Technical Report

UCAM-CL-TR-401
ISSN 1476-2986

Number 401

UNIVERSITY OF CAMBRIDGE
Computer Laboratory

A classical linear λ-calculus

Gavin Bierman

July 1996

15 JJ Thomson Avenue
Cambridge CB3 0FD
United Kingdom
phone +44 1223 763500

https://www.cl.cam.ac.uk/

Technical reports published by the University of Cambridge
Computer Laboratory are freely available via the Internet:

https://www.cl.cam.ac.uk/techreports/

ISSN 1476-2986

A Classical Linear λ-calculus*

G.M. Bierman

Gonville and Caius College,
Cambridge. England.

Abstract

This paper proposes and studies a typed λ-calculus for classical linear logic. I shall give an explanation of a multiple-conclusion formulation for classical logic due to Parigot and compare it to more traditional treatments by Prawitz and others. I shall use Parigot's method to devise a natural deduction formulation of classical linear logic. This formulation is compared in detail to the sequent calculus formulation. In an appendix I shall also demonstrate a somewhat hidden connexion with the paradigm of control operators for functional languages which gives a new computational interpretation of Parigot's techniques.

1 Introduction

In the past classical logic (**CL**) has often been dismissed as having no interesting proof theory. However following a rather pleasing interplay between theoretical computer science and practical computer science, there has been a renewed interest in **CL** and, in particular, the constructive content of classical proofs. This content appears to have links with, at the theoretical level, game theory [13] and at the practical level, certain control operators for functional programming languages [23]. To some extent Girard's linear logic [21] has also renewed interest in game theory and functional programming languages. The refined connectives of linear logic have helped shed new light on work on games [12]. The games models have proved useful for programming language semantics: the recent fully-abstract models of PCF [2, 26] are good examples of this. In addition, intuitionistic linear logic (**ILL**) has been proposed as a resource-sensitive foundation of functional programming languages. Thus it would seem useful to reconsider the work on **CL** in a linear setting, *viz.* to reconsider classical linear logic (**CLL**).

Gentzen's natural deduction is a very suitable deduction system for intuitionistic logic (**IL**) but seems less so for classical logic.[1] One could say that classical logic is a logic of symmetry whereas natural deduction is, by its very nature, an asymmetric system. To that extent Gentzen's alternative system, the sequent calculus, seems better suited as the system for **CL**.

The Curry-Howard correspondence [24] allows us to annotate natural deductions with terms. For **IL** this yields the typed λ-calculus. For the sequent calculus it is not entirely clear what the appropriate annotations are. In fact there are a number of choices and there is no real consensus on the best. It might seem prudent to revisit natural deduction, where

*A preliminary version of this paper appears as *Towards a Classical Linear λ-calculus* in the Proceedings of the Tokyo Conference on Linear Logic, Volume 3 of Electronic Notes in Theoretical Computer Science, Elsevier. 1996.

[1] "One may doubt that this is the proper way of analysing classical inferences." [32, Pages 244-5].

the question of syntax is settled, and see if we might be able to produce a more symmetric system. Shoesmith and Smiley [34] made an early attempt at this by defining a multiple-conclusion natural deduction system but unfortunately it was quite complicated. More recently Parigot [30] has introduced a variant of multiple-conclusion natural deduction which seems better suited for handling **CL**.

In this paper I shall consider in detail Parigot's formulation of **CL**, which I shall call $\mathbf{CL}_\mu$. I shall motivate it by considering more traditional formulations. This paper is organised as follows. In §2 I shall give two presentations of $\mathbf{CL}_\mu$; one which appears in Parigot's original work and the other a more type-theoretic one.[2] Via the Curry-Howard correspondence one derives a term calculus for $\mathbf{CL}_\mu$, which is called the (typed) $\lambda\mu$-calculus. In §2 I shall also study the particularly tricky area of reduction. In §3 I shall follow a similar method to derive a natural deduction formulation of **CLL** that I shall call $\mathbf{CLL}_\mu$. There are some surprises here to do with the exponential modality (!). Applying the Curry-Howard correspondence to $\mathbf{CLL}_\mu$ yields the (typed) linear $\lambda\mu$-calculus. In §4 I shall consider the process of reduction for the linear $\lambda\mu$-calculus. In §5 I shall show how to map sequent proofs in **CLL** to deductions in $\mathbf{CLL}_\mu$ and then use this to compare the process of cut-elimination with the term reduction process. In §6 I shall consider briefly the Q- and T-translations of Schellinx, at the level of terms, between $\mathbf{CL}_\mu$ and $\mathbf{CLL}_\mu$. In Appendix A, I shall propose a novel computational interpretation of the $\lambda\mu$-calculus, which suggests that it can be thought of as a programming language with catch and throw-like control operators. In Appendix B I shall briefly give another presentation of the linear $\lambda\mu$-calculus based on Benton's mixed presentation of the linear λ-calculus [6].

Before continuing I should clarify the rôle of this work. In his seminal paper [21], Girard presented *proof nets*, which are a succinct presentation of proofs in **CLL**. One important feature of proof nets is that formulae which are equivalent with respect to the dualities (for example, $\phi\otimes\psi$ and $(\phi^\perp ⅋ \psi^\perp)^\perp$) are considered to be equal. This cuts down considerably the number of proofs. What I am striving for here is a calculus which does *not* have these equivalences built-in. Consider an anologous situation for the λ-calculus with products and coproducts. We might consider the formulae $\phi \wedge (\psi \vee \upsilon)$ and $(\phi\wedge\psi)\vee(\phi\wedge\upsilon)$ to be equivalent and in (categorical) models they are *isomorphic*. However we certainly don't insist on them being equal (they are distinct types!). Indeed it is hard to imagine a programming language where this is so. Thus I suggest that the linear $\lambda\mu$-calculus is a more realistic foundation for a programming language based on **CLL**. A language based on proof nets would probably be some variant of Abramsky's proof expressions [1, Section 6].

2 From IL to CL

In this paper I shall only consider *propositional* formulations of the various logics. For both **IL** and **CL**, formulae are given by the grammar

$$\phi ::= p \mid \phi\wedge\phi \mid \phi\vee\phi \mid \phi\supset\phi,$$

where p is taken from a countable set of atomic formulae which includes a distinguished member, $\perp$, which denotes falsum.

[2] A similar presentation has been given independently by Ong [29].

$$\frac{}{\phi \vdash \phi}\ \mathit{Identity} \qquad \frac{}{\bot \vdash \phi}\ (\bot_{\mathcal{L}})$$

$$\frac{\Gamma \vdash \psi}{\Gamma, \phi \vdash \psi}\ \mathit{Weakening} \qquad \frac{\Gamma, \phi, \phi \vdash \psi}{\Gamma, \phi \vdash \psi}\ \mathit{Contraction}$$

$$\frac{\Gamma \vdash \phi \qquad \Delta, \phi \vdash \psi}{\Gamma, \Delta \vdash \psi}\ \mathit{Cut}$$

$$\frac{\Gamma, \phi \vdash \varphi}{\Gamma, \phi \wedge \psi \vdash \varphi}\ (\wedge_{\mathcal{L}}) \qquad \frac{\Gamma, \psi \vdash \varphi}{\Gamma, \phi \wedge \psi \vdash \varphi}\ (\wedge_{\mathcal{L}}) \qquad \frac{\Gamma \vdash \phi \qquad \Gamma \vdash \psi}{\Gamma \vdash \phi \wedge \psi}\ (\wedge_{\mathcal{R}})$$

$$\frac{\Gamma, \phi \vdash \varphi \qquad \Gamma, \psi \vdash \varphi}{\Gamma, \phi \vee \psi \vdash \varphi}\ (\vee_{\mathcal{L}}) \qquad \frac{\Gamma \vdash \phi}{\Gamma \vdash \phi \vee \psi}\ (\vee_{\mathcal{R}}) \qquad \frac{\Gamma \vdash \psi}{\Gamma \vdash \phi \vee \psi}\ (\vee_{\mathcal{R}})$$

$$\frac{\Gamma \vdash \phi \qquad \Gamma, \psi \vdash \varphi}{\Gamma, \phi \supset \psi \vdash \varphi}\ (\supset_{\mathcal{L}}) \qquad \frac{\Gamma, \phi \vdash \psi}{\Gamma \vdash \phi \supset \psi}\ (\supset_{\mathcal{R}})$$

Figure 1: Sequent Calculus formulation of **IL**

2.1 Sequent Calculus

Gentzen's sequent calculus [20] is a wonderfully symmetric system. Deductions form a tree of sequents of the form $\Gamma \vdash \Delta$ where both Γ and Δ represent collections of formulae (whether this 'collection' is a set, multiset, list or just a single formula is dependent on the formulation and the logic in question). Inference rules introduce connectives on the right and the left of the 'turnstile' (hence the symmetry). Although for the sequent calculus it is easier to see the formulation of **IL** as a subsystem of **CL**, I shall present them here in the opposite order, as that will match the discussion of the natural deduction formulation.

For **IL**, sequents are of the form $\Gamma \vdash \phi$ where Γ denotes a multiset of formulae. The formulation is given in Figure 1. For **CL**, sequents are of the form $\Gamma \vdash \Delta$ where both Γ and Δ are multisets of formulae. This formulation is given in Figure 2. I have chosen to give rules for negation directly in **CL**. In **IL**, negation is defined as $\neg\phi \stackrel{\text{def}}{=} \phi \supset \bot$.

To demonstrate that the formulation of Figure 2 does capture classical provability, here are derivations of Peirce's Law and the law of the excluded middle.

$$\frac{\dfrac{\dfrac{\dfrac{}{\phi \vdash \phi}\ \mathit{Identity}}{\phi \vdash \psi, \phi}\ \mathit{Weakening}_{\mathcal{R}}}{\vdash \phi \supset \psi, \phi}\ (\supset_{\mathcal{R}}) \qquad \dfrac{}{\phi \vdash \phi}\ \mathit{Identity}}{\dfrac{(\phi \supset \psi) \supset \phi \vdash \phi}{\vdash ((\phi \supset \psi) \supset \phi) \supset \phi}\ (\supset_{\mathcal{R}})}\ (\supset_{\mathcal{L}})$$

$$\frac{}{\phi \vdash \phi}\ \mathit{Identity}$$

$$\frac{\Gamma \vdash \Delta}{\Gamma, \phi \vdash \Delta}\ \mathit{Weakening}_{\mathcal{L}} \qquad \frac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta, \phi}\ \mathit{Weakening}_{\mathcal{R}}$$

$$\frac{\Gamma, \phi, \phi \vdash \Delta}{\Gamma, \phi \vdash \Delta}\ \mathit{Contraction}_{\mathcal{L}} \qquad \frac{\Gamma \vdash \Delta, \phi, \phi}{\Gamma \vdash \Delta, \phi}\ \mathit{Contraction}_{\mathcal{R}}$$

$$\frac{\Gamma \vdash \Delta, \phi \qquad \Gamma', \phi \vdash \Delta'}{\Gamma, \Gamma' \vdash \Delta, \Delta'}\ \mathit{Cut}$$

$$\frac{\Gamma, \phi \vdash \Delta}{\Gamma, \phi \wedge \psi \vdash \Delta}\ (\wedge_{\mathcal{L}}) \qquad \frac{\Gamma, \psi \vdash \Delta}{\Gamma, \phi \wedge \psi \vdash \Delta}\ (\wedge_{\mathcal{L}}) \qquad \frac{\Gamma \vdash \Delta, \phi \qquad \Gamma \vdash \Delta, \psi}{\Gamma \vdash \Delta, \phi \wedge \psi}\ (\wedge_{\mathcal{R}})$$

$$\frac{\Gamma, \phi \vdash \Delta \qquad \Gamma, \psi \vdash \Delta}{\Gamma, \phi \vee \psi \vdash \Delta}\ (\vee_{\mathcal{L}}) \qquad \frac{\Gamma \vdash \Delta, \phi}{\Gamma \vdash \Delta, \phi \vee \psi}\ (\vee_{\mathcal{R}}) \qquad \frac{\Gamma \vdash \Delta, \psi}{\Gamma \vdash \Delta, \phi \vee \psi}\ (\vee_{\mathcal{R}})$$

$$\frac{\Gamma \vdash \Delta, \phi \qquad \Gamma, \psi \vdash \Delta}{\Gamma, \phi \supset \psi \vdash \Delta}\ (\supset_{\mathcal{L}}) \qquad \frac{\Gamma, \phi \vdash \psi, \Delta}{\Gamma \vdash \phi \supset \psi, \Delta}\ (\supset_{\mathcal{R}})$$

$$\frac{\Gamma \vdash \phi, \Delta}{\Gamma, \neg\phi \vdash \Delta}\ (\neg_{\mathcal{L}}) \qquad \frac{\Gamma, \phi \vdash \Delta}{\Gamma \vdash \neg\phi, \Delta}\ (\neg_{\mathcal{R}})$$

Figure 2: Sequent Calculus formulation of **CL**

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\phi \vdash \phi}\ \mathit{Identity}}{\vdash \phi, \neg\phi}\ (\neg_{\mathcal{R}})}{\vdash \phi, \phi \vee \neg\phi}\ (\vee_{\mathcal{R}})}{\vdash \phi \vee \neg\phi, \phi \vee \neg\phi}\ (\vee_{\mathcal{R}})}{\vdash \phi \vee \neg\phi}\ \mathit{Contraction}_{\mathcal{R}}
$$

An important result for both these sequent calculus formulations is that instances of the *Cut* rule can be eliminated. In fact the proof of this assertion gives an algorithm for doing so (albeit a delicate one). This result is known as the *cut-elimination theorem* or Gentzen's *Hauptsatz.*

Theorem 1. (Gentzen) Given a derivation π of $\vdash_{\mathbf{IL}} \Gamma \vdash \phi$ ($\vdash_{\mathbf{CL}} \Gamma \vdash \Delta$), a derivation π' of $\Gamma \vdash \phi$ ($\Gamma \vdash \Delta$) can be found which contains no instances of the *Cut* rule.

Proof. For a nice presentation of the proof see the paper by Gallier [19]. ∎

2.2 Natural Deduction

Natural deduction was also originally proposed by Gentzen [20] and was later popularised by Prawitz [31]. Deductions proceed in a tree-like fashion where a single conclusion is derived from a (finite) number of assumption packets. More specifically, these packets contain a multiset of propositions and may be empty. Within a deduction we may 'discharge' any number of assumption packets. This discharging can be recorded in one of two ways. Gentzen originally proposed annotating assumption packets with (natural number) labels. Occurences of inference rules which discharge packets are then annotated with the labels of the packets they discharge. Thus a deduction is of the form

$$
\begin{array}{c}
\phi_1^{x_1} \cdots \phi_k^{x_k} \\
\vdots \\
\psi.
\end{array}
$$

The second alternative for annotations is to label every stage of the deduction with a complete list of the undischarged assumption packets. I shall refer to this as natural deduction in 'sequent-style'. I shall generally use the second method, although the first is sometimes used for clarity.

What distinguishes the natural deduction system from others is that there are rules for both introducing and eliminating a connective. Writing deductions in a sequent-style, $\Gamma \vdash \phi$; this means that the inference rules are solely concerned with the conclusion ϕ. This is what differentiates it with, for example, a sequent calculus formulation where we have rules for manipulating formulae on both sides of the turnstile. This is why it is said that the natural deduction system is essentially asymmetric. The natural deduction formulation of **IL** is given in Figure 3.

A good question is how one might extend this natural deduction formulation of **IL** to **CL**. We saw that for the sequent calculus the extension was to allow many conclusions. Extending natural deduction to allow for many conclusions seems to imply a graph-like structure. Alternatively we might consider simulating the multiple conclusions by storing them as a disjunction of formulae, which can then be treated as a single formula. Consider the sequent calculus implication-right rule from Figure 2.

$$\frac{}{\Gamma, \phi \vdash \phi}\ \mathit{Identity}$$

$$\frac{\Gamma \vdash \phi \qquad \Gamma \vdash \psi}{\Gamma \vdash \phi \wedge \psi}\ (\wedge_{\mathcal{I}})$$

$$\frac{\Gamma \vdash \phi \wedge \psi}{\Gamma \vdash \phi}\ (\wedge_{\mathcal{E}}) \qquad \frac{\Gamma \vdash \phi \wedge \psi}{\Gamma \vdash \psi}\ (\wedge_{\mathcal{E}})$$

$$\frac{\Gamma, \phi \vdash \psi}{\Gamma \vdash \phi \supset \psi}\ (\supset_{\mathcal{I}}) \qquad \frac{\Gamma \vdash \phi \supset \psi \qquad \Gamma \vdash \phi}{\Gamma \vdash \psi}\ (\supset_{\mathcal{E}})$$

$$\frac{\Gamma \vdash \phi}{\Gamma \vdash \phi \vee \psi}\ (\vee_{\mathcal{I}}) \qquad \frac{\Gamma \vdash \psi}{\Gamma \vdash \phi \vee \psi}\ (\vee_{\mathcal{I}})$$

$$\frac{\Gamma \vdash \phi \vee \psi \qquad \Gamma, \phi \vdash \varphi \qquad \Gamma, \psi \vdash \varphi}{\Gamma \vdash \varphi}\ (\vee_{\mathcal{E}})$$

$$\frac{\Gamma \vdash \bot}{\Gamma \vdash \phi}\ (\bot_{\mathcal{E}})$$

Figure 3: Natural Deduction Formulation of **IL**.

$$\frac{\Gamma, \phi \vdash \psi, \Delta}{\Gamma \vdash \phi \supset \psi, \Delta}\ (\supset_{\mathcal{R}})$$

If we consider simulating this in natural deduction, we have for the premiss

$$\begin{array}{c} \Gamma\ \phi \\ \vdots \\ (\bigvee \Delta) \vee \psi \end{array}$$

and clearly we wish to introduce an implication, but only over the formula ψ. The implication introduction rule will only allow

$$\begin{array}{c} \Gamma\ [\phi] \\ \vdots \\ \dfrac{(\bigvee \Delta) \vee \psi}{\phi \supset ((\bigvee \Delta) \vee \psi)}\ (\supset_{\mathcal{I}}). \end{array}$$

What is needed is the ability to abstract over just one of the conclusions. This seems to be precisely what we can *not* do in **IL**. Indeed the axiom

$$\textbf{ImpD}\colon (\phi \supset (\varphi \vee \psi)) \supset ((\phi \supset \varphi) \vee \psi)$$

is a sufficient (if unusual) addition to **IL** to give **CL**. Rather than continue with this 'simulation' of **CL**, traditional proof theory considers adding new rules to yield a formulation of **CL**. For example, Gentzen [20] and later in more detail, Prawitz [31], suggested adding either axioms of the form

$$\phi \vee \neg\phi$$

or a rule

$$\begin{array}{c} [\neg\phi] \\ \vdots \\ \bot \\ \hline \phi \end{array} RAA.$$

Parigot's system can be thought of as continuing with the simulation approach and adding sufficient extra machinery to make that method work. Thus we continue considering the many conclusions as a whole, but now where at most one of them will be distinguished as being 'active'. The others are 'passive' which is signified by being labelled (I shall label the active formula with a bullet). Thus deductions are of the form

$$\begin{array}{c} \Gamma \\ \vdots \\ \phi^{\bullet}, \psi_1^{a_1}, \ldots, \psi_n^{a_n} \end{array}$$

where ϕ is the active formula and the ψ_i are passive. The introduction and elimination rules are now considered to apply to just the active formula. To handle the example alluded to earlier, the system is extended with rules which enable active and passive rôles to be swapped. To facilitate this, two new rules are introduced

$$\begin{array}{c} \Gamma \\ \vdots \\ \phi^{\bullet}, \Sigma \\ \hline \phi^{a}, \Sigma \end{array} \textit{Passify} \qquad \text{and} \qquad \begin{array}{c} \Gamma \\ \vdots \\ \psi^{a}, \Sigma \\ \hline \psi^{\bullet}, \Sigma \end{array} \textit{Activate}.$$

It is important to realise that neither an active formula, ϕ, nor a passive formula, ψ, respectively, need to be present for these rules to be applied; or, in other words, we can consider the rules able to perform an implicit *Weakening* if necessary. (I shall overcome this slight messiness later by having explicit structural rules). This enables us to handle the earlier example, as follows

$$\dfrac{\dfrac{\dfrac{\begin{array}{c} \Gamma\ [\phi] \\ \vdots \\ \psi^{a}, \Delta, \varphi^{\bullet} \end{array}}{\psi^{a}, \Delta, \varphi^{b}} \textit{Passify}}{\psi^{\bullet}, \Delta, \varphi^{b}} \textit{Activate}}{(\phi \supset \psi)^{\bullet}, \Delta, \varphi^{b}} (\supset_{\mathcal{I}}).$$

Parigot's formulation of **CL** was given originally as follows.

