $\iota^\oplus p$ for*

$p : \mathcal{P}$ *are semantically positive.* $\square$

Theorem 5.15 (oblique_preduploid)**.** *The oblique duploid-to-be $\mathcal{O}[L \dashv R]$ is a split preduploid with the obvious polarity mapping $\omega\colon \mathsf{ob}\,\mathcal{O}[L \dashv R] \to \{\ominus, \oplus\}$.* $\square$

Remark 5.16 (catiso_kleisli_to_oblique_duploid, and dual [19]). The syntactically positive subcategory $\mathcal{O}[L \dashv R]^{\oplus\omega}$ of the oblique duploid-to-be $\mathcal{O}[L \dashv R : \mathcal{N} \to \mathcal{P}]$ is isomorphic to the Kleisli category $\mathbf{Kleisli}_{\mathsf{arr}}(\mathcal{P}, RL)$, and the syntactically negative subcategory $\mathcal{O}[L \dashv R]^{\ominus\omega}$ is isomorphic to the Co-Kleisli category $\mathbf{CoKleisli}_{\mathsf{arr}}(\mathcal{N}, LR)$. Note that these are specifically the non-univalent variants (see Example 2.11).

Definition 5.17 (oblique_negative_shift_data, and dual)**.** The oblique duploid-to-be $\mathcal{O}[L \dashv R]$ has polarity shift structure given on objects by

$$\Uparrow a \overset{\text{def}}{\equiv} \iota^{\ominus}(\pi^{\ominus}\, a) \quad \text{and} \quad \Downarrow a \overset{\text{def}}{\equiv} \iota^{\oplus}(\pi^{\oplus}\, a).$$

For the morphisms $\mathtt{force}_a$ and $\mathtt{delay}_a$, compute their types:

$$\begin{aligned} \mathtt{force}_a &: \mathcal{O}[L \dashv R][\![\Uparrow a, a]\!] \\ &\equiv \mathcal{O}[\![\pi^{\oplus}(\Uparrow a), \pi^{\ominus}\, a]\!] \\ &\equiv \mathcal{O}[\![\pi^{\oplus}(\Uparrow a), \pi^{\ominus}(\Uparrow a)]\!] \end{aligned} \qquad \begin{aligned} \mathtt{delay}_a &: \mathcal{O}[L \dashv R][\![a, \Uparrow a]\!] \\ &\equiv \mathcal{O}[\![\pi^{\oplus}\, a, \pi^{\ominus} \Uparrow a]\!] \\ &\equiv \mathcal{O}[\![\pi^{\oplus}\, a, \pi^{\ominus}\, a]\!]. \end{aligned}$$

Give $\mathtt{force}_a \overset{\text{def}}{\equiv} \mathsf{cross}(\Uparrow a) : \mathcal{O}[\![\pi^{\oplus}(\Uparrow a), \pi^{\ominus}(\Uparrow a)]\!]$ and $\mathtt{delay}_a \overset{\text{def}}{\equiv} \mathsf{cross}\, a : \mathcal{O}[\![\pi^{\oplus}\, a, \pi^{\ominus}\, a]\!]$.

Dually, define $\mathtt{wrap}_a \overset{\text{def}}{\equiv} \mathsf{cross}(\Downarrow a)$ of the type $\mathcal{O}[L \dashv R][\![a, \Downarrow a]\!] \equiv \mathcal{O}[\![\pi^{\oplus}(\Downarrow a), \pi^{\ominus}(\Downarrow a)]\!]$, and $\mathtt{unwrap}_a \overset{\text{def}}{\equiv} \mathsf{cross}\, a$ of the type $\mathcal{O}[L \dashv R][\![\Downarrow a, a]\!] \equiv \mathcal{O}[\![\pi^{\oplus}\, a, \pi^{\ominus}\, a]\!]$.

Lemma 5.18 (oblique_negative_shift_axioms, and dual)**.** *The polarity shift structure of Definition 5.17 for the oblique duploid-to-be $\mathcal{O}[L \dashv R]$ satisfies the axioms of polarity shifts, and respects the polarity mapping.* $\square$

Theorem 5.19 (oblique_duploid, oblique_split_duploid)**.** *The oblique duploid-to-be $\mathcal{O}[L \dashv R]$ is a split duploid. Hence, we now call it the* oblique duploid. $\square$

Similarly to Theorem 4.10, we have the following characterizations of linearity, thunkabilitiy and polarities in the oblique duploid.

Theorem 5.20 (is_thunkable_iff_oblique_unit_postcompose, and dual [19])**.** *Fix an adjunction $L \dashv_{(\eta,\epsilon)} R : \mathcal{N} \to \mathcal{P}$. A morphism $f\colon a \to \iota^{\oplus}_{\mathcal{O}}\, p$ in the oblique duploid $\mathcal{O}[L \dashv R]$ is thunkable if and only if the equation*

$$f^{\sharp} \mathbin{⨟} RL\eta_p = f^{\sharp} \mathbin{⨟} \eta_{RLp} \quad \textit{holds.}$$

Dually, a morphism $g\colon \iota^{\ominus}_{\mathcal{O}}\, n \to b$ in $\mathcal{O}[L \dashv R]$ is linear if and only if the equation

$$LR\epsilon_n \mathbin{⨟} g^{\flat} = \epsilon_{LRn} \mathbin{⨟} g^{\flat} \quad \textit{holds.}$$

Theorem 5.21 (is_negative_oblique_positive_iff_pre_fixed_point, and dual)**.** *Let $L \dashv_{(\eta,\epsilon)} R : \mathcal{N} \to \mathcal{P}$ be an adjunction. For all $p : \mathcal{P}$, the object $\iota^{\oplus}_{\mathcal{O}}\, p$ in $\mathcal{O}[L \dashv R]$ is* *semantically negative if and only if*

$$RL\eta_p = \eta_{RLp} \quad \textit{holds.}$$

Dually, for all $n : \mathcal{N}$*, the object* $\iota^{\ominus}_{\mathcal{O}}\, n$ *in* $\mathcal{O}[L \dashv R]$ *is semantically positive if and only if*

$$LR\epsilon_n = \epsilon_{LRn} \quad \textit{holds.}$$

Proof. Combine Theorems 4.12 and 5.20. □

It follows from Theorem 4.20 that the oblique duploid is not generally univalent as a single-sorted duploid. The following examples may be checked using Theorem 5.21.

