Higher-Order Focusing on Linearity and Effects

Siva Somayyajula

ssomayya@alumni.cmu.edu

Abstract

The relationship between effect calculi and focused logics has been extensively studied through their shared notion of polarization. We contribute another data point by extending Zeilberger's higher-order focused logic of delimited continuations to subsume the type structure of the enriched effect calculus. Then, we report ongoing work on modelling the linear usage of state via the interaction between the linear state monad and linear lenses.

Polarized calculi arise both as internal languages to adjunction models of effects [5] and as term assignments to focused logics [3]. The former, involving an adjunction between two categories, assigns a polarity to a type by the category from which it originates. In the latter, polarization arises from the assortation of logical rules on the basis of (non)invertibility, i.e., whether their application during proof search may require backtracking. The correspondence in type structure typically extends to terms and equational theories—for example, see Rioux and Zdancewic [20].

To phrase the question of interest, stated in the next paragraph, let us spell out the relationship between call-by-push-value (CBPV) [11] and focused intuitionistic logic [13, 21]. From the point of view of focusing, positive and negative types are allowed to respectively be right- and left-noninvertible. Dually, they *must* respectively be left- and right-invertible. This gives rise to four judgments: either inversion or focus on the left- or right-hand side of the sequent, each concerned with the iterated application of invertible or noninvertible rules. Modulo antecedent polarity [10], non-complex values and computations correspond to right focus resp. inversion terms. Non-complex stacks arising from the CK-machine semantics of CBPV [12] coincide with the left focus terms from *weakly* focused [22] intuitionistic logic.

Linearity in the enriched effect calculus (EEC) [7] arises by internalizing complex stacks via the linear function space, i.e., by reading a stack as a computation homomorphism. Unfortunately, the previous analogy appears to break down with the EEC when naively presented as a sequent calculus: the value types of (non)linear functions are not left-invertible. Moreover, the copower and computational sum types are not right-invertible. One solution is to decouple linearity from effects as Curien et al. [5] do with their effect calculi, which remain faithful to the focusing properties of intuitionistic linear logic, and whose models subsume those of the EEC. We approach the problem from a different angle: is there a judgmental reformulation of focused intuitionistic logic that can be *retrofitted* with linearity as conceived by the EEC?

Zeilberger lays the groundwork for a positive answer with the *higher-order* focused logic [23, 25] of delimited continuations [26]. Under higher-order focusing, inversion terms are metalevel maps out of focus terms. Not only are the derived logical rules trivially invertible, but admissibility of cut and identity is immediate. From there, our contribution is to "backpatch" into the logic the EEC connectives in question. Formally, fix right and left focus judgments $[P]$ and $N > P$ where P and N are positive and negative types. Then, the inversion judgments $\Gamma \vdash P$ and $\Gamma \vdash N$ are defined by the top row of inference rules in Figure 1 using the metalevel function space ($\Rightarrow$). In particular, a right inversion term is a stack-passing *value* parametric in the positive answer type; note that a negative answer type as in weak focusing would produce a circular definition. The formerly "problematic" connectives are given logical rules in the second and third rows of the same figure, renaming the positive ($\rightarrow$) to ($\supset$), I to $\mathbf{1}$ (not to be confused with 1, the positive unit), ($+$) to ($\oplus$), and ($\oplus$) to ($\invamp$).

$$
\begin{aligned}
\text{index sets } S &= \{\ell, k, \ldots\} \\
\text{positive types } P, Q &:= P \supset Q \mid 1 \mid P \times Q \mid \oplus\{\ell : P_\ell\}_{\ell \in S} \mid N \multimap\!\bullet\, P \mid N \multimap O \\
\text{negative types } M, N, O &:= P \to N \mid \mathbf{1} \mid P \otimes N \mid \parr\{\ell : N_\ell\}_{\ell \in S} \mid \&\{\ell : N_\ell\}_{\ell \in S} \\
\text{types } A &:= P \mid N \\
\text{contexts } \Gamma &:= \cdot \mid \Gamma, P \\
\text{judgments } &:= [P] \mid N > P \mid \Gamma \vdash A \mid N \gg O
\end{aligned}
$$

$$
\frac{[P]}{\cdot \vdash P} \qquad \frac{[P] \Rightarrow \Gamma \vdash Q}{\Gamma, P \vdash Q} \qquad \frac{\text{for all } P\text{: } N > P \Rightarrow \Gamma \vdash P}{\Gamma \vdash N} \qquad \frac{\text{for all } P\text{: } O > P \Rightarrow N > P}{N \gg O}
$$

