Kleisli Diagrams for Optics

Mario Román

March 31, 2020

Abstract

There are at least three different formal graphical representations of optics on the literature [Hed17, Ril18, Boi20]. They do not seem to provide a clear way of composing optics of different kinds in a faithful way: for instance, composing optics as setters is not necessarily faithful. We present a graphical formalism of optics that eases this task by observing that categories of optics are full subcategories of a Kleisli category for a monad.

We can link this to the elegant diagrams proposed by Boisseau [Boi20], which are able to describe the laws of optics. Because the monad is opmonoidal, its Eilenberg-Moore category is monoidal with the forgetful functor being strong monoidal. We can then use the fully faithful embedding of Kleisli categories into Eilenberg-Moore categories to interpret our diagrams in the promonoidal context. This is work in progress.

1 Summary

We apply general string diagrams for Kleisli categories to the particular case of optics, understood as the full Kleisli category on representables of the Pastro-Street monad [Rom19].

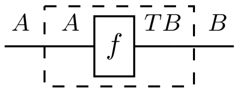
- The fact that every morphism in a Kleisli category can be decomposed into some *pure* part and some *effectful* part translates here to the decomposition of optics into transformations and cups reported by [Ril18, Boi20]. This also helps to conceptually explain teleological categories [Hed17].
- Kleisli categories fully faithfully embed into Eilenberg-Moore categories. The Pastro-Street monad is a monoidal monad. We observe that this is what gives monoidal structure to its Eilenberg-Moore category, the category of Tambara modules, and what lies at the core of Boisseau's string diagrams [Boi20]. This could allow us to regain monoidal structure for our Kleisli morphisms using the category of Tambara modules.
- However, Kleisli categories have a particularly nice description of the combined effects of a coproduct monad. This allows us to obtain a graphical calculus that is particularly suited for optic composition.

2 Prelude: Graphical calculus for Kleisli categories

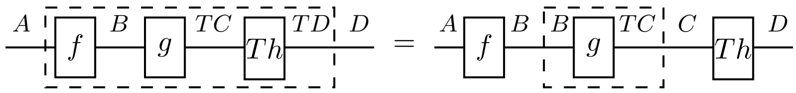
Monads (T, μ, η) can be alternatively expressed just in terms of the action of T on objects, their unit $\eta_A \colon A \to TA$ and a Kleisli extension operation

$$(-)^* \colon \mathcal{C}(A, TB) \cong \mathsf{Kl}(A, B)$$

that puts in correspondence every morphism with its associated homomorphism between free algebras. It is common to represent this assignment as a box and declare that every morphism in a Kleisli category can be drawn as in the following diagram.



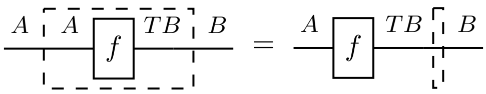
Note that, up to the equalities that already hold in $\mathcal{C}$, this way of writing the morphism is unique. The box is natural in both its outputs and two boxes inside of each other can be flattened using the monad multiplication.



This provides a graphical calculus for Kleisli categories with an extra benefit: it can be extended to accommodate for the effects of multiple monads. In other words, if we draw using boxes both for the monad T_1 and the monad T_2, the resulting diagram can be interpreted as a Kleisli morphism for the coproduct monad $T_1 \oplus T_2$, the least monad containing effects of both T_1 and T_2.

There are other niceties of this language: when the monad is monoidal, the Kleisli category can be made monoidal with the free functor being strong monoidal. Moreover, if the monad is opmonoidal, then we can fully faithfully embed into the Eilenberg-Moore category, which has a canonical monoidal structure that makes the forgetful functor being strong monoidal. However, it is in general only 1-dimensional, making it irrelevant in many cases.

Finally, this language allows us to express morphisms in $\mathsf{Kl}(\mathcal{C})$ by just adding a single piece (the box) to the language. This single piece can be always pushed to the right using naturality, and every morphism can be expressed in a unique way ending on a box thanks to the isomorphism $\mathcal{C}(A, TB) \cong \mathsf{Kl}(A, B)$.



Multiple boxes inside each other can be substituted by the multiplication of the monad. An example on how this calculus works can be found, for instance, in the work of Piróg and Wu [PW16].