$$\Gamma, \phi \vdash \phi^{\bullet}, \Sigma$$

$$\dfrac{\Gamma, \phi \vdash \psi^{\bullet}, \Sigma}{\Gamma \vdash (\phi \supset \psi)^{\bullet}, \Sigma} (\supset_{\mathcal{I}}) \qquad \dfrac{\Gamma \vdash (\phi \supset \psi)^{\bullet}, \Sigma \qquad \Gamma \vdash \phi^{\bullet}, \Sigma}{\Gamma \vdash \psi^{\bullet}, \Sigma} (\supset_{\mathcal{E}})$$

$$\dfrac{\Gamma \vdash \phi^{\bullet}, \Sigma}{\Gamma \vdash \phi^{a}, \Sigma} \textit{Passify} \qquad \dfrac{\Gamma \vdash \phi^{a}, \Sigma}{\Gamma \vdash \phi^{\bullet}, \Sigma} \textit{Activate}$$

$$\frac{}{\phi \vdash \phi^\bullet}\ \textit{Identity}$$

$$\frac{\Gamma \vdash \phi^\bullet, \Sigma}{\Gamma, \psi \vdash \phi^\bullet, \Sigma}\ \textit{Weakening} \qquad \frac{\Gamma \vdash \phi^\bullet, \Sigma}{\Gamma \vdash \phi^\bullet, \Sigma, \psi^a}\ \textit{Weakening}_p$$

$$\frac{\Gamma, \psi, \psi \vdash \phi^\bullet, \Sigma}{\Gamma, \psi \vdash \phi^\bullet, \Sigma}\ \textit{Contraction} \qquad \frac{\Gamma \vdash \phi^\bullet, \Sigma, \psi^a, \psi^a}{\Gamma \vdash \phi^\bullet, \Sigma, \psi^a}\ \textit{Contraction}_p$$

$$\frac{\Gamma, \phi \vdash \psi^\bullet, \Sigma}{\Gamma \vdash (\phi \supset \psi)^\bullet, \Sigma}\ (\supset_\mathcal{I}) \qquad \frac{\Gamma \vdash (\phi \supset \psi)^\bullet, \Sigma \qquad \Gamma' \vdash \phi^\bullet, \Sigma'}{\Gamma, \Gamma' \vdash \psi^\bullet, \Sigma, \Sigma'}\ (\supset_\mathcal{E})$$

$$\frac{\Gamma \vdash \phi^\bullet, \Sigma \qquad \Gamma' \vdash \psi^\bullet, \Sigma'}{\Gamma, \Gamma' \vdash (\phi \wedge \psi)^\bullet, \Sigma, \Sigma'}\ (\wedge_\mathcal{I})$$

$$\frac{\Gamma \vdash (\phi \wedge \psi)^\bullet, \Sigma}{\Gamma \vdash \phi^\bullet, \Sigma}\ (\wedge_{\mathcal{E}-1}) \qquad \frac{\Gamma \vdash (\phi \wedge \psi)^\bullet, \Sigma}{\Gamma \vdash \psi^\bullet, \Sigma}\ (\wedge_{\mathcal{E}-2})$$

$$\frac{\Gamma \vdash \phi^\bullet, \Sigma}{\Gamma \vdash (\phi \vee \psi)^\bullet, \Sigma}\ (\vee_{\mathcal{I}-1}) \qquad \frac{\Gamma \vdash \psi^\bullet, \Sigma}{\Gamma \vdash (\phi \vee \psi)^\bullet, \Sigma}\ (\vee_{\mathcal{I}-2})$$

$$\frac{\Gamma \vdash (\phi \vee \psi)^\bullet, \Sigma \qquad \Gamma', \phi \vdash \varphi^\bullet, \Sigma' \qquad \Gamma'', \psi \vdash \varphi^\bullet, \Sigma''}{\Gamma, \Gamma', \Gamma'' \vdash \varphi^\bullet, \Sigma, \Sigma', \Sigma''}\ (\vee_\mathcal{E})$$

$$\frac{\Gamma \vdash \phi^\bullet, \Sigma}{\Gamma \vdash \bot^\bullet, \phi^a, \Sigma}\ \textit{Passify} \qquad \frac{\Gamma \vdash \bot^\bullet, \phi^a, \Sigma}{\Gamma \vdash \phi^\bullet, \Sigma}\ \textit{Activate}$$

Figure 4: Natural Deduction formulation of **CL**: $\mathbf{CL}_\mu$

As it stands this formulation appears to be a conservative extension of the (implication fragment of the) natural deduction formulation in Figure 3. But this is a slight illusion because of the earlier caveat—in any instance of the rules (except the *Activate*) the active formula need not exist. Thus

$$\frac{\Gamma, \phi \vdash \Sigma}{\Gamma \vdash \phi \supset \psi, \Sigma}\ (\supset_\mathcal{I})$$

is a perfectly valid instance of the $\supset_\mathcal{I}$ rule where there is no active formula in the upper sequent and ψ is a completely fresh formula. To make the rôle of active and passive formulae more precise I shall present the formulation with explicit structural rules. This requires adding a new distinguished atomic formula, $\bot$. A result of this decision is that there is always exactly one active conclusion in any deduction. The resulting system, which I shall call $\mathbf{CL}_\mu$, is given in Figure 4. I have also added the conjunction and disjunction connectives to Parigot's original formulation.

Judgements in $\mathbf{CL}_\mu$ are of the form $\Gamma \vdash \phi^\bullet, \Sigma$ where Γ denotes a multiset of formulae and Σ denotes a multiset of formulae labelled with 'passification variables', which we write as ψ^a. The bullet annotation signifies that a formula is active. The *Passify* rule is not permitted if ϕ is $\bot$.

$\mathbf{CL}_\mu$ is a sound and complete formulation of **CL** in the following sense.

Theorem 2. (Parigot) $\vdash_{\mathbf{CL}} \Gamma \Vdash \Delta$ iff $\vdash_{\mathbf{CL}_\mu} \Gamma \Vdash \Delta$.

To demonstrate the power of $\mathbf{CL}_\mu$, here are derivations of Peirce's law and the law of the excluded middle.

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{(\phi \supset \psi) \supset \phi \Vdash (\phi \supset \psi) \supset \phi^\bullet}\,\textit{Identity} \qquad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\phi \Vdash \phi^\bullet}\,\textit{Identity}}{\phi \Vdash \bot^\bullet, \phi^a}\,\textit{Passify}}{\phi \Vdash \bot^\bullet, \phi^a, \psi^b}\,\textit{Weakening}_p}{\phi \Vdash \psi^\bullet, \phi^a}\,\textit{Activate}}{\Vdash \phi \supset \psi^\bullet, \phi^a}\,(\supset_\mathcal{I})}{(\phi \supset \psi) \supset \phi \Vdash \phi^\bullet, \phi^a}\,(\supset_\mathcal{E})}{(\phi \supset \psi) \supset \phi \Vdash \bot^\bullet, \phi^a, \phi^c}\,\textit{Passify}}{(\phi \supset \psi) \supset \phi \Vdash \bot^\bullet, \phi^d}\,\textit{Contraction}_p}{(\phi \supset \psi) \supset \phi \Vdash \phi^\bullet}\,\textit{Activate}}{\Vdash ((\phi \supset \psi) \supset \phi) \supset \phi^\bullet}\,(\supset_\mathcal{I})
$$

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\phi \Vdash \phi}\,\textit{Identity}}{\phi \Vdash (\phi \vee \neg\phi)^\bullet}\,(\vee_{\mathcal{I}-1})}{\phi \Vdash \bot^\bullet, (\phi \vee \neg\phi)^a}\,\textit{Passify}}{\Vdash (\phi \supset \bot)^\bullet, (\phi \vee \neg\phi)^a}\,(\supset_\mathcal{I})}{\Vdash (\phi \vee \neg\phi)^\bullet, (\phi \vee \neg\phi)^a}\,(\vee_{\mathcal{I}-2})}{\Vdash \bot^\bullet, (\phi \vee \neg\phi)^a, (\phi \vee \neg\phi)^b}\,\textit{Passify}}{\Vdash \bot^\bullet, (\phi \vee \neg\phi)^c}\,\textit{Contraction}_p}{\Vdash (\phi \vee \neg\phi)^\bullet}\,\textit{Activate}
$$

(It is quite instructive to compare these deductions to the sequent calculus derivations in §2.1.) We can apply the Curry-Howard (formulae-as-types) correspondence to $\mathbf{CL}_\mu$ to get what Parigot calls the (typed) $\lambda\mu$-calculus.[3] This is given in Figure 5.

[3] I have diverged a little from Parigot's original syntax. Rather than his $\mu a{:}\,\phi.M$, I write $\mathsf{pass}^\phi_a(M)$, and rather than his $[a{:}\,\phi]M$, I write $\mathsf{act}^\phi_a(M)$.

$$\frac{}{x{:}\,\phi \rhd x{:}\,\phi}\ \textit{Identity}$$

$$\frac{\Gamma \rhd M{:}\,\phi, \Sigma}{\Gamma, x{:}\,\psi \rhd M{:}\,\phi, \Sigma}\ \textit{Weakening} \qquad \frac{\Gamma \rhd M{:}\,\phi, \Sigma}{\Gamma \rhd M{:}\,\phi, \Sigma, a{:}\,\psi}\ \textit{Weakening}_p$$

$$\frac{\Gamma, x{:}\,\psi, y{:}\,\psi \rhd M{:}\,\phi, \Sigma}{\Gamma, z{:}\,\psi \rhd M[x, y := z]{:}\,\phi, \Sigma}\ \textit{Contraction} \qquad \frac{\Gamma \rhd M{:}\,\phi, \Sigma, a{:}\,\psi, b{:}\,\psi}{\Gamma \rhd M[a, b := c]{:}\,\phi, \Sigma, c{:}\,\psi}\ \textit{Contraction}_p$$

$$\frac{\Gamma, x{:}\,\phi \rhd M{:}\,\psi, \Sigma}{\Gamma \rhd \lambda x{:}\,\phi.M{:}\,\phi \supset \psi, \Sigma}\ (\supset_{\mathcal{I}}) \qquad \frac{\Gamma \rhd M{:}\,\phi \supset \psi, \Sigma \qquad \Gamma' \rhd N{:}\,\phi, \Sigma'}{\Gamma, \Gamma' \rhd MN{:}\,\psi, \Sigma, \Sigma'}\ (\supset_{\mathcal{E}})$$

$$\frac{\Gamma \rhd M{:}\,\phi, \Sigma \quad \Gamma' \rhd N{:}\,\psi, \Sigma'}{\Gamma, \Gamma' \rhd \langle M, N\rangle{:}\,\phi \wedge \psi, \Sigma, \Sigma'}\ (\wedge_{\mathcal{I}})$$

$$\frac{\Gamma \rhd M{:}\,\phi \wedge \psi, \Sigma}{\Gamma \rhd \mathsf{fst}(M){:}\,\phi, \Sigma}\ (\wedge_{\mathcal{E}-1}) \qquad \frac{\Gamma \rhd M{:}\,\phi \wedge \psi, \Sigma}{\Gamma \rhd \mathsf{snd}(M){:}\,\psi, \Sigma}\ (\wedge_{\mathcal{E}-2})$$

$$\frac{\Gamma \rhd M{:}\,\phi, \Sigma}{\Gamma \rhd \mathsf{inl}(M){:}\,\phi \vee \psi, \Sigma}\ (\vee_{\mathcal{I}-1}) \qquad \frac{\Gamma \rhd M{:}\,\psi, \Sigma}{\Gamma \rhd \mathsf{inr}(M){:}\,\phi \vee \psi, \Sigma}\ (\vee_{\mathcal{I}-2})$$

$$\frac{\Gamma \rhd M{:}\,\phi \vee \psi, \Sigma \quad \Gamma', x{:}\,\phi \rhd N{:}\,\varphi, \Sigma' \quad \Gamma'', y{:}\,\psi \rhd P{:}\,\varphi, \Sigma''}{\Gamma, \Gamma', \Gamma'' \rhd \mathsf{case}\ M\ \mathsf{of}\ \mathsf{inl}(x)\ \rightarrow\ N \parallel \mathsf{inr}(y)\ \rightarrow\ P{:}\,\varphi, \Sigma, \Sigma', \Sigma''}\ (\vee_{\mathcal{E}})$$

$$\frac{\Gamma \rhd M{:}\,\phi, \Sigma}{\Gamma \rhd \mathsf{pass}_a^{\phi}(M){:}\,\bot, a{:}\,\phi, \Sigma}\ \textit{Passify} \qquad \frac{\Gamma \rhd M{:}\,\bot, a{:}\,\phi, \Sigma}{\Gamma \rhd \mathsf{act}_a^{\phi}(M){:}\,\phi, \Sigma}\ \textit{Activate}$$

Figure 5: The $\lambda\mu$-calculus

Raw terms are then given by the grammar

$$
\begin{array}{rcll}
M & ::= & x & \text{Variable} \\
& | & \lambda x{:}\,\phi.M & \text{Abstraction} \\
& | & MM & \text{Application} \\
& | & \langle M, M\rangle & \text{Pair} \\
& | & \mathsf{fst}(M) & \text{First Projection} \\
& | & \mathsf{snd}(M) & \text{Second Projection} \\
& | & \mathsf{inl}(M) & \text{Left Injection} \\
& | & \mathsf{inr}(M) & \text{Right Injection} \\
& | & \mathsf{case}\ M\ \mathsf{of}\ \mathsf{inl}(x)\ \rightarrow\ M \parallel \mathsf{inr}(x)\ \rightarrow\ M & \text{Conditional} \\
& | & \mathsf{pass}_a^{\phi}(M) & \text{Passification} \\
& | & \mathsf{act}_a^{\phi}(M) & \text{Activation;}
\end{array}
$$

where x is taken from some countable set of variables, ϕ is a well-formed type (formula) and a is taken from some other countable set of passification variables.

Typing judgements are of the form, $\Gamma \triangleright M{:}\,\phi, \Sigma$, where Γ is a multiset of pairs of variables and types written $x{:}\,\psi$, M is a term from the above grammar and Σ denotes a multiset of pairs of passification variables and types written $a{:}\,\varphi$. For conciseness we drop the convention from the formulation of $\mathbf{CL}_{\mu}$ that the active formula is annotated with a bullet, whence the convention that the non-labelled formula on the right of the turnstile is the active formula.

2.3 Reduction Rules

There are β-rules corresponding to the introduction-elimination pairs, along with commuting conversions for the disjunction (as these are quite well known, I shall not give them here) and commuting conversions for the *Activate* rule. These are as follows.[4]

$$
\begin{array}{rcll}
(\lambda x{:}\,\phi.M)N & \rightsquigarrow_{\beta} & M[x := N] & \\
\mathsf{fst}(\langle M, N\rangle) & \rightsquigarrow_{\beta} & M & \\
\mathsf{snd}(\langle M, N\rangle) & \rightsquigarrow_{\beta} & N & \\
\mathsf{case}\ \mathsf{inl}(M)\ \mathsf{of}\ \mathsf{inl}(x)\ \rightarrow\ N \parallel \mathsf{inr}(y)\ \rightarrow\ P & \rightsquigarrow_{\beta} & N[x := M] & \\
\mathsf{case}\ \mathsf{inr}(M)\ \mathsf{of}\ \mathsf{inl}(x)\ \rightarrow\ N \parallel \mathsf{inr}(y)\ \rightarrow\ P & \rightsquigarrow_{\beta} & P[y := M] & \\
\mathsf{act}_a^{\phi}(\mathsf{pass}_a^{\phi}(M)) & \rightsquigarrow_{\beta} & M & \text{where } a \notin \textsc{fn}(M) \\
(\mathsf{act}_a^{\phi\supset\psi}(M))N & \rightsquigarrow_{c} & \mathsf{act}_a^{\psi}(M[\mathsf{pass}_a^{\phi\supset\psi}(P) \Leftarrow \mathsf{pass}_a^{\psi}(PN)]) & \\
\mathsf{fst}(\mathsf{act}_a^{\phi\wedge\psi}(M)) & \rightsquigarrow_{c} & \mathsf{act}_a^{\phi}(M[\mathsf{pass}_a^{\phi\wedge\psi}(P) \Leftarrow \mathsf{pass}_a^{\phi}(\mathsf{fst}(P))]) & \\
\mathsf{snd}(\mathsf{act}_a^{\phi\wedge\psi}(M)) & \rightsquigarrow_{c} & \mathsf{act}_a^{\psi}(M[\mathsf{pass}_a^{\phi\wedge\psi}(P) \Leftarrow \mathsf{pass}_a^{\psi}(\mathsf{snd}(P))]) & \\
\mathsf{case}\ (\mathsf{act}_a^{\phi\vee\psi}(M))\ \mathsf{of} & \rightsquigarrow_{c} & \mathsf{act}_a^{\varphi}(M[\mathsf{pass}_a^{\phi\vee\psi}(R) \Leftarrow & \\
\mathsf{inl}(x) \rightarrow N \parallel \mathsf{inr}(y) \rightarrow P & & \quad \mathsf{pass}_a^{\varphi}(\mathsf{case}\ R\ \mathsf{of}\ \mathsf{inl}(x)\ \rightarrow\ N \parallel \mathsf{inr}(y)\ \rightarrow\ P)]) &
\end{array}
$$

In the β-rule for the unit, $\textsc{fn}(M)$ denotes the set of free names, or passive formulae labels in the term M (I shall omit its rather obvious definition). In the commuting conversions for the *Activate* rule, I have used the notation $M[N \Leftarrow P]$ to denote the term M where all occurrences of the subterm N have been replaced by the term P. In fact this notation for term substitution is somewhat overloaded and is worth elucidation. Consider, for example, an instance of the first commuting conversion, *viz.*

[4]Ideally, following the type-theoretic approach for the presentation of the $\lambda\mu$-calculus, the reduction rules should be presented as equations-in-context.

$$\dfrac{\dfrac{\begin{array}{c}\pi_1\\ \vdots\\ \Gamma \rhd M\colon \bot, a\colon \phi \supset \psi, \Sigma\end{array}}{\Gamma \rhd \mathsf{act}_a^{\phi\supset\psi}(M)\colon \phi \supset \psi, \Sigma}\,\textit{Activate} \qquad \begin{array}{c}\pi_2\\ \vdots\\ \Gamma' \rhd N\colon \phi, \Sigma'\end{array}}{\Gamma, \Gamma' \rhd \mathsf{act}_a^{\phi\supset\psi}(M)N\colon \psi, \Sigma, \Sigma'}\,(\supset\varepsilon).$$

There are three distinct possibilities depending on the passive formula $a\colon \phi \supset \psi$.

1. If a is introduced by an instance of the *Weakening*$_p$ rule. In this case it is replaced with another instance where the weakened passive formula is $a\colon \psi$.

2. If a is introduced by an instance of the *Passify* rule. Thus π_1 is of the form

$$\begin{array}{c}\vdots\\ \dfrac{- \rhd P\colon \phi \supset \psi, +}{- \rhd \mathsf{pass}_a^{\phi\supset\psi}(P)\colon \bot, a\colon \phi \supset \psi, +}\,\textit{Passify}\\ \vdots\end{array}$$

   (note that this instance may appear several times in π_1). These applications are then replaced with

$$\begin{array}{c}\dfrac{\dfrac{\begin{array}{c}\vdots\\ - \rhd P\colon \phi \supset \psi, +\end{array} \qquad \begin{array}{c}\pi_2\\ \vdots\\ \Gamma' \rhd N\colon \phi, \Sigma'\end{array}}{-, \Gamma' \rhd PN\colon \psi, \Sigma', +}\,(\supset\varepsilon)}{-, \Gamma' \rhd \mathsf{pass}_a^{\psi}(PN)\colon \bot, a\colon \psi, \Sigma', +}\,\textit{Passify}\\ \vdots\end{array}$$

3. If a is of type $\phi \supset\bot$ then we get special instances of the above cases. In case (1) the instance of the *Weakening* rule would disappear altogether and in (2) the application would be rewritten instead to

$$\begin{array}{c}\dfrac{\begin{array}{c}\vdots\\ - \rhd P\colon \phi \supset\bot, +\end{array} \qquad \begin{array}{c}\pi_2\\ \vdots\\ \Gamma' \rhd N\colon \phi, \Sigma'\end{array}}{-, \Gamma' \rhd PN\colon \bot, \Sigma', +}\,(\supset\varepsilon)\\ \vdots\end{array}$$

   where, again, a disappears.

Despite the complexities of the commuting conversion, Parigot has (impressively) shown the following results for this reduction system.

Theorem 3. (Parigot)

1. The $\lambda\mu$-calculus is strongly normalising; and

2. The $\lambda\mu$-calculus is confluent.

Remark 1. The ability to introduce a passive formula by the *Weakening*$_p$ rule allows some strange behaviour, when considering the *untyped* $\lambda\mu$-calculus. As mentioned by Parigot, the term $\mathsf{act}_a(M)$, where a is not a free name of M, can be applied to any number of arguments and still give the same result, *viz.*

$$\mathsf{act}_a(M)N_1\cdots N_k \quad \leadsto_c^k \quad \mathsf{act}_a(M)$$

for any number of terms $N_1, \ldots, N_k$.

