# Normalization by Evaluation for the Lambek Calculus

Niccolò Veltri
Tallinn University of Technology, Tallinn, Estonia
niccolo@cs.ioc.ee

The syntactic calculus of Lambek is a deductive system for the multiplicative fragment of intuitionistic non-commutative linear logic. As a fine-grained calculus of resources, it has many applications, mostly in formal computational investigations of natural languages.

This paper introduces a calculus of $\beta\eta$-long normal forms for derivations in the Lambek calculus with multiplicative unit and conjunction among its logical connectives. Reduction to normal form follows the normalization by evaluation (NbE) strategy: (i) evaluate a derivation in a Kripke model of Lambek calculus; (ii) extract normal forms from the obtained semantic values. The implementation of the NbE algorithm requires the presence of a strong monad in the Kripke interpretation of positive formulae, in analogy with the extension of intuitionistic propositional logic with falsity and disjunction. The NbE algorithm can also be instantiated with minor variations to calculi for related substructural logics, such as multiplicative and dual intuitionistic linear logic (MILL and DILL).

## 1 Introduction

The syntactic calculus $\mathbf{L}$ of Lambek [17] is a deductive system which is primarily employed in mathematical studies of sentence structure in natural language. From a logical perspective, it provides a proof system for the multiplicative fragment of intuitionistic non-commutative linear logic [2, 21], comprising only of two ordered linear implications (or residuals) $\diagup$ and $\diagdown$, tensor product $\otimes$ and (often but not always) a unit $\mathsf{I}$ as logical connectives.

The metatheory of the Lambek calculus has been thoroughly developed in the past decades, in particular its categorical semantics by Lambek himself [18, 19]. The Lambek calculus enjoys cut elimination [17] and various normalization procedures, e.g. by Hepple for the implicational fragment [14] or more recently by Amblard and Retoré [6], aimed at the reduction of the proof search space and consequently the number of possible derivation of a given sequent. Various diagrammatic calculi and proof nets for the Lambek calculus have also been proposed [22, 16].

In this work, we study the natural deduction presentation of the Lambek calculus $\mathbf{L}$, together with a calculus $\mathbf{L}_{\beta\eta}$ consisting of $\beta\eta$-long normal forms, i.e. derivations that do not contain any redexes, and no further $\eta$-expansion is applicable. Sequents in $\mathbf{L}_{\beta\eta}$ have two shapes: $\Gamma \Uparrow A$, consisting of derivations in $\beta\eta$-long normal form, and $\Gamma \Downarrow A$, consisting of neutral derivations, i.e. (under Curry-Howard correspondence with a non-commutative linear variant of typed $\lambda$-calculus) a variable applied to other normal forms. The design of $\mathbf{L}_{\beta\eta}$ is inspired by the intercalation calculus for (non-commutative) linear logic [11]. It appears in particular as a fragment of the calculus of normal forms for ordered linear logic introduced by Polakov and Pfenning [21].

The normalization algorithm, sending each derivation in $\mathbf{L}$ to its $\beta\eta$-normal form in $\mathbf{L}_{\beta\eta}$, is an instance of *normalization by evaluation* (NbE) [10, 4]. NbE is a well-established normalization procedure for many variants of (typed and untyped) $\lambda$-calculus. The main idea behind NbE is that the effective normalization procedure factors through a categorical model of the syntactic calculus, usually a Kripke/presheaf model. Each formula $A$ can be interpreted as an object $[\![A]\!]$ of the model, and this

A. Indrzejczak, M. Zawidzki (Eds): 10th International Conference
on Non-Classical Logics. Theory and Applications (NCL 2022)
EPTCS 358, 2022, pp. 102–117, doi:10.4204/EPTCS.358.8

interpretation can be extended to contexts $[\![\Gamma]\!]$ as well. Moreover, each derivation $t$ of a sequent $\Gamma \vdash A$ in $\mathbf{L}$, for which we employ the syntax $t : \Gamma \vdash A$, is *evaluated* to a map $[\![t]\!] : [\![\Gamma]\!] \dot{\to} [\![A]\!]$. The Kripke model is precisely chosen so that the semantic value $[\![t]\!]$ contains enough information to allow the extraction of a normal form in $\mathbf{L}_{\beta\eta}$. This extraction process, sending semantic values to normal forms, is typically called *reification*. The presence of mixed-variance connectives, such as the ordered implications $\diagdown$ and $\diagup$, requires the construction of the reification function to be defined simultaneously with a *reflection* function, embedding neutrals in the Kripke model. The NbE procedure consists in the implementation of a function $\mathsf{nbe} : \Gamma \vdash A \to \Gamma \Uparrow A$ whose construction can be schematized as follows:

$$\begin{array}{ccccc} \text{(syntactic derivation)} & & \text{(Kripke semantic value)} & & (\beta\eta\text{-long normal form}) \\ t : \Gamma \vdash A & \longmapsto & [\![t]\!] : [\![\Gamma]\!] \dot{\to} [\![A]\!] & \longmapsto & \mathsf{nbe}\, t : \Gamma \Uparrow A \\ & \textbf{evaluation} & & \textbf{reification} & \end{array}$$

Normalization by evaluation for the implicational fragment of $\mathbf{L}$, not including unit and tensor, can be extrapolated from the normalization algorithm for the implicational fragment of ordered linear logic investigated by Polakov in his PhD thesis [21, 20]. Similarly, this could also be reconstructed from the recently developed NbE procedure for the natural deduction calculus of skew prounital closed categories [23].

Nevertheless, the presence of positive logical connectives $\mathsf{I}$ and $\otimes$ in combination with the negative implications $\diagdown$ and $\diagup$ does not make the NbE procedure for the implicational fragment of $\mathbf{L}$ directly extensible to the full calculus including also rules for $\mathsf{I}$ and $\otimes$. This situation sheds many similarities with the case of intuitionistic propositional logic with falsity and disjunction (or, equivalently, typed $\lambda$-calculus with empty type and sum types), where the standard Kripke semantics proves itself to be too weak for the extraction of normal forms. In the literature, the situation has been fixed in two different ways: employing a *strong monad* in the presheaf model for the interpretation of sums [5, 1, 24] or switching to more convoluted sheaf-theoretic models [3, 7]. In this paper we follow the first solution and similarly extend our presheaf model with a strong monad for recovering the reification procedure. This solution only helps with the implementation of NbE wrt. to a $\beta\eta$-conversion that does not include all possible permutative conversions, e.g. in the calculus $\mathbf{L}_{\beta\eta}$ the $\otimes$-introduction and $\otimes$-elimination rules do not commute. Defining NbE for a stronger conversion, removing all possible nondeterministic choices during proof search, is left to future work (more discussion on this in the conclusive section).

The main difference between NbE for the Lambek calculus $\mathbf{L}$ and intuitionistic propositional logic (or typed $\lambda$-calculus) is the extra bureaucracy that the substructural system $\mathbf{L}$, in which all the structural rules of weakening, contraction and exchange are absent, requires wrt. hypotheses/resources in context. Nevertheless, the presence or absence of the exchange rule does not seem to play a fundamental role. In fact, in the final part of the paper, we briefly discuss how to adapt NbE to the case of multiplicative intuitionistic linear logic (MILL) [9]. We also see a further extension to dual intuitionistic linear logic (DILL), which is a particular presentation of MILL extended with a linear exponential modality ! [8].

The material in the paper is organized as follows. Section 2 introduces the Lambek calculus $\mathbf{L}$ in natural deduction. Section 3 presents the calculus $\mathbf{L}_{\beta\eta}$ of $\beta\eta$-long normal forms. Section 4 discusses the NbE procedure: the construction of the Kripke model and the strong monad on it (Section 4.1), the interpretation of syntactic constructs (Section 4.2) and the reification/reflection algorithms for computing normal forms (Section 4.3). Section 5 proves the correctness of NbE using logical relations. Section 6 discusses extensions of NbE to MILL and DILL.

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash B \,/\, A}\, I_{/} \qquad \frac{A, \Gamma \vdash B}{\Gamma \vdash A \setminus B}\, I_{\setminus} \qquad \frac{\Gamma \vdash B \,/\, A \quad \Delta \vdash A}{\Gamma, \Delta \vdash B}\, E_{/} \qquad \frac{\Gamma \vdash A \quad \Delta \vdash A \setminus B}{\Gamma, \Delta \vdash B}\, E_{\setminus}$$

$$\frac{}{A \vdash A}\,\mathsf{ax} \qquad \frac{}{\vdash \mathsf{I}}\, I_{\mathsf{I}} \qquad \frac{\Gamma \vdash \mathsf{I} \quad \Delta_0, \Delta_1 \vdash C}{\Delta_0, \Gamma, \Delta_1 \vdash C}\, E_{\mathsf{I}}$$

$$\frac{\Gamma \vdash A \quad \Delta \vdash B}{\Gamma, \Delta \vdash A \otimes B}\, I_{\otimes} \qquad \frac{\Gamma \vdash A \otimes B \quad \Delta_0, A, B, \Delta_1 \vdash C}{\Delta_0, \Gamma, \Delta_1 \vdash C}\, E_{\otimes}$$

Figure 1: Inference rules of the Lambek calculus **L** in natural deduction

## 2 The Lambek Calculus in Natural Deduction

The Lambek calculus **L** has formulae generated by the grammar $A, B ::= p \mid \mathsf{I} \mid A \otimes B \mid B \,/\, A \mid A \setminus B$, where $p$ comes from a given set $\mathsf{At}$ of atomic formulae, $\mathsf{I}$ is the multiplicative unit, $\otimes$ is a multiplicative conjunction (a.k.a. tensor product), while $/$ and $\setminus$ are left and right implications (a.k.a. left and right residuals). Sequents in **L** are pairs $\Gamma \vdash A$, where $\Gamma$ is an ordered (possibly empty) list of formulae, that we call context, and $A$ is a single formula. The sets of formulae and contexts are called $\mathsf{Fma}$ and $\mathsf{Cxt}$, respectively. Derivations in **L** are generated by the inference rules in Figure 1. We write $t : \Gamma \vdash A$ to indicate that $t$ is a particular derivation of $\Gamma \vdash A$.