Example 5.22. Let $\mathcal{C}$ be a category with at least one object $a : \mathcal{C}$. Then the object $\iota^{\oplus}_{\mathcal{O}}\, a$ is both semantically negative and positive in the oblique duploid over the identity adjunction $\mathcal{O}[\mathsf{Id}_{\mathcal{C}} \dashv \mathsf{Id}_{\mathcal{C}}]$. Therefore the latter is not univalent as a single-sorted duploid.

Example 5.23. Consider a fixed inhabited set s : **Set**, and the state adjunction $(-\times s) \dashv (s \Rightarrow -) : \mathbf{Set} \to \mathbf{Set}$, given by the isomorphism $\mathbf{Set}[\![a \times s, b]\!] \simeq \mathbf{Set}[\![a, s \Rightarrow b]\!]$ natural in $a : \mathbf{Set}^{\mathsf{op}}$ and $b : \mathbf{Set}$. The object $\iota^{\oplus}_{\mathcal{O}}\,\texttt{empty}$ (and $\iota^{\ominus}_{\mathcal{O}}\,\texttt{empty}$) is both semantically negative and positive in $\mathcal{O}[(- \times s) \dashv (s \Rightarrow -)]$.

5.2 The Envelope Duploid

To define a truly single-sorted duploid arising from an adjunction, we must handle with care the objects that are both semantically negative and positive. For it to be univalent, we must (by Theorem 4.20) ensure that no polarity mapping can be defined when such an object exists. As anticipated in Remark 5.12, we obtain a perfectly serviceable variant of the oblique duploid by treating cross morphisms opaquely, only assuming that one of the mates is an isomorphism. Crucially, we forget which one by propositional truncation. In this section, we give the construction of the novel *envelope duploid*, a single-sorted counterpart to the oblique duploid.

★ **Definition 5.24** (`envelope_mor`)**.** Let $L \dashv R : \mathcal{N} \to \mathcal{P}$ be an adjunction. The *envelope duploid*-to-be $\mathcal{E}[L \dashv R]$ over $L \dashv R$ has objects, morphisms, and identities:

- The objects $\mathsf{ob}\,\mathcal{E}[L \dashv R]$ are envelope objects (Definition 5.8).
- Morphisms $\mathcal{E}[L \dashv R][\![a, b]\!]$ are the oblique morphisms $\mathcal{O}_{L\dashv R}[\![\pi^{\oplus}\, a, \pi^{\ominus}\, b]\!]$.
- For an object $a : \mathcal{E}[L \dashv R]$, its identity is given by its cross morphism $\mathsf{id}_a \overset{\text{def}}{\equiv} \mathsf{cross}\, a$.

We must now define composition in the envelope duploid. Like for the oblique duploid (Definition 5.11), we will do so by discriminating on the polarity of the intermediate object. Unlike the oblique duploid, we may not discriminate on the polarity of an envelope object unless we show the polarization choice to be irrelevant (Remark 5.9), which we do in Theorem 5.27.

★ **Definition 5.25** (envelope_compose'). Let a, b, and c be envelope pre-objects in $L \dashv R$, and let $f : \mathcal{O}[\![\pi^\oplus a, \pi^\ominus b]\!]$ and $g : \mathcal{O}[\![\pi^\oplus b, \pi^\ominus c]\!]$ be oblique morphisms between them. Given a polarization choice for b, define depending on it

$$\begin{aligned}
(f \mathbin{⨾} g)^\flat &\overset{\text{def}}{\equiv} L(\pi^\oplus a) \xrightarrow{Lf^\sharp} LR(\pi^\ominus b) \xrightarrow{L(\mathsf{cross}_\mathcal{E}\, b^\sharp)^{-1}} L(\pi^\oplus b) \xrightarrow{g^\flat} \pi^\ominus c && \text{and} \\
(f \mathbin{⨾} g)^\sharp &\overset{\text{def}}{\equiv} \pi^\oplus a \xrightarrow{f^\sharp} R(\pi^\ominus b) \xrightarrow{(\mathsf{cross}_\mathcal{E}\, b^\sharp)^{-1}} \pi^\oplus b \xrightarrow{g^\sharp} R(\pi^\ominus c) && \text{if negative, or} \\
(f \mathbin{⨾} g)^\flat &\overset{\text{def}}{\equiv} L(\pi^\oplus a) \xrightarrow{f^\flat} \pi^\ominus b \xrightarrow{(\mathsf{cross}_\mathcal{E}\, b^\flat)^{-1}} L(\pi^\oplus b) \xrightarrow{g^\flat} \pi^\ominus c && \text{and} \\
(f \mathbin{⨾} g)^\sharp &\overset{\text{def}}{\equiv} \pi^\oplus a \xrightarrow{f^\sharp} R(\pi^\ominus b) \xrightarrow{R(\mathsf{cross}_\mathcal{E}\, b^\flat)^{-1}} RL(\pi^\oplus b) \xrightarrow{Rg^\flat} R(\pi^\ominus c) && \text{if positive.}
\end{aligned}$$

Remark 5.26. Definition 5.25 is identical to Definition 5.11 of composition in the oblique duploid, but on envelope pre-objects instead.

★ **Theorem 5.27** (envelope_compose_irrel). *Let $f : a \to b$ and $g : b \to c$ be morphisms in the envelope duploid-to-be $\mathcal{E}[L \dashv R]$, with there being both a negative and a positive polarization choice for b. Then the two ways of composing $f \mathbin{⨾} g : a \to c$ using Definition 5.25 are equal.*

Proof. Let $\varphi_{p,n} : \mathcal{N}[\![Lp, n]\!] \cong \mathcal{P}[\![p, Rn]\!]$ be the natural isomorphism witnessing the adjunction $L \dashv R$. We show that the following morphisms are mates:

$$\begin{aligned}
&L(\pi^\oplus a) \xrightarrow{Lf^\sharp} LR(\pi^\ominus b) \xrightarrow{L(\mathsf{cross}\, b^\sharp)^{-1}} L(\pi^\oplus b) \xrightarrow{g^\flat} \pi^\ominus c && \text{(the negative choice's } (f \mathbin{⨾} g)^\flat) \\
&\pi^\oplus a \xrightarrow{f^\sharp} R(\pi^\ominus b) \xrightarrow{R(\mathsf{cross}\, b^\flat)^{-1}} RL(\pi^\oplus b) \xrightarrow{Rg^\flat} R(\pi^\ominus c) && \text{(the positive choice's } (f \mathbin{⨾} g)^\sharp).
\end{aligned}$$