2.4 Comparison with Cut-Elimination

It is folklore that the sequent calculus formulation of **CL** has the undesirable feature of several disastrous critical pairs. A simple example of this is the following derivation [22, Page 151].

$$\dfrac{\dfrac{\begin{array}{c}\pi_1\\ \vdots \\ \Gamma \vdash \Delta\end{array}}{\Gamma \vdash \Delta, \phi}\ \mathit{Weakening}_{\mathcal{R}} \qquad \dfrac{\begin{array}{c}\pi_2\\ \vdots \\ \Gamma \vdash \Delta\end{array}}{\Gamma, \phi \vdash \Delta}\ \mathit{Weakening}_{\mathcal{L}}}{\Gamma \vdash \Delta}\ \mathit{Cut}$$

Given the usual process of local cut-elimination, it is not clear whether to reduce this proof to π_1 or to π_2. It is interesting to note that this example translates (where I write $\mathcal{M}(\pi)$ to denote the translation of a sequent calculus derivation, π, to a deduction in $\mathbf{CL}_\mu$) to the following application of substitution in Parigot's formulation

$$\mathcal{M}(\pi_2)[x := \mathcal{M}(\pi_1)]$$

where x is *not* a free variable of $\mathcal{M}(\pi_2)$, and so by the definition of substitution, this is equal to

$$\mathcal{M}(\pi_2).$$

Thus Parigot's formulation resolves critical pairs essentially by its syntactic form for the structural rules.[5]

Another important property of Parigot's formulation is that ϕ and $\phi^{\perp\perp}$ are not forced to be equal by the proof theory. Of course we have the derived rules

$$\dfrac{\dfrac{\dfrac{}{x{:}\,\phi \supset \perp \rhd x{:}\,\phi \supset \perp} \qquad \begin{array}{c}\vdots\\ \Gamma \rhd M{:}\,\phi\end{array}}{\Gamma, x{:}\,\phi \supset \perp \rhd xM{:}\,\perp}\ (\supset_\mathcal{E})}{\Gamma \rhd \lambda x{:}\,\phi \supset \perp.xM{:}\,(\phi \supset \perp) \supset \perp}\ (\supset_\mathcal{I})$$

[5]A similar trick is used by de Paiva and Pereira in their multiple conclusion formulation of **IL** [15].

and

$$
\dfrac{\begin{array}{c}\vdots\\ \Gamma \rhd M\colon (\phi \supset \bot) \supset \bot\end{array} \qquad \dfrac{\dfrac{\overline{x\colon \phi \rhd x\colon \phi}}{x\colon \phi \rhd \mathsf{pass}^{\phi}_{a}(x)\colon \bot^{\bullet}, \phi^{a}}\,\textit{Passify}}{\rhd \lambda x.\mathsf{pass}^{\phi}_{a}(x)\colon \phi \supset \bot^{\bullet}, \phi^{a}}\,(\supset_{\mathcal{I}})}{\dfrac{\Gamma \rhd M(\lambda x.\mathsf{pass}^{\phi}_{a}(x))\colon \bot^{\bullet}, \phi^{a}}{\Gamma \rhd \mathsf{act}^{\phi}_{a}(M(\lambda x.\mathsf{pass}^{\phi}_{a}(x)))\colon \phi}\,\textit{Activate.}}\,(\supset_{\mathcal{E}})
$$

Composing the first with the second gives

$$
\begin{array}{rcl}
\mathsf{act}((\lambda x.xM)(\lambda x.\mathsf{pass}(x))) & \rightsquigarrow_{\beta} & \mathsf{act}((\lambda x.\mathsf{pass}(x))M) \\
& \rightsquigarrow_{\beta} & \mathsf{act}(\mathsf{pass}(M)) \\
& \rightsquigarrow_{\beta} & M;
\end{array}
$$

but composing the second with the first yields

$$
\lambda y.y(\mathsf{act}(M(\lambda x.\mathsf{pass}(x)))
$$

which is in (head) normal form.[6]

2.5 Further Consideration on Normal Forms

One motivation for the commuting conversion given in §2.3 is that the *Activate* rule can act as a barrier between an introduction-elimination pair and so we add a reduction to remove it. This has both a familiar and unfamiliar feel to it. We are used to this notion of commuting conversions to permit β-reductions when considering the disjunction in **IL**. However in this case, it introduces a new, unfamiliar, form of substitution, *textual* substitution, where whole subterms are replaced.

One could take these ideas further. Gentzen, as mentioned in §2.2, suggested adding the rule

$$
\dfrac{\begin{array}{c}[\neg\phi]\\ \vdots\\ \bot\end{array}}{\phi}\,\textit{RAA}
$$

to **IL** to get a formulation of **CL**. However, Prawitz [32] noted that applications of this rule can be restricted to cases where ϕ is *atomic*. This is achieved by both factoring formulae through the de Morgan dualities (thus eliminating certain problematic connectives) and by transformation. For example, an application of the above rule where $\phi = \phi \supset \psi$ is transformed to

[6] This property enables Ong [29] to define a categorical model. It is well known that a CCC with an isomorphism $A^{\bot\bot} \cong A$ collapses to a boolean algebra.

$$\begin{array}{c} \dfrac{\dfrac{\dfrac{[\phi \supset \psi] \qquad [\phi]}{\psi}\,(\supset_{\mathcal{E}}) \qquad [\neg\psi]}{\bot}\,(\supset_{\mathcal{E}})}{\neg(\phi \supset \psi)}\,(\supset_{\mathcal{I}}) \\ \vdots \\ \dfrac{\dfrac{\bot}{\psi}\,RAA}{\phi \supset \psi}\,(\supset_{\mathcal{I}}), \end{array}$$

where clearly the size of the formula used in the application of the *RAA* rule has been reduced. Prawitz suggests transforming all applications of this rule until they involve only atomic formulae. However the use of the de Morgan dualities is vital here; Prawitz [31, Footnote 1, Page 50] mentions that this technique does not extend to all the connectives (the problematic one being the disjunction).

Ong [29] suggests a similar strategy for the $\lambda\mu$-calculus by rewriting applications of the *Activate* rule until they are of atomic type, although his motivation is to ensure confluence when considering η-reduction. Given that this technique requires the use of the formula equivalences when considering all the connectives, I shall not consider it here.

3 From **ILL** to **CLL**

Linear logic is the logic obtained by removing the structural rules of *Weakening* and *Contraction*. This has the effect of refining the traditional connectives into two different kinds: multiplicative and additive. Of course what remains is a terribly weak logic. To regain full logical power the structural rules are re-introduced but in a controlled way, via the exponentials. A fuller introduction to linear logic can be found, for example, in Troelstra's book [35], the article by Lincoln [27] or the original article by Girard [21].

3.1 Sequent Calculus

Unlike the case for **IL** and **CL**, the grammar for intuitionistic linear and classical linear formulae are different. For **ILL** the grammar is

$$\phi ::= p \mid \phi \otimes \phi \mid \phi \multimap \phi \mid \phi \& \phi \mid \phi \oplus \phi \mid !\phi,$$

where p is taken from some countable set of atomic formulae which contains the distinguished elements I (the unit for $\otimes$), $\mathbf{t}$ (the unit for $\&$) and $\mathbf{f}$ (the unit for $\oplus$). The sequent calculus formulation of **ILL** is given in Figure 6.

As is the case for **CL**, to extend this formulation to **CLL** we add multiple conclusions. Interestingly this introduces three new connectives (and a unit). A multiplicative disjunction ('par', $⅋$), its unit $\bot$, an exponential ('why not', ?) and a linear negation $(-^{\bot})$.[7] The sequent calculus formulation is given in Figure 7.

[7]This is a slightly contentious point. There *is* a fragment of linear logic, *full intutionistic linear logic* [25, 10], which has multiple conclusions and these 'classical' connectives and, yet, can still be seen as an intuitionistic fragment.

$$\frac{}{\phi \vdash \phi}\ \textit{Identity}$$

$$\frac{\Gamma \vdash \phi \qquad \phi, \Delta \vdash \psi}{\Gamma, \Delta \vdash \psi}\ \textit{Cut}$$

$$\frac{}{\Gamma \vdash \mathbf{t}}\ (\mathbf{t}_{\mathcal{R}}) \qquad \frac{}{\Gamma, \mathbf{f} \vdash \phi}\ (\mathbf{f}_{\mathcal{L}})$$

$$\frac{\Gamma \vdash \phi}{\Gamma, I \vdash \phi}\ (I_{\mathcal{L}}) \qquad \frac{}{\vdash I}\ (I_{\mathcal{R}})$$

$$\frac{\Gamma, \phi, \psi \vdash \varphi}{\Gamma, \phi \otimes \psi \vdash \varphi}\ (\otimes_{\mathcal{L}}) \qquad \frac{\Gamma \vdash \phi \qquad \Delta \vdash \psi}{\Gamma, \Delta \vdash \phi \otimes \psi}\ (\otimes_{\mathcal{R}})$$

$$\frac{\Gamma \vdash \phi \qquad \Delta, \psi \vdash \varphi}{\Gamma, \Delta, \phi \multimap \psi \vdash \varphi}\ (\multimap_{\mathcal{L}}) \qquad \frac{\Gamma, \phi \vdash \psi}{\Gamma \vdash \phi \multimap \psi}\ (\multimap_{\mathcal{R}})$$

$$\frac{\Gamma, \phi \vdash \varphi}{\Gamma, \phi \& \psi \vdash \varphi}\ (\&_{\mathcal{L}-1}) \qquad \frac{\Gamma, \psi \vdash \varphi}{\Gamma, \phi \& \psi \vdash \varphi}\ (\&_{\mathcal{L}-2})$$

$$\frac{\Gamma \vdash \phi \qquad \Gamma \vdash \psi}{\Gamma \vdash \phi \& \psi}\ (\&_{\mathcal{R}})$$

$$\frac{\Gamma, \phi \vdash \varphi \qquad \Gamma, \psi \vdash \varphi}{\Gamma, \phi \oplus \psi \vdash \varphi}\ (\oplus_{\mathcal{L}})$$

$$\frac{\Gamma \vdash \phi}{\Gamma \vdash \phi \oplus \psi}\ (\oplus_{\mathcal{R}-1}) \qquad \frac{\Gamma \vdash \psi}{\Gamma \vdash \phi \oplus \psi}\ (\oplus_{\mathcal{R}-2})$$

$$\frac{\Gamma \vdash \psi}{\Gamma, !\phi \vdash \psi}\ \textit{Weakening} \qquad \frac{\Gamma, !\phi, !\phi \vdash \psi}{\Gamma, !\phi \vdash \psi}\ \textit{Contraction}$$

$$\frac{\Gamma, \phi \vdash \psi}{\Gamma, !\phi \vdash \psi}\ \textit{Dereliction} \qquad \frac{!\Gamma \vdash \phi}{!\Gamma \vdash !\phi}\ \textit{Promotion}$$

Figure 6: Sequent Calculus Formulation of **ILL**

$$\frac{}{\phi \vdash \phi}\ \textit{Identity} \qquad \frac{\Gamma_1 \vdash \phi, \Delta_1 \qquad \phi, \Gamma_2 \vdash \Delta_2}{\Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2}\ \textit{Cut}$$

$$\frac{}{\Gamma, \mathbf{f} \vdash \Delta}\ (\mathbf{f}_{\mathcal{L}}) \qquad \frac{}{\bot \vdash}\ (\bot_{\mathcal{L}}) \qquad \frac{\Gamma \vdash \Delta}{\Gamma \vdash \bot, \Delta}\ (\bot_{\mathcal{R}})$$

$$\frac{\Gamma \vdash \Delta}{\Gamma, I \vdash \Delta}\ (I_{\mathcal{L}}) \qquad \frac{}{\vdash I}\ (I_{\mathcal{R}}) \qquad \frac{}{\Gamma \vdash \mathbf{t}, \Delta}\ (\mathbf{t}_{\mathcal{R}})$$

$$\frac{\Gamma, \phi, \psi \vdash \Delta}{\Gamma, \phi \otimes \psi \vdash \Delta}\ (\otimes_{\mathcal{L}}) \qquad \frac{\Gamma_1 \vdash \phi, \Delta_1 \qquad \Gamma_2 \vdash \psi, \Delta_2}{\Gamma_1, \Gamma_2 \vdash \phi \otimes \psi, \Delta_1, \Delta_2}\ (\otimes_{\mathcal{R}})$$

$$\frac{\Gamma_1 \vdash \phi, \Delta_1 \qquad \Gamma_2, \psi \vdash \Delta_2}{\Gamma_1, \phi \multimap \psi, \Gamma_2 \vdash \Delta_1, \Delta_2}\ (\multimap_{\mathcal{L}}) \qquad \frac{\Gamma, \phi \vdash \psi, \Delta}{\Gamma \vdash \phi \multimap \psi, \Delta}\ (\multimap_{\mathcal{R}})$$

$$\frac{\Gamma, \phi \vdash \Delta}{\Gamma, \phi \& \psi \vdash \Delta}\ (\&_{\mathcal{L}-1}) \qquad \frac{\Gamma, \psi \vdash \Delta}{\Gamma, \phi \& \psi \vdash \Delta}\ (\&_{\mathcal{L}-2}) \qquad \frac{\Gamma \vdash \phi, \Delta \qquad \Gamma \vdash \psi, \Delta}{\Gamma \vdash \phi \& \psi, \Delta}\ (\&_{\mathcal{R}})$$

$$\frac{\Gamma, \phi \vdash \Delta \qquad \Gamma, \psi \vdash \Delta}{\Gamma, \phi \oplus \psi \vdash \Delta}\ (\oplus_{\mathcal{L}}) \qquad \frac{\Gamma \vdash \phi, \Delta}{\Gamma \vdash \phi \oplus \psi, \Delta}\ (\oplus_{\mathcal{R}-1}) \qquad \frac{\Gamma \vdash \psi, \Delta}{\Gamma \vdash \phi \oplus \psi, \Delta}\ (\oplus_{\mathcal{R}-2})$$

$$\frac{\Gamma_1, \phi \vdash \Delta_1 \qquad \Gamma_2, \psi \vdash \Delta_2}{\Gamma_1, \Gamma_2, \phi \parr \psi \vdash \Delta_1, \Delta_2}\ (\parr_{\mathcal{L}}) \qquad \frac{\Gamma \vdash \phi, \psi, \Delta}{\Gamma \vdash \phi \parr \psi, \Delta}\ (\parr_{\mathcal{R}})$$

$$\frac{\Gamma \vdash \phi, \Delta}{\Gamma, \phi^{\perp} \vdash \Delta}\ ({}^{\perp}{}_{\mathcal{L}}) \qquad \frac{\Gamma, \phi \vdash \Delta}{\Gamma \vdash \phi^{\perp}, \Delta}\ ({}^{\perp}{}_{\mathcal{R}})$$

$$\frac{\Gamma \vdash \Delta}{\Gamma, !\phi \vdash \Delta}\ \textit{Weakening}_{\mathcal{L}} \qquad \frac{\Gamma \vdash \Delta}{\Gamma \vdash ?\phi, \Delta}\ \textit{Weakening}_{\mathcal{R}}$$

$$\frac{\Gamma, !\phi, !\phi \vdash \Delta}{\Gamma, !\phi \vdash \Delta}\ \textit{Contraction}_{\mathcal{L}} \qquad \frac{\Gamma \vdash ?\phi, ?\phi, \Delta}{\Gamma \vdash ?\phi, \Delta}\ \textit{Contraction}_{\mathcal{R}}$$

$$\frac{\Gamma, \phi \vdash \Delta}{\Gamma, !\phi \vdash \Delta}\ \textit{Dereliction}_{\mathcal{L}} \qquad \frac{\Gamma \vdash \phi, \Delta}{\Gamma \vdash ?\phi, \Delta}\ \textit{Dereliction}_{\mathcal{R}}$$

$$\frac{!\Gamma, \phi \vdash ?\Delta}{!\Gamma, ?\phi \vdash ?\Delta}\ \textit{Promotion}_{\mathcal{L}} \qquad \frac{!\Gamma \vdash \phi, ?\Delta}{!\Gamma \vdash !\phi, ?\Delta}\ \textit{Promotion}_{\mathcal{R}}$$

Figure 7: Two-Sided Sequent Calculus Formulation of **CLL**

The linear negation is extremely well-behaved and gives a number of formulae equivalences, reminiscent of the de Morgan dualities of **CL**, *viz.*

$$\begin{array}{rcl} (\phi^{\perp})^{\perp} & \equiv & \phi, \\ (\phi \otimes \psi)^{\perp} & \equiv & \phi^{\perp} \parr \psi^{\perp}, \\ (\phi \parr \psi)^{\perp} & \equiv & \phi^{\perp} \otimes \psi^{\perp}, \\ (\phi \oplus \psi)^{\perp} & \equiv & \phi^{\perp} \& \psi^{\perp}, \\ (\phi \& \psi)^{\perp} & \equiv & \phi^{\perp} \oplus \psi^{\perp}, \\ (!\phi)^{\perp} & \equiv & ?\phi^{\perp}, \\ (?\phi)^{\perp} & \equiv & !\phi^{\perp}, \\ I^{\perp} & \equiv & \perp, \\ \mathbf{f}^{\perp} & \equiv & \mathbf{t}. \end{array}$$

As is the case for **CL**, the (linear) implication can be seen as a defined connective, *viz.*

$$\phi \multimap \psi \stackrel{\text{def}}{=} \phi^{\perp} \parr \psi.$$

These equivalences give the possibility of writing any sequent $\Gamma \vdash \Delta$ as $\vdash \Gamma^{\perp}, \Delta$ and so give a one-sided sequent calculus formulation of **CLL**. This leads to the notion of a proof net, but this is not explored here. Again the reader is refered to the paper by Girard [21].

3.2 Natural Deduction

The natural deduction formulation of **ILL** proved harder to formulate and is studied quite closely in my thesis [8]. The difficulty is in giving the correct formulation of the exponential, !. The feature of this natural deduction formulation is that linearity entails that packets contain exactly one formula. The natural deduction formulation is given in Figure 8. (I shall drop any further consideration of the additive units, **t** and **f**, as their computational content appears limited.)

It is possible to extend the natural deduction formulation of **ILL** using Parigot's methodology, outlined in §2.2. However this process is not entirely straightforward. Firstly it has to be established what the unit is in the linear equivalent of the *Passify* and *Activate* rules. It turns out that it is, $\perp$, the unit for Par ($\parr$).[8] There is no way to build-in the classical connectives directly; rather they have to be defined as follows

$$\begin{array}{rcl} \phi^{\perp} & \stackrel{\text{def}}{=} & \phi \multimap \perp, \\ ?\phi & \stackrel{\text{def}}{=} & (!\phi^{\perp})^{\perp}, \text{and} \\ \phi \parr \psi & \stackrel{\text{def}}{=} & ((\phi^{\perp}) \otimes (\psi^{\perp}))^{\perp}. \end{array}$$

A surprise is that the *Promotion* rule has to be extended for the classical formulation. It seems that, rather, its **ILL** formulation is a particular instance of the full classical formulation.

The natural deduction formulation of **CLL**, $\mathbf{CLL}_{\mu}$, is given in Figure 9. Again the $\perp_{\mathcal{I}}$ is only permitted if the formula being passified is not $\perp$. This formulation is sound and complete in the usual sense.

Theorem 4. $\vdash_{\mathbf{CLL}} \Gamma \vdash \Delta$ iff $\vdash_{\mathbf{CLL}_{\mu}} \Gamma \vdash \Delta$.

Applying the Curry-Howard correspondence to $\mathbf{CLL}_{\mu}$ yields the (typed) linear $\lambda\mu$-calculus, which is given in Figure 10.