Substitution, i.e. the *cut rule*, is admissible in **L** [17]. As is common in the NbE literature, our cut rule allows the simultaneous substitution of multiple variables in context:

$$\frac{\Gamma \triangleright \Delta \quad \Delta \vdash A}{\Gamma \vdash A}\,\mathsf{cut} \tag{1}$$

where $\Gamma \triangleright \Delta$ denotes a set containing lists of derivations, called *environments*: given $\Gamma = \Gamma_1, \ldots, \Gamma_n$ and $\Delta = \langle A_1, \ldots, A_n \rangle$, an element of $\Gamma \triangleright \Delta$ consists of derivations of sequents $\Gamma_i \vdash A_i$, for all $i = 1, \ldots, n$. The relation $\triangleright$ can be defined inductively by the rules:

$$\frac{}{\langle\rangle \triangleright \langle\rangle}\,\langle\rangle \qquad \frac{\Gamma \vdash A \quad \Gamma' \triangleright \Delta}{\Gamma, \Gamma' \triangleright A, \Delta}\,\mathsf{cons}$$

In other words, the only environment in $\langle\rangle \triangleright \langle\rangle$ is the empty list of terms, denoted also by $\langle\rangle$. An environment in $\Gamma, \Gamma' \triangleright A, \Delta$ is a pair of a term $t : \Gamma \vdash A$ and another environment $\sigma : \Gamma' \triangleright \Delta$. We always write the pair as $(t, \sigma)$ instead of the more verbose $\mathsf{cons}\, t\, \sigma$. More generally, given two environments $\sigma_1 : \Gamma_1 \triangleright \Delta_1$ and $\sigma_2 : \Gamma_2 \triangleright \Delta_2$, we write $(\sigma_1, \sigma_2) : \Gamma_1, \Gamma_2 \triangleright \Delta_1, \Delta_2$ for their concatenation. Given a context $\Gamma$, recursively define the lists of variables $\mathsf{ids}_\Gamma : \Gamma \triangleright \Gamma$ by induction on $\Gamma$: $\mathsf{ids}_{\langle\rangle} =_{\mathrm{df}} \langle\rangle$ and $\mathsf{ids}_{A,\Gamma} =_{\mathrm{df}} (\mathsf{ax}, \mathsf{ids}_\Gamma)$. A more familiar-looking cut rule is derivable from (1) as follows:

$$\frac{t : \Gamma \vdash C \quad u : \Delta_0, C, \Delta_1 \vdash A}{\Delta_0, \Gamma, \Delta_1 \vdash A}\,\mathsf{sub} \quad =_{\mathrm{df}} \quad \frac{(\mathsf{ids}_{\Delta_0}, t, \mathsf{ids}_{\Delta_1}) : \Delta_0, \Gamma, \Delta_1 \triangleright \Delta_0, C, \Delta_1 \quad u : \Delta_0, C, \Delta_1 \vdash A}{\Delta_0, \Gamma, \Delta_1 \vdash A}\,\mathsf{cut}$$

The employment of $\mathsf{cut}$ in place of $\mathsf{sub}$ in NbE facilitates the statement and proof of correctness discussed in Section 5.

Derivations in **L** can be identified modulo a certain $\beta\eta$-equivalence relation $\sim$, which is the least congruence generated by the pairs of derivations in Figure 2. Equations include $\beta$- and $\eta$-conversions for all the logical connectives. Due to space limitations, derivations in these equations are displayed using an inline, term-like notation, but hopefully the reader will not struggle too hard in reconstructing

($\beta$-CONVERSIONS)

$$E_{/}\,(I_{/}\,t)\,u \sim \mathsf{sub}\,u\,t \qquad (t : \Gamma, A \vdash B,\ u : \Delta \vdash A)$$
$$E_{\backslash}\,u\,(I_{\backslash}\,t) \sim \mathsf{sub}\,u\,t \qquad (u : \Gamma \vdash A,\ t : A, \Delta \vdash B)$$
$$E_{\mathsf{I}}\,I_{\mathsf{I}}\,t \sim t \qquad (t : \Delta_0, \Delta_1 \vdash C)$$
$$E_{\otimes}\,(I_{\otimes}\,s_1\,s_2)\,t \sim \mathsf{sub}\,s_1\,(\mathsf{sub}\,s_2\,t) \qquad (s_i : \Gamma_i \vdash A_i,\ t : \Delta_0, A_1, A_2, \Delta_1 \vdash C)$$

($\eta$-CONVERSIONS)

$$t \sim I_{/}\,(E_{/}\,t\,\mathsf{ax}) \qquad (t : \Gamma \vdash B / A)$$
$$t \sim I_{\backslash}\,(E_{\backslash}\,\mathsf{ax}\,t) \qquad (t : \Gamma \vdash A \backslash B)$$
$$s \sim E_{\mathsf{I}}\,s\,I_{\mathsf{I}} \qquad (s : \Gamma \vdash \mathsf{I})$$
$$s \sim E_{\otimes}\,s\,(I_{\otimes}\,\mathsf{ax}\,\mathsf{ax}) \qquad (s : \Gamma \vdash A \otimes B)$$

(PERMUTATIVE CONVERSIONS)

$$E_{\mathsf{I}}\,s\,(I_{/}\,t) \sim I_{/}\,(E_{\mathsf{I}}\,s\,t) \qquad (s : \Gamma \vdash \mathsf{I},\ t : \Delta_0, \Delta_1, A \vdash B)$$
$$E_{\mathsf{I}}\,s\,(I_{\backslash}\,t) \sim I_{\backslash}\,(E_{\mathsf{I}}\,s\,t) \qquad (s : \Gamma \vdash \mathsf{I},\ t : A, \Delta_0, \Delta_1 \vdash B)$$
$$E_{/}\,(E_{\mathsf{I}}\,s\,t)\,u \sim E_{\mathsf{I}}\,s\,(E_{/}\,t\,u) \qquad (s : \Gamma \vdash \mathsf{I},\ t : \Delta_0, \Delta_1 \vdash B / A,\ u : \Omega \vdash A)$$
$$E_{/}\,t\,(E_{\mathsf{I}}\,s\,u) \sim E_{\mathsf{I}}\,s\,(E_{/}\,t\,u) \qquad (s : \Gamma \vdash \mathsf{I},\ t : \Omega \vdash B / A,\ u : \Delta_0, \Delta_1 \vdash A)$$
$$E_{\backslash}\,(E_{\mathsf{I}}\,s\,t)\,u \sim E_{\mathsf{I}}\,s\,(E_{\backslash}\,t\,u) \qquad (s : \Gamma \vdash \mathsf{I},\ t : \Delta_0, \Delta_1 \vdash A,\ u : \Omega \vdash A \backslash B)$$
$$E_{\backslash}\,t\,(E_{\mathsf{I}}\,s\,u) \sim E_{\mathsf{I}}\,s\,(E_{\backslash}\,t\,u) \qquad (s : \Gamma \vdash \mathsf{I},\ t : \Omega \vdash A,\ u : \Delta_0, \Delta_1 \vdash A \backslash B)$$
$$E_{\mathsf{I}}\,(E_{\mathsf{I}}\,s_1\,s_2)\,t \sim E_{\mathsf{I}}\,s_1\,(E_{\mathsf{I}}\,s_2\,t) \qquad (s_1 : \Gamma \vdash \mathsf{I},\ s_2 : \Delta_0, \Delta_1 \vdash \mathsf{I},\ t : \Omega_0, \Omega_1 \vdash C)$$
$$E_{\otimes}\,s\,(I_{/}\,t) \sim I_{/}\,(E_{\otimes}\,s\,t) \qquad (s : \Gamma \vdash A' \otimes B',\ t : \Delta_0, A', B', \Delta_1, A \vdash B)$$
$$E_{\otimes}\,s\,(I_{\backslash}\,t) \sim I_{\backslash}\,(E_{\otimes}\,s\,t) \qquad (s : \Gamma \vdash A' \otimes B',\ t : A, \Delta_0, A', B', \Delta_1 \vdash B)$$
$$E_{/}\,(E_{\otimes}\,s\,t)\,u \sim E_{\otimes}\,s\,(E_{/}\,t\,u) \qquad (s : \Gamma \vdash A' \otimes B',\ t : \Delta_0, A', B', \Delta_1 \vdash B / A,\ u : \Omega \vdash A)$$
$$E_{/}\,t\,(E_{\otimes}\,s\,u) \sim E_{\otimes}\,s\,(E_{/}\,t\,u) \qquad (s : \Gamma \vdash A' \otimes B',\ t : \Omega \vdash B / A,\ u : \Delta_0, A', B', \Delta_1 \vdash A)$$
$$E_{\backslash}\,(E_{\otimes}\,s\,t)\,u \sim E_{\otimes}\,s\,(E_{\backslash}\,t\,u) \qquad (s : \Gamma \vdash A' \otimes B',\ t : \Delta_0, A', B', \Delta_1 \vdash A,\ u : \Omega \vdash A \backslash B)$$
$$E_{\backslash}\,t\,(E_{\otimes}\,s\,u) \sim E_{\otimes}\,s\,(E_{\backslash}\,t\,u) \qquad (s : \Gamma \vdash A' \otimes B',\ t : \Omega \vdash A,\ u : \Delta_0, A', B', \Delta_1 \vdash A \backslash B)$$
$$E_{\otimes}\,(E_{\otimes}\,s_1\,s_2)\,t \sim E_{\otimes}\,s_1\,(E_{\otimes}\,s_2\,t) \qquad (s_1 : \Gamma \vdash A \otimes B,\ s_2 : \Delta_0, A, B, \Delta_1 \vdash A' \otimes B',\ t : \Omega_0, A', B', \Omega_1 \vdash C)$$

Figure 2: The $\beta\eta$-equivalence of derivations in $\mathbf{L}$

the associated proof trees. E.g. the reconstructed $\beta$-rules for right implication $/$ and multiplicative conjunction $\otimes$ look are:

$$\dfrac{\dfrac{t : \Gamma, A \vdash B}{\Gamma \vdash B / A}\, I_{/} \qquad u : \Delta \vdash A}{\Gamma, \Delta \vdash B}\, E_{/} \quad \sim \quad \dfrac{u : \Delta \vdash A \qquad t : \Gamma, A \vdash B}{\Gamma, \Delta \vdash B}\, \mathsf{sub}$$

$$\dfrac{\dfrac{s_1 : \Gamma_1 \vdash A_1 \qquad s_2 : \Gamma_2 \vdash A_2}{\Gamma_1, \Gamma_2 \vdash A_1 \otimes A_2}\, I_{\otimes} \qquad t : \Delta_0, A_1, A_2, \Delta_1 \vdash C}{\Delta_0, \Gamma_1, \Gamma_2, \Delta_1 \vdash C}\, E_{\otimes} \sim \dfrac{s_1 : \Gamma_1 \vdash A_1 \qquad \dfrac{s_2 : \Gamma_2 \vdash A_2 \qquad t : \Delta_0, A_1, A_2, \Delta_1 \vdash C}{\Delta_0, A_1, \Gamma_2, \Delta_1 \vdash C}\, \mathsf{sub}}{\Delta_0, \Gamma_1, \Gamma_2, \Delta_1 \vdash C}\, \mathsf{sub}$$

Equations in Figure 2 include also many permutative conversions, but not all of them. For example, the following equations are missing, stating that $\otimes$-introduction commutes with $\otimes$-elimination, and that two $\otimes$-eliminations can be swapped if they are independent, i.e. they act on disjoint collections of formulae

$$\frac{\Gamma, A \Uparrow B}{\Gamma \Uparrow B \diagup A}\, I_{\diagup} \qquad \frac{A, \Gamma \Uparrow B}{\Gamma \Uparrow A \diagdown B}\, I_{\diagdown} \qquad \frac{\Gamma \Downarrow B \diagup A \quad \Delta \Uparrow A}{\Gamma, \Delta \Downarrow B}\, E_{\diagup} \qquad \frac{\Gamma \Uparrow A \quad \Delta \Downarrow A \diagdown B}{\Gamma, \Delta \Downarrow B}\, E_{\diagdown}$$

$$\frac{}{A \Downarrow A}\, \mathsf{ax} \qquad \frac{}{\Uparrow \mathsf{I}}\, I_{\mathsf{I}} \qquad \frac{\Gamma \Downarrow \mathsf{I} \quad \Delta_0, \Delta_1 \Uparrow \gamma^+}{\Delta_0, \Gamma, \Delta_1 \Uparrow \gamma^+}\, E_{\mathsf{I}}$$

$$\frac{\Gamma \Uparrow A \quad \Delta \Uparrow B}{\Gamma, \Delta \Uparrow A \otimes B}\, I_{\otimes} \qquad \frac{\Gamma \Downarrow A \otimes B \quad \Delta_0, A, B, \Delta_1 \Uparrow \gamma^+}{\Delta_0, \Gamma, \Delta_1 \Uparrow \gamma^+}\, E_{\otimes} \qquad \frac{\Gamma \Downarrow p}{\Gamma \Uparrow p}\, \mathsf{sw}_{\Downarrow}$$