In fact, we need only check that $L(\mathsf{cross}\, b^\sharp)^{-1}$ and $R(\mathsf{cross}\, b^\flat)^{-1}$ are mates, since it follows by naturality of φ that

$$\varphi_{(\pi^\oplus a),(\pi^\ominus a)} \left(Lf^\sharp \mathbin{⨾} L(\mathsf{cross}\, b^\sharp)^{-1} \mathbin{⨾} g^\flat\right) = f^\sharp \mathbin{⨾} R(\mathsf{cross}\, b^\flat)^{-1} \mathbin{⨾} Rg^\flat.$$

Therefore we must show $R(\mathsf{cross}\, b^\flat)^{-1} = \varphi_{R(\pi^\ominus b),L(\pi^\oplus b)}(L(\mathsf{cross}\, b^\sharp)^{-1})$. By isomorphism of $\mathsf{cross}\, b^\sharp$ and φ, it suffices to consider

$$\begin{aligned}
&\varphi^{-1}_{(\pi^\oplus a),L(\pi^\oplus a)} \left(\mathsf{cross}\, b^\sharp \mathbin{⨾} R(\mathsf{cross}\, b^\flat)^{-1}\right) \\
&\quad = \varphi^{-1}_{(\pi^\oplus a),L(\pi^\oplus a)} \left(\mathsf{cross}\, b^\sharp \mathbin{⨾} \varphi_{R(\pi^\ominus b),L(\pi^\oplus b)}(L(\mathsf{cross}\, b^\sharp)^{-1})\right).
\end{aligned}$$

Both sides are equal to $\mathsf{id}_{L(\pi^\oplus a)}$ using naturality of φ^{-1}. □

★ **Definition 5.28** (envelope_compose, envelope_unital_magmoid). The envelope duploid-to-be $\mathcal{E}[L \dashv R]$ has composition determined by Definition 5.25 from the polarization choice that exists for each envelope object. By Theorem 5.27 the choice of polarization is irrelevant, thus this definition is unique.

★ **Lemma 5.29** (is_negative_of_envelope_chosen_negative, and dual)**.** *Objects in the envelope duploid-to-be $\mathcal{E}[L \dashv R]$ with a positive polarization choice are positive, and those with a negative polarization choice are negative.*

Proof. Same as Lemma 5.14. □

★ **Definition 5.30** (envelope_preob_of_negative, and dual)**.** Define the negative and positive injections for $n : \mathcal{N}$ or $p : \mathcal{P}$ via Definition 5.5 as

$$\iota^{\ominus}_{\mathcal{E}}\, n \overset{\text{def}}{\equiv} \widehat{\iota^{\ominus}_{\mathcal{O}}\, n} \quad \text{and} \quad \iota^{\oplus}_{\mathcal{E}}\, p \overset{\text{def}}{\equiv} \widehat{\iota^{\oplus}_{\mathcal{O}}\, p}.$$

★ **Lemma 5.31** (envelope_chosen_negative_of_negative, and dual)**.** *For an object $a : \mathcal{O}[L \dashv R]$ of the oblique duploid, its corresponding envelope pre-object $\hat{a}$ has the corresponding polarization choice: negative if $a \equiv \iota^{\ominus}_{\mathcal{O}}\, n$ and positive if $a \equiv \iota^{\oplus}_{\mathcal{O}}\, p$.*

Proof. Compute $\mathsf{cross}_{\mathcal{O}}(\iota^{\ominus}_{\mathcal{O}}\, n)^{\sharp} \equiv \mathsf{id}_{Rn}$ and $\mathsf{cross}_{\mathcal{O}}(\iota^{\oplus}_{\mathcal{O}}\, p)^{\flat} \equiv \mathsf{id}_{Lp}$. □

★ **Lemma 5.32** (envelope_has_negative_shifts, and dual)**.** *The envelope duploid-to-be $\mathcal{E}[L \dashv R]$ has polarity shifts given the same way as the oblique duploid. That is:*

$$\begin{aligned}
{\Uparrow} a &\overset{\text{def}}{\equiv} \iota^{\ominus}(\pi^{\ominus}\, a) : \mathcal{E}[L \dashv R] & {\Downarrow} a &\overset{\text{def}}{\equiv} \iota^{\oplus}(\pi^{\oplus}\, a) : \mathcal{E}[L \dashv R] \\
\texttt{force}_a &\overset{\text{def}}{\equiv} \mathsf{cross}({\Uparrow} a) : \mathcal{E}[L \dashv R][\![{\Uparrow} a, a]\!] & \texttt{wrap}_a &\overset{\text{def}}{\equiv} \mathsf{cross}({\Downarrow} a) : \mathcal{E}[L \dashv R][\![a, {\Downarrow} a]\!] \\
\texttt{delay}_a &\overset{\text{def}}{\equiv} \mathsf{cross}\, a : \mathcal{E}[L \dashv R][\![a, {\Uparrow} a]\!] & \texttt{unwrap}_a &\overset{\text{def}}{\equiv} \mathsf{cross}\, a : \mathcal{E}[L \dashv R][\![{\Downarrow} a, a]\!].
\end{aligned}$$

Proof. Same as Lemma 5.18. □

★ **Theorem 5.33** (envelope_duploid)**.** *The envelope duploid-to-be $\mathcal{E}[L \dashv R]$ is a single-sorted duploid. Hence, we now call it the* envelope duploid. □

Similarly to the oblique duploid's Theorem 5.20, we have the following characterizations in the envelope duploid.