[8] As the passification rule really introduces the par unit and the activation rule eliminates it, they shall be referred to as introduction and elimination rules.

$$\frac{}{\phi \vdash \phi}\ \textit{Identity}$$

$$\frac{\Gamma, \phi \vdash \psi}{\Gamma \vdash \phi \multimap \psi}\ (\multimap_{\mathcal{I}}) \qquad \frac{\Gamma \vdash \phi \multimap \psi \qquad \Delta \vdash \phi}{\Gamma, \Delta \vdash \psi}\ (\multimap_{\mathcal{E}})$$

$$\frac{}{\vdash I}\ (I_{\mathcal{I}}) \qquad \frac{\Gamma \vdash I \qquad \Delta \vdash \phi}{\Gamma, \Delta \vdash \phi}\ (I_{\mathcal{E}})$$

$$\frac{\Gamma \vdash \phi \qquad \Delta \vdash \psi}{\Gamma, \Delta \vdash \phi \otimes \psi}\ (\otimes_{\mathcal{I}}) \qquad \frac{\Gamma \vdash \phi \otimes \psi \qquad \Delta, \phi, \psi \vdash \varphi}{\Gamma, \Delta \vdash \varphi}\ (\otimes_{\mathcal{E}})$$

$$\frac{\Gamma \vdash \phi \qquad \Gamma \vdash \psi}{\Gamma \vdash \phi \& \psi}\ (\&_{\mathcal{I}}) \qquad \frac{\Gamma \vdash \phi \& \psi}{\Gamma \vdash \phi}\ (\&_{\mathcal{E}-1}) \qquad \frac{\Gamma \vdash \phi \& \psi}{\Gamma \vdash \psi}\ (\&_{\mathcal{E}-2})$$

$$\frac{\Gamma \vdash \phi}{\Gamma \vdash \phi \oplus \psi}\ (\oplus_{\mathcal{I}-1}) \qquad \frac{\Gamma \vdash \psi}{\Gamma \vdash \phi \oplus \psi}\ (\oplus_{\mathcal{I}-2})$$

$$\frac{\Gamma \vdash \phi \oplus \psi \qquad \Delta, \phi \vdash \varphi \qquad \Delta, \psi \vdash \varphi}{\Gamma, \Delta \vdash \varphi}\ (\oplus_{\mathcal{E}})$$

$$\frac{\Gamma_1 \vdash !\phi_1 \quad \cdots \quad \Gamma_n \vdash !\phi_n \qquad !\phi_1, \ldots, !\phi_n \vdash \psi}{\Gamma_1, \ldots, \Gamma_n \vdash !\psi}\ \textit{Promotion}$$

$$\frac{\Gamma \vdash !\phi \qquad \Delta \vdash \psi}{\Gamma, \Delta \vdash \psi}\ \textit{Weakening}$$

$$\frac{\Gamma \vdash !\phi \qquad \Delta, !\phi, !\phi \vdash \psi}{\Gamma, \Delta \vdash \psi}\ \textit{Contraction}$$

$$\frac{\Gamma \vdash !\phi}{\Gamma \vdash \phi}\ \textit{Dereliction}$$

Figure 8: Natural Deduction Formulation of **ILL**

$$\frac{}{\phi \vdash \phi^\bullet}\ \textit{Identity}$$

$$\frac{\Gamma, \phi \vdash \psi^\bullet, \Sigma}{\Gamma \vdash (\phi \multimap \psi)^\bullet, \Sigma}\ (\multimap_{\mathcal{I}}) \qquad \frac{\Gamma \vdash (\phi \multimap \psi)^\bullet, \Sigma \qquad \Delta \vdash \phi^\bullet, \Sigma'}{\Gamma, \Delta \vdash \psi^\bullet, \Sigma, \Sigma'}\ (\multimap_{\mathcal{E}})$$

$$\frac{}{\vdash I^\bullet}\ (I_{\mathcal{I}}) \qquad \frac{\Gamma \vdash I^\bullet, \Sigma \qquad \Delta \vdash \phi^\bullet, \Sigma'}{\Gamma, \Delta \vdash \phi^\bullet, \Sigma, \Sigma'}\ (I_{\mathcal{E}})$$

$$\frac{\Gamma \vdash \phi^\bullet, \Sigma \qquad \Delta \vdash \psi^\bullet, \Sigma'}{\Gamma, \Delta \vdash (\phi \otimes \psi)^\bullet, \Sigma, \Sigma'}\ (\otimes_{\mathcal{I}})$$

$$\frac{\Gamma \vdash (\phi \otimes \psi)^\bullet, \Sigma \qquad \Delta, \phi, \psi \vdash \varphi^\bullet, \Sigma'}{\Gamma, \Delta \vdash \varphi^\bullet, \Sigma, \Sigma'}\ (\otimes_{\mathcal{E}})$$

$$\frac{\Gamma \vdash \phi^\bullet, \Sigma \qquad \Gamma \vdash \psi^\bullet, \Sigma}{\Gamma \vdash (\phi \& \psi)^\bullet, \Sigma}\ (\&_{\mathcal{I}})$$

$$\frac{\Gamma \vdash (\phi \& \psi)^\bullet, \Sigma}{\Gamma \vdash \phi^\bullet, \Sigma}\ (\&_{\mathcal{E}-1}) \qquad \frac{\Gamma \vdash (\phi \& \psi)^\bullet, \Sigma}{\Gamma \vdash \psi^\bullet, \Sigma}\ (\&_{\mathcal{E}-2})$$

$$\frac{\Gamma \vdash \phi^\bullet, \Sigma}{\Gamma \vdash (\phi \oplus \psi)^\bullet, \Sigma}\ (\oplus_{\mathcal{I}-1}) \qquad \frac{\Gamma \vdash \psi^\bullet, \Sigma}{\Gamma \vdash (\phi \oplus \psi)^\bullet, \Sigma}\ (\oplus_{\mathcal{I}-2})$$

$$\frac{\Gamma \vdash (\phi \oplus \psi)^\bullet, \Sigma \qquad \Delta, \phi \vdash \varphi^\bullet, \Sigma' \qquad \Delta, \psi \vdash \varphi^\bullet, \Sigma'}{\Gamma, \Delta \vdash \varphi^\bullet, \Sigma, \Sigma'}\ (\oplus_{\mathcal{E}})$$

$$\frac{\begin{array}{ll} \Gamma_1 \vdash !\phi_1^\bullet, \Sigma_1 & \Delta_1 \vdash !((!\varphi_1 \multimap \bot) \multimap \bot)^\bullet, \Upsilon_1 \\ \Gamma_n \vdash !\phi_n^\bullet, \Sigma_n & \Delta_m \vdash !((!\varphi_m \multimap \bot) \multimap \bot)^\bullet, \Upsilon_m \\ \multicolumn{2}{c}{!\phi_1, \ldots, !\phi_n \vdash \psi^\bullet, (!\varphi_1 \multimap \bot)^{a_1}, \ldots, (!\varphi_m \multimap \bot)^{a_m}} \end{array}}{\vec{\Gamma}, \vec{\Delta} \vdash !\psi^\bullet, \vec{\Sigma}, \vec{\Upsilon}}\ \textit{Promotion}$$

$$\frac{\Gamma \vdash !\phi^\bullet, \Sigma}{\Gamma \vdash \phi^\bullet, \Sigma}\ \textit{Dereliction}$$

$$\frac{\Gamma \vdash !\phi^\bullet, \Sigma \qquad \Delta \vdash \psi^\bullet, \Sigma'}{\Gamma, \Delta \vdash \psi^\bullet, \Sigma, \Sigma'}\ \textit{Weakening}$$

$$\frac{\Gamma \vdash !\phi^\bullet, \Sigma \qquad \Delta, !\phi, !\phi \vdash \psi^\bullet, \Sigma'}{\Gamma, \Delta \vdash \psi^\bullet, \Sigma, \Sigma'}\ \textit{Contraction}$$

$$\frac{\Gamma \vdash \phi^\bullet, \Sigma}{\Gamma \vdash \bot^\bullet, \phi^a, \Sigma}\ (\bot_{\mathcal{I}})$$

$$\frac{\Gamma \vdash \bot^\bullet, \phi^a, \Sigma}{\Gamma \vdash \phi^\bullet, \Sigma}\ (\bot_{\mathcal{E}})$$

Figure 9: Natural Deduction Formulation of **CLL**: $\mathbf{CLL}_\mu$

$$\frac{}{x\colon\phi \rhd x\colon\phi}\ \textit{Identity}$$

$$\frac{\Gamma, x\colon\phi \rhd M\colon\psi, \Sigma}{\Gamma \rhd \lambda x\colon\phi.M\colon\phi\multimap\psi, \Sigma}\ (\multimap_{\mathcal{I}}) \qquad \frac{\Gamma \rhd M\colon\phi\multimap\psi, \Sigma \qquad \Delta \rhd N\colon\phi, \Sigma'}{\Gamma, \Delta \rhd MN\colon\psi, \Sigma, \Sigma'}\ (\multimap_{\mathcal{E}})$$

$$\frac{}{\rhd *\colon I}\ (I_{\mathcal{I}}) \qquad \frac{\Gamma \rhd M\colon I, \Sigma \qquad \Delta \rhd N\colon\phi, \Sigma'}{\Gamma, \Delta \rhd \mathsf{let}\ M\ \mathsf{be}\ *\ \mathsf{in}\ N\colon\phi, \Sigma, \Sigma'}\ (I_{\mathcal{E}})$$

$$\frac{\Gamma \rhd M\colon\phi, \Sigma \qquad \Delta \rhd N\colon\psi, \Sigma'}{\Gamma, \Delta \rhd M\otimes N\colon\phi\otimes\psi, \Sigma, \Sigma'}\ (\otimes_{\mathcal{I}})$$

$$\frac{\Gamma \rhd M\colon\phi\otimes\psi, \Sigma \qquad \Delta, x\colon\phi, y\colon\psi \rhd N\colon\varphi, \Sigma'}{\Gamma, \Delta \rhd \mathsf{let}\ M\ \mathsf{be}\ x\otimes y\ \mathsf{in}\ N\colon\varphi, \Sigma, \Sigma'}\ (\otimes_{\mathcal{E}})$$

$$\frac{\Gamma \rhd M\colon\phi, \Sigma \qquad \Gamma \rhd N\colon\psi, \Sigma}{\Gamma \rhd \langle M, N\rangle\colon\phi\&\psi, \Sigma}\ (\&_{\mathcal{I}})$$

$$\frac{\Gamma \rhd M\colon\phi\&\psi, \Sigma}{\Gamma \rhd \mathsf{fst}(M)\colon\phi, \Sigma}\ (\&_{\mathcal{E}-1}) \qquad \frac{\Gamma \rhd M\colon\phi\&\psi, \Sigma}{\Gamma \rhd \mathsf{snd}(M)\colon\psi, \Sigma}\ (\&_{\mathcal{E}-2})$$

$$\frac{\Gamma \rhd M\colon\phi, \Sigma}{\Gamma \rhd \mathsf{inl}(M)\colon\phi\oplus\psi, \Sigma}\ (\oplus_{\mathcal{I}-1}) \qquad \frac{\Gamma \rhd M\colon\psi, \Sigma}{\Gamma \rhd \mathsf{inr}(M)\colon\phi\oplus\psi, \Sigma}\ (\oplus_{\mathcal{I}-2})$$

$$\frac{\Gamma \rhd M\colon\phi\oplus\psi, \Sigma \qquad \Delta, x\colon\phi \rhd N\colon\varphi, \Sigma' \qquad \Delta, y\colon\psi \rhd P\colon\varphi, \Sigma'}{\Gamma, \Delta \rhd \mathsf{case}\ M\ \mathsf{of}\ \mathsf{inl}(x)\ \to\ N \parallel \mathsf{inr}(y)\ \to\ P\colon\varphi, \Sigma, \Sigma'}\ (\oplus_{\mathcal{E}})$$

$$\frac{\begin{array}{ll} \Gamma_1 \rhd M_1\colon !\phi_1, \Sigma_1 & \Delta_1 \rhd P_1\colon !((!\varphi_1 \multimap \bot)\multimap \bot), \Upsilon_1 \\ \Gamma_n \rhd M_n\colon !\phi_n, \Sigma_n & \Delta_m \rhd P_m\colon !((!\varphi_m \multimap \bot)\multimap \bot), \Upsilon_m \\ \multicolumn{2}{c}{x_1\colon !\phi_1, \ldots, x_n\colon !\phi_n \rhd N\colon\psi, a_1\colon !\varphi_1 \multimap \bot, \ldots, a_m\colon !\varphi_m \multimap \bot} \end{array}}{\vec{\Gamma}, \vec{\Delta} \rhd \mathsf{promote}\ \vec{M}|\vec{P}\ \mathsf{for}\ \vec{x}|\vec{a}\ \mathsf{in}\ N\colon !\psi, \vec{\Sigma}, \vec{\Upsilon}}\ \textit{Promotion}$$

$$\frac{\Gamma \rhd M\colon !\phi, \Sigma}{\Gamma \rhd \mathsf{derelict}(M)\colon\phi, \Sigma}\ \textit{Dereliction}$$

$$\frac{\Gamma \rhd M\colon !\phi, \Sigma \qquad \Delta \rhd N\colon\psi, \Sigma'}{\Gamma, \Delta \rhd \mathsf{discard}\ M\ \mathsf{in}\ N\colon\psi, \Sigma, \Sigma'}\ \textit{Weakening}$$

$$\frac{\Gamma \rhd M\colon !\phi, \Sigma \qquad \Delta, x\colon !\phi, y\colon !\phi \rhd N\colon\psi, \Sigma'}{\Gamma, \Delta \rhd \mathsf{copy}\ M\ \mathsf{as}\ x, y\ \mathsf{in}\ N\colon\psi, \Sigma, \Sigma'}\ \textit{Contraction}$$

$$\frac{\Gamma \rhd M\colon\phi, \Sigma}{\Gamma \rhd \mathsf{unit}_a^{\phi}(M)\colon\bot, a\colon\phi, \Sigma}\ (\bot_{\mathcal{I}})$$

$$\frac{\Gamma \rhd M\colon\bot, a\colon\phi, \Sigma}{\Gamma \rhd \mathsf{deunit}_a^{\phi}(M)\colon\phi, \Sigma}\ (\bot_{\mathcal{E}})$$

Figure 10: The linear $\lambda\mu$-calculus

Raw terms are then given by the grammar

$$
\begin{array}{rcll}
M & ::= & x & \text{Variable} \\
& | & \lambda x{:}\,\phi.M & \text{Abstraction} \\
& | & MM & \text{Application} \\
& | & M{\otimes}M & \text{Multiplicative Pair} \\
& | & \mathsf{let}\ M\ \mathsf{be}\ x{\otimes}x\ \mathsf{in}\ M & \text{Depairing} \\
& | & \langle M, M\rangle & \text{Additive Pair} \\
& | & \mathsf{fst}(M) & \text{First Projection} \\
& | & \mathsf{snd}(M) & \text{Second Projection} \\
& | & \mathsf{inl}(M) & \text{Left Injection} \\
& | & \mathsf{inr}(M) & \text{Right Injection} \\
& | & \mathsf{case}\ M\ \mathsf{of}\ \mathsf{inl}(x)\ \to\ M \parallel \mathsf{inr}(x)\ \to\ M & \text{Conditional} \\
& | & \mathsf{promote}\ \vec{M}|\vec{M}\ \mathsf{for}\ \vec{x}|\vec{a}\ \mathsf{in}\ M & \text{Promote} \\
& | & \mathsf{derelict}(M) & \text{Derelict} \\
& | & \mathsf{discard}\ M\ \mathsf{in}\ M & \text{Discarding} \\
& | & \mathsf{copy}\ M\ \mathsf{as}\ x, x\ \mathsf{in}\ M & \text{Duplication} \\
& | & \mathsf{unit}_a^{\phi}(M) & \text{Passification} \\
& | & \mathsf{deunit}_a^{\phi}(M) & \text{Activation;}
\end{array}
$$

where, as for $\lambda\mu$-calculus, x is taken from some countable set of variables, ϕ is a well-formed type (formula) and a is taken from some countable set of passification variables.

Typing judgements, again as for $\lambda\mu$-calculus, are of the form, $\Gamma \triangleright M{:}\,\phi, \Sigma$, where Γ is a multiset of pairs of variables and types, written $x{:}\,\psi$, M is a term from the above grammar and Σ denotes a multiset of pairs of passification variables and types, written $a{:}\,\varphi$. As is the case for **ILL**, in well-typed terms of the multiplicative-exponential fragment $(\otimes, \multimap, \parr, !, ?)$ variables occur exactly once.

4 Reduction Rules

From the linear λ-calculus (**ILL**) there are both β-rules and commuting conversions. Of course, the reductions for the *Promotion* rule have to be suitably extended. The β-rules are as follows.

$$
\begin{array}{rcl}
(\lambda x{:}\,\phi.M)\ N & \leadsto_\beta & M[x := N] \\
\mathsf{let}\ M{\otimes}N\ \mathsf{be}\ x{\otimes}y\ \mathsf{in}\ P & \leadsto_\beta & P[x := M, y := N] \\
\mathsf{fst}(\langle M, N\rangle) & \leadsto_\beta & M \\
\mathsf{snd}(\langle M, N\rangle) & \leadsto_\beta & N \\
\mathsf{case}\ \mathsf{inl}(M)\ \mathsf{of}\ \mathsf{inl}(x)\ \to\ N \parallel \mathsf{inr}(y)\ \to\ P & \leadsto_\beta & N[x := M] \\
\mathsf{case}\ \mathsf{inr}(M)\ \mathsf{of}\ \mathsf{inl}(x)\ \to\ N \parallel \mathsf{inr}(y)\ \to\ P & \leadsto_\beta & P[y := M] \\
\mathsf{derelict}(\mathsf{promote}\ \vec{M}|\vec{P}\ \mathsf{for}\ \vec{x}|\vec{a}\ \mathsf{in}\ N) & \leadsto_\beta & N\ [x_i := M_i, \\
& & \quad \mathsf{unit}_{a_j}^{!\varphi_j \multimap \perp}(R) \Leftarrow \mathsf{derelict}(P_j)R] \\
\mathsf{discard}\ (\mathsf{promote}\ \vec{M}|\vec{P}\ \mathsf{for}\ \vec{x}|\vec{a}\ \mathsf{in}\ N)\ \mathsf{in}\ R & \leadsto_\beta & \mathsf{discard}\ \vec{M}, \vec{P}\ \mathsf{in}\ R \\
\mathsf{copy}\ (\mathsf{promote}\ \vec{M}|\vec{P}\ \mathsf{for}\ \vec{x}|\vec{a}\ \mathsf{in}\ N)\ \mathsf{as}\ y, z\ \mathsf{in}\ R & \leadsto_\beta & \mathsf{copy}\ \vec{M}\ \mathsf{as}\ \vec{x'}, \vec{x''}\ \mathsf{in} \\
& & \mathsf{copy}\ \vec{P}\ \mathsf{as}\ \vec{w'}, \vec{w''}\ \mathsf{in} \\
& & \ R\ [y := \mathsf{promote}\ \vec{x'}|\vec{w'}\ \mathsf{for}\ \vec{x}|\vec{a}\ \mathsf{in}\ N, \\
& & \quad z := \mathsf{promote}\ \vec{x''}|\vec{w''}\ \mathsf{for}\ \vec{x}|\vec{a}\ \mathsf{in}\ N]
\end{array}
$$

Rather than give all the commuting conversions for the **ILL** connectives (they are given in full in my thesis), I shall only give the 'promote-of-promote' one, which for ease of typesetting I shall present as two reduction rules depending on where the inner *Promotion* rule occurs.

$$\begin{array}{l}\mathsf{promote}\ \vec{Q}|(\mathsf{promote}\ \vec{L}|\vec{M}\ \mathsf{for}\ \vec{x}|\vec{a}\ \mathsf{in}\ N)\ \mathsf{for}\ \vec{y}|b\ \mathsf{in}\ P\\ \rightsquigarrow_c\ \mathsf{promote}\ \vec{Q},\vec{L}|\vec{M}\ \mathsf{for}\ \vec{x},\vec{y}|\vec{a}\ \mathsf{in}\ P[\mathsf{unit}_b^{!\varphi\multimap\bot}(R)\Leftarrow NR]\end{array}$$

and

$$\begin{array}{l}\mathsf{promote}\ (\mathsf{promote}\ \vec{L}|\vec{M}\ \mathsf{for}\ \vec{x}|\vec{a}\ \mathsf{in}\ N)|\vec{Q}\ \mathsf{for}\ y|\vec{b}\ \mathsf{in}\ P\\ \rightsquigarrow_c\ \mathsf{promote}\ \vec{L}|\vec{M},\vec{Q}\ \mathsf{for}\ \vec{x'}|\vec{a'},\vec{b}\ \mathsf{in}\ P[y:=\mathsf{promote}\ \vec{x'}|\mathcal{P}_{\vec{a'}}\ \mathsf{for}\ \vec{x}|\vec{a}\ \mathsf{in}\ N].\end{array}$$

(The term $\mathcal{P}_{\vec{a}}$ is defined in the following section.) There is a β-rule corresponding to the introduction-elimination pair for the $\bot$,[9] and a number of commuting conversions for this unit (as per the discussion in §2.3). These are as follows.

$$\begin{array}{rcl}
\mathsf{deunit}_a^{\phi}(\mathsf{unit}_a^{\phi}(M)) & \rightsquigarrow_\beta & M\\
(\mathsf{deunit}_a^{\phi\multimap\psi}(M))N & \rightsquigarrow_c & \mathsf{deunit}_a^{\psi}(M[\mathsf{unit}_a^{\phi\multimap\psi}(P)\Leftarrow\mathsf{unit}_a^{\psi}(PN)])\\
\mathsf{let}\ \mathsf{deunit}_a^{\phi\otimes\psi}(M)\ \mathsf{be}\ x\otimes y\ \mathsf{in}\ N & \rightsquigarrow_c & \mathsf{deunit}_a^{\varphi}(M[\mathsf{unit}_a^{\phi\otimes\psi}(P)\Leftarrow\mathsf{unit}_a^{\varphi}(\mathsf{let}\ P\ \mathsf{be}\ x\otimes y\ \mathsf{in}\ N)])\\
\mathsf{fst}(\mathsf{deunit}_a^{\phi\&\psi}(M)) & \rightsquigarrow_c & \mathsf{deunit}_a^{\phi}(M[\mathsf{unit}_a^{\phi\&\psi}(P)\Leftarrow\mathsf{unit}_a^{\phi}(\mathsf{fst}(P))])\\
\mathsf{snd}(\mathsf{deunit}_a^{\phi\&\psi}(M)) & \rightsquigarrow_c & \mathsf{deunit}_a^{\psi}(M[\mathsf{unit}_a^{\phi\&\psi}(P)\Leftarrow\mathsf{unit}_a^{\psi}(\mathsf{snd}(P))])\\
\mathsf{case}\ (\mathsf{deunit}_a^{\phi\oplus\psi}(M))\ \mathsf{of} & \rightsquigarrow_c & \mathsf{deunit}_a^{\varphi}(M[\mathsf{unit}_a^{\phi\oplus\psi}(R)\Leftarrow\\
\mathsf{inl}(x)\to N\ \|\ \mathsf{inr}(y)\to P & & \qquad\mathsf{unit}_a^{\varphi}(\mathsf{case}\ R\ \mathsf{of}\ \mathsf{inl}(x)\ \to\ N\ \|\ \mathsf{inr}(y)\ \to\ P)])\\
\mathsf{derelict}(\mathsf{deunit}_a^{!\phi}(M)) & \rightsquigarrow_c & \mathsf{deunit}_a^{\phi}(M[\mathsf{unit}_a^{!\phi}(P)\Leftarrow\mathsf{unit}_a^{\phi}(\mathsf{derelict}(P))])\\
\mathsf{copy}\ (\mathsf{deunit}_a^{!\phi}(M))\ \mathsf{as}\ x,y\ \mathsf{in}\ N & \rightsquigarrow_c & \mathsf{deunit}_a^{\varphi}(M[\mathsf{unit}_a^{!\phi}(P)\Leftarrow\mathsf{unit}_a^{\varphi}(\mathsf{copy}\ P\ \mathsf{as}\ x,y\ \mathsf{in}\ N)])\\
\mathsf{discard}\ (\mathsf{deunit}_a^{!\psi}(M))\ \mathsf{in}\ N & \rightsquigarrow_c & \mathsf{deunit}_a^{\varphi}(M[\mathsf{unit}_a^{!\phi}(P)\Leftarrow\mathsf{unit}_a^{\varphi}(\mathsf{discard}\ P\ \mathsf{in}\ N)])
\end{array}$$

Of course there are η-rules for the connectives. Those for the **ILL** connectives have appeared elsewhere [8, Figure 4.3] and the new one, for the $\bot$, is

$$\mathsf{unit}_b^{\phi}(\mathsf{deunit}_a^{\phi}(M))\quad\rightsquigarrow_\eta\quad M[a:=b]$$

A vital property of this formulation is the so-called subject reduction property.