$$
\frac{[P] \Rightarrow [Q]}{[P \supset Q]} \supset\text{R} \quad \frac{N > P}{[N \multimap\!\bullet\, P]} \multimap\!\bullet\,\text{R} \quad \frac{N \gg O}{[N \multimap O]} \multimap\text{R} \quad \frac{[P]}{\mathbf{1} > P} \mathbf{1}\text{L} \quad \frac{[P] \Rightarrow N > Q}{P \otimes N > Q} \otimes\text{L}
$$

$$
\frac{\{N_\ell > P\}_{\ell \in S}}{\parr\{\ell : N_\ell\}_{\ell \in S} > P} \parr\text{L} \quad \frac{N \gg O \quad O > P}{N > P} \text{cut}^{\gg} \quad \frac{N \gg M \quad M \gg O}{N \gg O} \text{cut}^{-}_{-} \quad \frac{\Gamma \vdash N \quad N > P}{\Gamma \vdash P} \text{cut}_{>}
$$

$$
\frac{\Gamma \vdash N \quad N \gg O}{\Gamma \vdash O} \text{cut}_{\gg} \qquad \frac{\Gamma \vdash P \quad \Gamma, P \vdash A}{\Gamma \vdash A} \text{cut}^{+} \qquad \frac{}{\Gamma, P \vdash P} \text{id}^{+} \qquad \frac{}{N \gg N} \text{id}^{-}
$$

Figure 1: Types, Judgments, and Selected Rules

The first step toward the linear function space is to note that the positive type ($\multimap\!\bullet$), which internalizes stacks, can be read as a generalized negation. To recover the negative codomain, our core observation is to define a new judgment $N \gg O$, internalized by the positive type $N \multimap O$, by applying contraposition, i.e., $N \multimap O \simeq (O \multimap\!\bullet\, P) \supset (N \multimap\!\bullet\, P)$ parametrically in P. The resulting terms are stack transformers [8] related to the `where` construct of complex stacks but opposite to linear continuation transformers [4].

On the one hand, this calculus subsumes the type structure of the EEC, but also diverges in the term assignment. On the other, we also depart from Zeilberger [26] by omitting explicit pattern variables and have the shift modalities between polarities $\downarrow N \triangleq \mathbf{1} \multimap N$ and $\uparrow P \triangleq P \otimes \mathbf{1}$ be definable rather than be primitive. In terms of metatheory, the expected cut and identity rules are *immediately* admissible—indicated by double lines, they are enumerated in the last two rows. For example, each cut rule amounts to function composition in the metatheory. One goal of this talk is to solicit feedback concerning the denotational and categorical semantics of this calculus; under *defunctionalization*, we review an option in the related work.

One practical and initial motivation of this work concerned tying stacks to the linear usage of state [17] by way of *lenses*. In particular, the type of *linear lenses* [19] $N \rightsquigarrow O$, representing a functional reference of type O within the type N, is definable as $N \multimap ((O \multimap N) \otimes O)$. As do Møgelberg and Staton [17], we can first define the linear state monad in P as $\text{State}_N(P) \triangleq N \multimap P \otimes N$ where N is the state type with the associated operations. From there, we can define a function of type $(N \rightsquigarrow O) \supset \text{State}_O(P) \supset \text{State}_N(P)$ that lifts mutations of an O-state into those of an N-state. However, we unsuccessfully attempted to systematically generate linear lenses by the structure of N, i.e., by converting $N \multimap\!\bullet\, P$ to $N \rightsquigarrow \uparrow P$. Looking forward, we are interested in resolving this issue as well as in determining the formal relationship of this calculus to its contemporaries (see below).

Related Work L-calculi [6, 18] and their descendant effect calculi [5] are related to higher-order focusing under defunctionalization [24], dualizing the exposition: reduction is *a priori* and the observation about inversion terms mapping out of focus terms is derived. Thus, it should be possible to determine the exact relationship between the presented calculus and the models of *ops. cit.* Separately, Zeilberger's original presentation [26] included type variables to enforce parametric use of answer types; if internalized, the resulting calculus would be comparable to polymorphic CBPV [20] with first-class stacks and stack manipulation [16, 9]. Lastly, an extension to dependent types would draw a correspondence to eMLTT [2, 1] and other polarized/focused dependent type theories [14, 15].

References

[1] Danel Ahman. Fibred Computational Effects. *CoRR*, abs/1710.02594, 2017.

[2] Danel Ahman, Neil Ghani, and Gordon D. Plotkin. Dependent Types and Fibred Computational Effects. In Bart Jacobs and Christof Löding, editors, *Foundations of Software Science and Computation Structures*, pages 36–54, Berlin, Heidelberg, 2016. Springer Berlin Heidelberg.