★ **Lemma 5.34** (is_thunkable_from_downshift_iff_envelope_unit_postcompose, and dual [19])**.** *Let $L \dashv_{(\eta,\epsilon)} R : \mathcal{N} \to \mathcal{P}$ be an adjunction. A morphism $f\colon a \to \iota^{\oplus}_{\mathcal{E}}\, p$ in the envelope duploid $\mathcal{E}[L \dashv R]$ is thunkable if and only if the equation*

$$f^{\sharp} \mathbin{⨾} RL\eta_p = f^{\sharp} \mathbin{⨾} \eta_{RLp} \quad \textit{holds.}$$

Dually, a morphism $g\colon \iota^{\ominus}_{\mathcal{E}}\, n \to b$ in $\mathcal{E}[L \dashv R]$ is linear if and only if the equation

$$LR\epsilon_n \mathbin{⨾} g^{\flat} = \epsilon_{LRn} \mathbin{⨾} g^{\flat} \quad \textit{holds.}$$

Proof. Same as Theorem 5.20. □

★ **Theorem 5.35** (is_negative_envelope_downshift_iff_pre_fixed_point, and dual)**.** *Let $L \dashv_{(\eta,\epsilon)} R : \mathcal{N} \to \mathcal{P}$ be an adjunction. For all $p : \mathcal{P}$, the object $\iota^{\oplus}_{\mathcal{E}} p$ is negative in $\mathcal{E}[L \dashv R]$ if and only if the equation*

$$RL\eta_p = \eta_{RLp} \quad \textit{holds.}$$

Dually, for $n : \mathcal{N}$, the object $\iota^{\ominus}_{\mathcal{E}} n$ is positive in $\mathcal{E}[L \dashv R]$ if and only if the equation

$$LR\epsilon_n = \epsilon_{LRn} \quad \textit{holds.}$$

Proof. Combine Theorems 4.12 and 5.34. □

Every two-sorted duploid is isomorphic to the oblique duploid over some adjunction [20, Proposition II.35].[2] We have the same for the envelope duploid, but with equivalence of unital magmoids instead of isomorphism.[3] In fact, the oblique and envelope duploids are equivalent as unital magmoids.

★ **Theorem 5.36** (weak_equiv_oblique_to_envelope_duploid)**.** *The oblique duploid $\mathcal{O}[L \dashv R]$ is weakly equivalent to the corresponding envelope duploid $\mathcal{E}[L \dashv R]$.*

Proof (sketch). The mapping $a \mapsto \hat{a}$ on objects $\mathcal{O}[L \dashv R] \to \mathcal{E}[L \dashv R]$ extends to a weak equivalence. □

★ **Theorem 5.37** (weak_dupoid_equivalence_duploid_to_envelope_on_shifts)**.** *The polarity shift functors of a duploid $\mathcal{D}$ take part in an adjunction $\Uparrow \dashv_{(\eta,\epsilon)} \Downarrow : \mathcal{D}^{\ominus}_{l} \to \mathcal{D}^{\oplus}_{t}$ with unit $\eta_p \overset{\text{def}}{\equiv} \texttt{delay}_p \mathbin{⨾} \texttt{wrap}_{\Uparrow p}$ and counit $\epsilon_n \overset{\text{def}}{\equiv} \texttt{force}_{\Downarrow n} \mathbin{⨾} \texttt{unwrap}_n$. The duploid $\mathcal{D}$ is weakly equivalent to the envelope duploid $\mathcal{E}[\Uparrow \dashv \Downarrow]$.*

Proof (sketch). The functor $F : \mathcal{D} \to \mathcal{E}[\Uparrow \dashv \Downarrow]$ is determined by its action on morphisms

$$\begin{aligned}
F_1\left(a \xrightarrow{f} b\right) &: \mathcal{O}_{\Uparrow\dashv\Downarrow}[\![\Downarrow a, \Uparrow b]\!] \\
F_1\left(a \xrightarrow{f} b\right)^{\flat} &\overset{\text{def}}{\equiv} \Uparrow\left(\Downarrow a \xrightarrow{\texttt{unwrap}_a} a \xrightarrow{f} b\right) \\
F_1\left(a \xrightarrow{f} b\right)^{\sharp} &\overset{\text{def}}{\equiv} \Downarrow\left(a \xrightarrow{f} b \xrightarrow{\texttt{delay}_a} \Uparrow b\right).
\end{aligned}$$

On objects, we then have $F_0(a) \overset{\text{def}}{\equiv} (\Downarrow a \xrightarrow{F_1(\mathsf{id}_a)} \Uparrow a)$. The fibre over each $b : \mathcal{E}[\Uparrow \dashv \Downarrow]$ is inhabited with object part $\pi^{\ominus} b$ or $\pi^{\oplus} b$ depending on the polarity of b. If b is negative, we must find an isomorphism $F(\pi^{\ominus} b) \cong_{lt} b$. We have $\texttt{force}_b : \Uparrow b \cong_{lt} b$ and, since F can be shown to preserve polarities, $\texttt{delay}_{F(\pi^{\ominus} b)} : F(\pi^{\ominus} b) \cong_{lt} \Uparrow F(\pi^{\ominus} b)$. We may

[2] The isomorphism was proven for two-sorted duploids, rather than split duploids, and with an appropriate definition for a functor of two-sorted duploids.

[3] Munch-Maccagnoni [19] requires duploid functors to preserve **syntactic** polarities and linearity/thunkability. We note that weak equivalences always preserve **semantic** polarities and linearity/thunkability.

then compute $\Uparrow F(\pi^\ominus b) \equiv \iota^\ominus(\Uparrow(\pi^\ominus b))$ and $\iota^\ominus(\pi^\ominus b) \equiv \Uparrow b$ to obtain $\iota^\ominus(\mathtt{force}_{\pi^\ominus b}) : \iota^\ominus(\Uparrow(\pi^\ominus b)) \cong_{lt} \iota^\ominus(\pi^\ominus b)$. Ultimately, the isomorphism part is:

$$F(\pi^\ominus b) \overset{\mathtt{delay}}{\cong_{lt}} \Uparrow F(\pi^\ominus b) \equiv \iota^\ominus(\Uparrow(\pi^\ominus b)) \overset{\iota^\ominus(\mathtt{force})}{\cong_{lt}} \iota^\ominus(\pi^\ominus b) \equiv \Uparrow b \overset{\mathtt{force}}{\cong_{lt}} b.$$

We omit plenty of details. □

Remark 5.38 (`equiv_oblique_to_envelope_duploid_from_LEM`, `dupoid_equivalence_duploid_to_envelope_on_shifts_from_LEM`). The equivalences of Theorems 5.36 and 5.37 are strong when assuming the law of excluded middle. This is because we can "decide" whether, say, a negative polarization choice for $a : \mathcal{E}[L \dashv R]$ exists, and take an arbitrary bias towards $\pi^\ominus a$ when it does.