Theorem 5. If $\Gamma\rhd M{:}\,\phi^\bullet,\Sigma$ and $M\rightsquigarrow_{\beta,c}N$ then $\Gamma\rhd N{:}\,\phi^\bullet,\Sigma$.

I conjecture that, as for the $\lambda\mu$-calculus, the properties of strong normalisation and confluence hold for the linear $\lambda\mu$-calculus.

5 Comparison with Cut-Elimination

In this section I shall first show how to translate derivations in the sequent calculus formulation to deductions in the $\mathbf{CLL}_\mu$. Given this translation I shall consider the principal steps in the cut-elimination process for **CLL** and show how they are reflected in $\mathbf{CLL}_\mu$.[10]

Before giving the translation from the sequent calculus formulation of **CLL** to $\mathbf{CLL}_\mu$, I shall identify a particularly useful term. Its derivation is

[9] Here is an advantage of linearity: we need no side-condition for this rule as we do for the non-linear system.

[10] In fact, for ease of reference, I shall use the term annotations, *viz.* the linear $\lambda\mu$-calculus.

$$
\dfrac{\dfrac{\dfrac{\overline{x\colon !(\phi\multimap\bot)\triangleright x\colon !(\phi\multimap\bot)} \qquad \dfrac{\dfrac{\overline{f\colon !(\phi\multimap\bot)\multimap\bot \triangleright f\colon !(\phi\multimap\bot)\multimap\bot} \qquad \overline{x\colon !(\phi\multimap\bot)\triangleright x\colon !(\phi\multimap\bot)}}{f\colon !(\phi\multimap\bot)\multimap\bot, x\colon !(\phi\multimap\bot)\triangleright fx\colon \bot}\,(\multimap_{\mathcal{E}})}{x\colon !(\phi\multimap\bot)\triangleright \lambda f.fx\colon (!(\phi\multimap\bot)\multimap\bot)\multimap\bot}\,(\multimap_{\mathcal{I}})}{x\colon !(\phi\multimap\bot)\triangleright \mathsf{promote}\ x|-\ \mathsf{for}\ x|-\ \mathsf{in}\ \lambda f.fx\colon !((!(\phi\multimap\bot)\multimap\bot)\multimap\bot)}\,\mathit{Promote}}{\dfrac{x\colon !(\phi\multimap\bot)\triangleright \mathsf{unit}(\mathsf{promote}\ x|-\ \mathsf{for}\ x|-\ \mathsf{in}\ \lambda f.fx)\colon\bot, b\colon !((!(\phi\multimap\bot)\multimap\bot)\multimap\bot)}{\dfrac{\triangleright \lambda x.\mathsf{unit}(\mathsf{promote}\ x|-\ \mathsf{for}\ x|-\ \mathsf{in}\ \lambda f.fx)\colon !(\phi\multimap\bot)\multimap\bot, b\colon !((!(\phi\multimap\bot)\multimap\bot)\multimap\bot)}{\dfrac{\triangleright \mathsf{unit}(\lambda x.\mathsf{unit}(\mathsf{promote}\ x|-\ \mathsf{for}\ x|-\ \mathsf{in}\ \lambda f.fx))\colon\bot, b\colon !((!(\phi\multimap\bot)\multimap\bot)\multimap\bot), a\colon !(\phi\multimap\bot)\multimap\bot}{\triangleright \mathsf{deunit}(\mathsf{unit}(\lambda x.\mathsf{unit}(\mathsf{promote}\ x|-\ \mathsf{for}\ x|-\ \mathsf{in}\ \lambda f.fx)))\colon !((!(\phi\multimap\bot)\multimap\bot)\multimap\bot), a\colon !(\phi\multimap\bot)\multimap\bot}\,(\bot_{\mathcal{E}}).}\,(\bot_{\mathcal{I}})}\,(\multimap_{\mathcal{I}})}\,(\bot_{\mathcal{I}})
$$

I shall refer to this term as $\mathcal{P}_a$, where a is the final passive variable. I shall also use the shorthand $\mathcal{P}_{\vec{a}}$ to represent the obvious extension of the above term. An important property of this term is the following.

Lemma 1. For all appropriately typed terms M, $(\mathsf{derelict}(\mathcal{P}_a))M \rightsquigarrow^*_{\beta,c,\eta} \mathsf{unit}(M)$.

Proof.

$$
\begin{array}{ll}
 & (\mathsf{derelict}(\mathsf{deunit}(\mathsf{unit}(\lambda x.\mathsf{unit}(\mathsf{promote}\ x|-\ \mathsf{for}\ x|-\ \mathsf{in}\ \lambda f.fx)))))M \\
\rightsquigarrow_c & (\mathsf{deunit}(\mathsf{unit}(\lambda x.\mathsf{unit}(\mathsf{derelict}(\mathsf{promote}\ x|-\ \mathsf{for}\ x|-\ \mathsf{in}\ \lambda f.fx)))))M \\
\rightsquigarrow_\beta & (\mathsf{deunit}(\mathsf{unit}(\lambda x.\mathsf{unit}(\lambda f.fx))))M \\
\rightsquigarrow_c & \mathsf{unit}(\lambda x.((\lambda f.fx)M)) \\
\rightsquigarrow_\beta & \mathsf{unit}(\lambda x.Mx) \\
\rightsquigarrow_\eta & \mathsf{unit}(M)
\end{array}
$$

■

I shall show how to translate sequent derivations in **CLL** to deductions in $\mathbf{CLL}_\mu$ by defining a procedure, $\mathcal{M}$, inductively over the sequent derivation.

- A proof of the form

$$
\dfrac{}{\phi \vdash \phi}\,\mathit{Identity}
$$

  is translated to

$$
\dfrac{}{x\colon\phi \triangleright x\colon\phi}\,\mathit{Identity}
$$

- A proof of the form

$$
\dfrac{\begin{array}{c}\pi_1\\ \vdots \\ \Gamma,\phi,\psi \vdash \Delta\end{array}}{\Gamma,\phi\otimes\psi \vdash \Delta}\,(\otimes_{\mathcal{L}})
$$

  is translated to

$$
\dfrac{\overline{z\colon\phi\otimes\psi \triangleright z\colon\phi\otimes\psi} \qquad \begin{array}{c}\mathcal{M}(\pi_1)\\ \vdots \\ \Gamma, x\colon\phi, y\colon\psi \triangleright M\colon\Delta\end{array}}{\Gamma, z\colon\phi\otimes\psi \triangleright \mathsf{let}\ z\ \mathsf{be}\ x\otimes y\ \mathsf{in}\ M\colon\Delta}\,(\otimes_{\mathcal{E}})
$$

- A proof of the form

$$\dfrac{\begin{array}{c}\pi_1\\ \vdots\\ \Gamma \vdash \phi, \Delta\end{array} \qquad \begin{array}{c}\pi_2\\ \vdots\\ \Gamma' \vdash \psi, \Delta'\end{array}}{\Gamma, \Gamma' \vdash \phi \otimes \psi, \Delta, \Delta'}\,(\otimes_{\mathcal{R}})$$

is translated to

$$\dfrac{\begin{array}{c}\mathcal{M}(\pi_1)\\ \vdots\\ \Gamma \triangleright M\colon \phi, \Delta\end{array} \qquad \begin{array}{c}\mathcal{M}(\pi_2)\\ \vdots\\ \Gamma' \triangleright N\colon \psi, \Delta'\end{array}}{\Gamma, \Gamma' \triangleright M \otimes N\colon \phi \otimes \psi, \Delta, \Delta'}\,(\otimes_{\mathcal{I}})$$

- A proof of the form

$$\dfrac{\begin{array}{c}\pi_1\\ \vdots\\ \Gamma \vdash \phi, \Delta\end{array} \qquad \begin{array}{c}\pi_2\\ \vdots\\ \Gamma', \psi \vdash \Delta'\end{array}}{\Gamma, \phi \multimap \psi, \Gamma' \vdash \Delta, \Delta'}\,(\multimap_{\mathcal{L}})$$

is translated to

$$\dfrac{\dfrac{\dfrac{}{f\colon \phi \multimap \psi \triangleright f\colon \phi \multimap \psi} \qquad \begin{array}{c}\mathcal{M}(\pi_1)\\ \vdots\\ \Gamma \triangleright M\colon \phi, \Delta\end{array}}{\Gamma, f\colon \phi \multimap \psi \triangleright fM\colon \psi, \Delta}\,(\multimap_{\mathcal{E}}) \qquad \begin{array}{c}\mathcal{M}(\pi_2)\\ \vdots\\ \Gamma', x\colon \psi \triangleright N\colon \Delta'\end{array}}{\Gamma, \Gamma' \triangleright N[x := fM]\colon \Delta', \Delta}\,\textit{Substitution}$$

- A proof of the form

$$\dfrac{\begin{array}{c}\pi_1\\ \vdots\\ \Gamma, \phi \vdash \psi, \Delta\end{array}}{\Gamma \vdash \phi \multimap \psi, \Delta}\,(\multimap_{\mathcal{R}})$$

is translated to

$$\dfrac{\begin{array}{c}\mathcal{M}(\pi_1)\\ \vdots\\ \Gamma, x\colon \phi \triangleright M\colon \psi, \Delta\end{array}}{\Gamma \triangleright \lambda x.M\colon \phi \multimap \psi, \Delta}\,(\multimap_{\mathcal{I}})$$

- A proof of the form

$$\dfrac{\begin{array}{c}\pi_1\\ \vdots\\ \Gamma, \phi \vdash \Delta\end{array} \qquad \begin{array}{c}\pi_2\\ \vdots\\ \Gamma', \psi \vdash \Delta'\end{array}}{\Gamma, \Gamma', \phi \parr \psi \vdash \Delta, \Delta'}\,(\parr_{\mathcal{L}})$$

is translated to

$$
\dfrac{\dfrac{}{z{:}\,\phi\parr\psi \rhd z{:}\,\phi\parr\psi} \qquad \dfrac{\dfrac{\dfrac{\begin{matrix}\mathcal{M}(\pi_1)\\ \vdots \\ \Gamma, x{:}\,\phi \rhd M{:}\,\Delta\end{matrix}}{\Gamma, x{:}\,\phi \rhd \mathsf{unit}(M){:}\,\bot, \Delta}\,(\bot_{\mathcal{I}})}{\Gamma \rhd \lambda x.\mathsf{unit}(M){:}\,\phi\multimap\bot, \Delta}\,(\multimap_{\mathcal{I}}) \qquad \dfrac{\dfrac{\begin{matrix}\mathcal{M}(\pi_2)\\ \vdots \\ \Gamma', y{:}\,\psi \rhd N{:}\,\Delta'\end{matrix}}{\Gamma', y{:}\,\psi \rhd \mathsf{unit}(N){:}\,\bot, \Delta'}\,(\bot_{\mathcal{I}})}{\Gamma' \rhd \lambda y.\mathsf{unit}(N){:}\,\psi\multimap\bot, \Delta'}\,(\multimap_{\mathcal{I}})}{\Gamma, \Gamma' \rhd (\lambda x.\mathsf{unit}(M))\otimes(\lambda y.\mathsf{unit}(M)){:}\,(\phi\multimap\bot)\otimes(\psi\multimap\bot), \Delta, \Delta'}\,(\otimes_{\mathcal{I}})}{\dfrac{\Gamma, \Gamma', z{:}\,\phi\parr\psi \rhd z\;((\lambda x.\mathsf{unit}(M))\otimes(\lambda y.\mathsf{unit}(M))){:}\,\bot, \Delta, \Delta'}{\Gamma, \Gamma', z{:}\,\phi\parr\psi \rhd \mathsf{deunit}(z\;((\lambda x.\mathsf{unit}(M))\otimes(\lambda y.\mathsf{unit}(M)))){:}\,\Delta, \Delta'}\,(\bot_{\mathcal{E}})}\,(\multimap_{\mathcal{E}})
$$

- A proof of the form

$$
\dfrac{\begin{matrix}\pi_1\\ \vdots \\ \Gamma \vdash \phi, \psi, \Delta\end{matrix}}{\Gamma \vdash \phi\parr\psi, \Delta}\,(\parr_{\mathcal{R}})
$$

is translated to

$$
\dfrac{\dfrac{\dfrac{}{z{:}\,(\phi\multimap\bot)\otimes(\psi\multimap\bot) \rhd z{:}} \qquad \dfrac{\dfrac{}{y{:}\,\psi\multimap\bot \rhd y{:}\,\psi\multimap\bot} \qquad \dfrac{\dfrac{\dfrac{}{x{:}\,\phi\multimap\bot \rhd x{:}\,\phi\multimap\bot} \qquad \dfrac{\dfrac{\begin{matrix}\mathcal{M}(\pi_1)\\ \vdots \\ \Gamma \rhd M{:}\,\varphi, c{:}\,\phi, b{:}\,\psi, \Delta\end{matrix}}{\Gamma \rhd \mathsf{unit}(M){:}\,\bot, c{:}\,\phi, b{:}\,\psi, a{:}\,\varphi, \Delta}\,(\bot_{\mathcal{I}})}{\Gamma \rhd \mathsf{deunit}(\mathsf{unit}(M)){:}\,\phi, b{:}\,\psi, a{:}\,\varphi, \Delta}\,(\bot_{\mathcal{E}})}{\Gamma, x{:}\,(\phi\multimap\bot) \rhd x\;\mathsf{deunit}(\mathsf{unit}(M)){:}\,\bot, b{:}\,\psi, a{:}\,\varphi, \Delta(\multimap_{\mathcal{E}})}}{\Gamma, x{:}\,(\phi\multimap\bot) \rhd \mathsf{deunit}(x\;\mathsf{deunit}(\mathsf{unit}(M))){:}\,\psi, a{:}\,\varphi, \Delta}\,(\bot_{\mathcal{E}})}{\Gamma, x{:}\,(\phi\multimap\bot), y{:}\,(\psi\multimap\bot) \rhd y\;\mathsf{deunit}(x\;\mathsf{deunit}(\mathsf{unit}(M))){:}\,\bot, a{:}\,\varphi, \Delta}\,(\multimap_{\mathcal{E}})}{\Gamma, z{:}\,(\phi\multimap\bot)\otimes(\psi\multimap\bot) \rhd \mathsf{let}\;z\;\mathsf{be}\;x\otimes y\;\mathsf{in}\;y\;\mathsf{deunit}(x\;\mathsf{deunit}(\mathsf{unit}(M))){:}\,\bot, a{:}\,\varphi, \Delta}\,(\otimes_{\mathcal{E}})}{\Gamma \rhd \lambda z.\mathsf{let}\;z\;\mathsf{be}\;x\otimes y\;\mathsf{in}\;y\;\mathsf{deunit}(x\;\mathsf{deunit}(\mathsf{unit}(M))){:}\,\phi\parr\psi, a{:}\,\varphi, \Delta}\,(\multimap_{\mathcal{I}})
$$

- A proof of the form

$$
\dfrac{\begin{matrix}\pi_1\\ \vdots \\ \Gamma \vdash \phi, \Delta\end{matrix}}{\Gamma, \phi^{\bot} \vdash \Delta}\,({}^{\bot}\mathcal{L})
$$

is translated to

$$
\dfrac{\dfrac{\dfrac{}{x{:}\,\phi^{\bot} \rhd x{:}\,\phi^{\bot}} \qquad \begin{matrix}\mathcal{M}(\pi_1)\\ \vdots \\ \Gamma \rhd M{:}\,\phi, \Delta\end{matrix}}{\Gamma, x{:}\,\phi^{\bot} \rhd xM{:}\,\bot, \Delta}\,(\multimap_{\mathcal{E}})}{\Gamma, x{:}\,\phi^{\bot} \rhd \mathsf{deunit}(xM){:}\,\Delta}\,(\bot_{\mathcal{E}})
$$

- A proof of the form

$$\dfrac{\begin{array}{c}\pi_1\\ \vdots\\ \Gamma, \phi \vdash \Delta\end{array}}{\Gamma \vdash \phi^{\perp}, \Delta}\,(^{\perp}\mathcal{R})$$

is translated to

$$\dfrac{\dfrac{\begin{array}{c}\mathcal{M}(\pi_1)\\ \vdots\\ \Gamma, x{:}\,\phi \rhd M{:}\,\Delta\end{array}}{\Gamma, x{:}\,\phi \rhd \mathsf{unit}(M){:}\,\bot, \Delta}\,(\bot_{\mathcal{I}})}{\Gamma \rhd \lambda x{:}\,\phi.\mathsf{unit}(M){:}\,\phi^{\perp}, \Delta}\,(\multimap_{\mathcal{I}})$$

- A proof of the form

$$\dfrac{\begin{array}{c}\pi_1\\ \vdots\\ \Gamma, \phi \vdash \Delta\end{array}}{\Gamma, !\phi \vdash \Delta}\,\textit{Dereliction}_{\mathcal{L}}$$

is translated to

$$\dfrac{\dfrac{\dfrac{}{x{:}\,!\phi \rhd x{:}\,!\phi}}{x{:}\,!\phi \rhd \mathsf{derelict}(x){:}\,\phi}\,\textit{Derelict} \qquad \begin{array}{c}\mathcal{M}(\pi_1)\\ \vdots\\ \Gamma, y{:}\,\phi \rhd M{:}\,\Delta\end{array}}{\Gamma, x{:}\,!\phi \rhd M[y := \mathsf{derelict}(x)]{:}\,\Delta}\,\textit{Substitution}$$

- A proof of the form

$$\dfrac{\begin{array}{c}\pi_1\\ \vdots\\ \Gamma \vdash \phi, \Delta\end{array}}{\Gamma \vdash ?\phi, \Delta}\,\textit{Dereliction}_{\mathcal{R}}$$

is translated to

$$\dfrac{\dfrac{\dfrac{\dfrac{}{x{:}\,!(\phi\multimap\bot) \rhd x{:}\,!(\phi\multimap\bot)}}{x{:}\,!(\phi\multimap\bot) \rhd \mathsf{derelict}(x){:}\,\phi\multimap\bot}\,\textit{Dereliction} \qquad \begin{array}{c}\mathcal{M}(\pi_1)\\ \vdots\\ \Gamma \rhd M{:}\,\phi, \Delta\end{array}}{\Gamma, x{:}\,!(\phi\multimap\bot) \rhd (\mathsf{derelict}(x))M{:}\,\bot, \Delta}\,(\multimap_{\mathcal{E}})}{\Gamma \rhd \lambda x.(\mathsf{derelict}(x))M{:}\,?\phi, \Delta}\,(\multimap_{\mathcal{I}})$$

- A proof of the form

$$\dfrac{\begin{array}{c}\pi_1\\ \vdots\\ \Gamma \vdash \Delta\end{array}}{\Gamma, !\phi \vdash \Delta}\,\textit{Weakening}_{\mathcal{L}}$$

is translated to

$$\dfrac{\dfrac{}{x{:}\,!\phi \rhd x{:}\,!\phi} \qquad \begin{array}{c}\mathcal{M}(\pi_1)\\ \vdots\\ \Gamma \rhd M{:}\,\Delta\end{array}}{\Gamma, x{:}\,!\phi \rhd \mathsf{discard}\ x\ \mathsf{in}\ M{:}\,\Delta}\,\textit{Weakening}$$