3 The case of optics

We fix an action $\oslash \colon \mathcal{M} \times \mathcal{C} \to \mathcal{C}$ Optics are the full subcategory on profunctors of the form $よ^A_B := \mathcal{C}(A, -) \times \mathcal{C}(-, B)$ of the Kleisli category for a monad $\Phi_\oslash \colon \mathit{Prof}(\mathcal{C}) \to \mathit{Prof}(\mathcal{C})$ defined as

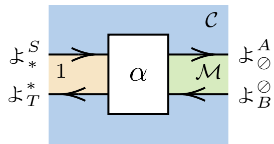
$$\Phi_\oslash P(S, T) := \int^{X,Y,M} \mathcal{C}(S, M \oslash X) \times \mathcal{C}(M \oslash Y) \times P(X, Y).$$

This monad, when evaluated in $\Phi\, よ^A_B$, and after applying the Yoneda lemma, yields the usual description of optics

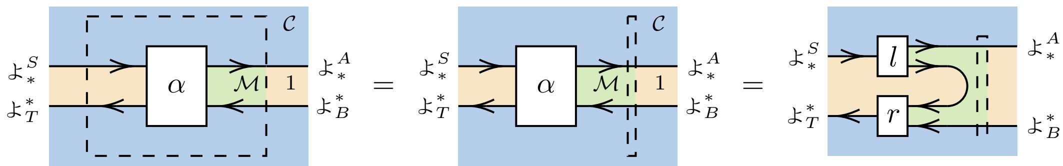
$$\Phi_\oslash よ^A_B(S, T) \cong \int^M \mathcal{C}(S, M \oslash A) \times \mathcal{C}(M \oslash B, T).$$

This description automatically provides a graphical calculus for optics as a particular case of the graphical calculus for Kleisli categories. Every diagram $\mathsf{Kl}(よ_{S,T}, よ_{A,B})$ is, by definition, an optic from (S, T) to (A, B).

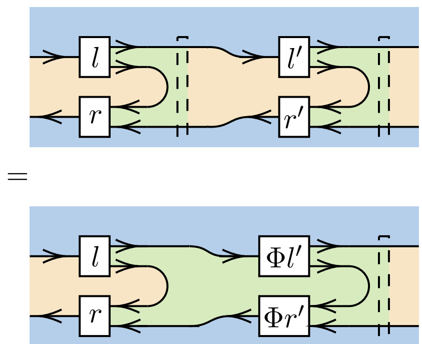
We can use the richer graphical language of the bicategory *Prof* for representing the involved profunctors. For a monoidal action $\oslash \colon \mathcal{M} \times \mathcal{C} \to \mathcal{C}$, we call $よ^{A}_{\oslash} \coloneqq \mathcal{C}(= \oslash A, -)$ and $よ^{\oslash}_{B} \coloneqq \mathcal{C}(= \oslash -, B)$, and we denote these by directed arrows. Observe that $よ^{A}_{B} \cong よ^{A}_{*} \diamond よ^{*}_{B}$, where $* \colon \mathbf{1} \times \mathcal{C} \to \mathcal{C}$ is the trivial action from the terminal category; but also that $\Phi_{\oslash} よ^{A}_{B} \cong よ^{A}_{\oslash} \diamond よ^{\oslash}_{B}$. All of this justifies the following representation of a generic optic for an action $(\oslash)$ on this calculus. Note how the monad can be said to just *change the color* of an internal region.



Note that then, in the Kleisli category, each morphism it can be written in a canonical way as follows. This is a recurrent observation that is reported multiple times independently in the literature on optics [Ril18, Boi20].

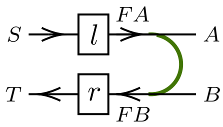


Composition of optics for the same family uses the multiplication of the monad, as in any Kleisli category.



3.1 Making it practical

In practice, we will want to relax the calculus by ommiting the colors of the regions. Moreover, we will draw the combination of a cap and a box as a colored regions. This simplification is justified by the fact that any optic (any morphism in a Kleisli category) can be rewritten in order to have a box after the cap (accumulate effects at the end). Now, $\mathcal{C}^{op} \times \mathcal{C}$ embeds fully faithfully into *Prof*$(\mathcal{C})$; so we can write the first part of the morphism always as two wires. On the other hand, we can introduce specific notation for the Kleisli effect as a cup, and end up writing every optic as follows.



We obtain a calculus of optics that consists of two wires: one covariant, another one contravariant; and caps between them joining the effects of an action ($\oslash$). This is, after all, the graphical language one wants to use for practical applications (§4). If pressed to justify the notation of every component, one can always go back to the full colored diagrams in the Kleisli category or their equivalent diagrams in the Eilenberg-Moore category of Tambara modules [Boi20].

