URL: http://www.elsevier.nl/locate/entcs/volume3.html 13 pages

# Towards a Classical Linear $\lambda$-calculus (Preliminary Report)

G.M. Bierman

*Gonville and Caius College,*
*Cambridge, United Kingdom.*

**Abstract**

This paper considers a typed $\lambda$-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. I shall also demonstrate a somewhat hidden connexion with the continuation-passing paradigm which gives a new computational interpretation of Parigot's techniques and possibly a new style of continuation programming.

## 1 Introduction

Recently there has been renewed interest in classical logic, or rather in the constructive content of classical proofs. This appears to have links with, on the theoretical side, game theory and on the practical side, certain extensions to functional programming languages. Intuitionistic linear logic (**ILL**) can be seen as a foundation of functional programming languages and so it would seem interesting to consider extensions of it to classical linear logic (**CLL**). In particular as it has been suggested that **CLL** has strong links with concurrent computation.

## 2 Parigot's Method

Gentzen's natural deduction is a very suitable deduction system for intuitionistic logic (**IL**) but seems less so for classical logic[1] (**CL**). 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 allows us to annotate natural deductions with terms. For **IL** this yields the typed $\lambda$-calculus. For sequent calculus it is not entirely clear what the appropriate annotations are. In fact there are

[1] "One may doubt that this is the proper way of analysing classical inferences." [12, Pages 244-5].

a number of choices and there is no real consensus on the best. It might seem prudent to revisit natural deduction, where the question of syntax is settled, and see if we might be able to produce a more symmetric system.

Shoesmith and Smiley [13] made an early attempt at this by defining a multiple-conclusion natural deduction system but unfortunately this was quite complicated. More recently Parigot [10] has introduced a variant of multiple-conclusion natural deduction which seems better suited for handling **CL**. I hope to provide an alternative explanation of his method and, in a later section, to utilise it to produce a formulation of **CLL**.

# 3 From Intuitionistic Logic to Classical Logic

Traditionally **IL** can be presented in a sequent calculus formulation where sequents have the form $\Gamma \vdash \psi$. Thus from many assumptions (which are to be thought of as being conjoined) one deduces $\psi$. To extend this to **CL** we allow the conclusion to contain many formulae (which are to be thought of as being disjoined).

In the natural deduction system, deductions take the form of (inverted) trees, *viz.*

$$\begin{array}{c}\Gamma\\ \vdots\\ \psi\end{array}$$

which is clearly well suited when we have just one conclusion. Extending this 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 formulae. Consider the implication-right rule of **CL**

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

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 $\psi$. The implication introduction rule will only allow

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

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

**ImpD**: $(\phi \supset (\delta \vee \psi)) \supset ((\phi \supset \delta) \vee \psi)$

is a sufficient addition to **IL** to give **CL**. Rather than continue with this 'simulation' of **CL**, traditional proof theory considers new rules to add to the system to yield **CL**. For example, Prawitz [11] suggests either adding axioms of the form

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

or a rule

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

Parigot's system can be thought of as continuing with the simulation approach and adding what is sufficient to make that method work. Thus we continue with 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

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

where $\phi$ is the active formula and the $\psi_i$ are passive. I shall write $\Sigma$ to represent a multiset of passive formulae. 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

$$\cfrac{\begin{array}{c}\Gamma \\ \vdots \\ \phi^{\bullet}, \Sigma\end{array}}{\phi^{\alpha}, \Sigma}\ \mathit{Freeze} \qquad \text{and} \qquad \cfrac{\begin{array}{c}\Gamma \\ \vdots \\ \psi^{\alpha}, \Sigma\end{array}}{\psi^{\bullet}, \Sigma}\ \mathit{Unfreeze}.$$

It is important to realise that neither an active formula, $\phi$, nor a passive formula, $\psi$, 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.[2] (Of course, when we come to the linear calculus, this will not be the case.) This enables us to handle the earlier example, as follows

$$\cfrac{\cfrac{\cfrac{\begin{array}{c}\Gamma\ [\phi] \\ \vdots \\ \psi^{\alpha}, \Delta, \chi^{\bullet}\end{array}}{\psi^{\alpha}, \Delta, \chi^{\beta}}\ \mathit{Freeze}}{\psi^{\bullet}, \Delta, \chi^{\beta}}\ \mathit{Unfreeze}}{(\phi \supset \psi)^{\bullet}, \Delta, \chi^{\beta}}\ (\supset_{\mathcal{I}}).$$