Figure 3: Inference rules of the calculus $\mathbf{L}_{\beta\eta}$ of $\beta\eta$-long normal forms

in the context:

$$\begin{array}{rr} I_\otimes\ (E_\otimes\ s\ t)\ u \sim E_\otimes\ s\ (I_\otimes\ t\ u) & (s : \Gamma \vdash A' \otimes B',\ t : \Delta_0, A', B', \Delta_1 \vdash A,\ u : \Omega \vdash B) \\ I_\otimes\ t\ (E_\otimes\ s\ u) \sim E_\otimes\ s\ (I_\otimes\ t\ u) & (s : \Gamma \vdash A' \otimes B',\ t : \Omega \vdash A,\ u : \Delta_0, A', B', \Delta_1 \vdash B) \\ E_\otimes\ s\ (E_\otimes\ s'\ t) \sim E_\otimes\ s'\ (E_\otimes\ s\ t) & (s : \Gamma \vdash A \otimes B,\ s' : \Gamma' \vdash A' \otimes B',\ t : \Delta_0, A, B, \Delta_1, A', B', \Delta_2 \vdash C) \end{array} \quad (2)$$

E.g. the proof trees in the first equation of (2) are:

$$\frac{\dfrac{s : \Gamma \vdash A' \otimes B' \quad t : \Delta_0, A', B', \Delta_1 \vdash A}{\Delta_0, \Gamma, \Delta_1 \vdash A}\, E_\otimes \quad u : \Omega \vdash B}{\Delta_0, \Gamma, \Delta_1, \Omega \vdash A \otimes B}\, I_\otimes \sim \frac{s : \Gamma \vdash A' \otimes B' \quad \dfrac{t : \Delta_0, A', B', \Delta_1 \vdash A \quad u : \Gamma' \vdash B}{\Delta_0, A', B', \Delta_1, \Omega \vdash A \otimes B}\, I_\otimes}{\Delta_0, \Gamma, \Delta_1, \Omega \vdash A \otimes B}\, E_\otimes$$

Analogous permutative conversions involving $I_\otimes$ and $E_{\mathsf{I}}$, two independent applications of $E_{\mathsf{I}}$, and an application of $E_{\mathsf{I}}$ independent from an application of $E_\otimes$, are missing as well. The study of the Lambek calculus with derivations identified by a richer system of equations, including the missing axioms discussed above, such as the ones in (2), is left for future work. Categorically speaking, the equational theory resulting from adding all the missing axioms corresponds to the one of monoidal biclosed categories. The Lambek calculus **L**, with derivations quotiented by the resulting richer congruence relation, would therefore be a presentation of the free monoidal biclosed category on the set $\mathsf{At}$.

## 3 A Calculus of Normal Forms

Canonical representatives of equivalence classes for the congruence $\sim$, i.e. $\beta\eta$-long normal forms, can be organized in a calculus called $\mathbf{L}_{\beta\eta}$. Sequents in $\mathbf{L}_{\beta\eta}$ have two shapes, $\Gamma \Uparrow A$ and $\Gamma \Downarrow A$. We employ the notation using $\Uparrow$ and $\Downarrow$ from [21], which is more generally used in focused sequent calculi for linear logic [11]. In the literature on intuitionistic propositional logic (i.e. typed $\lambda$-calculus, via Curry-Howard correspondence), derivations of $\Gamma \Uparrow A$ are called *normal forms*, while derivations of $\Gamma \Downarrow A$ are called *neutrals*. A calculus with similar backward- and forward-chaining phases is also typically called an *intercalation calculus*.

Derivations in $\mathbf{L}_{\beta\eta}$ are generated by the inference rules in Figure 3. The metavariable $\gamma^+$ indicates a *non-negative* formula, i.e. a formula which is not of the form $A \diagdown B$ nor $B \diagup A$. Elimination rules $E_{\mathsf{I}}$ and $E_\otimes$ are only applicable when the goal formula $\gamma^+$ in the conclusion is non-negative.

When restricting the attention on the implicational fragment of $\mathbf{L}_{\beta\eta}$, with only one implication $\diagdown$, it is possible to recognize in the rules of Figure 3 a goal-directed proof search strategy attempting to build a derivation in **L**. The construction of a valid derivation would proceed as follows: first eagerly apply the $\diagdown$-introduction rule, until the succedent becomes atomic; then switch to the neutral phase $\Downarrow$ and apply a sequence of $\diagdown$-eliminations rules to arguments in normal form, ultimately closing the derivation using

the ax rule. Under the Curry-Howard correspondence between **L** and a non-commutative linear (some people might say "planar" [23]) variant of $\lambda$-calculus, the derivations in this implicational fragment of $\mathbf{L}_{\beta\eta}$ correspond to weak head normal forms. The rules for the other implication $\diagup$ have analogous roles.

The introduction and elimination rules of $\otimes$ and $\mathsf{I}$ both produce normal forms, but the first premises of $E_{\otimes}$ and $E_{\mathsf{I}}$ are neutrals. The fact that all the rules involving the positive connectives $\otimes$ and $\mathsf{I}$ are in the same phase $\Uparrow$ shows that permutative conversions such as those in (2) do not hold (as syntactic equalities) in $\mathbf{L}_{\beta\eta}$. The formula $\gamma^+$ appearing in the rules $E_{\otimes}$ and $E_{\mathsf{I}}$ is required to be different from an implication, which fixes the relative position of $\diagdown$- and $\diagup$-introduction wrt. the elimination rules for the positive connectives $\otimes$ and $\mathsf{I}$.

Soundness of $\mathbf{L}_{\beta\eta}$ wrt. **L** is evident: each $\mathbf{L}_{\beta\eta}$-derivation can be embedded into **L** via functions $\mathsf{emb}_{\Uparrow} : \Gamma \Uparrow A \to \Gamma \vdash A$ and $\mathsf{emb}_{\Downarrow} : \Gamma \Downarrow A \to \Gamma \vdash A$, which simply change $\Uparrow$ and $\Downarrow$ to $\vdash$ and erase all uses of the rule $\mathsf{sw}_{\Downarrow}$.

# 4 Normalization by Evaluation

$\mathbf{L}_{\beta\eta}$ is also complete wrt. **L**. The proof of completeness corresponds to the construction of a normalization algorithm nbe, taking a derivation of $\Gamma \vdash A$ and returning a derivation of $\Gamma \Uparrow A$, satisfying the three following properties, for all derivations $t, u : \Gamma \vdash A$ and $n : \Gamma \Uparrow A$:

1. $t \sim u \to \mathsf{nbe}\, t = \mathsf{nbe}\, u$, which means that the normalization algorithm sends $\sim$-related derivations in **L** to syntactically equal derivations in $\mathbf{L}_{\beta\eta}$;
2. $t \sim \mathsf{emb}_{\Uparrow}\ (\mathsf{nbe}\, t)$, which means that each derivation in **L** is $\sim$-related to (the embedding of) its normal form;
3. $\mathsf{nbe}\ (\mathsf{emb}_{\Uparrow}\ n) = n$, which implies that each derivation in $\mathbf{L}_{\beta\eta}$ corresponds uniquely to an equivalence class of the relation $\sim$ in **L**.

The procedure nbe is defined following the *normalization by evaluation (NbE)* methodology [10, 4], consisting in the following steps: $(i)$ Find a model of **L** and its equational theory $\sim$, in our case (and in the majority of applications of NbE) a Kripke/presheaf model. This provides the existence of an element $[\![t]\!]$ in the model, for each derivation $t$ in **L**, such that $[\![t]\!] = [\![u]\!]$ whenever $t \sim u$; $(ii)$ Define a reification function, sending a semantic value in the Kripke model to a normal form in $\mathbf{L}_{\beta\eta}$, so that $\mathsf{nbe}\, t$ is defined as the reification of $[\![t]\!]$.

## 4.1 The Kripke Model

The model is defined as the presheaf category $\mathsf{Set}^{\mathsf{Cxt}}$. Explicitly, an object $P$ of the category $\mathsf{Set}^{\mathsf{Cxt}}$ is a Cxt-indexed family of sets, i.e. a presheaf: for any context $\Gamma$, $P\,\Gamma$ is a set[1]. A map $f$ between $P$ and $Q$ in $\mathsf{Set}^{\mathsf{Cxt}}$ is a Cxt-indexed family of functions, i.e. a natural transformation: for any context $\Gamma$, $f\,\Gamma$ is a function between $P\,\Gamma$ and $Q\,\Gamma$. We typically omit the index $\Gamma$, and simply write $f : P\,\Gamma \to Q\,\Gamma$. The set of maps between $P$ and $Q$ is denoted $P \dot{\to} Q$.

The category $\mathsf{Set}^{\mathsf{Cxt}}$ is *monoidal biclosed* (or, using Lambek terminology, *residual*) [18], with unit,

[1] We think of Cxt as a discrete category, which is why there is no mention of $P$'s action of maps.

tensor and internal homs given by

$$\begin{array}{rcl} \widehat{\mathsf{I}}\,\Gamma & =_{\mathrm{df}} & (\Gamma = \langle\rangle) \\ (P \mathbin{\widehat{\otimes}} Q)\,\Gamma & =_{\mathrm{df}} & \{\Gamma_0,\Gamma_1 : \mathsf{Cxt}\} \times \{\Gamma = \Gamma_0,\Gamma_1\} \times P\,\Gamma_0 \times Q\,\Gamma_1 \\ (P \mathbin{\widehat{/}} Q)\,\Gamma & =_{\mathrm{df}} & \{\Delta : \mathsf{Cxt}\} \to Q\,\Delta \to P\,(\Gamma,\Delta) \\ (Q \mathbin{\widehat{\backslash}} P)\,\Gamma & =_{\mathrm{df}} & \{\Delta : \mathsf{Cxt}\} \to Q\,\Delta \to P\,(\Delta,\Gamma) \end{array} \tag{3}$$

The unit $\widehat{\mathsf{I}}\,\Gamma$ is the singleton set which contains an element if and only if $\Gamma$ is empty. The tensor $(P \mathbin{\widehat{\otimes}} Q)\,\Gamma$ consists of pairs of an element of $P\,\Gamma_0$ and an element of $Q\,\Gamma_1$, for some contexts $\Gamma_0$ and $\Gamma_1$ such that $\Gamma = \Gamma_0,\Gamma_1$. The tensor product $\mathbin{\widehat{\otimes}}$ is often called *Day convolution*. The left (resp. right) internal hom $(Q \mathbin{\widehat{\backslash}} P)\,\Gamma$ (resp. $(P \mathbin{\widehat{/}} Q)\,\Gamma$) consists of functions from $Q\,\Delta$, for a given context $\Delta$, to $P\,(\Delta,\Gamma)$ (resp. $P\,(\Gamma,\Delta)$). The tensor and internal homs in $\mathsf{Set}^{\mathsf{Cxt}}$ form two adjunctions: there are natural bijective correspondences between the set of maps $P \mathbin{\widehat{\otimes}} Q \mathbin{\dot{\to}} R$ and the sets of maps $P \mathbin{\dot{\to}} R \mathbin{\widehat{/}} Q$ and $Q \mathbin{\dot{\to}} P \mathbin{\widehat{\backslash}} R$.