4 Composing optics, an example

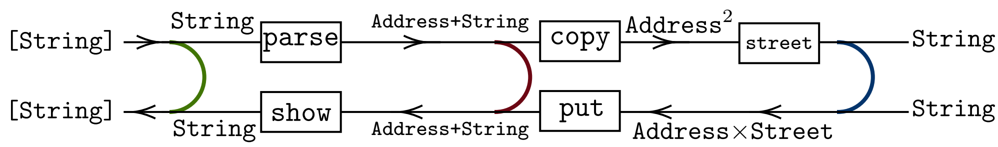
Arguably, what makes optics interesting is that we can compose optics of different *families* into a new optic. In other words, given two monoidal actions $\textcircled{M}\colon \mathcal{M} \times \mathcal{C} \to \mathcal{C}$ and $\textcircled{N}\colon \mathcal{N} \times \mathcal{C} \to \mathcal{C}$, the two comonads $\Theta_{\textcircled{M}}$ and $\Theta_{\textcircled{N}}$ have a coproduct that is again a monad of the same form. Explicitly, $\Theta_{\textcircled{M}+\textcircled{N}} \cong \Theta_{\textcircled{M}} \oplus \Theta_{\textcircled{N}}$ [Rom19].

```
let venues =
  [ "19 Albany Street, Pasadena, MD 21122"
  , "7 Hamilton Court Park, NY 11374"
  , "very.common@example.com"
  , "48 Grove Lane, Wallingford, CT 06492"
  ]

each :: Traversal [String] String
address :: Prism String Address
street :: Lens Address String

>>> venues ^. each . address . street %~ uppercase
[ "19 ALBANY STREET, Pasadena, MD 21122"
, "7 HAMILTON COURT PARK, NY 11374"
, "Error! not a postal address"
, "48 GROVE LANE, Wallingford, CT 06492"
]
```

Figure 1: A traversal (`each`), a prism (`address`) and a lens (`street`) are composed into a single optic that iterates over a list of strings, parses each one of them into some data structure, and modifies one of their subfields.

Because the graphical calculus of Kleisli categories helps with composing optics of different families, this author considers it particularly well-suited for describing applications of optics in functional programming. In functional programming, **optics** are a compositional representation of bidirectional data accessors, provided by libraries such as [Kme18]. Optics are divided into various *families*; each one of them encapsulating some data accessing pattern. For instance, *lenses* access subfields, *prisms* pattern match, and *traversals* iterate over containers. The usefulness of optics comes from the fact that any two of them can be composed, even if they are from different *families*. Optics function as building blocks for constructing complex data accessors, as it is exemplified in Figure 4.

Let us consider what the diagram for this optic composition looks on the newly described graphical calculus.



Because we are using traversals, prisms and lenses, we are describing an optic for any action that interprets polynomials, sums and products. Polynomials themselves offer such an interpretation, and thus, the complete optic can be interpreted as a product.

5 Related work

5.1 String diagrams for optics

The Pastro-Street monad $\Phi_{\oslash}$ is oplax [PS08]. It is routine to see that this is true by checking that its left adjoint comonad is lax. This makes its Eilenberg-Moore category to be monoidal, with the monoidal structure inherited from $\mathit{Prof}(\mathcal{C})$. The Kleisli category embeds fully faithfully into the Eilenberg-Moore category, but what makes this interesting for optics is that it allows us to express the *laws* of optics described by Riley [Ril18]. This is precisely the graphical calculus studied by Boisseau [Boi20]. We can embed diagrams on the Kleisli category to a particular Eilenberg-Moore category by the obvious inclusion sending every profunctor P to its free algebra ΦP. One can see that our previously defined $\Phi よ^{A}_{B}$ coincides with the pairs $R_A \diamond L_B$ in [Boi20].

How do both compare? The graphical calculus of the Eilenberg-Moore category is definitely more suited for studying lawfulness and the promonoidal structure of optics. Kleisli diagrams seem to be better suited for composing optics of different families sequentially in a less noisy way. In any case, one embeds directly into the other, so they can be regarded as different notation for essentially the same construction. Finally, there is an important difference in that the string diagrams of [Boi20] extend the Eilenberg-Moore category to mixed Pastro-Street comonads. We could repeat the same formalism for the Kleisli category of a mixed Pastro-Street comonad, but we have decided to avoid this added complexity on this note.

Acknowledgements

The author wants to thank Guillaume Boisseau and the reviewers of SYCO8 for their comments on a related work on composition of optics.

Mario Román was supported by the European Union through the ESF funded Estonian IT Academy research measure (project 2014-2020.4.05.19-0001).

References

[Boi20] Guillaume Boisseau. String diagrams for optics. *arXiv preprint arXiv:2002.11480*, 2020.

[Hed17] Jules Hedges. Coherence for lenses and open games. *CoRR*, abs/1704.02230, 2017.

[Kme18] Edward Kmett. lens library, version 4.16. Hackage `https://hackage.haskell.org/package/lens-4.16`, 2012–2018.

[PS08] Craig Pastro and Ross Street. Doubles for monoidal categories. *Theory and applications of categories*, 21(4):61–75, 2008.

[PW16] Maciej Piróg and Nicolas Wu. String diagrams for free monads (functional pearl). In *Proceedings of the 21st ACM SIGPLAN International Conference on Functional Programming*, pages 490–501, 2016.

[Ril18] Mitchell Riley. Categories of Optics. *arXiv preprint arXiv:1809.00738*, 2018.

[Rom19] Mario Román. Profunctor optics and traversals. *Master's thesis, University of Oxford*, 2019.