Parigot's system (which he calls the $\lambda\mu$-calculus) is given below, where for compactness I have presented the deductions in a sequent-style and added a

[2] This is clarified if Parigot's system is presented with explicit structural rules.

term syntax.

$$\Gamma, x{:}\,\phi \triangleright x{:}\,\phi^\bullet, \Sigma$$

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

$$\frac{\Gamma \triangleright M{:}\,\phi^\bullet, \Sigma}{\Gamma \triangleright \mathsf{freeze}^\phi_\alpha(M){:}\,\phi^\alpha, \Sigma}\,\mathit{Freeze} \qquad \frac{\Gamma \triangleright M{:}\,\phi^\alpha, \Sigma}{\Gamma \triangleright \mathsf{unfreeze}^\phi_\alpha(M){:}\,\phi^\bullet, \Sigma}\,\mathit{Unfreeze}$$

Thus judgements are of the form $\Gamma \triangleright M{:}\,\phi^\bullet, \Sigma$ where $\Gamma$ denotes a set of formulae labelled with variable names, written $x{:}\,\psi$ and $\Sigma$ denotes a set of (passive) formulae labelled with 'passification variables', which we write as $\psi^\alpha$. To demonstrate the full power of Parigot's system, below is a derivation of the famous Peirce's law.

$$\dfrac{\dfrac{\dfrac{y{:}\,(\phi \supset \psi) \supset \phi \triangleright y{:}\,(\phi \supset \psi) \supset \phi^\bullet \qquad \dfrac{\dfrac{\dfrac{x{:}\,\phi \triangleright x{:}\,\phi^\bullet}{x{:}\,\phi \triangleright \mathsf{freeze}^\phi_\alpha(x){:}\,\phi^\alpha}\,\mathit{Freeze}}{x{:}\,\phi \triangleright \mathsf{unfreeze}^\psi_\beta(\mathsf{freeze}^\phi_\alpha(x)){:}\,\psi^\bullet, \phi^\alpha}\,\mathit{Unfreeze}\ (1)}{\triangleright \lambda x{:}\,\phi.\mathsf{unfreeze}^\psi_\beta(\mathsf{freeze}^\phi_\alpha(x)){:}\,\phi \supset \psi^\bullet, \phi^\alpha}\,(\supset_\mathcal{I})}{\dfrac{y{:}\,(\phi \supset \psi) \supset \phi \triangleright y(\lambda x{:}\,\phi.\mathsf{unfreeze}^\psi_\beta(\mathsf{freeze}^\phi_\alpha(x))){:}\,\phi^\bullet, \phi^\alpha}{y{:}\,(\phi \supset \psi) \supset \phi \triangleright \mathsf{freeze}^\phi_\alpha(y(\lambda x{:}\,\phi.\mathsf{unfreeze}^\psi_\beta(\mathsf{freeze}^\phi_\alpha(x)))){:}\,\phi^\alpha}\,\mathit{Freeze}\ (2)}\,(\supset_\mathcal{E})}{y{:}\,(\phi \supset \psi) \supset \phi \triangleright \mathsf{unfreeze}^\phi_\alpha(\mathsf{freeze}^\phi_\alpha(y(\lambda x{:}\,\phi.\mathsf{unfreeze}^\psi_\beta(\mathsf{freeze}^\phi_\alpha(x))))){:}\,\phi^\bullet}\,\mathit{Unfreeze}}{\triangleright \lambda y{:}\,(\phi \supset \psi) \supset \phi.\mathsf{unfreeze}^\phi_\alpha(\mathsf{freeze}^\phi_\alpha(y(\lambda x{:}\,\phi.\mathsf{unfreeze}^\psi_\beta(\mathsf{freeze}^\phi_\alpha(x))))){:}\,((\phi \supset \psi) \supset \phi) \supset \phi^\bullet}\,(\supset_\mathcal{I})$$

The reader familiar with the sequent calculus formulation of **CL** will spot where Parigot's formulation mimics the sequent proof. More specifically the application of the *Unfreeze* rule, marked (1), corresponds to the *Weakening-Right* rule and the application of the *Freeze* rule marked (2), corresponds to the *Contraction-Right* rule.