5.3 Univalence and the Equalizing Requirement

Under what conditions are the oblique and envelope duploids of an adjunction $L \dashv R : \mathcal{N} \to \mathcal{P}$ univalent? It seems reasonable to assume that $\mathcal{N}$ and $\mathcal{P}$ must be univalent. Then, we must somehow show that $\mathcal{O}[L \dashv R]_l^{\ominus\omega}/\mathcal{O}[L \dashv R]_t^{\oplus\omega}$ and $\mathcal{E}[L \dashv R]_l^{\ominus}/\mathcal{E}[L \dashv R]_t^{\oplus}$ are univalent. We shall do so using Theorem 3.25, by exhibiting isomorphisms of categories with $\mathcal{N}/\mathcal{P}$. To do so, we will need an *equalizing requirement* [19].

★ **Lemma 5.39** (`negative_to_oblique_duploid`, `negative_category_to_envelope_duploid`, *etc.*)**.** *Let $L \dashv R : \mathcal{N} \to \mathcal{P}$ be an adjunction. (Recall that $\mathcal{O}[L \dashv R]_l^{\ominus\omega}$ is the **syntactic** negative-linear subcategory, and $\mathcal{E}[L \dashv R]_l^{\ominus}$ the **semantic** negative-linear subcategory.) The inclusions $\iota_{\mathcal{O}}^\ominus$, $\iota_{\mathcal{O}}^\oplus$, $\iota_{\mathcal{E}}^\ominus$, and $\iota_{\mathcal{E}}^\oplus$ extend to functors:*

$$\begin{aligned} &\iota_{\mathcal{O}}^\ominus : \mathcal{N} \to \mathcal{O}[L \dashv R]_l^{\ominus\omega} && \iota_{\mathcal{E}}^\ominus : \mathcal{N} \to \mathcal{E}[L \dashv R]_l^{\ominus} \\ &\iota_{\mathcal{O}}^\oplus : \mathcal{P} \to \mathcal{O}[L \dashv R]_t^{\oplus\omega} && \iota_{\mathcal{E}}^\oplus : \mathcal{P} \to \mathcal{E}[L \dashv R]_t^{\oplus}. \end{aligned}$$

These are given on morphisms by

$$\begin{aligned} &\iota^\ominus\left(n \xrightarrow{g} m\right)^\flat \overset{\text{def}}{\equiv} LRn \xrightarrow{\epsilon_n \mathbin{⨾} f} m && \iota^\oplus\left(p \xrightarrow{f} q\right)^\flat \overset{\text{def}}{\equiv} Lp \xrightarrow{Lf} Lq \\ &\iota^\ominus\left(n \xrightarrow{g} m\right)^\sharp \overset{\text{def}}{\equiv} Rn \xrightarrow{Rf} Rm && \iota^\oplus\left(p \xrightarrow{f} q\right)^\sharp \overset{\text{def}}{\equiv} p \xrightarrow{f \mathbin{⨾} \eta_q} RLq. \end{aligned}$$

Proof. The linearity and thunkability conditions follow from naturality of ϵ/η (by Theorems 5.20 and 5.34). The functor laws follow from those of R/L. □

Lemma 5.40 (`isweq_on_objects_negative_to_oblique_duploid`, and dual)**.** *Let $L \dashv R : \mathcal{N} \to \mathcal{P}$ be an adjunction. The inclusions $\iota_{\mathcal{O}}^\ominus : \mathcal{N} \to \mathcal{O}[L \dashv R]_l^{\ominus\omega}$ and $\iota_{\mathcal{O}}^\oplus : \mathcal{P} \to \mathcal{O}[L \dashv R]_t^{\oplus\omega}$ are equivalences on objects.* □

★ **Lemma 5.41** (`envelope_ob_eq_negative'`, and dual)**.** *Let $L \dashv R : \mathcal{N} \to \mathcal{P}$ be an adjunction between **univalent** categories $\mathcal{N}$ and $\mathcal{P}$. Every object $a : \mathcal{E}[L \dashv R]$ with a* *negative polarization choice is identified with* $\Uparrow a$*. Dually, every object* $b : \mathcal{E}[L \dashv R]$ *with a positive polarization choice is identified with* $\Downarrow b$*.*

Proof. We show the former. We need to identify

$$\left(\pi^\oplus\, a \xrightarrow{\mathsf{cross}\, a} \pi^\ominus\, a\right) = \left(R(\pi^\ominus\, a) \xrightarrow{\mathsf{cross}(\Uparrow a)} \pi^\ominus\, a\right). \tag{5.1}$$

By assumption, $\mathsf{cross}\, a^\sharp : \mathcal{P}[\![\pi^\oplus\, a, R(\pi^\ominus\, a)]\!]$ is an isomorphism. Thus by Theorem 4.18 we have $\langle \pi^\oplus\, a, \mathsf{cross}\, a^\sharp\rangle =_{\{R(\pi^\ominus\, a)\}^{\Leftarrow}_{\mathcal{P}}} \langle R(\pi^\ominus\, a), \mathsf{id}_{R(\pi^\ominus\, a)}\rangle$. Since $\mathsf{id}_{R(\pi^\ominus\, a)} \equiv \mathsf{cross}(\Uparrow a)^\sharp$, Equation 5.1 follows from injectivity of $\sharp$. ☐

Remark 5.42. Unlike for the oblique duploid, Lemma 5.41 is insufficient to make $\iota^\ominus_{\mathcal{E}}$ and $\iota^\oplus_{\mathcal{E}}$ equivalences on objects, which are only essentially surjective in general. This is because, say, a semantically negative object $\iota^\oplus_{\mathcal{E}}\, p$ in $\mathcal{E}[L \dashv_{(\eta,\epsilon)} R]$ need only have $RL\eta_p = \eta_{RLp}$ (by Theorem 5.35), which does not generally imply that η_p has an inverse. The equalizing requirements of Definition 5.45 will remedy this.