In (3) we have employed notation from Agda/Martin Löf type theory for the dependent sum and dependent product operations: $(x : A) \times B\,x$ and $(x : A) \to B\,x$ stand for $\sum_{x:A} B\,x$ and $\prod_{x:A} B\,x$ respectively, where $A$ is a set and $B$ is a family of sets indexed by $A$. Curly brackets indicate implicit arguments, e.g. when giving an element of $\{x : A\} \times B\,x$ it is sufficient to give an element $y : B\,x$ for some implicit $x : A$. For the readers more familiar with set-theoretic notation, the definitions in (3) can be rephrased as follows, where for clarity all the appearing arguments have been made explicit:

$$\begin{array}{rcl} \widehat{\mathsf{I}}\,\Gamma & =_{\mathrm{df}} & \{* \mid \Gamma \text{ is empty}\} \\ (P \mathbin{\widehat{\otimes}} Q)\,\Gamma & =_{\mathrm{df}} & \{(\Gamma_0,\Gamma_1,x,y) \mid \Gamma_0,\Gamma_1 : \mathsf{Cxt} \text{ such that } \Gamma = \Gamma_0,\Gamma_1, \text{ and } x : P\,\Gamma_0 \text{ and } y : Q\,\Gamma_1\} \\ (P \mathbin{\widehat{/}} Q)\,\Gamma & =_{\mathrm{df}} & \prod_{\Delta:\mathsf{Cxt}} Q\,\Delta \to P\,(\Gamma,\Delta) \\ (Q \mathbin{\widehat{\backslash}} P)\,\Gamma & =_{\mathrm{df}} & \prod_{\Delta:\mathsf{Cxt}} Q\,\Delta \to P\,(\Delta,\Gamma) \end{array}$$

The tensor $\mathbin{\widehat{\otimes}}$ is associative and unital wrt. $\widehat{\mathsf{I}}$ only up to natural isomorphism, i.e. $(P \mathbin{\widehat{\otimes}} Q) \mathbin{\widehat{\otimes}} S \cong P \mathbin{\widehat{\otimes}} (Q \mathbin{\widehat{\otimes}} S)$ and $\widehat{\mathsf{I}} \mathbin{\widehat{\otimes}} P \cong P \cong P \mathbin{\widehat{\otimes}} \widehat{\mathsf{I}}$, and the same is not true if we replace $\cong$ with $=$. To ease the readability of the forthcoming constructions, we leave implicit all applications of the natural isomorphisms of associativity and unitality in the paper.

The monoidal biclosed category $\mathsf{Set}^{\mathsf{Cxt}}$ is not completely suitable for the construction of an algorithm satisfying the NbE specification (more on this in Section 4.3). In analogy with the case of intuitionistic propositional logic with falsity and disjunction [1], we introduce a monad $T$ on $\mathsf{Set}^{\mathsf{Cxt}}$. For each presheaf $P$ and context $\Gamma$, the elements of the set $T\,P\,\Gamma$ are inductively generated by the following constructors:

$$\frac{P\,\Gamma}{T\,P\,\Gamma}\ \eta \qquad \frac{\Gamma \Downarrow \mathsf{I} \quad T\,P\,(\Delta_0,\Delta_1)}{T\,P\,(\Delta_0,\Gamma,\Delta_1)}\ E^T_{\mathsf{I}} \qquad \frac{\Gamma \Downarrow A \otimes B \quad T\,P\,(\Delta_0,A,B,\Delta_1)}{T\,P\,(\Delta_0,\Gamma,\Delta_1)}\ E^T_{\otimes} \tag{4}$$

Notice the similarity of the $E^T_{\mathsf{I}}$ and $E^T_{\otimes}$ constructors with the elimination rules $E_{\mathsf{I}}$ and $E_{\otimes}$ of $\mathbf{L}_{\beta\eta}$. In other words, elements of $T\,P\,\Gamma$ are lists of neutral terms with a positive formula (i.e. unit or tensor) in the succedent, ending with an element of $P\,\Gamma$. As it is usual in category theory, we also write $T : P \mathbin{\dot{\to}} Q \to T\,P \mathbin{\dot{\to}} T\,Q$ for the action on maps of $T$, and $\mu : T\,(T\,P) \mathbin{\dot{\to}} T\,P$ for the monad multiplication. The monad $T$ is left- and right-strong wrt. the monoidal structure $(\widehat{\mathsf{I}}, \mathbin{\widehat{\otimes}})$. Left strength is defined by recursion:

$$\begin{array}{l} \mathsf{lmst} : P \mathbin{\widehat{\otimes}} T\,Q \mathbin{\dot{\to}} T\,(P \mathbin{\widehat{\otimes}} Q) \\ \mathsf{lmst}\,(x, \eta\,y) =_{\mathrm{df}} \eta\,(x,y) \\ \mathsf{lmst}\,(x, E^T_{\mathsf{I}}\,t\,y) =_{\mathrm{df}} E^T_{\mathsf{I}}\,t\,(\mathsf{lmst}\,(x,y)) \\ \mathsf{lmst}\,(x, E^T_{\otimes}\,t\,y) =_{\mathrm{df}} E^T_{\otimes}\,t\,(\mathsf{lmst}\,(x,y)) \end{array} \tag{5}$$

Right-monoidal strength $\mathsf{rmst} : T\,P \mathbin{\widehat{\otimes}} Q \mathrel{\dot{\to}} T\,(P \mathbin{\widehat{\otimes}} Q)$ is defined analogously. The monoidal strengths are interdefinable with closed strengths $\mathsf{lcst}_{/} : P \mathbin{\widehat{/}} Q \mathrel{\dot{\to}} T\,P \mathbin{\widehat{/}} T\,Q$ and $\mathsf{rcst}_{/} : T\,(P \mathbin{\widehat{/}} Q) \mathrel{\dot{\to}} T\,P \mathbin{\widehat{/}} Q$, and also $\mathsf{lcst}_{\backslash}$ and $\mathsf{rcst}_{\backslash}$ obtained by turning $\widehat{/}$ into $\widehat{\backslash}$.

$$
\begin{array}{ll}
\mathsf{lcst}_{/} : P \mathbin{\widehat{/}} Q \mathrel{\dot{\to}} T\,P \mathbin{\widehat{/}} T\,Q & \mathsf{rcst}_{/} : T\,(P \mathbin{\widehat{/}} Q) \mathrel{\dot{\to}} T\,P \mathbin{\widehat{/}} Q \\
\mathsf{lcst}_{/}\ f\ (\eta\ x) \quad =_{\mathrm{df}} \eta\ (f\ x) & \mathsf{rcst}_{/}\ (\eta\ f)\ x \quad =_{\mathrm{df}} \eta\ (f\ x) \\
\mathsf{lcst}_{/}\ f\ (E^T_{\mathsf{I}}\ t\ x) =_{\mathrm{df}} E^T_{\mathsf{I}}\ t\ (\mathsf{lcst}_{/}\ f\ x) & \mathsf{rcst}_{/}\ (E^T_{\mathsf{I}}\ t\ f)\ x =_{\mathrm{df}} E^T_{\mathsf{I}}\ t\ (\mathsf{rcst}_{/}\ f\ x) \\
\mathsf{lcst}_{/}\ f\ (E^T_{\otimes}\ t\ x) =_{\mathrm{df}} E^T_{\otimes}\ t\ (\mathsf{lcst}_{/}\ f\ x) & \mathsf{rcst}_{/}\ (E^T_{\otimes}\ t\ f)\ x =_{\mathrm{df}} E^T_{\otimes}\ t\ (\mathsf{rcst}_{/}\ f\ x)
\end{array}
$$

The monad $T$ is reminiscent of a proof-relevant variant of the closure operator employed in phase semantics of non-commutative intuitionistic linear logic, appearing in the definition of non-commutative intuitionistic phase space [2].

## 4.2 Interpretation of Syntax

Each formula $A$ is interpreted as a presheaf $[\![A]\!]$:

$$
\begin{array}{ccc}
[\![p]\!] =_{\mathrm{df}} - \Uparrow p \qquad [\![\mathsf{I}]\!] =_{\mathrm{df}} T\,\widehat{\mathsf{I}} \qquad [\![A \otimes B]\!] =_{\mathrm{df}} T\,([\![A]\!] \mathbin{\widehat{\otimes}} [\![B]\!]) \\
[\![B \mathbin{/} A]\!] =_{\mathrm{df}} [\![B]\!] \mathbin{\widehat{/}} [\![A]\!] \qquad [\![A \mathbin{\backslash} B]\!] =_{\mathrm{df}} [\![A]\!] \mathbin{\widehat{\backslash}} [\![B]\!]
\end{array}
\tag{6}
$$

Notice the presence of the monad $T$ in the interpretation of the positive formulae $\mathsf{I}$ and $\otimes$. The interpretation of formulae extends to contexts via the monoidal structure of $\mathsf{Set}^{\mathsf{Cxt}}$: $[\![\langle\rangle]\!] =_{\mathrm{df}} \widehat{\mathsf{I}}$ and $[\![A,\Gamma]\!] =_{\mathrm{df}} [\![A]\!] \mathbin{\widehat{\otimes}} [\![\Gamma]\!]$. We use the same notation $[\![-]\!]$ for the interpretation of formulae and contexts (and later for the interpretation of derivations and environments), but it should always be clear which interpretation function is employed in each situation.

The interpretation of derivations of $\mathbf{L}$ in the Kripke model requires the monad $T$ to be *runnable* on each interpreted formula $[\![A]\!]$, i.e. we want a natural transformation $\mathsf{run}_A : T\,[\![A]\!] \mathrel{\dot{\to}} [\![A]\!]$. In turn, this requires the monad $T$ to be runnable on presheaves of the form $- \Uparrow A$. The function $\mathsf{run}_A$ is defined by induction on $A$, the interesting cases are $/$ and $\backslash$, which figure an application of right closed strengths $\mathsf{rcst}_{/}$ and $\mathsf{rcst}_{\backslash}$.

$$
\begin{array}{ll}
\mathsf{run}^{\Uparrow} : T\,(- \Uparrow A) \mathrel{\dot{\to}} - \Uparrow A & \mathsf{run}_A : T\,[\![A]\!] \mathrel{\dot{\to}} [\![A]\!] \\
\mathsf{run}^{\Uparrow}\ (\eta\ t) \quad =_{\mathrm{df}} \quad t & \mathsf{run}_p\ t \quad =_{\mathrm{df}} \quad \mathsf{run}^{\Uparrow}\ t \\
\mathsf{run}^{\Uparrow}\ (E^T_{\mathsf{I}}\ t\ u) \quad =_{\mathrm{df}} \quad E_{\mathsf{I}}\ t\ (\mathsf{run}^{\Uparrow}\ u) & \mathsf{run}_{\mathsf{I}}\ t \quad =_{\mathrm{df}} \quad \mu\ t \\
\mathsf{run}^{\Uparrow}\ (E^T_{\otimes}\ t\ u) \quad =_{\mathrm{df}} \quad E_{\otimes}\ t\ (\mathsf{run}^{\Uparrow}\ u) & \mathsf{run}_{A \otimes B}\ t \quad =_{\mathrm{df}} \quad \mu\ t \\
 & \mathsf{run}_{B/A}\ t \quad =_{\mathrm{df}} \quad \lambda x.\, \mathsf{run}_B\ (\mathsf{rcst}_{/}\ t\ x) \\
 & \mathsf{run}_{A\backslash B}\ t \quad =_{\mathrm{df}} \quad \lambda x.\, \mathsf{run}_B\ (\mathsf{rcst}_{\backslash}\ t\ x)
\end{array}
\tag{7}
$$