- A proof of the form

$$\dfrac{\begin{matrix}\pi_1\\ \vdots\\ \Gamma \vdash \Delta\end{matrix}}{\Gamma \vdash ?\phi, \Delta}\ \textit{Weakening}_{\mathcal{R}}$$

is translated to

$$\dfrac{\dfrac{\dfrac{\dfrac{}{x{:}\,!(\phi\multimap\bot) \rhd x{:}\,!(\phi\multimap\bot)} \qquad \begin{matrix}\mathcal{M}(\pi_1)\\ \vdots\\ \Gamma \rhd M{:}\,\Delta\end{matrix}}{\Gamma, x{:}\,!(\phi\multimap\bot) \rhd \mathsf{discard}\ x\ \mathsf{in}\ M{:}\,\Delta}\ \textit{Weakening}}{\Gamma, x{:}\,!(\phi\multimap\bot) \rhd \mathsf{unit}(\mathsf{discard}\ x\ \mathsf{in}\ M){:}\,\bot, \Delta}\ (\bot_{\mathcal{I}})}{\Gamma \rhd \lambda x.\mathsf{unit}(\mathsf{discard}\ x\ \mathsf{in}\ M){:}\,?\phi, \Delta}\ (\multimap_{\mathcal{I}})$$

- A proof of the form

$$\dfrac{\begin{matrix}\pi_1\\ \vdots\\ \Gamma, !\phi, !\phi \vdash \Delta\end{matrix}}{\Gamma, !\phi \vdash \Delta}\ \textit{Contraction}_{\mathcal{L}}$$

is translated to

$$\dfrac{\dfrac{}{z{:}\,!\phi \rhd x{:}\,!\phi} \qquad \begin{matrix}\mathcal{M}(\pi_1)\\ \vdots\\ \Gamma, x{:}\,!\phi, y{:}\,!\phi \rhd M{:}\,\Delta\end{matrix}}{\Gamma, z{:}\,!\phi \rhd \mathsf{copy}\ z\ \mathsf{as}\ x, y\ \mathsf{in}\ M{:}\,\Delta}\ \textit{Contraction}$$

- A proof of the form

$$\dfrac{\begin{matrix}\pi_1\\ \vdots\\ \Gamma \vdash ?\phi, ?\phi, \Delta\end{matrix}}{\Gamma \vdash ?\phi, \Delta}\ \textit{Contraction}_{\mathcal{R}}$$

is translated to

$$\dfrac{\dfrac{\dfrac{}{z{:}\,!(\phi\multimap\bot) \rhd z{:}\,!(\phi\multimap\bot)} \qquad \dfrac{\dfrac{\dfrac{\begin{matrix}\mathcal{M}(\pi_1)\\ \vdots\\ \Gamma \rhd M{:}\,?\phi, c{:}\,?\phi, \Delta\end{matrix} \qquad \dfrac{}{x{:}\,!(\phi\multimap\bot) \rhd x{:}\,!(\phi\multimap\bot)}}{\Gamma, x{:}\,!(\phi\multimap\bot) \rhd Mx{:}\,\bot, c{:}\,?\phi, \Delta}\ (\multimap_{\mathcal{E}})}{\Gamma, x{:}\,!(\phi\multimap\bot) \rhd \mathsf{deunit}(Mx){:}\,?\phi, \Delta}\ (\bot_{\mathcal{E}}) \qquad \dfrac{}{y{:}\,!(\phi\multimap\bot) \rhd y{:}\,!(\phi\multimap\bot)}}{\Gamma, x{:}\,!(\phi\multimap\bot), y{:}\,!(\phi\multimap\bot) \rhd (\mathsf{deunit}(Mx))y{:}\,\bot, \Delta}\ (\multimap_{\mathcal{E}})}{\Gamma, z{:}\,!(\phi\multimap\bot) \rhd \mathsf{copy}\ z\ \mathsf{as}\ x, y\ \mathsf{in}\ (\mathsf{deunit}(Mx))y{:}\,\bot, \Delta}\ \textit{Contr.}}{\Gamma \rhd \lambda z.\mathsf{copy}\ z\ \mathsf{as}\ x, y\ \mathsf{in}\ (\mathsf{deunit}(Mx))y{:}\,?\phi, \Delta}\ (\multimap_{\mathcal{I}})$$

- A proof of the form

$$\dfrac{\begin{matrix}\pi_1\\ \vdots\\ !\Gamma, \phi \vdash ?\Delta\end{matrix}}{!\Gamma, ?\phi \vdash ?\Delta}\ \textit{Promotion}_{\mathcal{L}}$$

(where $?\Delta = ?\varphi, ?\Delta'$) is translated to

$$\dfrac{\overline{z{:}\,?\phi \rhd z{:}\,?\phi} \qquad \dfrac{\dfrac{\overline{\vec{x}{:}\,!\Gamma \rhd \vec{x}{:}\,!\Gamma} \quad \overline{\rhd \mathcal{P}_{\vec{a}}{:}\,!(?\Delta \multimap \bot), \vec{a}{:}\,?\Delta} \quad \dfrac{\dfrac{\begin{array}{c}\mathcal{M}(\pi_1)\\ \vdots \\ \vec{x}{:}\,!\Gamma, y{:}\,\phi \rhd M{:}\,?\varphi, ?\Delta'\end{array}}{\vec{x}{:}\,!\Gamma, y{:}\,\phi \rhd \mathsf{unit}(M){:}\,\bot, \vec{a}{:}\,?\Delta}\,(\bot_{\mathcal{I}})}{\vec{x}{:}\,!\Gamma \rhd \lambda y.\mathsf{unit}(M){:}\,\phi \multimap \bot, \vec{a}{:}\,?\Delta}\,(\multimap_{\mathcal{I}})}{\vec{x}{:}\,!\Gamma \rhd \mathsf{promote}\ \vec{x}|\mathcal{P}_{\vec{a}}\ \mathsf{for}\ \vec{x}|\vec{a}\ \mathsf{in}\ \lambda y.\mathsf{unit}(M){:}\,!(\phi \multimap \bot), \vec{a}{:}\,?\Delta}\,\mathit{Prom.}}{\dfrac{\vec{x}{:}\,!\Gamma, z{:}\,?\phi \rhd z(\mathsf{promote}\ \vec{x}|\mathcal{P}_{\vec{a}}\ \mathsf{for}\ \vec{x}|\vec{a}\ \mathsf{in}\ \lambda y.\mathsf{unit}(M)){:}\,\bot, \vec{a}{:}\,?\Delta}{\vec{x}{:}\,!\Gamma, z{:}\,?\phi \rhd \mathsf{deunit}(z(\mathsf{promote}\ \vec{x}|\mathcal{P}_{\vec{a}}\ \mathsf{for}\ \vec{x}|\vec{a}\ \mathsf{in}\ \lambda y.\mathsf{unit}(M))){:}\,?\varphi, ?\Delta'}\,(\bot_{\mathcal{E}})}\,(\multimap_{\mathcal{E}})$$

- A proof of the form

$$\dfrac{\begin{array}{c}\pi_1\\ \vdots \\ !\Gamma \vdash \phi, ?\Delta\end{array}}{!\Gamma \vdash !\phi, ?\Delta}\,\mathit{Promotion}_{\mathcal{R}}$$

is translated to

$$\dfrac{\overline{\vec{x}{:}\,!\Gamma \rhd \vec{x}{:}\,!\Gamma} \qquad \overline{\rhd \mathcal{P}_{\vec{a}}{:}\,!(?\Delta \multimap \bot), \vec{a}{:}\,?\Delta} \qquad \begin{array}{c}\mathcal{M}(\pi_1)\\ \vdots \\ \vec{x}{:}\,!\Gamma \rhd M{:}\,\phi, \vec{a}{:}\,?\Delta\end{array}}{\vec{x}{:}\,!\Gamma \rhd \mathsf{promote}\ \vec{x}|\mathcal{P}_{\vec{a}}\ \mathsf{for}\ \vec{x}|\vec{a}\ \mathsf{in}\ M{:}\,!\phi, \vec{a}{:}\,?\Delta}\,\mathit{Promotion}$$

One of the features of **CLL** is that the cut-elimination process is much better behaved than it is for **CL**. For example, trying to construct the critical pair of §2.4 flounders, *i.e.*

$$\dfrac{\dfrac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta, ?\phi}\,\mathit{Weakening}_{\mathcal{R}} \qquad \dfrac{\Gamma' \vdash \Delta'}{\Gamma', !\phi \vdash \Delta'}\,\mathit{Weakening}_{\mathcal{L}}}{\Gamma, \Gamma' \vdash \Delta, \Delta'}\,\mathit{?Cut?}$$

where the instance of *Cut* is not even valid! Hence the problematic critical pairs from **CL** are removed by moving to the linear framework with its more refined connectives.

It is now possible to reconsider the (better behaved) process of cut-elimination for **CLL**, by translating the steps across to $\mathbf{CLL}_\mu$. I shall demonstrate this by considering four instances of principal cuts.

- $(⅋_{\mathcal{R}}, ⅋_{\mathcal{L}})$-cut.

$$\dfrac{\dfrac{\Gamma \vdash \phi, \psi, \Delta}{\Gamma \vdash \phi ⅋ \psi, \Delta}\,(⅋_{\mathcal{R}}) \qquad \dfrac{\Gamma', \phi \vdash \Delta' \quad \Gamma'', \psi \vdash \Delta''}{\Gamma', \Gamma'', \phi ⅋ \psi \vdash \Delta', \Delta''}\,(⅋_{\mathcal{L}})}{\Gamma, \Gamma', \Gamma'' \vdash \Delta, \Delta', \Delta''}\,\mathit{Cut}$$

$$\rightsquigarrow_{cut} \quad \dfrac{\dfrac{\Gamma \vdash \phi, \psi, \Delta \quad \Gamma', \phi \vdash \Delta'}{\Gamma, \Gamma' \vdash \Delta, \Delta', \psi}\,\mathit{Cut} \qquad \Gamma'', \psi \vdash \Delta''}{\Gamma, \Gamma', \Gamma'' \vdash \Delta, \Delta', \Delta''}\,\mathit{Cut}$$

The former deduction is translated to

$$\begin{array}{ll} & (N[y := z(\lambda x.\mathsf{unit}(M))])[z := \lambda u.\mathsf{deunit}(uP)] \\ \equiv & N[y := (\lambda u.\mathsf{deunit}(uP))(\lambda x.\mathsf{unit}(M))] \\ \leadsto_\beta & N[y := \mathsf{deunit}((\lambda x.\mathsf{unit}(M))P)] \\ \leadsto_\beta & N[y := \mathsf{deunit}(\mathsf{unit}(M[x := P]))] \end{array}$$

which is the translation of the latter.

- $(\mathit{Promotion}_\mathcal{R}, \mathit{Dereliction}_\mathcal{L})$-cut

$$\dfrac{\dfrac{!\Gamma \vdash \phi, ?\Delta}{!\Gamma \vdash !\phi, ?\Delta}\ \mathit{Promotion}_\mathcal{R} \qquad \dfrac{\Gamma', \phi \vdash \Delta'}{\Gamma', !\phi \vdash \Delta'}\ \mathit{Dereliction}_\mathcal{L}}{!\Gamma, \Gamma' \vdash ?\Delta, \Delta'}\ \mathit{Cut}$$

$$\leadsto_{cut} \dfrac{!\Gamma \vdash \phi, ?\Delta \qquad \Gamma', \phi \vdash \Delta'}{!\Gamma, \Gamma' \vdash ?\Delta, \Delta'}\ \mathit{Cut}$$

The former deduction is translated to

$$\begin{array}{ll} & N[z := \mathsf{derelict}(w)][w := \mathsf{promote}\ \vec{y}|\mathcal{P}_{\vec{a}}\ \mathsf{for}\ \vec{y}|\vec{a}\ \mathsf{in}\ M] \\ \equiv & N[z := \mathsf{derelict}(\mathsf{promote}\ \vec{y}|\mathcal{P}_{\vec{a}}\ \mathsf{for}\ \vec{y}|\vec{a}\ \mathsf{in}\ M)] \\ \leadsto_\beta & N[z := M[\vec{y} := \vec{y}, \mathsf{unit}(R) \Leftarrow \mathsf{derelict}(\mathcal{P}_{\vec{a}})R]] \\ \leadsto_{\beta,c,\eta} & N[z := M] \end{array}$$

which is the translation of the latter. (The last step holds by Lemma 1.)

- $(\mathit{Promotion}_\mathcal{R}, \mathit{Weakening}_\mathcal{L})$-cut.

$$\dfrac{\dfrac{!\Gamma \vdash \phi, ?\Delta}{!\Gamma \vdash !\phi, ?\Delta}\ \mathit{Promotion}_\mathcal{R} \qquad \dfrac{\Gamma' \vdash \Delta'}{\Gamma', \phi \vdash \Delta'}\ \mathit{Weakening}_\mathcal{L}}{!\Gamma, \Gamma' \vdash ?\Delta, \Delta'}\ \mathit{Cut}$$

$$\leadsto_{cut} \dfrac{\dfrac{\Gamma' \vdash \Delta'}{\Gamma', !\Gamma \vdash \Delta'}\ \mathit{Weakening}^*_\mathcal{L}}{\Gamma', !\Gamma \vdash \Delta', ?\Delta}\ \mathit{Weakening}^*_\mathcal{R}$$

The former deduction is translated to

$$\begin{array}{ll} & \mathsf{discard}\ (\mathsf{promote}\ \vec{x}|\mathcal{P}_{\vec{a}}\ \mathsf{for}\ \vec{x}|\vec{a}\ \mathsf{in}\ M)\ \mathsf{in}\ N \\ \leadsto^*_\beta & \mathsf{discard}\ \mathcal{P}_{\vec{a}}\ \mathsf{in}\ \mathsf{discard}\ \vec{x}\ \mathsf{in}\ N \end{array}$$

which is the translation of the latter.

- $({}^\perp{}_\mathcal{R}, {}^\perp{}_\mathcal{L})$-cut.

$$\dfrac{\dfrac{\Gamma, \phi \vdash \Delta}{\Gamma \vdash \Delta, \phi^\perp}\ ({}^\perp{}_\mathcal{R}) \qquad \dfrac{\Gamma' \vdash \phi, \Delta'}{\Gamma', \phi^\perp \vdash \Delta'}\ ({}^\perp{}_\mathcal{L})}{\Gamma, \Gamma' \vdash \Delta, \Delta'}\ \mathit{Cut}$$

$$\leadsto_{cut} \dfrac{\Gamma' \vdash \phi, \Delta' \qquad \Gamma, \phi \vdash \Delta}{\Gamma, \Gamma' \vdash \Delta, \Delta'}\ \mathit{Cut}$$

The former deduction is translated to

$$\begin{array}{rl} & \mathsf{deunit}^{\varphi}_{b}(zN)[z := \lambda x.\mathsf{unit}^{\phi}_{a}(M)] \\ \equiv & \mathsf{deunit}^{\varphi}_{b}((\lambda x.\mathsf{unit}^{\phi}_{a}(M))N) \\ \rightsquigarrow_{\beta} & \mathsf{deunit}^{\varphi}_{b}(\mathsf{unit}^{\phi}_{a}(M[x := N])) \end{array}$$

which is the translation of the latter.

Thus we can show that term reduction captures the steps of cut-elimination in the following sense.

Theorem 6. If $\Gamma \overset{\pi_1}{\vdash} \Delta \rightsquigarrow_{cut} \Gamma \overset{\pi_2}{\vdash} \Delta$ then $\mathcal{M}(\pi_1) \rightsquigarrow^{*}_{\beta,c,\eta} \mathcal{M}(\pi_2)$.

6 TRANSLATIONS

Just as there are translations from **IL** into **ILL** [8, Chapter 2, §5], there are also translations from **CL** into **CLL**. These have been studied by Schellinx [33] in his thesis. As he points out they are (necessarily) quite complicated, requiring a large number of exponentials. Interestingly there is no unique 'optimal' solution as is the case for **IL**. Rather there are two candidates. The T-translation which is based on a linear decomposition of $\phi \supset \psi$ as $!?\phi \multimap ?\psi$; and the Q-translation which interprets $\phi \supset \psi$ as $!\phi \multimap ?!\psi$. They are defined as follows.

$$\begin{array}{rcll} p^{\mathsf{Q}} & \stackrel{\text{def}}{=} & p & (p \text{ atomic}) \\ \phi \supset \psi^{\mathsf{Q}} & \stackrel{\text{def}}{=} & !\phi^{\mathsf{Q}} \multimap ?!\phi^{\mathsf{Q}} & \\ \phi \wedge \psi^{\mathsf{Q}} & \stackrel{\text{def}}{=} & !\phi^{\mathsf{Q}} \otimes !\psi^{\mathsf{Q}} & \\ \phi \vee \psi^{\mathsf{Q}} & \stackrel{\text{def}}{=} & ?!\phi^{\mathsf{Q}} \parr ?!\psi^{\mathsf{Q}}; & \end{array}$$

$$\begin{array}{rcll} p^{\mathsf{T}} & \stackrel{\text{def}}{=} & p & (p \text{ atomic}) \\ \phi \supset \psi^{\mathsf{T}} & \stackrel{\text{def}}{=} & !?\phi^{\mathsf{T}} \multimap ?\psi^{\mathsf{T}} & \\ \phi \wedge \psi^{\mathsf{T}} & \stackrel{\text{def}}{=} & !?\phi^{\mathsf{T}} \otimes !?\psi^{\mathsf{T}} & \\ \phi \vee \psi^{\mathsf{T}} & \stackrel{\text{def}}{=} & ?\phi^{\mathsf{T}} \parr ?\psi^{\mathsf{T}}. & \end{array}$$

Theorem 7. $\vdash_{\mathbf{CLL}} !\Gamma^{\mathsf{Q}} \vdash ?!\Delta^{\mathsf{Q}}$ iff $\vdash_{\mathbf{CL}} \Gamma \vdash \Delta$ iff $\vdash_{\mathbf{CLL}} !?\Gamma^{\mathsf{T}} \vdash ?\Delta^{\mathsf{T}}$.

These equivalences can be presented for $\mathbf{CL}_{\mu}$ and $\mathbf{CLL}_{\mu}$. Rather than give all the details I shall show how the implication rules for $\mathbf{CL}_{\mu}$ are translated using both the Q and T strategies.