### *3.1 Reduction Rules*

There are two $\beta$-rules corresponding to introduction-elimination pairs, along with a commuting conversion for the *Unfreeze* rule.

$$\begin{aligned} (\lambda x{:}\,\phi.M)N &\rightsquigarrow_\beta M[x := N] \\ \mathsf{unfreeze}^\phi_\alpha(\mathsf{freeze}^\phi_\alpha(M)) &\rightsquigarrow_\beta M \\ (\mathsf{unfreeze}^{\phi\supset\psi}_\alpha(M))N &\rightsquigarrow_c \mathsf{unfreeze}^\psi_\alpha(M[\mathsf{freeze}^{\phi\supset\psi}_\alpha(P) \Leftarrow \mathsf{freeze}^\psi_\alpha(PN)]) \end{aligned}$$

In the last commuting conversion rule, I have used the notation $M[N \Leftarrow P]$ to denote the term $M$ where (inductively) all occurrences of the subterm $N$ have been replaced by the term $P$.

Parigot has (impressively) shown the following results for this reduction system.

**Theorem 3.1**

(i) *The λμ-calculus is strongly normalising; and*

(ii) *The λμ-calculus is confluent.*

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 [8, 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 $\pi_1$ or to $\pi_2$. It is interesting to note that this example translates to the following application of substitution in Parigot's formulation

$$[\![\pi_2]\!][x := [\![\pi_1]\!]]$$

where $x$ is *not* a free variable of $[\![\pi_2]\!]$, and so by the definition of substitution, this is equal to

$$[\![\pi_2]\!].$$

Thus Parigot's formulation automatically avoids critical pairs essentially by its syntactic form for the structural rules.

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

$$\dfrac{\dfrac{\dfrac{}{x\colon \phi \supset \perp \rhd x\colon \phi \supset \perp} \qquad \Gamma \rhd M\colon \phi}{\Gamma, x\colon \phi \supset \perp \rhd xM\colon \perp}\,(\supset_{\mathcal{E}})}{\Gamma \rhd \lambda x\colon \phi \supset \perp.xM\colon (\phi \supset \perp) \supset \perp}\,(\supset_{\mathcal{I}})$$

and

$$\dfrac{\dfrac{\Gamma \rhd M\colon (\phi \supset \perp) \supset \perp \qquad \dfrac{\dfrac{\dfrac{}{x\colon \phi \rhd x\colon \phi}}{x\colon \phi \rhd \mathsf{freeze}^{\phi}_{\alpha}(x)\colon \perp^{\bullet}, \phi^{\alpha}}\,\mathit{Freeze}}{\rhd \lambda x.\mathsf{freeze}^{\phi}_{\alpha}(x)\colon \phi \supset \perp^{\bullet}, \phi^{\alpha}}\,(\supset_{\mathcal{I}})}{\Gamma \rhd M(\lambda x.\mathsf{freeze}^{\phi}_{\alpha}(x))\colon \perp^{\bullet}, \phi^{\alpha}}\,(\supset_{\mathcal{E}})}{\Gamma \rhd \mathsf{unfreeze}^{\phi}_{\alpha}(M(\lambda x.\mathsf{freeze}^{\phi}_{\alpha}(x)))\colon \phi}\,\mathit{Unfreeze.}$$

Composing the first with the second gives

$$\begin{aligned}\mathsf{unfreeze}((\lambda x.xM)(\lambda x.\mathsf{freeze}(x))) &\rightsquigarrow_{\beta} \mathsf{unfreeze}((\lambda x.\mathsf{freeze}(x))M)\\ &\rightsquigarrow_{\beta} \mathsf{unfreeze}(\mathsf{freeze}(M))\\ &\rightsquigarrow_{\beta} M;\end{aligned}$$

but composing the second with the first yields

$$\lambda y.y(\mathsf{unfreeze}(M(\lambda x.\mathsf{freeze}(x)))$$

which is in (head) normal form.[3]