[3] Jean-Marc Andreoli. Logic programming with focusing proofs in linear logic. *Journal of Logic and Computation*, 2(3):297–347, 06 1992.

[4] Josh Berdine, Peter O'Hearn, Uday Reddy, and Hayo Thielecke. Linear Continuation-Passing. *Higher-Order and Symbolic Computation*, 15(2/3):181–208, 2002.

[5] Pierre-Louis Curien, Marcelo Fiore, and Guillaume Munch-Maccagnoni. A Theory of Effects and Resources: Adjunction Models and Polarised Calculi. In *Proceedings of the 43rd Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, POPL '16, pages 44–56, New York, NY, USA, 2016. Association for Computing Machinery.

[6] Pierre-Louis Curien and Guillaume Munch-Maccagnoni. The Duality of Computation under Focus. In Cristian S. Calude and Vladimiro Sassone, editors, *Theoretical Computer Science*, pages 165–181, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

[7] Jeff Egger, Rasmus Ejlers Møgelberg, and Alex Simpson. The enriched effect calculus: syntax and semantics. *Journal of Logic and Computation*, 24(3):615–654, 06 2012.

[8] Willem Heijltjes. The Functional Machine Calculus. *Electronic Notes in Theoretical Informatics and Computer Science*, Volume 1 - Proceedings of MFPS XXXVIII, Feb 2023.

[9] Yuchen Jiang, Runze Xue, and Max S. New. Notions of Stack-manipulating Computation and Relative Monads (Extended Version), 2025.

[10] Neel Krishnaswami. Focusing is not Call-by-Push-Value, 2014.

[11] Paul Blain Levy. Call-by-Push-Value: A Subsuming Paradigm. In Jean-Yves Girard, editor, *Typed Lambda Calculi and Applications*, pages 228–243, Berlin, Heidelberg, 1999. Springer Berlin Heidelberg.

[12] Paul Blain Levy. Adjunction Models For Call-By-Push-Value With Stacks. *Electronic Notes in Theoretical Computer Science*, 69:248–271, 2003. CTCS '02, Category Theory and Computer Science.

[13] Chuck Liang and Dale Miller. Focusing and Polarization in Intuitionistic Logic. In *Computer Science Logic*, Lausanne, Switzerland, September 2007.

[14] Daniel R. Licata and Robert Harper. Positively Dependent Types. In *Proceedings of the 3rd Workshop on Programming Languages Meets Program Verification*, PLPV '09, page 314, New York, NY, USA, 2009. Association for Computing Machinery.

[15] Étienne Miquey, Xavier Montillet, and Guillaume Munch-Maccagnoni. Dependent Type Theory in Polarised Sequent Calculus. In *TYPES 2020 - 26th International Conference on Types for Proofs and Programs*, pages 1–3, Torino, Italy, March 2020.

[16] Rasmus Ejlers Møgelberg and Alex Simpson. Relational Parametricity for Computational Effects. *Logical Methods in Computer Science*, Volume 5, Issue 3, Aug 2009.

[17] Rasmus Ejlers Møgelberg and Sam Staton. Linear usage of state. *Logical Methods in Computer Science*, Volume 10, Issue 1, Mar 2014.

[18] Guillaume Munch-Maccagnoni and Gabriel Scherer. Polarised Intermediate Representation of Lambda Calculus with Sums. In *Thirtieth Annual ACM/IEEE Symposium on Logic In Computer Science (LICS 2015)*, Kyoto, Japan, July 2015. Dec. 2015: see the added footnote on page 7.

[19] Mitchell Riley. Categories of Optics, 2018.

[20] Nick Rioux and Steve Zdancewic. Computation Focusing. *Proc. ACM Program. Lang.*, 4(ICFP), August 2020.

[21] Robert J. Simmons. Structural Focalization. *ACM Trans. Comput. Logic*, 15(3), September 2014.

[22] Robert J. Simmons and Frank Pfenning. Weak Focusing for Ordered Linear Logic. June 2008.

[23] Noam Zeilberger. On the unity of duality. *Annals of Pure and Applied Logic*, 153(1):66–96, 2008.

[24] Noam Zeilberger. Defunctionalizing Focusing Proofs. In *Proceedings of the 2009 International Workshop on Proof-Search in Type Theories*, PSTT 2009, 2009.

[25] Noam Zeilberger. *The Logical Basis of Evaluation Order and Pattern-Matching*. PhD thesis, USA, 2009. AAI3358066.

[26] Noam Zeilberger. Polarity and the Logic of Delimited Continuations. In *25th Annual IEEE Symposium on Logic in Computer Science*, pages 219–227, 2010.