The implication introduction rule

$$\frac{\Gamma, x{:}\,\phi \rhd M{:}\,\psi, \vec{a}{:}\,\Delta}{\Gamma \rhd \lambda x.M{:}\,\phi \supset \psi, \vec{a}{:}\,\Delta}\,(\supset_{\mathcal{I}})$$

is T-translated to

$$\dfrac{\dfrac{\dfrac{}{z{:}\,!((!?\phi^{\mathsf{T}} \multimap ?\psi^{\mathsf{T}}) \multimap \bot) \rhd z{:}\,!((!?\phi^{\mathsf{T}} \multimap ?\psi^{\mathsf{T}}) \multimap \bot)}}{z{:}\,!((!?\phi^{\mathsf{T}} \multimap ?\psi^{\mathsf{T}}) \multimap \bot) \rhd \mathsf{derelict}(z){:}\,(!?\phi^{\mathsf{T}} \multimap ?\psi^{\mathsf{T}}) \multimap \bot} \quad \dfrac{!?\Gamma^{\mathsf{T}}, x{:}\,!?\phi^{\mathsf{T}} \rhd M^{\mathsf{T}}{:}\,?\psi^{\mathsf{T}}, \vec{a}{:}\,?\Delta^{\mathsf{T}}}{!?\Gamma^{\mathsf{T}} \rhd \lambda x.M^{\mathsf{T}}{:}\,!?\phi^{\mathsf{T}} \multimap ?\psi^{\mathsf{T}}, \vec{a}{:}\,?\Delta^{\mathsf{T}}}}{\dfrac{!?\Gamma^{\mathsf{T}}, z{:}\,!((!?\phi^{\mathsf{T}} \multimap ?\psi^{\mathsf{T}}) \multimap \bot) \rhd \mathsf{derelict}(z)(\lambda x.M^{\mathsf{T}}){:}\,\bot, \vec{a}{:}\,?\Delta}{!?\Gamma^{\mathsf{T}} \rhd \lambda z.\mathsf{derelict}(z)(\lambda x.M^{\mathsf{T}}){:}\,?(!?\phi^{\mathsf{T}} \multimap ?\psi^{\mathsf{T}}), \vec{a}{:}\,?\Delta^{\mathsf{T}}}}$$

The implication elimination rule

$$\frac{\Gamma \rhd M\colon \phi \supset \psi, \Delta \qquad \Gamma' \rhd N\colon \phi, \Delta'}{\Gamma, \Gamma' \rhd MN\colon \psi, \Delta, \Delta'}\ (\supset\mathcal{E})$$

is T-translated to

$$\dfrac{\dfrac{!?\Gamma^{\mathsf{T}} \rhd M^{\mathsf{T}}\colon ?(!?\phi^{\mathsf{T}} \multimap ?\psi^{\mathsf{T}}), ?\Delta^{\mathsf{T}} \qquad \vec{y}\colon !?\Gamma'^{\mathsf{T}} \rhd A\colon !((!?\phi^{\mathsf{T}} \multimap ?\psi^{\mathsf{T}}) \multimap \bot), \vec{b}\colon ?\Delta'^{\mathsf{T}}, c\colon ?\psi^{\mathsf{T}}}{!?\Gamma^{\mathsf{T}}, !?\Gamma'^{\mathsf{T}} \rhd M^{\mathsf{T}}A\colon \bot, ?\psi^{\mathsf{T}}, ?\Delta^{\mathsf{T}}, ?\Delta'^{\mathsf{T}}}}{!?\Gamma^{\mathsf{T}}, !?\Gamma'^{\mathsf{T}} \rhd \mathsf{deunit}(M^{\mathsf{T}}A)\colon ?\psi^{\mathsf{T}}, ?\Delta^{\mathsf{T}}, ?\Delta'^{\mathsf{T}}}$$

where A is given by the deduction

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{z\colon !?\phi^{\mathsf{T}} \multimap ?\psi^{\mathsf{T}} \rhd z\colon !?\phi^{\mathsf{T}} \multimap ?\psi^{\mathsf{T}}} \qquad \dfrac{\vec{y}\colon !?\Gamma'^{\mathsf{T}} \rhd N^{\mathsf{T}}\colon ?\phi^{\mathsf{T}}, \vec{b}\colon ?\Delta'^{\mathsf{T}}}{\vec{y}\colon !?\Gamma'^{\mathsf{T}} \rhd \mathsf{promote}\ \vec{y}|\mathcal{P}_{\vec{b}}\ \mathsf{for}\ \vec{y}|\vec{b}\ \mathsf{in}\ N^{\mathsf{T}}\colon !?\phi^{\mathsf{T}}, \vec{b}\colon ?\Delta'^{\mathsf{T}}}}{\vec{y}\colon !?\Gamma'^{\mathsf{T}}, z\colon !?\phi^{\mathsf{T}} \multimap ?\psi^{\mathsf{T}} \rhd z(\mathsf{promote}\ \vec{y}|\mathcal{P}_{\vec{b}}\ \mathsf{for}\ \vec{y}|\vec{b}\ \mathsf{in}\ N^{\mathsf{T}})\colon ?\psi^{\mathsf{T}}, \vec{b}\colon ?\Delta'^{\mathsf{T}}}}{\vec{y}\colon !?\Gamma'^{\mathsf{T}}, z\colon !?\phi^{\mathsf{T}} \multimap ?\psi^{\mathsf{T}} \rhd \mathsf{unit}(z(\mathsf{promote}\ \vec{y}|\mathcal{P}_{\vec{b}}\ \mathsf{for}\ \vec{y}|\vec{b}\ \mathsf{in}\ N^{\mathsf{T}}))\colon \bot, \vec{b}\colon ?\Delta'^{\mathsf{T}}, c\colon ?\psi^{\mathsf{T}}}}{\vec{y}\colon !?\Gamma'^{\mathsf{T}} \rhd \lambda z.\mathsf{unit}(z(\mathsf{promote}\ \vec{y}|\mathcal{P}_{\vec{b}}\ \mathsf{for}\ \vec{y}|\vec{b}\ \mathsf{in}\ N^{\mathsf{T}}))\colon (!?\phi^{\mathsf{T}} \multimap ?\psi^{\mathsf{T}}) \multimap \bot, \vec{b}\colon ?\Delta'^{\mathsf{T}}, c\colon ?\psi^{\mathsf{T}}}}{\vec{y}\colon !?\Gamma'^{\mathsf{T}} \rhd \mathsf{promote}\ \vec{y}|\mathcal{P}_{\vec{b},c}\ \mathsf{for}\ \vec{y}|\vec{b}, c\ \mathsf{in}\ \lambda z.\mathsf{unit}(z(\mathsf{promote}\ \vec{y}|\mathcal{P}_{\vec{b}}\ \mathsf{for}\ \vec{y}|\vec{b}\ \mathsf{in}\ N^{\mathsf{T}}))\colon !((!?\phi^{\mathsf{T}} \multimap ?\psi^{\mathsf{T}}) \multimap \bot), \vec{b}\colon ?\Delta'^{\mathsf{T}}, c\colon ?\psi^{\mathsf{T}}}$$

The implication introduction rule

$$\frac{\vec{y}\colon \Gamma, x\colon \phi \rhd M\colon \psi, \vec{a}\colon \Delta}{\vec{y}\colon \Gamma \rhd \lambda x.M\colon \phi \supset \psi, \vec{a}\colon \Delta}\ (\supset\mathcal{I})$$

is Q-translated to

$$\dfrac{\dfrac{\dfrac{\dfrac{}{z\colon !(!(!\phi^{\mathsf{Q}} \multimap ?!\psi^{\mathsf{Q}}) \multimap \bot) \rhd z\colon !(!(!\phi^{\mathsf{Q}} \multimap ?!\psi^{\mathsf{Q}}) \multimap \bot)}}{z\colon !(!(!\phi^{\mathsf{Q}} \multimap ?!\psi^{\mathsf{Q}}) \multimap \bot) \rhd \mathsf{derelict}(z)\colon !(!\phi^{\mathsf{Q}} \multimap ?!\psi^{\mathsf{Q}}) \multimap \bot} \qquad \dfrac{\dfrac{\vec{y}\colon !\Gamma^{\mathsf{Q}}, x\colon !\phi^{\mathsf{Q}} \rhd M^{\mathsf{Q}}\colon ?!\psi^{\mathsf{Q}}, \vec{a}\colon ?!\Delta^{\mathsf{Q}}}{\vec{y}\colon !\Gamma^{\mathsf{Q}} \rhd \lambda x.M^{\mathsf{Q}}\colon !\phi^{\mathsf{Q}} \multimap ?!\psi^{\mathsf{Q}}, \vec{a}\colon ?!\Delta^{\mathsf{Q}}}}{\vec{y}\colon !\Gamma^{\mathsf{Q}} \rhd \mathsf{promote}\ \vec{y}|\mathcal{P}_{\vec{a}}\ \mathsf{for}\ \vec{y}|\vec{a}\ \mathsf{in}\ \lambda x.M^{\mathsf{Q}}\colon !(!\phi^{\mathsf{Q}} \multimap ?!\psi^{\mathsf{Q}}), \vec{a}\colon ?!\Delta^{\mathsf{Q}}}}{!\Gamma^{\mathsf{Q}}, z\colon !(!(!\phi^{\mathsf{Q}} \multimap ?!\psi^{\mathsf{Q}}) \multimap \bot) \rhd \mathsf{derelict}(z)(\mathsf{promote}\ \vec{y}|\mathcal{P}_{\vec{a}}\ \mathsf{for}\ \vec{y}|\vec{a}\ \mathsf{in}\ \lambda x.M^{\mathsf{Q}})\colon \bot, \vec{a}\colon ?!\Delta}}{!\Gamma^{\mathsf{Q}} \rhd \lambda z.\mathsf{derelict}(z)(\mathsf{promote}\ \vec{y}|\mathcal{P}_{\vec{a}}\ \mathsf{for}\ \vec{y}|\vec{a}\ \mathsf{in}\ \lambda x.M^{\mathsf{Q}})\colon ?!(!\phi^{\mathsf{Q}} \multimap ?!\psi^{\mathsf{Q}}), \vec{a}\colon ?!\Delta^{\mathsf{Q}}}$$

Finally, the implication elimination rule

$$\frac{\Gamma \rhd M\colon \phi \supset \psi, \Delta \qquad \Gamma' \rhd N\colon \phi, \Delta'}{\Gamma, \Gamma' \rhd MN\colon \psi, \Delta, \Delta'}\ (\supset\mathcal{E})$$

is Q-translated to

$$\dfrac{\dfrac{!\Gamma^{\mathsf{Q}} \rhd M\colon ?!(!\phi^{\mathsf{Q}} \multimap ?!\psi^{\mathsf{Q}}), ?!\Delta^{\mathsf{Q}} \qquad !\Gamma'^{\mathsf{Q}} \rhd A\colon !(!(!\phi^{\mathsf{Q}} \multimap ?!\psi^{\mathsf{Q}}) \multimap \bot), a\colon ?!\psi^{\mathsf{Q}}, ?!\Delta'^{\mathsf{Q}}}{!\Gamma^{\mathsf{Q}}, !\Gamma'^{\mathsf{Q}} \rhd M^{\mathsf{Q}}A\colon \bot, a\colon ?!\psi^{\mathsf{Q}}, ?!\Delta^{\mathsf{Q}}, ?!\Delta'^{\mathsf{Q}}}}{!\Gamma^{\mathsf{Q}}, !\Gamma'^{\mathsf{Q}} \rhd \mathsf{deunit}(M^{\mathsf{Q}}A)\colon ?!\psi^{\mathsf{Q}}, ?!\Delta^{\mathsf{Q}}, ?!\Delta'^{\mathsf{Q}}}$$

where A is given by the deduction

$$
\dfrac{\dfrac{\dfrac{!\Gamma'^{\mathsf{Q}} \rhd N^{\mathsf{Q}}\colon ?!\phi^{\mathsf{Q}}, ?!\Delta'^{\mathsf{Q}} \qquad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{x\colon !(!\phi^{\mathsf{Q}} \multimap ?!\psi^{\mathsf{Q}}) \rhd x\colon !(!\phi^{\mathsf{Q}} \multimap ?!\psi^{\mathsf{Q}})}}{x\colon !(!\phi^{\mathsf{Q}} \multimap ?!\psi^{\mathsf{Q}}) \rhd \mathsf{derelict}(x)\colon !\phi^{\mathsf{Q}} \multimap ?!\psi^{\mathsf{Q}}} \quad \dfrac{}{y\colon !\phi^{\mathsf{Q}} \rhd y\colon !\phi^{\mathsf{Q}}}}{x\colon !(!\phi^{\mathsf{Q}} \multimap ?!\psi^{\mathsf{Q}}), y\colon \phi^{\mathsf{Q}} \rhd \mathsf{derelict}(x)y\colon ?!\psi^{\mathsf{Q}}}}{x\colon !(!\phi^{\mathsf{Q}} \multimap ?!\psi^{\mathsf{Q}}), y\colon \phi^{\mathsf{Q}} \rhd \mathsf{unit}(\mathsf{derelict}(x)y)\colon \bot, a\colon ?!\psi^{\mathsf{Q}}}}{x\colon !(!\phi^{\mathsf{Q}} \multimap ?!\psi^{\mathsf{Q}}) \rhd \lambda y.\mathsf{unit}(\mathsf{derelict}(x)y)\colon !\phi^{\mathsf{Q}} \multimap \bot, a\colon ?!\psi^{\mathsf{Q}}}}{x\colon !(!\phi^{\mathsf{Q}} \multimap ?!\psi^{\mathsf{Q}}) \rhd \mathsf{promote}\ x|\mathcal{P}_a\ \mathsf{for}\ x|a\ \mathsf{in}\ \lambda y.\mathsf{unit}(\mathsf{derelict}(x)y)\colon !(!\phi^{\mathsf{Q}} \multimap \bot), a\colon ?!\psi^{\mathsf{Q}}}}{!\Gamma'^{\mathsf{Q}}, x\colon !(!\phi^{\mathsf{Q}} \multimap ?!\psi^{\mathsf{Q}}) \rhd N^{\mathsf{Q}}\ \mathsf{promote}\ x|\mathcal{P}_a\ \mathsf{for}\ x|a\ \mathsf{in}\ \lambda y.\mathsf{unit}(\mathsf{derelict}(x)y)\colon \bot, a\colon ?!\psi^{\mathsf{Q}}, ?!\Delta'^{\mathsf{Q}}}}{!\Gamma'^{\mathsf{Q}} \rhd \lambda x.N^{\mathsf{Q}}\ \mathsf{promote}\ x|\mathcal{P}_a\ \mathsf{for}\ x|a\ \mathsf{in}\ \lambda y.\mathsf{unit}(\mathsf{derelict}(x)y)\colon !(!\phi^{\mathsf{Q}} \multimap ?!\psi^{\mathsf{Q}}) \multimap \bot, a\colon ?!\psi^{\mathsf{Q}}, ?!\Delta'^{\mathsf{Q}}}}{!\Gamma'^{\mathsf{Q}} \rhd \mathsf{promote}\ \vec{z}|\mathcal{P}_{\vec{b},a}\ \mathsf{for}\ \vec{z}|\vec{b}, a\ \mathsf{in}\ \lambda x.N^{\mathsf{Q}}\ \mathsf{promote}\ x|\mathcal{P}_a\ \mathsf{for}\ x|a\ \mathsf{in}\ \lambda y.\mathsf{unit}(\mathsf{derelict}(x)y)\colon !(!(!\phi^{\mathsf{Q}} \multimap ?!\psi^{\mathsf{Q}}) \multimap \bot), a\colon ?!\psi^{\mathsf{Q}}, ?!\Delta'^{\mathsf{Q}}}
$$

Filling in all the details gives the following theorem.

Theorem 8. $\vdash_{\mathbf{CLL}_\mu} !\Gamma^{\mathsf{Q}} \vdash ?!\Delta^{\mathsf{Q}}$ iff $\vdash_{\mathbf{CL}_\mu} \Gamma \vdash \Delta$ iff $\vdash_{\mathbf{CLL}_\mu} !?\Gamma^{\mathsf{T}} \vdash ?\Delta^{\mathsf{T}}$.

In fact these translations preserve reductions as well, although I shall not give any details here. Unlike the case for the various translations of **IL** into **ILL** [7], it is quite hard to determine computational interpretations of these two translation strategies.

7 Towards a Programming Language

In the formulations of $\mathbf{CL}_\mu$ and $\mathbf{CLL}_\mu$, I have included all of the connectives separately. Of course the formulae equivalences of both logics mean that, in fact, we could trim this down. For $\mathbf{CL}_\mu$ both Parigot [30] and Ong [29] restrict their attention to a fragment with just the implication connective (and the *Activate* and *Passify* rules). For $\mathbf{CLL}_\mu$ the most obvious fragment includes just the linear implication ($\multimap$) and the exponential (!). As an illustration of how this might work, I shall show how the tensor ($\otimes$) and its unit (I) can be simulated with just these connectives. Term formation for these connectives is then defined as

$$
\begin{aligned}
M \otimes N &\stackrel{\mathrm{def}}{=} \lambda x.(xM)N, \\
\mathsf{let}\ M\ \mathsf{be}\ x \otimes y\ \mathsf{in}\ N &\stackrel{\mathrm{def}}{=} \mathsf{deunit}^{\varphi}_{a}(M(\lambda x.\lambda y.\mathsf{unit}^{\varphi}_{a}(N))), \\
* &\stackrel{\mathrm{def}}{=} \lambda x\colon \bot\ .x, \\
\mathsf{let}\ M\ \mathsf{be}\ *\ \mathsf{in}\ N &\stackrel{\mathrm{def}}{=} \mathsf{deunit}^{\varphi}_{a}(M(\mathsf{unit}^{\varphi}_{a}(N))).
\end{aligned}
$$

The β-rules are preserved by this translation, *viz.*

$$
\begin{aligned}
\mathsf{let}\ M \otimes N\ \mathsf{be}\ x \otimes y\ \mathsf{in}\ P &\stackrel{\mathrm{def}}{=} \mathsf{deunit}^{\varphi}_{a}((\lambda z.(zM)N)(\lambda x.\lambda y.\mathsf{unit}^{\varphi}_{a}(P))) \\
&\leadsto_\beta \mathsf{deunit}^{\varphi}_{a}((\lambda x.\lambda y.\mathsf{unit}^{\varphi}_{a}(P))M\ N) \\
&\leadsto^*_\beta \mathsf{deunit}^{\varphi}_{a}(\mathsf{unit}^{\varphi}_{a}(P[x := M, y := N])) \\
&\leadsto_\beta P[x := M, y := N]; \\[1em]
\mathsf{let}\ *\ \mathsf{be}\ *\ \mathsf{in}\ M &\stackrel{\mathrm{def}}{=} \mathsf{deunit}^{\varphi}_{a}((\lambda x.x)\mathsf{unit}^{\varphi}_{a}(M)) \\
&\leadsto_\beta \mathsf{deunit}^{\varphi}_{a}(\mathsf{unit}^{\varphi}_{a}(M)) \\
&\leadsto_\beta M.
\end{aligned}
$$

If either $\mathbf{CL}_\mu$ or $\mathbf{CLL}_\mu$ were to be made into a programming language, a design decision would have to be made as to which connectives were built-in and which ones were defined.

Although experience might tell otherwise, it would seem likely that any programming language would be as verbose as possible, whereas an intermediate language might well profit for having only a few connectives.

8 Conclusions and Future Work

In this paper I have demonstrated how Parigot's techniques can be applied to the linear case to yield a classical linear λ-calculus. In addition I hope to have at least shed some new light on the relationship between Parigot's work and more traditional treatments of classical logic in natural deduction. I would claim that the linear $\lambda\mu$-calculus, considered as a programming language, is of more use than one based on proof nets. As mentioned earlier, proof nets rely on equivalent types being considered equal—this would present an unusual programming paradigm where, for example, the type inference mechanism would have to be adapted to factor all types by the various equivalences. In the linear $\lambda\mu$-calculusthere are explicit coercion terms.

Others have proposed natural deduction formulations of **CLL**. Troelstra [35] presents linear versions of Gentzen's original proposals. Martini and Masini [28] present a different formulation with the motivation of having the par connective as fundamental and not, as it is in this paper, derived. Albrecht *et al.* [3] give yet another formulation which is very compact and appears to be closely related to a proof net formulation (in particular, the formulae equivalences are essential and implicit).

In particular I would promote the computational interpretation suggested in Appendix A, for both the linear and non-linear calculus. It provides (after sugaring) a programming language with catch and throw-like control operators; but one which has a correspondence with a proof theory. This alone makes it worthy of further study. Other work on relationships between classical logic and control operators for functional languages tends to be in the other direction, *viz.* using the control operators to understand classical logic (e.g. [4]).

A semantic study would also be desirable. Ong [29] has proposed a categorical semantics and a class of game-theoretic models for $\mathbf{CL}_\mu$. It would be interesting to see if a similar extension of linear categories [9] would produce some sort of $\star$-autonomous category [5]. I should also like to investigate to what extent this work can be adapted to give a natural deduction formulation of classical **S4**, in the same way that work on **ILL** can be adapted for *intuitionistic* **S4** [11].

Acknowledgements

I am grateful to Martin Hyland for many interesting discussions and for his suggestion to investigate a linear version of Parigot's work (his suggestion of a π-calculus interpretation still remains work in progress). I should also like to take this opportunity to thank Nick Benton for his friendship and critical advice. I am grateful to Luke Ong for many conversations about this work and for his patient explanation of his work and its relationship to mine. Nick Benton, Luke Ong and Valeria de Paiva all offered useful criticisms of various drafts of this paper.

A preliminary version of this paper appears in the Proceedings of the Tokyo Conference on Linear Logic. I am grateful to Mitsu Okada for his kind invitation.

References

[1] S. Abramsky. Computational interpretations of linear logic. *Theoretical Computer Science*, 111(1–2):3–57, 1993. Previously Available as Department of Computing, Imperial College Technical Report 90/20, 1990.

[2] S. Abramsky, R. Jagadeesan, and P. Malacaria. Full abstraction for PCF (Extended Abstract). In *Proceedings of Conference on Theoretical Aspects of Computer Software*, volume 789 of *Lecture Notes in Computer Science*, pages 1–15, 1994.

[3] D. Albrecht, J.N. Crossley, and J.S. Jeavons. New Curry-Howard terms for full linear logic. Unpublished Manuscript, May 1996.

[4] F. Barbanera and S. Berardi. A strong normalisation result for classical logic. *Annals of Pure and Applied Logic*, 76:99–116, 1995.

[5] M. Barr. $\star$-autonomous categories and linear logic. *Mathematical Structures in Computer Science*, 1:159–178, 1991.

[6] P.N. Benton. A mixed linear and non-linear logic: Proofs, terms and models. Technical Report 352, Computer Laboratory, University of Cambridge, 1994.

[7] P.N. Benton and P. Wadler. Linear logic, monads and the lambda calculus. To appear in LICS, 1996.

[8] G.M. Bierman. *On Intuitionistic Linear Logic*. PhD thesis, Computer Laboratory, University of Cambridge, December 1993. Published as Computer Laboratory Technical Report 346, August 1994.

[9] G.M. Bierman. What is a categorical model of intuitionistic linear logic? In *Proceedings of Second International Conference on Typed λ-calculi and applications*, volume 902 of *Lecture Notes in Computer Science*, pages 78–93, April 1995. Previously available as Technical Report 333, University of Cambridge Computer Laboratory, March 1994.

[10] G.M. Bierman. A note on full intuitionistic linear logic. To appear in Annals of Pure and Applied Logic, 1996.

[11] G.M. Bierman and V.C.V. de Paiva. Intuitionistic necessity revisited. Technical Report CSR–96–10, School of Computer Science, University of Birmingham, June 1996.

[12] A. Blass. A game semantics for linear logic. *Annals of Pure and Applied Logic*, 56:183–220, 1992.

[13] Th. Coquand. A semantics of evidence of classical arithmetic. *Journal of Symbolic Logic*, 60:325–337, 1995.