### 3.2 Normal Forms

Before I consider a linear version of Parigot's system, it seems prudent to highlight a slightly tricky area: that of normal forms. Parigot takes the opinion that the *Unfreeze* rule can act as a barrier between an introduction-elimination pair and so adds a commuting conversion to remove it. This has both a familiar and unfamiliar feel to it. We are used to this notion of commuting conversions to permit $\beta$-reductions from the writings of Prawitz. 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. Prawitz, as mentioned earlier, suggests adding the rule

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

to **IL** to get a formulation of **CL**. However, he notes that applications of this rule can be restricted to cases where $\phi$ 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

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

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 [11, Footnote 1, Page 50] mentions that this technique does not extend to all the connectives (the problematic one being the disjunction).

Ong [9] suggests a similar strategy for Parigot's system by rewriting applications of *Unfreeze* until they are of atomic type, although he advocates it to ensure confluence when considering $\eta$-reduction. Given that this technique requires the use of the de Morgan dualities when considering all the connectives,

[3] This property enables Ong [9] 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.

I shall not consider it here.

## 4 From Intuitionistic Linear Logic to Classical Linear Logic

I shall extend the natural deduction formulation of **ILL** from my thesis [3] (which has appeared in other places e.g. [2]) using Parigot's techniques as explained in the previous section. The resulting system is given below.

$$x\colon\phi \rhd x\colon\phi^{\bullet}$$

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

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

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

$$\frac{\begin{array}{ll}\Gamma_1 \rhd M_1\colon!\phi_1^{\bullet}, \Sigma_1 & \Delta_1 \rhd P_1\colon!((!\sigma_1\multimap\bot)\multimap\bot)^{\bullet}, \Upsilon_1\\ \Gamma_n \rhd M_n\colon!\phi_n^{\bullet}, \Sigma_n & \Delta_m \rhd P_m\colon!((!\sigma_m\multimap\bot)\multimap\bot)^{\bullet}, \Upsilon_m\\ \multicolumn{2}{l}{x_1\colon!\phi_1, \ldots, x_n\colon!\phi_n \rhd N\colon\psi^{\bullet}, (!\sigma_1\multimap\bot)^{\alpha_1}, \ldots, (!\sigma_m\multimap\bot)^{\alpha_m}}\end{array}}{\vec{\Gamma}, \vec{\Delta} \rhd \mathsf{promote}\ \vec{M}|\vec{P}\ \mathsf{for}\ \vec{x}|\vec{\alpha}\ \mathsf{in}\ N\colon!\psi^{\bullet}, \vec{\Sigma}, \vec{\Upsilon}}\,\textit{Promotion}$$

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

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

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

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

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

A few observations need to be made before continuing. In this formulation the par unit, $\bot$, must be introduced. The other classical connectives become defined as follows

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

It is interesting that, $I$, the tensor unit now becomes a defined formula, the details are given below. Applications of the $\bot_{\mathcal{I}}$ rule are restricted such that the upper active formula, $\phi$, is not equal to $\bot$. The $\bot_{\mathcal{E}}$ rule is similarly restricted.

A further property of this formulation is that the *Promotion* rule is *not* the same as that for **ILL**. It seems that rather the **ILL** formulation is a particular instance of the full classical formulation.

However the formulation is sound and complete in the following sense.

**Theorem 4.1** *$\phi$ is provable in* **CLL** *if and only if there is a term $M$ such that $\triangleright M : \phi^{\bullet}$.*

It is worth discussing further the nature of negation in this system. Consider the formula $\phi^{\bot\bot} \multimap \phi$. In the standard presentation of proof nets this is just the identity function as all formulae are factored by the equivalences for negation; in particular, $\phi^{\bot\bot} \equiv \phi$. In systems based on Parigot's method this will not be the case, the formula is $((\phi \multimap \bot) \multimap \bot) \multimap \phi$. In some senses one could say that negation retains here a more constructive nature. This does not seem that unreasonable, however. By analogy consider the formulae $\phi \wedge (\psi \vee \sigma)$ and $(\phi \wedge \psi) \vee (\phi \wedge \sigma)$. In both **CL** and **IL** these are equivalent and, for example, in a cartesian closed category (with coproducts) they are isomorphic. However we wouldn't necessarily expect to collapse them. Indeed in the simply typed $\lambda$-calculus, there are distinct functions which map from one to the other (as they represent distinct datatypes!).