Each derivation $t : \Gamma \vdash A$ in $\mathbf{L}$ is interpreted as a natural transformation $[\![t]\!] : [\![\Gamma]\!] \mathrel{\dot{\to}} [\![A]\!]$. The evaluation function is defined by recursion on the argument $t$. The interesting cases are: $I_{\mathsf{I}}$ and $I_{\otimes}$, where an extra application of the monad unit $\eta$ is required; $E_{\mathsf{I}}$ and $E_{\otimes}$, which involves both monoidal strengths $\mathsf{lmst}$ and

rmst, and the run function.

$$
\begin{array}{lll}
[\![\mathsf{ax}]\!]\ a & =_{\mathrm{df}} & a \\
[\![I_{/}\ t]\!]\ \gamma & =_{\mathrm{df}} & \lambda x.\,[\![t]\!]\ (\gamma, x) \\
[\![I_{\backslash}\ t]\!]\ \gamma & =_{\mathrm{df}} & \lambda x.\,[\![t]\!]\ (x, \gamma) \\
[\![I_{\mathsf{I}}]\!]\ \gamma & =_{\mathrm{df}} & \eta\ \gamma \\
[\![I_{\otimes}\ t\ u]\!]\ (\gamma, \delta) & =_{\mathrm{df}} & \eta\ ([\![t]\!]\ \gamma, [\![u]\!]\ \delta) \\
[\![E_{/}\ t\ u]\!]\ (\gamma, \delta) & =_{\mathrm{df}} & [\![t]\!]\ \gamma\ ([\![u]\!]\ \delta) \\
[\![E_{\backslash}\ t\ u]\!]\ (\gamma, \delta) & =_{\mathrm{df}} & [\![u]\!]\ \delta\ ([\![t]\!]\ \gamma) \\
[\![E_{\mathsf{I}}\ t\ u]\!]\ (\delta_0, \gamma, \delta_1) & =_{\mathrm{df}} & \mathsf{run}\ (T\ [\![u]\!]\ (\mathsf{rmst}\ (\mathsf{lmst}\ (\delta_0, [\![t]\!]\ \gamma), \delta_1))) \\
[\![E_{\otimes}\ t\ u]\!]\ (\delta_0, \gamma, \delta_1) & =_{\mathrm{df}} & \mathsf{run}\ (T\ [\![u]\!]\ (\mathsf{rmst}\ (\mathsf{lmst}\ (\delta_0, [\![t]\!]\ \gamma), \delta_1)))
\end{array}
\tag{8}
$$

Pictorially, $[\![E_{\otimes}\ t\ u]\!]$ is the following composite natural transformation (remember that in our definitions, and in particular in the definition of $[\![E_{\otimes}\ t\ u]\!]$ above, applications of the associativity natural isomorphism of $\widehat{\otimes}$ are omitted):

$$
\begin{array}{l}
[\![\Delta_0, \Gamma, \Delta_1]\!] \xrightarrow{\cong} ([\![\Delta_0]\!] \mathbin{\widehat{\otimes}} [\![\Gamma]\!]) \mathbin{\widehat{\otimes}} [\![\Delta_1]\!] \xrightarrow{(\mathsf{id}\widehat{\otimes}[\![t]\!])\widehat{\otimes}\mathsf{id}} ([\![\Delta_0]\!] \mathbin{\widehat{\otimes}} T\ ([\![A]\!] \mathbin{\widehat{\otimes}} [\![B]\!])) \mathbin{\widehat{\otimes}} [\![\Delta_1]\!] \\
\qquad\qquad \xrightarrow{\mathsf{lmst}\widehat{\otimes}\mathsf{id}} T\ ([\![\Delta_0]\!] \mathbin{\widehat{\otimes}} ([\![A]\!] \mathbin{\widehat{\otimes}} [\![B]\!])) \mathbin{\widehat{\otimes}} [\![\Delta_1]\!] \\
\qquad\qquad \xrightarrow{\mathsf{rmst}} T\ (([\![\Delta_0]\!] \mathbin{\widehat{\otimes}} ([\![A]\!] \mathbin{\widehat{\otimes}} [\![B]\!])) \mathbin{\widehat{\otimes}} [\![\Delta_1]\!]) \\
\qquad\qquad \xrightarrow{\cong} T\ ([\![\Delta_0, A, B, \Delta_1]\!]) \\
\qquad\qquad \xrightarrow{T\ [\![u]\!]} T\ [\![C]\!] \xrightarrow{\mathsf{run}_C} [\![C]\!]
\end{array}
$$

The evaluation function $[\![-]\!]$ is well-defined on $\sim$-equivalence classes: given two derivations $t, u : \Gamma \vdash A$, if $t \sim u$ then $[\![t]\!] = [\![u]\!]$. The proof of this fact relies on $T$ being a strong monad and run being an algebra for the monad $T$. For example $\mathsf{run}_A\ (\eta\ x) = x$, for all $x : [\![A]\!]\ \Gamma$.

The interpretation of derivations can be readily extended to environments $[\![\sigma]\!] : [\![\Gamma]\!] \dot{\to} [\![\Delta]\!]$, for each $\sigma : \Gamma \triangleright \Delta$: $[\![\langle\rangle]\!]\ x =_{\mathrm{df}} x$ and $[\![(t, \sigma)]\!]\ (\gamma, \gamma') =_{\mathrm{df}} ([\![t]\!]\ \gamma, [\![\sigma]\!]\ \gamma')$. The evaluation function behaves well wrt. substitution: $[\![\mathsf{cut}\ \sigma\ t]\!] = [\![t]\!] \circ [\![\sigma]\!]$, i.e. the cut rule is interpreted as function composition in the Kripke model.

### 4.3 The Normalization Function

The last phase of the NbE procedure is the extraction of normal forms from elements of the Kripke model. Concretely, this corresponds to the construction of a *reification* function $\downarrow_A : [\![A]\!] \dot{\to} - \Uparrow A$. In order to deal with the cases of mixed-variance connectives $/$ and $\backslash$, it is necessary to simultaneously define a *reflection* procedure $\uparrow_A : - \Downarrow A \dot{\to} [\![A]\!]$, embedding neutrals in the presheaf model. This is the crucial point were the interpretation of the unit $[\![\mathsf{I}]\!] =_{\mathrm{df}} T\ \widehat{\mathsf{I}}$ and the tensor product $[\![A \otimes B]\!] =_{\mathrm{df}} T\ ([\![A]\!] \mathbin{\widehat{\otimes}} [\![B]\!])$ in (6) works, while naïve interpretations $[\![\mathsf{I}]\!] =_{\mathrm{df}} \widehat{\mathsf{I}}$ and $[\![A \otimes B]\!] =_{\mathrm{df}} [\![A]\!] \mathbin{\widehat{\otimes}} [\![B]\!]$ without the application of the monad $T$ would fail. With the latter interpretation, $\uparrow_{\mathsf{I}}\ t$ would be required to have type $\widehat{\mathsf{I}}\ \Gamma$, which in turn will force us to show that the context $\Gamma$ is empty. But this is generally false, e.g. when $t$ is of the form $\mathsf{ax} : \mathsf{I} \Downarrow \mathsf{I}$. Analogously, $\uparrow_{A \otimes B}\ t$ would be required to have type $([\![A]\!] \mathbin{\widehat{\otimes}} [\![B]\!])\ \Gamma$, which in turn will force us to split the context $\Gamma = \Gamma_0, \Gamma_1$ and provide elements of type $[\![A]\!]\ \Gamma_0$ and $[\![B]\!]\ \Gamma_1$. But this split is generally impossible to achieve, e.g. for $t$ of the form $\mathsf{ax} : A \otimes B \Downarrow A \otimes B$. This problematic splitting in the definition of reflection $\uparrow_A$ is avoided by the use of the monad $T$, and the cases of $A$ being unit or tensor are dealt via the application of the constructors $E^T_{\mathsf{I}}$ and $E^T_{\otimes}$. The employment of a strong monad seems to be a general

pattern in NbE for calculi including positive logical connectives, e.g. consider falsity and disjunction in intuitionistic propositional logic [5, 1].

$$
\begin{array}{llll}
\downarrow_p t & =_{\mathrm{df}} & t & \uparrow_p t \quad =_{\mathrm{df}} \quad \mathsf{sw}_{\Downarrow} t \\
\downarrow_{B/A} t & =_{\mathrm{df}} & I_{/}\,(\downarrow_B (t\,(\uparrow_A \mathsf{ax}))) & \uparrow_{B/A} t \quad =_{\mathrm{df}} \quad \lambda x.\, \uparrow_B (E_{/}\, t\, (\downarrow_A x)) \\
\downarrow_{A\backslash B} t & =_{\mathrm{df}} & I_{\backslash}\,(\downarrow_B (t\,(\uparrow_A \mathsf{ax}))) & \uparrow_{A\backslash B} t \quad =_{\mathrm{df}} \quad \lambda x.\, \uparrow_B (E_{\backslash}\, (\downarrow_A x)\, t) \\
\downarrow_{\mathsf{I}} t & =_{\mathrm{df}} & \mathsf{run}^{\Uparrow}\,(T\,(\lambda\, \mathsf{refl}.\, I_{\mathsf{I}})\, t) & \uparrow_{\mathsf{I}} t \quad =_{\mathrm{df}} \quad E^T_{\mathsf{I}}\, t\, (\eta\ \mathsf{refl}) \\
\downarrow_{A\otimes B} t & =_{\mathrm{df}} & \mathsf{run}^{\Uparrow}\,(T\,(\lambda(x,y).I_{\otimes}\,(\downarrow_A x)\,(\downarrow_B y))\,t) & \uparrow_{A\otimes B} t \quad =_{\mathrm{df}} \quad E^T_{\otimes}\, t\, (\eta\ (\uparrow_A \mathsf{ax}, \uparrow_B \mathsf{ax}))
\end{array}
\tag{9}
$$

The reflection function $\uparrow_A$ can also be used for the definition of an element $\mathsf{fresh}_\Gamma : [\![\Gamma]\!]\,\Gamma$, for each context $\Gamma$: $\mathsf{fresh}_{\langle\rangle} =_{\mathrm{df}} \mathsf{refl}$ and $\mathsf{fresh}_{A,\Gamma} =_{\mathrm{df}} (\uparrow_A \mathsf{ax}, \mathsf{fresh}_\Gamma)$. When $\Gamma = \langle\rangle$, the element $\mathsf{fresh}_{\langle\rangle}$ has type $[\![\langle\rangle]\!]\,\langle\rangle$, which reduces to $\langle\rangle = \langle\rangle$, and refl is the proof of reflexivity of equality.

The normalization function $\mathsf{nbe} : \Gamma \vdash A \to \Gamma \Uparrow A$ is then definable as the reification of the evaluation of a derivation $t : \Gamma \vdash A$ in the Kripke model:

$$\mathsf{nbe}\, t \quad =_{\mathrm{df}} \quad \downarrow_A ([\![t]\!]\ \mathsf{fresh}_\Gamma)$$

Here we consider the interpretation $[\![t]\!] : [\![\Gamma]\!]\,\Gamma \to [\![A]\!]\,\Gamma$, which can be applied to $\mathsf{fresh}_\Gamma : [\![\Gamma]\!]\,\Gamma$. Since $[\![t]\!] = [\![u]\!]$ for all $t \sim u$, then the function nbe sends $\sim$-related derivations in $\mathbf{L}$ to equal derivations in $\mathbf{L}_{\beta\eta}$, i.e. $\mathsf{nbe}\, t = \mathsf{nbe}\, u$ whenever $t \sim u$.