[14] O. Danvy and A. Filinski. A functional abstraction of typed contexts. Technical Report 89/12, DIKU, University of Copenhagen, July 1989.

[15] V.C.V. de Paiva and L.C. Pereira. A new proof system for intuitionistic logic. Paper presented at the Logic Colloquium 93, Keele, England. Abstract appears in Bulletin of Symbolic Logic, 1(1):101–102, April 1993.

[16] B.F. DUBA, R. HARPER, AND D. MACQUEEN. Typing first-class continuations in ML. In *Proceedings of Symposium on Principles of Programming Languages*, pages 163–173, January 1991.

[17] M. FELLEISEN. The theory and practice of first-class prompts. In *Proceedings of Symposium on Principles of Programming Languages*, pages 180–190, January 1988.

[18] A. FILINSKI. Linear continuations. In *Proceedings of Symposium on Principles of Programming Languages*, pages 27–38, January 1992.

[19] J. GALLIER. Constructive logics part I: A tutorial on proof systems and typed λ-calculi. *Theoretical Computer Science*, 110(2):249–339, March 1993.

[20] G. GENTZEN. Investigations into logical deduction. In M.E. Szabo, editor, *The Collected Papers of Gerhard Gentzen*, pages 68–131. North-Holland, 1969. English Translation of 1935 German original.

[21] J.-Y. GIRARD. Linear logic. *Theoretical Computer Science*, 50:1–101, 1987.

[22] J.-Y. GIRARD, Y. LAFONT, AND P. TAYLOR. *Proofs and Types*, volume 7 of *Cambridge Tracts in Theoretical Computer Science.* Cambridge University Press, 1989.

[23] T.G. GRIFFIN. A formulae-as-types notion of control. In *Proceedings of Symposium on Principles of Programming Languages*, pages 47–58, 1990.

[24] W.A. HOWARD. The formulae-as-types notion of construction. In J.R. Hindley and J.P. Seldin, editors, *To H.B. Curry: Essays on combinatory logic, lambda calculus and formalism.* Academic Press, 1980.

[25] J.M.E. HYLAND AND V.C.V. DE PAIVA. Full intuitionistic linear logic. *Annals of Pure and Applied Logic*, 64:273–291, 1993.

[26] J.M.E. HYLAND AND C.-H.L. ONG. Pi-calculus, dialogue games and PCF. In *Proceedings of Conference on Functional Programming Languages and Computer Architecture*, pages 96 – 107, 1995.

[27] P. LINCOLN. Linear logic. *ACM SIGACT News*, 23(2):29–37, 1992.

[28] S. MARTINI AND A. MASINI. Experiments in linear natural deduction. Unpublished Manuscript, March 1995.

[29] C.-H.L. ONG. A semantic view of classical proofs: type-theoretic, categorical and denotational characterizations. To appear in LICS, 1996.

[30] M. PARIGOT. $\lambda\mu$-calculus: an algorithmic interpretation of classical natural deduction. In *Proceedings of Conference on Logic Programming and Automated Reasoning*, volume 624 of *Lecture Notes in Computer Science*, pages 190–201, 1992.

[31] D. PRAWITZ. *Natural Deduction*, volume 3 of *Stockholm Studies in Philosophy.* Almqvist and Wiksell, 1965.

[32] D. PRAWITZ. Ideas and results in proof theory. In J.E. Fenstad, editor, *Proceedings of Second Scandinavian Logic Symposium*, pages 235–307, 1971.

[33] H. SCHELLINX. *The Noble Art of Linear Decorating.* PhD thesis, University of Amsterdam, February 1994.

[34] D.J. SHOESMITH AND T.J. SMILEY. *Multiple-Conclusion Logic.* Cambridge University Press, 1978.

[35] A.S. TROELSTRA. *Lectures on Linear Logic*, volume 29 of *Lecture Notes.* CSLI, 1992.

A A Computational Interpretation

In §2.2 Parigot's formulation was motivated in terms of proof theory, but a worthwhile question is whether there is a more convincing computer science explanation. Consider again the $\bot_\mathcal{I}$ rule,

$$\frac{\Gamma \rhd M \colon \phi, \Sigma}{\Gamma \rhd \mathsf{unit}_a^\phi(M) \colon \bot, a \colon \phi, \Sigma}\ (\bot_\mathcal{I}).$$

A key to understanding this rule is to give a computational explanation of the passive formulae. To do so I shall rewrite it as the following

$$\frac{\Gamma \rhd M \colon \phi, \Sigma}{\Gamma \rhd \kappa M \colon \bot, \kappa \colon \phi \multimap \bot, \Sigma}\ \mathit{Catch}.$$

Here κ is to be thought of as a *continuation variable*; and thus *Catch* can be seen as a special kind of application. A judgement $\vec{x} \colon \Gamma \rhd M \colon \phi, \vec{\kappa} \colon \Sigma$ consists of a term, M, with (typed) free variables, $\vec{x}$, and (typed) free continuation variables, $\vec{\kappa}$. (Hence Σ is now a multiset of continuation variables.) The $\bot_\mathcal{E}$ rule can similarly be rewritten as

$$\frac{\Gamma \rhd M \colon \bot, \kappa \colon \phi \multimap \bot, \Sigma}{\Gamma \rhd \mathsf{throw}_\phi^\kappa(M) \colon \phi, \Sigma}\ \mathit{Throw}.$$

Before understanding this rule we need to introduce some standard terminology from work in continuation-passing, e.g. [17]. To formalise the notion of an evaluation order, Felleisen [*op. cit.*], defined an *evaluation context*. This is essentially a term with a 'hole' in it, written $E[\ ]$. Placing a term, M, in that hole is written $E[M]$. These contexts are devised so that every closed term, M, is either a canonical value or can be written *uniquely* as $E[N]$, where N is a redex. Reduction then proceeds towards a canonical value as follows.

$$M \equiv E[N] \Rightarrow E[N'] \Rightarrow \cdots \Rightarrow V.$$

The context $E[-]$ can be thought of as representing the rest of the computation that remains to be done after N has evaluated (to a value). In this sense it can be thought of as a *continuation* of N. The various continuation, or control, operators which have been introduced (e.g. [14, 16]), can be explained with reference to this continuation.

To understand this computational interpretation of the (linear) $\lambda\mu$-calculus we need to introduce an additional context which contains a multiset of labelled terms (the continuations). For example given a closed term

$$\rhd M \colon \phi, \kappa_1 \colon \varphi_1 \multimap \bot, \ldots, \kappa_n \colon \varphi_n \multimap \bot,$$

we need a multiset of continuations $\mathcal{E} = [M_1, \ldots, M_n]$, where $\rhd M_i \colon \varphi_i \multimap \bot$. Evaluation is then written as

$$E[N]\,\mathcal{E} \Rightarrow M'$$

where $M \equiv E[N]$, as discussed above. The important evaluation rules are

$$\begin{array}{rcl} E[\kappa M]\,\mathcal{E}\dagger\{\kappa \colon N[-]\} & \Rightarrow & E[N[M]]\,\mathcal{E}, \\ E[\mathsf{throw}^\kappa(M)]\,\mathcal{E} & \Rightarrow & M\,\mathcal{E}\dagger\{\kappa \colon E[-]\}; \end{array}$$

where $\mathcal{E} \dagger \{k{:}\, N[-]\}$ denotes the extension of the continuation multiset $\mathcal{E}$ with $k{:}\, N[-]$. Thus *Throw* captures the current continuation and places it in the continuation multiset, labelled with κ. The *Catch* catches a continuation[11] from the multiset and replaces the continuation variable with the caught term.

Whilst this context view of continuation passing is by far the most intuitive (it essentially describes an abstract machine), it has traditionally been presented in an untyped setting (e.g. [17]). When moving to a typed setting it becomes a little messy. The problem is that for the second evaluation rule given above to be type correct, the continuation $E[-]$ must be of type $\perp$. Of course there are no closed terms of type $\perp$ so, at first sight, this rule appears useless. Griffin [23] noticed a similar problem for Felleisen's calculus and suggested that instead of evaluating a term M of type ϕ; the expression $\mathcal{C}(\lambda k{:}\, \phi \supset\perp .kM)$ is evaluated with the evaluation rule being applied to the inner term M.[12] The term $\mathcal{C}$ is that of type $\neg\neg\phi \supset \phi$. Applying these ideas to the linear $\lambda\mu$-calculus, the term $\mathcal{C}$ is given by

$$\mathcal{C} \stackrel{\text{def}}{=} \lambda y{:}\, (\phi\multimap \perp)\multimap \perp .\mathsf{deunit}^{\phi}_{a}(y(\lambda x{:}\, \phi.\mathsf{unit}^{\phi}_{a}(x))).$$

The two rules given earlier are adjusted suitably and we add a new rule

$$\mathcal{C}(\lambda k{:}\, \phi\multimap \perp .kV) \Rightarrow V,$$

where V is a value. This reduction rule is justified by the following reasoning

$$\begin{array}{ll} & (\lambda y{:}\, (\phi\multimap \perp)\multimap \perp .\mathsf{deunit}^{\phi}_{a}(y(\lambda x{:}\, \phi.\mathsf{unit}^{\phi}_{a}(x))))(\lambda k{:}\, \phi\multimap \perp .kV) \\ \rightsquigarrow_{\beta} & \mathsf{deunit}^{\phi}_{a}((\lambda k{:}\, \phi\multimap \perp .kV)(\lambda x{:}\, \phi.\mathsf{unit}^{\phi}_{a}(x))) \\ \rightsquigarrow_{\beta} & \mathsf{deunit}^{\phi}_{a}((\lambda x{:}\, \phi.\mathsf{unit}^{\phi}_{a}(x))V) \\ \rightsquigarrow_{\beta} & \mathsf{deunit}^{\phi}_{a}(\mathsf{unit}^{\phi}_{a}(V)) \\ \rightsquigarrow_{\beta} & V. \end{array}$$

The rather complicated formulation of the *Promotion* rule from §3.2 becomes slightly clearer with this continuation interpretation. The rule is rewritten as

$$\frac{\begin{array}{ll} \Gamma_1 \triangleright M_1{:}\, !\phi_1, \Sigma_1 & \Delta_1 \triangleright P_1{:}\, !((!\varphi_1\multimap \perp)\multimap \perp), \Upsilon_1 \\ \Gamma_n \triangleright M_n{:}\, !\phi_n, \Sigma_n & \Delta_m \triangleright P_m{:}\, !((!\varphi_m\multimap \perp)\multimap \perp), \Upsilon_m \\ \multicolumn{2}{c}{x_1{:}\, !\phi_1, \ldots, x_n{:}\, !\phi_n \triangleright N{:}\, \psi, \kappa_1{:}\, (!\varphi_1\multimap \perp)\multimap \perp, \ldots, \kappa_m{:}\, (!\varphi_m\multimap \perp)\multimap \perp} \end{array}}{\vec{\Gamma}, \vec{\Delta} \triangleright \mathsf{promote}\ \vec{M}|\vec{P}\ \mathsf{for}\ \vec{x}|\vec{\kappa}\ \mathsf{in}\ N{:}\, !\psi, \vec{\Sigma}, \vec{\Upsilon}}\ \textit{Promotion}.$$

Thus the promoted term can be seen not only as a sort of closure for the free variables, as is the case for **ILL**, but also for the continuation variables; where we build in substitution for both classes of variable. As this closure can be freely duplicated and discarded, the continuation terms, P_i, must be of a non-linear type.

It should be emphasized that this interpretation is not dependent on linearity, *viz.* it can be formulated for the $\lambda\mu$-calculus as well. In comparison to other works where authors have used continuation-passing work to explain classical logic, this interpretation is essentially in the other direction, *viz.* using classical logic to suggest a continuation-passing technique. The advantage here is that a quite complicated programming feature is given directly by a proof theory. Filinski [18] has suggested that linear versions of conventional continuation-passing ideas are of some use, and I would hope that these advantages apply to this system.

[11] Linearity guarantees that the continuation exists.

[12] An alternative solution is to devise a system of reduction rules which somehow matches the context view of evaluation. After completing this paper, Luke Ong informed me of his (as yet unpublished) work, where he defines such a system for both call-by-value and call-by-name versions of the $\lambda\mu$-calculus.

$$\frac{}{\Theta; x\colon\phi \rhd_l x\colon\phi; \Upsilon}\ \mathit{Identity}_l \qquad \frac{}{\Theta, x\colon\alpha \rhd_n x\colon\alpha, \Upsilon}\ \mathit{Identity}_n$$

$$\frac{\Theta;\Gamma, x\colon\phi \rhd_l M\colon\psi, \Sigma;\Upsilon}{\Theta;\Gamma \rhd_l \lambda x\colon\phi.M\colon\phi\multimap\psi, \Sigma;\Upsilon}\ (\multimap_{\mathcal{I}}) \qquad \frac{\Theta, x\colon\alpha \rhd_n M\colon\beta, \Upsilon}{\Theta \rhd_n \lambda x\colon\alpha.M\colon\alpha\supset\beta, \Upsilon}\ (\supset_{\mathcal{I}})$$

$$\frac{\Theta;\Gamma \rhd_l M\colon\phi\multimap\psi, \Sigma;\Upsilon \qquad \Theta;\Gamma' \rhd_l N\colon\phi, \Sigma';\Upsilon}{\Theta;\Gamma,\Gamma' \rhd_l MN\colon\psi, \Sigma,\Sigma';\Upsilon}\ (\multimap_{\mathcal{E}}) \qquad \frac{\Theta \rhd_n M\colon\alpha\supset\beta, \Upsilon \qquad \Theta \rhd_n N\colon\beta, \Upsilon}{\Theta \rhd_n MN\colon\beta, \Upsilon}\ (\supset_{\mathcal{E}})$$

$$\frac{\Theta;\Gamma \rhd_l M\colon\phi, \Sigma;\Upsilon}{\Theta;\Gamma \rhd_l \mathsf{unit}^{\phi}_{a}(M)\colon\bot, a\colon\phi, \Sigma;\Upsilon}\ (\bot_{\mathcal{I}}) \qquad \frac{\Theta \rhd_n M\colon\alpha, \Upsilon}{\Theta \rhd_n \mathsf{pass}^{\alpha}_{a}(M)\colon\mathbf{f}, a\colon\alpha, \Upsilon}\ (\mathbf{f}_{\mathcal{I}})$$

$$\frac{\Theta;\Gamma \rhd_l M\colon\bot, a\colon\phi, \Sigma;\Upsilon}{\Theta;\Gamma \rhd_l \mathsf{deunit}^{\phi}_{a}(M)\colon\phi, \Sigma;\Upsilon}\ (\bot_{\mathcal{E}}) \qquad \frac{\Theta \rhd_n M\colon\mathbf{f}, a\colon\alpha, \Upsilon}{\Theta \rhd_n \mathsf{act}^{\alpha}_{a}(M)\colon\alpha, \Upsilon}\ (\mathbf{f}_{\mathcal{E}})$$

$$\frac{\Theta \rhd_n M\colon\alpha, \Upsilon}{\Theta; - \rhd_l \mathsf{F}(M)\colon\mathsf{F}(\alpha);\Upsilon}\ (\mathsf{F}_{\mathcal{I}}) \qquad \frac{\Theta; - \rhd_l M\colon\phi;\Upsilon}{\Theta \rhd_n \mathsf{G}(M)\colon\mathsf{G}(\phi), \Upsilon}\ (\mathsf{G}_{\mathcal{I}})$$

$$\frac{\Theta;\Gamma \rhd_l M\colon\mathsf{F}(\alpha), \Sigma;\Upsilon \qquad \Theta, x\colon\alpha;\Gamma' \rhd_l N\colon\phi, \Sigma';\Upsilon}{\Theta;\Gamma,\Gamma' \rhd_l \mathsf{let}\ M\ \mathsf{be}\ \mathsf{F}(x)\ \mathsf{in}\ N\colon\phi, \Sigma,\Sigma';\Upsilon}\ (\mathsf{F}_{\mathcal{E}})$$

$$\frac{\Theta \rhd_n M\colon\mathsf{G}(\phi), \Upsilon}{\Theta; - \rhd_l \mathsf{derelict}(M)\colon\phi;\Upsilon}\ (\mathsf{G}_{\mathcal{E}})$$

Figure 11: A linear/non-linear presentation of the linear $\lambda\mu$-calculus

B An Alternative Formulation

In this appendix I shall sketch rather briefly how one can apply the ideas of Benton [6] to the linear $\lambda\mu$-calculus (for brevity I shall only discuss the $(\multimap, !)$-fragment). Benton proposed (following a categorical insight) to present **ILL** in three parts: a linear subsystem, a non-linear subsystem and a third part containing operations to move between the two subsystems. The exponential can then be thought of as a composite of these operations. I shall not go into any real detail here, the reader is referred to Benton's paper [*op. cit.*].

I shall use the following conventions: α to range over non-linear formulae, ϕ to range over linear formulae, Γ to range over linear contexts, Θ to range over non-linear contexts, Σ to range over linear passive contexts and Υ to range over non-linear passive contexts. Formulae are then defined by the grammars

$$\begin{array}{rcl} \alpha & ::= & p \mid \alpha\supset\alpha \mid \mathsf{G}(\phi), \text{ and} \\ \phi & ::= & q \mid \phi\multimap\phi \mid \mathsf{F}(\alpha); \end{array}$$

where p ranges over some countable set of non-linear atomic formulae including a distinguished member $\mathbf{f}$, and q ranges over some countable set of linear atomic formulae including the distinguished member $\bot$.

We have two forms of deduction, linear and non-linear, which are of the form $\Theta;\Gamma \rhd_l M\colon\phi, \Sigma;\Upsilon$ and $\Theta \rhd_n M\colon\alpha, \Upsilon$, respectively. Rather than explain these forms of deduction I shall simply give the term assignment rules in Figure 11.

The β-rules for this formulation are then quite succinct

$$
\begin{array}{rcll}
(\lambda x{:}\,\phi.M)N & \rightsquigarrow_\beta & M[x:=N], & \\
(\lambda x{:}\,\alpha.M)N & \rightsquigarrow_\beta & M[x:=N], & \\
\mathsf{deunit}_a^\phi(\mathsf{unit}_a^\phi(M)) & \rightsquigarrow_\beta & M, & \\
\mathsf{act}_a^\alpha(\mathsf{pass}_a^\alpha(M)) & \rightsquigarrow_\beta & M & \text{where } a \notin \mathrm{FN}(M), \\
\mathsf{let}\ \mathsf{F}(M)\ \mathsf{be}\ \mathsf{F}(x)\ \mathsf{in}\ N & \rightsquigarrow_\beta & N[x:=M], & \\
\mathsf{derelict}(\mathsf{G}(M)) & \rightsquigarrow_\beta & M. &
\end{array}
$$

There are also commuting conversations, which I leave to the reader to discover. It is possible to translate between this linear/non-linear formulation and the linear $\lambda\mu$-calculus. First we need some translations between types.

$$
\begin{array}{rcl}
p^\circ & \stackrel{\text{def}}{=} & p, \\
(\phi \multimap \psi)^\circ & \stackrel{\text{def}}{=} & \phi^\circ \multimap \psi^\circ, \\
(!\phi)^\circ & \stackrel{\text{def}}{=} & \mathsf{F}(\mathsf{G}(\phi^\circ));
\end{array}
$$

$$
\begin{array}{rcl}
q^\star & \stackrel{\text{def}}{=} & q, \\
(\phi \multimap \psi)^\star & \stackrel{\text{def}}{=} & \phi^\star \multimap \psi^\star, \\
(\mathsf{F}(\alpha))^\star & \stackrel{\text{def}}{=} & !(\alpha^\star), \\
p^\star & \stackrel{\text{def}}{=} & \begin{cases} \bot & \text{if } p = \mathbf{f} \\ p & \text{otherwise,} \end{cases} \\
(\alpha \supset \beta)^\star & \stackrel{\text{def}}{=} & !\alpha^\star \multimap \beta^\star, \\
(\mathsf{G}(\phi))^\star & \stackrel{\text{def}}{=} & \phi^\star.
\end{array}
$$

Theorem 9.

1. If $\Gamma \triangleright M{:}\,\phi, \Sigma$ then there is a term M° such that $-; \Gamma^\circ \rhd_l M^\circ{:}\,\phi^\circ, \Sigma^\circ; -$.
2. If $\Theta; \Gamma \rhd_l M{:}\,\phi, \Sigma; \Upsilon$ then there is a linear $\lambda\mu$-term $M^\star$ such that $!\Theta^\star, \Gamma^\star \triangleright M^\star{:}\,\phi^\star, \Sigma^\star, !\Upsilon^\star$.
3. If $\Theta \rhd_n M{:}\,\alpha, \Upsilon$ then there is a linear $\lambda\mu$-term $M^\star$ such that $!\Theta^\star \triangleright M^\star{:}\,\alpha^\star, !\Upsilon^\star$.

However, as is the case for **ILL**, it is not immediately clear how much of an improvement this formulation is. A smaller set of reduction rules has been gained at the expense of a loss of information about *Weakening* and *Contraction*, which surely are the *raison d'être* of linear proof theory. Of course, at the level of a programming language, explicit duplication and erasure of data structures would be quite tiresome and it seems that this mixed presentation might be of some practical value.