Lemma 5.43 (`split_essentially_surjective_negative_category_to_envelope_duploid`)**.** *Let* $L \dashv R : \mathcal{N} \to \mathcal{P}$ *be an adjunction. The inclusions* $\iota^\ominus_{\mathcal{E}} : \mathcal{N} \to \mathcal{E}[L \dashv R]^\ominus_l$ *and* $\iota^\oplus_{\mathcal{E}} : \mathcal{P} \to \mathcal{E}[L \dashv R]^\oplus_t$ *are split essentially surjective.*

Proof. The fibre of $\iota^\ominus_{\mathcal{E}}$ over $b : \mathcal{E}[L \dashv R]^\ominus_l$ has an element with object part $\pi^\ominus_{\mathcal{E}}\, b$. We have $\iota^\ominus_{\mathcal{E}}(\pi^\ominus_{\mathcal{E}}\, b) \equiv \Uparrow b$, so the isomorphism is just $\mathtt{force}_b : \Uparrow b \cong_{lt} b$. Dualize for $\iota^\oplus_{\mathcal{E}}$. ☐

Theorem 5.44 (`fully_faithful_iff_negative_equalizing_negative_to_oblique_duploid`, `fully_faithful_negative_category_to_envelope_duploid_iff`, *etc.*)**.** *Let* $L \dashv R : \mathcal{N} \to \mathcal{P}$ *be an adjunction. The following are equivalent:*

1. *The functor* $\iota^\ominus_{\mathcal{O}} : \mathcal{N} \to \mathcal{O}[L \dashv R]^{\ominus\omega}_l$ *is fully-faithful.*
2. ______________________ *is an isomorphism of categories.*
3. *The functor* $\iota^\ominus_{\mathcal{E}} : \mathcal{N} \to \mathcal{E}[L \dashv R]^\ominus_l$ *is fully-faithful.*
4. ______________________ *is a strong equivalence of categories.*[4]
5. *For all* $n : N$*, the fork* $LRLRn \overset{LR\epsilon_n}{\underset{\epsilon_{LRn}}{\rightrightarrows}} LRn \xrightarrow{\epsilon_n} n$ *is a coequalizer:*

$$\begin{tikzcd} LRLRn \arrow[r, shift left, "LR\epsilon_n"] \arrow[r, shift right, "\epsilon_{LRn}"'] & LRn \arrow[r, "\epsilon_n"] \arrow[dr, "f"'] & n \arrow[d, dashed, "\exists! f^*"] \\ & & m \end{tikzcd}$$

That is, for all objects $n, m : \mathcal{N}$ *and morphisms* $f : LRn \to m$ *satisfying* $LR\epsilon_n \mathbin{⨾} f = \epsilon_{LRn} \mathbin{⨾} f$*, there is a unique morphism* $f^* : n \to m$ *such that* $\epsilon_n \mathbin{⨾} f^* = f$*.*

[4] Conditions 4 and D are not propositions in general, so "equivalent" means "if and only if" for these.

Proof. The implications (2) $\Rightarrow$ (1) and (4) $\Rightarrow$ (3) are immediate. Their converses (1) $\Rightarrow$ (2) and (3) $\Rightarrow$ (4) follow from Lemmas 5.40 and 5.43. The equivalences (1) $\Leftrightarrow$ (5) and (3) $\Leftrightarrow$ (5) follow from the characterizations of linear morphisms in terms of the counit ϵ of the adjunction (Theorems 5.20 and 5.34). $\square$

Theorem 5.44 (dual part)**.** *Dually, the following are equivalent:*

A. *The functor $\iota^{\oplus}_{\mathcal{O}}\colon \mathcal{N} \to \mathcal{O}[L \dashv R]^{\oplus\omega}_{t}$ is fully-faithful.*

B. ______________________________ *is an isomorphism of categories.*

C. *The functor $\iota^{\oplus}_{\mathcal{E}}\colon \mathcal{N} \to \mathcal{E}[L \dashv R]^{\oplus}_{t}$ is fully-faithful.*

D. ______________________________ *is a strong equivalence of categories.*[4]

E. *For all $p : P$, the fork $q \xrightarrow[\eta_p]{} RLq \underset{\eta_{RLq}}{\overset{RL\eta_q}{\rightrightarrows}} RLRLq$ is an equalizer:*

$$
\begin{tikzcd}
p \arrow[d, dashed, "\exists! g^*"'] \arrow[dr, "g"] & & \\
q \arrow[r, "\eta_p"'] & RLq \arrow[r, shift left, "RL\eta_q"] \arrow[r, shift right, "\eta_{RLq}"'] & RLRLq
\end{tikzcd}
$$

That is, for all objects $p, q : \mathcal{P}$ and morphisms $g\colon p \to RLq$ satisfying $g \fatsemi RL\eta_q = g \fatsemi \eta_{RLq}$, there is a unique morphism $g^* : p \to q$ such that $g^* \fatsemi \eta_q = g$. $\square$

Definition 5.45 (is_negative_equalizing, is_fully_equalizing, and dual [19])**.** An adjunction satisfying conditions 1–5 of Theorem 5.44 is *negative equalizing*, and an adjunction satisfying conditions A–E is *positive equalizing*. An adjunction which satisfies conditions 1–5 **and** conditions A–E is *fully equalizing*.

We now prove univalence for both the oblique and envelope duploids.

Theorem 5.46 (is_univalent_split_oblique_duploid, is_univalent_split_oblique_duploid_inv)**.** *The oblique duploid $\mathcal{O}[L \dashv R : \mathcal{N} \to \mathcal{P}]$ over a fully equalizing adjunction $L \dashv R$ is univalent as a split duploid if and only if $\mathcal{N}$ and $\mathcal{P}$ are univalent.*

Proof. Recall that univalence as a split duploid means that $\mathcal{O}[L \dashv R]^{\ominus\omega}_{l}$ and $\mathcal{O}[L \dashv R]^{\oplus\omega}_{t}$ are univalent. Using Theorem 3.25, univalence may be transported (in both directions) across the isomorphisms $\iota^{\ominus}_{\mathcal{O}} : \mathcal{N} \cong \mathcal{O}[L \dashv R]^{\ominus\omega}_{l}$ and $\iota^{\oplus}_{\mathcal{O}} : \mathcal{P} \cong \mathcal{O}[L \dashv R]^{\oplus\omega}_{t}$. $\square$