### *4.1 Reduction Rules*

We have the $\beta$-rules for the linear $\lambda$-calculus, suitably extended for the *Promotion* rule, as well as the $\beta$-rule for the new unit $\bot$. In addition, I shall give the commuting conversions, as per the discussion in §3.2.

$$\begin{aligned} (\lambda x : \phi . M)\, N &\leadsto_{\beta} M[x := N] \\ \mathsf{let}\ M \otimes N\ \mathsf{be}\ x \otimes y\ \mathsf{in}\ P &\leadsto_{\beta} P[x := M, y := N] \\ \mathsf{derelict}(\mathsf{promote}\ \vec{M} | \vec{P}\ \mathsf{for}\ \vec{x} | \vec{\alpha}\ \mathsf{in}\ N) &\leadsto_{\beta} N\,[x_i := M_i, \\ & \qquad \mathsf{unit}_{\alpha_j}^{\sigma_j \multimap \bot}(R) \Leftarrow \mathsf{derelict}(P_j) R] \\ \mathsf{discard}\ (\mathsf{promote}\ \vec{M} | \vec{P}\ \mathsf{for}\ \vec{x} | \vec{\alpha}\ \mathsf{in}\ N)\ \mathsf{in}\ R &\leadsto_{\beta} \mathsf{discard}\ \vec{M}, \vec{P}\ \mathsf{in}\ R \end{aligned}$$

$$\begin{array}{rl}\mathsf{copy}\ (\mathsf{promote}\ \vec{M}|\vec{P}\ \mathsf{for}\ \vec{x}|\vec{\alpha}\ \mathsf{in}\ N)\ \mathsf{as}\ y,z\ \mathsf{in}\ R \rightsquigarrow_\beta & \mathsf{copy}\ \vec{M}\ \mathsf{as}\ \vec{x'},\vec{x''}\ \mathsf{in}\\ & \mathsf{copy}\ \vec{P}\ \mathsf{as}\ \vec{w'},\vec{w''}\ \mathsf{in}\\ & R\,[y := \mathsf{promote}\ \vec{x'}|\vec{w'}\ \mathsf{for}\ \vec{x}|\vec{\alpha}\ \mathsf{in}\ N,\\ & \quad z := \mathsf{promote}\ \vec{x''}|\vec{w''}\ \mathsf{for}\ \vec{x}|\vec{\alpha}\ \mathsf{in}\ N]\end{array}$$