**Example.** Let $p, q, r$ be atomic formulae and let $t : p \otimes q, r \vdash p \otimes (q \otimes r)$ be the following derivation:

$$
\dfrac{\dfrac{\dfrac{}{p\otimes q \vdash p\otimes q}\,\mathsf{ax}}{p\otimes q, r \vdash p\otimes(q\otimes r)}\,E_{\otimes}}{p\otimes q, r \vdash p\otimes(q\otimes r)}\,E_{\otimes}
\qquad
\begin{array}{c}
\dfrac{\dfrac{}{p\vdash p}\,\mathsf{ax} \quad \dfrac{}{q\vdash q}\,\mathsf{ax}}{p,q\vdash p\otimes q}\,I_{\otimes} \quad \dfrac{\dfrac{}{p\vdash p}\,\mathsf{ax} \quad \dfrac{\dfrac{}{q\vdash q}\,\mathsf{ax} \quad \dfrac{}{r\vdash r}\,\mathsf{ax}}{q,r\vdash q\otimes r}\,I_{\otimes}}{p,q,r\vdash p\otimes(q\otimes r)}\,I_{/}}{p,q\vdash (p\otimes(q\otimes r))/r}\,E_{\otimes} \quad \dfrac{}{r\vdash r}\,\mathsf{ax}\\
\hline
p,q\vdash (p\otimes(q\otimes r))/r
\end{array}
\,E_{/}
$$

It is possible to check that the evaluation $[\![t]\!]$, when applied to the element $\mathsf{fresh}_{p\otimes q,r} : [\![p\otimes q, r]\!]\,(p\otimes q, r)$, returns the following element of $T\,([\![p]\!]\,\widehat{\otimes}\,(T\,([\![q]\!]\,\widehat{\otimes}\,[\![r]\!]))\,(p\otimes q, r)$:

$$
[\![t]\!]\,(\mathsf{fresh}_{p\otimes q,r}) \;=\;
\dfrac{\dfrac{}{p\otimes q \Downarrow p\otimes q}\,\mathsf{ax} \quad \dfrac{\dfrac{\dfrac{}{p\Downarrow p}\,\mathsf{ax}}{p\Uparrow p}\,\mathsf{sw}_{\Downarrow} \quad \dfrac{\dfrac{\dfrac{}{q\Downarrow q}\,\mathsf{ax}}{q\Uparrow q}\,\mathsf{sw}_{\Downarrow} \quad \dfrac{\dfrac{}{r\Downarrow r}\,\mathsf{ax}}{r\Uparrow r}\,\mathsf{sw}_{\Downarrow}}{T\,([\![q]\!]\,\widehat{\otimes}\,[\![r]\!])\,(q,r)}\,\eta}{T\,([\![p]\!]\,\widehat{\otimes}\,(T\,([\![q]\!]\,\widehat{\otimes}\,[\![r]\!])))\,(p,q,r)}\,\eta}{T\,([\![p]\!]\,\widehat{\otimes}\,(T\,([\![q]\!]\,\widehat{\otimes}\,[\![r]\!])))\,(p\otimes q, r)}\,E^T_{\otimes}
\tag{10}
$$

(In the applications of the monad unit $\eta$ above, the definition of $\widehat{\otimes}$ is automatically unfolded). The application of the function nbe to $t$, corresponding to the application of the reification map $\downarrow_{p\otimes(q\otimes r)}$ to the element in (10), returns the following $\beta\eta$-long normal form in $\mathbf{L}_{\beta\eta}$:

$$
\mathsf{nbe}\, t \;=\;
\dfrac{\dfrac{}{p\otimes q \Downarrow p\otimes q}\,\mathsf{ax} \quad \dfrac{\dfrac{\dfrac{}{p\Downarrow p}\,\mathsf{ax}}{p\Uparrow p}\,\mathsf{sw}_{\Downarrow} \quad \dfrac{\dfrac{\dfrac{}{q\Downarrow q}\,\mathsf{ax}}{q\Uparrow q}\,\mathsf{sw}_{\Downarrow} \quad \dfrac{\dfrac{}{r\Downarrow r}\,\mathsf{ax}}{r\Uparrow r}\,\mathsf{sw}_{\Downarrow}}{q,r\Uparrow q\otimes r}\,I_{\otimes}}{p,q,r\Uparrow p\otimes(q\otimes r)}\,E_{\otimes}}{p\otimes q, r\Uparrow p\otimes(q\otimes r)}
$$

## 5 Correctness of NbE

So far we have constructed an effective normalization procedure sending $\beta\eta$-related derivations in $\mathbf{L}$ to the same normal form in $\mathbf{L}_{\beta\eta}$. It remains to show that this procedure is *correct*, which is to say that $\mathsf{nbe}$ is a bijective function with $\mathsf{emb}_{\Uparrow}$ as its inverse, up to the equivalence of derivations $\sim$. This in turn implies that each derivation in $\mathbf{L}$ is $\beta\eta$-equivalent to (the embedding of) its normal form, and each derivation in $\mathbf{L}_{\beta\eta}$ is the unique representative of a $\sim$-equivalence class in $\mathbf{L}$.

The latter property of the normalization function, corresponding to the surjectivity of $\mathsf{nbe}$, is straightforward. Formally, it is possible to prove the following statements by structural induction on the input normal forms and neutrals.

**(surjectivity of** $\mathsf{nbe}$**)** For any neutral $n : \Gamma \Downarrow A$, we have $\uparrow_A n = [\![\mathsf{emb}_{\Downarrow}\, n]\!]\, \mathsf{fresh}_\Gamma$.
For any normal form $n : \Gamma \Uparrow A$, we have $n = \mathsf{nbe}\,(\mathsf{emb}_{\Uparrow} n)$.

The injectivity of the normalization function $\mathsf{nbe}$ can be proved employing a *logical relation* $\approx_A$, which relates each derivation of a sequent $\Gamma \vdash A$ in $\mathbf{L}$ with the corresponding denoting values of $[\![A]\!]\,\Gamma$ in the Kripke model. It is mutually-inductively defined together with another relation $\approx^{\otimes}$, whose definition is given below. The generating inference rules of $\approx_A$ are:

$$\frac{t \sim \mathsf{emb}_{\Uparrow} n}{t \approx_p n} \qquad \frac{t\,(\overline{T}\approx^{\mathsf{I}})\,v}{t \approx_{\mathsf{I}} v} \qquad \frac{t\,(\overline{T}\approx^{\otimes})\,v}{t \approx_{A\otimes B} v}$$
$$\frac{\{\Delta : \mathsf{Cxt}\}\ (u : \Delta \vdash A)\ (a : [\![A]\!]\,\Delta) \to u \approx_A a \to E_{/}\, t\, u \approx_B f\, a}{t \approx_{B/A} f} \tag{11}$$
$$\frac{\{\Delta : \mathsf{Cxt}\}\ (u : \Delta \vdash A)\ (a : [\![A]\!]\,\Delta) \to u \approx_A a \to E_{\backslash}\, u\, t \approx_B f\, a}{t \approx_{A\backslash B} f}$$

The definition of the logical relation is very similar to the one used to prove correctness of normalization for (the implicational fragment of) ordered linear logic [20], but the cases of the positive connectives $\mathsf{I}$ and $\otimes$ require additional care because of the presence of the monad $T$ in the interpretation of formulae. To this end, the rules in (11) make use of the following auxiliary notions:

- Given a relation $R \subseteq (\Gamma \vdash C) \times (P\,\Gamma)$ between syntactic derivations and semantic values, the *lifting* of $R$ to the monad $T$ is a relation $\overline{T}R \subseteq (\Gamma \vdash C) \times (T\ P\,\Gamma)$, whose validity is specified inductively by the following rules:

$$\frac{t\,R\,x}{t\,(\overline{T}R)\,(\eta\ x)} \qquad \frac{t' : \Delta_0, \Delta_1 \vdash C \quad t \sim E_{\mathsf{I}}\,(\mathsf{emb}_{\Downarrow} u)\,t' \quad t'\,(\overline{T}R)\,v}{t\,(\overline{T}R)\,E^T_{\mathsf{I}}\ u\ v}$$
$$\frac{t' : \Delta_0, A, B, \Delta_1 \vdash C \quad t \sim E_{\otimes}\,(\mathsf{emb}_{\Downarrow} u)\,t' \quad t'\,(\overline{T}R)\,v}{t\,(\overline{T}R)\,E^T_{\otimes}\ u\ v}$$

  The second rule should be read as: a derivation $t : \Delta_0, \Gamma, \Delta_1 \vdash C$ is $(\overline{T}R)$-related to $E^T_{\mathsf{I}}\ u\ v$, for some neutral $u : \Gamma \Downarrow \mathsf{I}$ and monadic value $v : T\ P\ (\Delta_0, \Delta_1)$, if and only if $t$ is $\beta\eta$-equivalent to a unit-elimination $E_{\mathsf{I}}\ (\mathsf{emb}_{\Downarrow} u)\ t'$, for some derivation $t' : \Delta_0, \Delta_1 \vdash C$, such that $t'\,(\overline{T}R)\,v$. The third rule should be read analogously.

- The relation $\approx^{\mathsf{I}} \subseteq (\Gamma \vdash I) \times (\widehat{\mathsf{I}}\,\Gamma)$ is defined as follows: $t \approx^{\mathsf{I}} v$ if and only if $t \approx I_{\mathsf{I}}$ (notice that $v : \widehat{\mathsf{I}}\,\Gamma$ forces $\Gamma$ to be empty).

- The relation $\approx^{\otimes} \subseteq (\Gamma \vdash A \otimes B) \times ((\llbracket A \rrbracket \mathbin{\widehat{\otimes}} \llbracket B \rrbracket)\,\Gamma)$ is defined as follows: $t \approx^{\otimes} (a,b)$ if and only if $u \approx_A a$ and $v \approx_B b$ for some derivations $u : \Gamma_0 \vdash A$ and $v : \Gamma_1 \vdash B$ such that $t \sim I_{\otimes}\, u\, v$ (notice that by assumption $\Gamma$ is of the form $\Gamma_0, \Gamma_1$, so that $a : \llbracket A \rrbracket\, \Gamma_0$ and $b : \llbracket B \rrbracket\, \Gamma_1$). In particular, the relation $\approx^{\otimes}$ is defined simultaneously with $\approx$.

The logical relation $\approx$ is invariant under $\beta\eta$-conversion, formally:

**(fundamental theorem of logical relations)** For any derivations $t, u : \Gamma \vdash A$ and value $v : \llbracket A \rrbracket\, \Gamma$, if $t \sim u$ and $u \approx_A v$, then also $t \approx_A v$.

This fact can be proved by induction on the proof of $u \approx_A v$, together with similar closure properties of the relation lifting $\overline{T}$ and the auxiliary relations $\approx^{\mathsf{I}}$ and $\approx^{\otimes}$.

It is possible to define a relation $\approx^{\triangleright}_{\Gamma} \subseteq (\Gamma \triangleright \Delta) \times (\llbracket \Gamma \rrbracket\, \Delta)$ which naturally extends the logical relation $\approx$ to environments and their interpretations. Using the fundamental theorem, one can show that the syntactic notion of substitution given by the admissible $\mathsf{cut}$ rule is $\approx$-related to the semantic notion of substitution performed during the interpretation of a formula in the Kripke model. Formally:

**(correctness of evaluation)** Given a derivation $t : \Gamma \vdash A$, a syntactic environment $\sigma : \Gamma \triangleright \Delta$ and a semantic environment $\gamma : \llbracket \Gamma \rrbracket\, \Delta$ such that $\sigma \approx^{\triangleright}_{\Gamma} \gamma$, we have $\mathsf{cut}\, \sigma\, t \approx_A \llbracket t \rrbracket\, \gamma$.