Lemma 5.47 (is_negative_fixed_point_iff_pre_fixed_point, and dual)**.** *If an adjunction $L \dashv R : \mathcal{N} \to \mathcal{P}$ is negative equalizing, then for all $n : \mathcal{N}$,*

$$LR\epsilon_n = \epsilon_{LRn} \quad \textit{implies that } \epsilon_n \textit{ is an isomorphism.}$$

Dually, if $L \dashv R : \mathcal{N} \to \mathcal{P}$ is positive equalizing, then for all $p : \mathcal{N}$,

$$RL\eta_p = \eta_{RLp} \quad \textit{implies that } \eta_p \textit{ is an isomorphism.}$$

Proof (sketch). This follows from the coequalizer and equalizer conditions of Theorem 5.44, instantiated with $f \equiv \mathsf{id}_{LRn}$ and $g \equiv \mathsf{id}_{RLp}$ respectively. $\square$

★ **Lemma 5.48** (envelope_chosen_negative_of_is_negative, and dual)**.** *In the envelope duploid $\mathcal{E}[L \dashv R]$ over a fully equalizing adjunction $L \dashv R$, every negative/positive object has a negative/positive polarization choice.*

Proof. We show the negative case. Let $\varphi_{p,n} : \mathcal{N}[\![Lp, n]\!] \simeq \mathcal{P}[\![p, Rn]\!]$ be the natural isomorphism witnessing $L \dashv R : \mathcal{N} \to \mathcal{P}$, and let a be a negative object in $\mathcal{E}[L \dashv R]$. Being an envelope object, a has some polarization choice. If this is negative then we are done, otherwise assume that a has a positive polarization choice, so $\mathsf{cross}\, a^\flat$ is an isomorphism. We show that $\mathsf{cross}\, a^\sharp = \varphi_{(\pi^\oplus a),(\pi^\ominus a)}(\mathsf{cross}\, a^\flat) \equiv \eta_{\pi^\oplus a} \mathbin{⨾} R(\mathsf{cross}\, a^\flat)$ is the composite of two isomorphisms. $R(\mathsf{cross}\, a^\flat)$ being an isomorphism follows from our assumption. Since a is both negative and positive, $\Downarrow a \equiv \iota^\oplus(\pi^\oplus a)$ is also negative. Finally, $\eta_{\pi^\oplus a}$ being an isomorphism follows from Theorem 5.35 and Lemma 5.47. $\square$

From Lemmas 5.48 and 5.41, we obtain for the envelope duploid results characterizing its subcategories (Remark 5.16 and Lemma 5.40 for the oblique duploid).

★ **Theorem 5.49** (catiso_kleisli_to_envelope_duploid, and dual)**.** *Let $L \dashv R$ be a fully equalizing adjunction between **univalent** categories $\mathcal{N}$ and $\mathcal{P}$. The positive and negative subcategories of $\mathcal{E}[L \dashv R]$ are isomorphic to the Kleisli and Co-Kleisli categories: $\mathcal{E}[L \dashv R]^\oplus \cong$ **Kleisli**$_{\mathsf{arr}}(\mathcal{P}, RL)$ and $\mathcal{E}[L \dashv R]^\ominus \cong$ **CoKleisli**$_{\mathsf{arr}}(\mathcal{N}, LR)$.* $\square$

★ **Theorem 5.50** (is_catiso_negative_category_to_envelope_duploid, and dual)**.** *Let $L \dashv R$ be a fully equalizing adjunction between **univalent** categories $\mathcal{N}$ and $\mathcal{P}$. The inclusions $\iota^\ominus_{\mathcal{E}} : \mathcal{N} \to \mathcal{E}[L \dashv R]^\ominus_l$ and $\iota^\oplus_{\mathcal{E}} : \mathcal{P} \to \mathcal{E}[L \dashv R]^\oplus_t$ are isomorphisms of categories.*

Proof. Fully-faithfulness is by Definition 5.45, equivalence on objects is by Lemmas 5.48 and 5.41. $\square$

★ **Corollary 5.51** (is_univalent_envelope_duploid)**.** *Let $L \dashv R$ be a fully equalizing adjunction between **univalent** categories. The envelope duploid $\mathcal{E}[L \dashv R]$ is univalent.*

6 Conclusions and Future Work

The contributions of this work are twofold. First, we have formalized part of Munch-Maccagnoni's work on duploids [19, 20, Ch. II] in Rocq. The formalization covers unital magmoids and two variants of duploids: single-sorted duploids and split duploids. Second, we have made novel contributions to their study in univalent foundations. We have defined what it means to be univalent for unital magmoids, and for both variants of duploids. We define two versions of the duploid arising from an adjunction: the original "oblique duploid" construction of Munch-Maccagnoni [19] which is naturally a split duploid, and a novel "envelope duploid" construction which is a single-sorted duploid. Finally, we have shown both of these to be univalent—in their respective senses—when arising from a fully equalizing adjunction between univalent categories.

Our formalization of Munch-Maccagnoni's work ends at showing that each duploid is equivalent to one arising from an adjunction. This result is the first part of the full "structure theorem" [19]: that the category of duploids is equivalent to the category of fully equalizing adjunctions. As alluded to in Section 1.2, the hom-set law of categories (Definition 3.1.5) prevents even defining the 1-category of duploids in univalent foundations, and suggests that duploids ought to be treated as a 2-category instead (see Non-example 3.8). Future work could include a study of the 2-category of duploids, and a 2-categorical structure theorem for duploids.

We fall short of defining a Rezk completion for duploids: a weakly equivalent univalent duploid for any duploid $\mathcal{D}$. One way could be to use Theorem 5.37, but Rezk-complete the categories $\mathcal{D}_l^{\ominus}$ and $\mathcal{D}_t^{\oplus}$ before assembling $\mathcal{E}[\Uparrow \dashv \Downarrow]$. This procedure eluded our formalization due to complexity arising from the 2-categorical nature of adjunctions.

Lastly, there is more work to be done studying use cases for univalence of duploids. For example, we proved directly that polarity shifts are unique in a univalent duploid (Theorem 4.23). Future work could generalize this result to other universal properties on univalent unital magmoids.