From the fundamental theorem it follows that the reification and reflection functions behave correctly:

**(correctness of reification)** For any derivation $t : \Gamma \vdash C$ and semantic value $v : \llbracket A \rrbracket\, \Gamma$, if $t \approx v$ then $t \sim \mathsf{emb}_{\Uparrow}\, (\downarrow_A v)$. This means that whenever $v$ denotes $t$, then the reification of $v$ (seen as a derivation in $\mathbf{L}$ via $\mathsf{emb}_{\Uparrow}$) is correctly $\beta\eta$-related to $t$.

**(correctness of reflection)** For any neutral $n : \Gamma \Downarrow A$, we have $\mathsf{emb}_{\Downarrow}\, n \approx \uparrow_A n$. This means that the reflected neutral $n$ correctly denotes $n$ (seen as a derivation in $\mathbf{L}$ via $\mathsf{emb}_{\Downarrow}$).

The desired correctness property of $\mathsf{nbe}$ can therefore be formally stated as:

**(injectivity of $\mathsf{nbe}$)** For all derivations $t : \Gamma \vdash A$, we have $t \sim \mathsf{emb}_{\Uparrow}\, (\mathsf{nbe}\, t)$.

This can be proved by connecting all the correctness results obtained so far:

$$
\begin{array}{lrll}
t \sim \mathsf{emb}_{\Uparrow}\, (\mathsf{nbe}\, t) & \text{iff} & t \sim \mathsf{emb}_{\Uparrow}(\downarrow_A (\llbracket t \rrbracket\, \mathsf{fresh}_{\Gamma})) & \text{(unfolding the definition of } \mathsf{nbe}) \\
& \text{true if} & t \approx_A \llbracket t \rrbracket\, \mathsf{fresh}_{\Gamma} & \text{(correctness of reification)} \\
& \text{true if} & \mathsf{cut}\, \mathsf{ids}_{\Gamma}\, t \approx_A \llbracket t \rrbracket\, \mathsf{fresh}_{\Gamma} & \text{(fundamental theorem applied to } t \sim \mathsf{cut}\, \mathsf{ids}_{\Gamma}\, t) \\
& \text{true if} & \mathsf{ids}_{\Gamma} \approx^{\triangleright}_{\Gamma} \mathsf{fresh}_{\Gamma} & \text{(correctness of evaluation)}
\end{array}
$$

and $\mathsf{ids}_{\Gamma} \approx^{\triangleright}_{\Gamma} \mathsf{fresh}_{\Gamma}$ is a simple consequence of the correctness of reflection.

# 6 NbE for Fragments of Intuitionistic Linear Logic

The NbE strategy described so far for the Lambek calculus works also for other related substructural systems, such as multiplicative intuitionistic linear logic (MILL) and dual intuitionistic linear logic (DILL).

## 6.1 Multiplicative Intuitionistic Linear Logic

A natural deduction calculus for MILL [9] is obtained from $\mathbf{L}$ by adding an explicit exchange rule allowing to swap the positions of any two formulae in the context. Alternatively, and this is the modification we choose, one could also change the definition of a sequent $\Gamma \vdash A$ by requiring the context $\Gamma$ to be a finite multiset of formulae instead of an ordered list. In particular, the comma notation in $\Gamma, \Delta$ now stands

for multiset concatenation. The change of nature of antecedents makes the presence of two implications $\backslash$ and $/$ redundant, since they become equivalent connectives. The two ordered implications are then replaced by a unique linear implication $\multimap$, and only one occurrence of the two implication introduction and elimination rules is preserved, similarly for the equations in Figure 2. All the other rules and equations are the same. Similar modifications occur in the calculus of normal forms $\mathbf{L}_{\beta\eta}$.

The construction of the Kripke model and the implementation of the NbE procedure for $\mathbf{L}$ can then be directly adapted to the case of MILL. The main difference is that the presheaf model $\mathsf{Set}^{\mathsf{Cxt}}$ is now *symmetric* monoidal closed, which means that there is a natural isomorphism $P \mathbin{\widehat{\otimes}} Q \cong Q \mathbin{\widehat{\otimes}} P$.

## 6.2 Dual Intuitionistic Linear Logic

Our NbE strategy extends with some modifications also to the case of dual intuitionistic linear logic (DILL) [8], which is a particular presentation of MILL with the exponential modality !. Its peculiarity comes from the design of sequents, which are triples of the form $\Gamma;\Delta \vdash A$, where $\Gamma$ is a list of formulae (the *intuitionistic context*) and $\Delta$ is a finite multiset of formulae (the *linear context*). The rules of DILL are listed in Figure 4 (the $\beta\eta$-equivalence on DILL derivations is omitted from the paper). The rules for $\beta\eta$-long normal forms and neutrals are presented in Figure 5.

The presence of the intuitionistic context makes the construction of the Kripke model slightly more involved. Let $\mathsf{LCxt}$ be the set of linear contexts and $\mathsf{ICxt}$ be the category with intuitionistic contexts as objects and *renamings* as maps, i.e. injective functions $f : \Gamma \to \Gamma'$ sending each occurrence of a formula in $\Gamma$ to a distinct occurrence of a formula in $\Gamma'$. The Kripke model for DILL is the (covariant) presheaf category $\mathsf{Set}^{\mathsf{ICxt}\times\mathsf{LCxt}}$. This category has again a (symmetric) monoidal closed structure, which is defined analogously to (3). For example, the (action on objects of the) semantics tensor product is $(P \mathbin{\widehat{\otimes}} Q)\ (\Gamma;\Delta) =_{\mathrm{df}} \{\Delta_0,\Delta_1 : \mathsf{LCxt}\} \times \{\Delta = \Delta_0,\Delta_1\} \times P\ (\Gamma;\Delta_0) \times Q\ (\Gamma;\Delta_1)$. This category has an exponential modality $\widehat{!}$ whose action on objects is given by

$$\widehat{!}\,P\ (\Gamma;\Delta) =_{\mathrm{df}} (\Delta = \langle\rangle) \times P\ (\Gamma;\ )$$

The definition of the monad $T$ on $\mathsf{Set}^{\mathsf{ICxt}\times\mathsf{LCxt}}$ is analogous to the one in (4) with the following addition

$$\frac{\Gamma;\Delta \Downarrow !A \quad T\ P\ (\Gamma,A;\Delta')}{T\ P\ (\Gamma;\Delta,\Delta')}\ E^T_!$$

and the interpretation $[\![!A]\!]$ is given by $T\ (\widehat{!}\ [\![A]\!])$. The interpretation of formulae extends to contexts $[\![\Gamma;\Delta]\!](\Gamma';\Delta') =_{\mathrm{df}} [\![\Gamma]\!]^{\times}\Gamma' \times [\![\Delta]\!]^{\otimes}(\Gamma';\Delta')$, where the interpretation of linear contexts $[\![\Delta]\!]^{\otimes}$ is as before (i.e. defined using tensor $\widehat{\otimes}$) and the interpretation of intuitionistic contexts is defined using the Cartesian product of the Kripke model, i.e. $[\![A_1,\ldots,A_n]\!]^{\times}\ \Gamma' =_{\mathrm{df}} [\![A_1]\!]\ (\Gamma';\ ) \times \cdots \times [\![A_n]\!]\ (\Gamma';\ )$.

The definition of evaluation $[\![t]\!] : [\![\Gamma;\Delta]\!] \dot{\to} [\![A]\!]$ in (8) needs to take into account the three new rules

$$\begin{array}{lll}
[\![\mathsf{ax}_{Int}]\!]\ (\gamma,a,\gamma') & =_{\mathrm{df}} & a \\
[\![I_!\ t]\!]\ (\gamma,\delta) & =_{\mathrm{df}} & \eta\ ([\![t]\!]\ (\gamma,\delta)) \\
[\![E_!\ t\ u]\!]\ (\gamma,\delta_0,\delta_1) & =_{\mathrm{df}} & \mathsf{run}\ (T\ [\![u]\!]\ (\mathsf{rmst}\ (\mathsf{lmst}\ (\gamma,[\![t]\!]\ (\gamma,\delta_0)),\delta_1)))
\end{array}$$

where the function $\mathsf{run}$ is as in (7), with the extra case of $E^T_!$ dealt similarly to $E^T_{\mathsf{I}}$ and $E^T_{\otimes}$. The left strength $\mathsf{lmst}$ (and the equivalent right strength $\mathsf{rmst}$) is defined as in (5), but the case of $E^T_!$ requires some extra care: $\mathsf{lmst}\ (x,E^T_!\ t\ y) =_{\mathrm{df}} E^T_!\ t\ (\mathsf{lmst}\ (P\ \iota\ x,y))$, where $P\ \iota : P\ (\Gamma;\Delta) \to P\ (\Gamma,A;\Delta)$ is the action

$$\frac{\Gamma;\Delta,A\vdash B}{\Gamma;\Delta\vdash A\multimap B}\ \mathsf{I}_{\multimap} \qquad \frac{\Gamma;\Delta\vdash A\multimap B \quad \Gamma;\Delta'\vdash A}{\Gamma;\Delta,\Delta'\vdash B}\ \mathsf{E}_{\multimap} \qquad \frac{\Gamma;\ \vdash A}{\Gamma;\ \vdash !A}\ I_{!} \qquad \frac{\Gamma;\Delta\vdash !A \quad \Gamma,A;\Delta'\vdash C}{\Gamma;\Delta,\Delta'\vdash C}\ E_{!}$$

$$\frac{}{\Gamma;A\vdash A}\ \mathsf{ax}_{Lin} \qquad \frac{}{\Gamma_0,A,\Gamma_1;\ \vdash A}\ \mathsf{ax}_{Int} \qquad \frac{}{\Gamma;\ \vdash \mathsf{I}}\ I_{\mathsf{I}} \qquad \frac{\Gamma;\Delta\vdash \mathsf{I} \quad \Gamma;\Delta'\vdash C}{\Gamma;\Delta,\Delta'\vdash C}\ E_{\mathsf{I}}$$

$$\frac{\Gamma;\Delta\vdash A \quad \Gamma;\Delta'\vdash B}{\Gamma;\Delta,\Delta'\vdash A\otimes B}\ I_{\otimes} \qquad \frac{\Gamma;\Delta\vdash A\otimes B \quad \Gamma;\Delta',A,B\vdash C}{\Gamma;\Delta,\Delta'\vdash C}\ E_{\otimes}$$