Bibliography

[1] B. Ahrens, D. Frumin, M. Maggesi, N. Veltri, and N. van der Weide, "Bicategories in univalent foundations," *Mathematical Structures in Computer Science*, vol. 31, no. 10, pp. 1232–1269, 2021. DOI: 10.1017/S0960129522000032

[2] B. Ahrens, K. Kapulkin, and M. Shulman, "Univalent categories and the Rezk completion," *Mathematical Structures in Computer Science*, vol. 25, no. 5, pp. 1010–1039, 2015. DOI: 10.1017/S0960129514000486

[3] B. Ahrens, P. R. North, M. Shulman, and D. Tsementzis, *The Univalence Principle* (Memoirs of the American Mathematical Society). 2025, vol. 305. DOI: 10.1090/memo/1541 arXiv: 2102.06275 [math.CT].

[4] AlDanial et al., *AlDanial/cloc*, version v2.08, Jan. 2026. DOI: 10.5281/zenodo.18364352

[5] A. Bordg, *Univalent foundations and the unimath library*, 2019. DOI: 10.1007/978-3-030-15655-8_8 arXiv: 1710.02723 [math.LO].

[6] P. Clairambault and G. Munch-Maccagnoni, "Duploid situations in concurrent games," in *12th Workshop on Games for Logic and Programming Languages (GaLoP XII)*, Juha Kontinen, Marina Lenisa, Uppsala, Sweden, Apr. 2017. [Online]. Available: https://inria.hal.science/hal-01991555

[7] T. Coquand and N. A. Danielsson, "Isomorphism is equality," *Indagationes Mathematicae*, vol. 24, no. 4, pp. 1105–1120, 2013, In memory of N.G. (Dick) de Bruijn (1918–2012), ISSN: 0019-3577. DOI: 10.1016/j.indag.2013.09.002 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0019357713000694

[8] C. Führmann, "Direct models of the computational lambda-calculus," *Electronic Notes in Theoretical Computer Science*, vol. 20, pp. 245–292, 1999, MFPS XV, Mathematical Foundations of Progamming Semantics, Fifteenth Conference, ISSN: 1571-0661. DOI: 10.1016/S1571-0661(04)80078-1 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1571066104800781

[9] J.-Y. Girard, "Linear logic," *Theoretical Computer Science*, vol. 50, no. 1, pp. 1–101, 1987, ISSN: 0304-3975. DOI: 10.1016/0304-3975(87)90045-4 [Online]. Available: https://www.sciencedirect.com/science/article/pii/0304397587900454

[10] D. R. Grayson et al., *UniMath/UniMath*, version v20250923, Sep. 2025. DOI: 10.5281/zenodo.17186647 [Online]. Available: https://github.com/UniMath/UniMath

[11] J. Lambek and P. J. Scott, *Introduction to higher order categorical logic.* USA: Cambridge University Press, 1986, ISBN: 0521246652.

[12] P. B. Levy, "Adjunction models for call-by-push-value with stacks," *Electronic Notes in Theoretical Computer Science*, vol. 69, pp. 248–271, 2003, CTCS'02, Category Theory and Computer Science, ISSN: 1571-0661. DOI: 10.1016/S1571-0661(04)80568-1

[13] P. B. Levy, "Call-by-push-value: A subsuming paradigm," in *Typed Lambda Calculi and Applications*, J.-Y. Girard, Ed., Berlin, Heidelberg: Springer Berlin Heidelberg, 1999, pp. 228–243, ISBN: 978-3-540-48959-7.

[14] P. B. Levy, "Contextual isomorphisms," *SIGPLAN Not.*, vol. 52, no. 1, pp. 400–414, Jan. 2017, ISSN: 0362-1340. DOI: 10.1145/3093333.3009898

[15] É. Mangel, P.-A. Melliès, and G. Munch-Maccagnoni, "Classical notions of computation and the Hasegawa-Thielecke theorem," *Proc. ACM Program. Lang.*, vol. 10, no. POPL, Jan. 2026. DOI: 10.1145/3776715

[16] P. Martin-Löf, "Constructive mathematics and computer programming," in *Logic, methodology and philosophy of science, VI (Hannover, 1979)*, ser. Stud. Logic Found. Math. Vol. 104, North-Holland, Amsterdam, 1982, pp. 153–175. DOI: 10.1016/S0049-237X(09)70189-2

[17] E. Moggi, "Computational lambda-calculus and monads," in *[1989] Proceedings. Fourth Annual Symposium on Logic in Computer Science*, 1989, pp. 14–23. DOI: 10.1109/LICS.1989.39155

[18] E. Moggi, "Notions of computation and monads," *Information and Computation*, vol. 93, no. 1, pp. 55–92, 1991, Selections from 1989 IEEE Symposium on Logic in Computer Science, ISSN: 0890-5401. DOI: 10.1016/0890-5401(91)90052-4

[19] G. Munch-Maccagnoni, "Models of a Non-Associative Composition," in *Foundations of Software Science and Computation Structures*, A. Muscholl, Ed., Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 396–410, ISBN: 978-3-642-54830-7. DOI: 10.1007/978-3-642-54830-7_26

[20] G. Munch-Maccagnoni, "Syntax and Models of a non-Associative Composition of Programs and Proofs," Theses, Université Paris-Diderot - Paris VII, Dec. 2013. [Online]. Available: https://theses.hal.science/tel-00918642

[21] E. Rijke, *Introduction to Homotopy Type Theory* (Cambridge Studies in Advanced Mathematics). Cambridge University Press, 2025.

[22] The Rocq Development Team. "The Rocq prover 9.1.1 documentation." version 9.1. [Online]. Available: https://rocq-prover.org/doc/v9.1/refman/index.html

[23] The Univalent Foundations Program, *Homotopy Type Theory: Univalent Foundations of Mathematics.* Institute for Advanced Study, 2013. [Online]. Available: https://homotopytypetheory.org/book

[24] V. Voevodsky, *Experimental library of univalent formalization of mathematics*, 2014. arXiv: 1401.0053 [math.HO].

[25] N. van der Weide, “The formal theory of monads, univalently,” *Logical Methods in Computer Science*, vol. Volume 21, Issue 1, Feb. 2025, ISSN: 1860-5974. DOI: 10.46298/lmcs-21(1:16)2025 arXiv: 2212.08515 [cs.LO].

[26] K. Wullaert, R. Matthes, and B. Ahrens, “Univalent monoidal categories,” en, vol. 269, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2023, 15:1–15:21. DOI: 10.4230/LIPICS.TYPES.2022.15 [Online]. Available: https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.TYPES.2022.15

Index