Figure 4: Inference rules of DILL

$$\frac{\Gamma;\Delta,A\Uparrow B}{\Gamma;\Delta\Uparrow A\multimap B}\ \mathsf{I}_{\multimap} \qquad \frac{\Gamma;\Delta\Downarrow A\multimap B \quad \Gamma;\Delta'\Uparrow A}{\Gamma;\Delta,\Delta'\Downarrow B}\ \mathsf{E}_{\multimap} \qquad \frac{\Gamma;\ \Uparrow A}{\Gamma;\ \Uparrow !A}\ I_{!} \qquad \frac{\Gamma;\Delta\Downarrow !A \quad \Gamma,A;\Delta'\Uparrow C}{\Gamma;\Delta,\Delta'\Uparrow C}\ E_{!}$$

$$\frac{}{\Gamma;A\Downarrow A}\ \mathsf{ax}_{Lin} \qquad \frac{}{\Gamma,A,\Gamma';\ \Downarrow A}\ \mathsf{ax}_{Int} \qquad \frac{}{\Gamma;\ \Uparrow \mathsf{I}}\ I_{\mathsf{I}} \qquad \frac{\Gamma;\Delta\Downarrow \mathsf{I} \quad \Gamma;\Delta'\Uparrow \gamma^{+}}{\Gamma;\Delta,\Delta'\Uparrow \gamma^{+}}\ E_{\mathsf{I}}$$

$$\frac{\Gamma;\Delta\Uparrow A \quad \Gamma;\Delta'\Uparrow B}{\Gamma;\Delta,\Delta'\Uparrow A\otimes B}\ I_{\otimes} \qquad \frac{\Gamma;\Delta\Downarrow A\otimes B \quad \Gamma;\Delta',A,B\Uparrow \gamma^{+}}{\Gamma;\Delta,\Delta'\Uparrow \gamma^{+}}\ E_{\otimes} \qquad \frac{\Gamma;\Delta\Downarrow p}{\Gamma;\Delta\Uparrow p}\ \mathsf{sw}_{\Downarrow}$$

Figure 5: Normal forms and neutrals of DILL

of the presheaf $P$ on the renaming $\iota : \Gamma \to \Gamma,A$, which is the inclusion of $\Gamma$ into $\Gamma,A$. Reflection $\uparrow_A$ and reification $\downarrow_A$ in (9) also need to cover the extra case of the modality !:

$$\downarrow_{!A}\ t \quad =_{\mathrm{df}} \quad \mathsf{run}^{\Uparrow}\ (T\ (\lambda(\mathsf{refl},y).I_{!}\ (\downarrow_A y))\ t) \qquad\qquad \uparrow_{!A} t \quad =_{\mathrm{df}} \quad E_{!}^{T}\ t\ (\eta\ (\mathsf{refl},\uparrow_A \mathsf{ax}_{Int}))$$

This implies that the $\mathsf{nbe}$ function can also be defined on DILL derivations. Similarly to the case of $\mathbf{L}$, the correctness of $\mathsf{nbe}$ can be proved via logical relations. A thorough description of the NbE procedure for DILL, together with a proof of its correctness, will appear in a future extended version of this paper.

## 7 Conclusions

We have studied normalization by evaluation for the Lambek calculus and other related substructural logics such as MILL and DILL. The presence of positive connectives, i.e. the multiplicative unit $\mathsf{I}$ and tensor $\otimes$ (and exponential ! in DILL), requires a modification of the NbE technique for typed $\lambda$-calculus employing a strong monad on the Kripke model, a situation sharing many similarities with the extension of typed $\lambda$-calculus with empty and sum types [5, 1].

The normalization algorithm developed in this paper works for "weak" unit and tensor, in the sense that the equational theory $\sim$ does not satisfy equations of the form (2). In other words, $\mathbf{L}_{\beta\eta}$ is not a presentation of the free monoidal biclosed category on the set of atomic formulae $\mathsf{At}$. The extension of NbE to "strong" unit and tensor is left for future work. We should take inspiration from the methodologies adopted in the NbE literature to deal with strong sums in typed $\lambda$-calculus [3, 7], which require a modification of the Kripke semantics using sheaves on a particularly chosen site instead of presheaves. The normal forms for the stronger equational theory would also serve as the target calculus of the normalization procedure of Amblard and Retoré [6]

This paper should be considered as a starting point in understanding the NbE methodology in the realm of substructural logics. We are especially interested in the connection between Kripke logical relations for linear logic [13, 15] and categorical reformulations of normalization by evaluation [4, 12].

**Acknowledgments** The author was supported by the ESF funded Estonian IT Academy research measure (project 2014-2020.4.05.19-0001) and the Estonian Research Council grants PSG659 and PSG749. Participation in the conference was supported by the EU COST action CA19135 (CERCIRAS).

## References

[1] Andreas Abel & Christian Sattler (2019): *Normalization by Evaluation for Call-by-Push-Value and Polarized Lambda Calculus*. In Ekaterina Komendantskaya, editor: *Proceedings of the 21st International Symp. on Principles and Practice of Programming Languages, PPDP 2019*, ACM, pp. 3:1–3:12, doi: 10.1145/3354166.3354168.

[2] Vito M. Abrusci (1990): *Non-Commutative Intuitionistic Linear Logic*. *Mathematical Logic Quarterly* 36(4), pp. 297–318, doi: 10.1002/malq.19900360405.

[3] Thorsten Altenkirch, Peter Dybjer, Martin Hofmann & Philip J. Scott (2001): *Normalization by Evaluation for Typed Lambda Calculus with Coproducts*. In: *Proceedings of the 16th Annual IEEE Symp. on Logic in Computer Science, LICS 2001*, IEEE Computer Society, pp. 303–310, doi: 10.1109/LICS.2001.932506.

[4] Thorsten Altenkirch, Martin Hofmann & Thomas Streicher (1995): *Categorical Reconstruction of a Reduction Free Normalization Proof*. In David H. Pitt, David E. Rydeheard & Peter T. Johnstone, editors: *Proceedings of the 6th International Conference on Category Theory and Computer Science, CTCS 1995*, *Lecture Notes in Computer Science* 953, Springer, pp. 182–199, doi: 10.1007/3-540-60164-3_27.

[5] Thorsten Altenkirch & Tarmo Uustalu (2004): *Normalization by Evaluation for* $\lambda^{\to 2}$. In Yukiyoshi Kameyama & Peter J. Stuckey, editors: *Proceedings of the 7th International Symp. on Functional and Logic Programming, FLOPS 2004*, *Lecture Notes in Computer Science* 2998, Springer, pp. 260–275, doi: 10.1007/978-3-540-24754-8_19.

[6] Maxime Amblard & Christian Retoré (2014): *Partially Commutative Linear Logic and Lambek Caculus with Product: Natural Deduction, Normalisation, Subformula Property*. *FLAP* 1(1), pp. 53–94. Available at `http://www.collegepublications.co.uk/downloads/ifcolog00001.pdf`.

[7] Vincent Balat, Roberto Di Cosmo & Marcelo P. Fiore (2004): *Extensional Normalisation and Type-Directed Partial Evaluation for Typed Lambda Calculus with Sums*. In Neil D. Jones & Xavier Leroy, editors: *Proceedings of the 31st ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2004*, ACM, pp. 64–76, doi: 10.1145/964001.964007.

[8] Andrew G. Barber (1997): *Linear Type Theories, Semantics and Action Calculi*. Ph.D. thesis, University of Edinburgh, UK. Available at `http://hdl.handle.net/1842/392`.

[9] Nick Benton, Gavin M. Bierman, Valeria de Paiva & Martin Hyland (1993): *A Term Calculus for Intuitionistic Linear Logic*. In Marc Bezem & Jan Friso Groote, editors: *Proceedings of the 1st International Conference on Typed Lambda Calculi and Applications, TLCA '93*, *Lecture Notes in Computer Science* 664, Springer, pp. 75–90, doi: 10.1007/BFb0037099.

[10] Ulrich Berger & Helmut Schwichtenberg (1991): *An Inverse of the Evaluation Functional for Typed lambda-calculus*. In: *Proceedings of the 6th Annual Symposium on Logic in Computer Science, LICS 1991*, IEEE Computer Society, pp. 203–211, doi: 10.1109/LICS.1991.151645.

[11] Taus Brock-Nannestad & Carsten Schürmann (2010): *Focused Natural Deduction*. In Christian G. Fermüller & Andrei Voronkov, editors: *Proceedings of the 17th International Conference on Logic for Programming, Artificial Intelligence, and Reasoning, LPAR 2010*, *Lecture Notes in Computer Science* 6397, Springer, pp. 157–171, doi: 10.1007/978-3-642-16242-8_12.

[12] Marcelo P. Fiore (2002): *Semantic Analysis of Normalisation by Evaluation for Typed Lambda Calculus*. In: *Proceedings of the 4th International ACM SIGPLAN Conference on Principles and Practice of Declarative Programming, PPDP 2002*, ACM, pp. 26–37, doi: 10.1145/571157.571161.

[13] Masahito Hasegawa (1999): *Logical Predicates for Intuitionistic Linear Type Theories*. In Jean-Yves Girard, editor: *Proceedings of the 4th International Conference on Typed Lambda Calculi and Applications, TLCA 1999*, *Lecture Notes in Computer Science* 1581, Springer, pp. 198–212, doi: 10.1007/3-540-48959-2_15.

[14] Mark Hepple (1990): *Normal Form Theorem Proving for the Lambek Calculus*. In: *Proceedings of the 13th International Conference on Computational Linguistics, COLING 1990*, pp. 173–178. Available at `https://aclanthology.org/C90-2030/`.

[15] Martin Hyland & Andrea Schalk (2003): *Glueing and Orthogonality for Models of Linear Logic*. *Theorerical Computer Science* 294(1/2), pp. 183–231, doi: 10.1016/S0304-3975(01)00241-9.

[16] Francois Lamarche & Christian Retoré (1996): *Proof Nets for the Lambek Calculus – an Overview*. In: *Proceedings of the 3rd Roma Workshop on Proofs and Linguistic Categories 1998*, pp. 241–262. Available at `https://hal.archives-ouvertes.fr/inria-00098442`.

[17] Joachim Lambek (1958): *The Mathematics of Sentence Structure*. *The American Mathematical Monthly* 65(3), pp. 154–170, doi: 10.2307/2310058.

[18] Joachim Lambek (1968): *Deductive Systems and Categories I: Syntactic Calculus and Residuated Categories*. *Mathematical Systems Theory* 2(4), pp. 287–318, doi: 10.1007/bf01703261.

[19] Joachim Lambek (1969): *Deductive Systems and Categories II: Standard Constructions and Closed Categories*. In: *Category theory, Homology Theory and Their Applications I*, Springer, pp. 76–122, doi: 10.1007/bfb0079385.

[20] Jeff Polakov (2001): *Ordered Linear Logic and Applications*. Ph.D. thesis, Carnegie Mellon University, USA. Available at `https://www.cs.cmu.edu/~jpolakow/diss.ps`.

[21] Jeff Polakow & Frank Pfenning (1999): *Natural Deduction for Intuitionistic Non-communicative Linear Logic*. In Jean-Yves Girard, editor: *Proceedings of the 4th International Conference on Typed Lambda Calculi and Applications, TLCA 1999*, *Lecture Notes in Computer Science* 1581, Springer, pp. 295–309, doi: 10.1007/3-540-48959-2_21.

[22] Dirk Roorda (1992): *Proof Nets for Lambek Calculus*. *Journal of Logic and Computation* 2(2), pp. 211–231, doi: 10.1093/logcom/2.2.211.

[23] Tarmo Uustalu, Niccoló Veltri & Noam Zeilberger (2021): *Deductive Systems and Coherence for Skew Prounital Closed Categories*. In Claudio Sacerdoti Coen & Alwen Tiu, editors: *Proceedings of the 15th Workshop on Logical Frameworks and Meta-Languages: Theory and Practice, LFMTP 2020*, *Electronic Proceedings in Theoretical Computer Science* 332, Open Publishing Assoc., pp. 35–53, doi: 10.4204/eptcs.332.3.

[24] Nachiappan Valliappan & Alejandro Russo (2019): *Exponential Elimination for Bicartesian Closed Categorical Combinators*. In Ekaterina Komendantskaya, editor: *Proceedings of the 21st International Symp. on Principles and Practice of Programming Languages, PPDP 2019*, ACM, pp. 20:1–20:13, doi: 10.1145/3354166.3354185.