$$\begin{array}{rcl}\mathsf{deunit}^{\phi}_{\alpha}(\mathsf{unit}^{\phi}_{\alpha}(M)) & \rightsquigarrow_\beta & M\\ (\mathsf{deunit}^{\phi\multimap\psi}_{\alpha}(M))N & \rightsquigarrow_c & \mathsf{deunit}^{\psi}_{\alpha}(M[\mathsf{unit}^{\phi\multimap\psi}_{\alpha}(P) \Leftarrow \mathsf{unit}^{\psi}_{\alpha}(PN)])\\ \mathsf{let}\ \mathsf{deunit}^{\phi\otimes\psi}_{\alpha}(M)\ \mathsf{be}\ x\otimes y\ \mathsf{in}\ N & \rightsquigarrow_c & \mathsf{deunit}^{\theta}_{\alpha}(M[\mathsf{unit}^{\phi\otimes\psi}_{\alpha}(P) \Leftarrow \mathsf{unit}^{\theta}_{\alpha}(\mathsf{let}\ P\ \mathsf{be}\ x\otimes y\ \mathsf{in}\ N)])\\ \mathsf{derelict}(\mathsf{deunit}^{!\phi}_{\alpha}(M)) & \rightsquigarrow_c & \mathsf{deunit}^{\phi}_{\alpha}(M[\mathsf{unit}^{!\phi}_{\alpha}(P) \Leftarrow \mathsf{unit}^{\phi}_{\alpha}(\mathsf{derelict}(P))])\\ \mathsf{copy}\ (\mathsf{deunit}^{!\phi}_{\alpha}(M))\ \mathsf{as}\ x,y\ \mathsf{in}\ N & \rightsquigarrow_c & \mathsf{deunit}^{\theta}_{\alpha}(M[\mathsf{unit}^{!\phi}_{\alpha}(P) \Leftarrow \mathsf{unit}^{\theta}_{\alpha}(\mathsf{copy}\ P\ \mathsf{as}\ x,y\ \mathsf{in}\ N)])\\ \mathsf{discard}\ (\mathsf{deunit}^{!\psi}_{\alpha}(M))\ \mathsf{in}\ N & \rightsquigarrow_c & \mathsf{deunit}^{\theta}_{\alpha}(M[\mathsf{unit}^{!\phi}_{\alpha}(P) \Leftarrow \mathsf{unit}^{\theta}_{\alpha}(\mathsf{discard}\ P\ \mathsf{in}\ N)])\end{array}$$

It is important to realise that the discussion earlier concerning the restrictions of the $\perp$-rules is relevant in the formulation of the commuting conversions. For example, a special case of the first commuting conversion is

$$(\mathsf{deunit}^{\phi\multimap\perp}_{\alpha}(M))N \rightsquigarrow_c M[\mathsf{unit}^{\phi\multimap\perp}_{\alpha}(P) \Leftarrow PN].$$

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

**Theorem 4.2** *If* $\Gamma \triangleright M{:}\,\phi^\bullet, \Sigma$ *and* $M \rightsquigarrow_{\beta,c} N$ *then* $\Gamma \triangleright N{:}\,\phi^\bullet, \Sigma$.

I conjecture that the properties of strong normalisation and confluence hold for this linear system.

My original motivation in devising this formulation was to study syntactically the process of cut elimination for **CLL**. This will be given in detail in the full version of this paper [5]. For now I shall show how the laws for the tensor unit, $I$, can be derived.

First the introduction rule can be derived as

$$\frac{x{:}\perp \triangleright x{:}\perp}{\triangleright \lambda x{:}\perp .x{:}\perp \multimap \perp}\ (\multimap_{\mathcal{I}})$$

and the elimination rule as

$$\frac{\dfrac{\Gamma \triangleright M{:}\,(\perp \multimap \perp)^\bullet, \Sigma \qquad \dfrac{\Delta \triangleright N{:}\,\phi^\bullet, \Sigma'}{\Delta \triangleright \mathsf{unit}^{\phi}_{\alpha}(N){:}\,\perp^\bullet, \phi^\alpha, \Sigma'}\ (\perp_{\mathcal{I}})}{\Gamma, \Delta \triangleright M(\mathsf{unit}^{\phi}_{\alpha}(N)){:}\,\perp^\bullet, \phi^\alpha, \Sigma, \Sigma'}\ (\multimap_{\mathcal{E}})}{\Gamma, \Delta \triangleright \mathsf{deunit}^{\phi}_{\alpha}(M(\mathsf{unit}^{\phi}_{\alpha}(N))){:}\,\phi^\bullet, \Sigma, \Sigma'}\ (\perp_{\mathcal{E}}).$$

The $\beta$-rule then holds as follows.

$$\begin{array}{rcl}\mathsf{let}\ *\ \mathsf{be}\ *\ \mathsf{in}\ N & \stackrel{\text{def}}{=} & \mathsf{deunit}^{\phi}_{\alpha}((\lambda x.x)(\mathsf{unit}^{\phi}_{\alpha}(N)))\\ & \rightsquigarrow_\beta & \mathsf{deunit}^{\phi}_{\alpha}(\mathsf{unit}^{\phi}_{\alpha}(N))\\ & \rightsquigarrow_\beta & N\end{array}$$

The commuting conversions associated with the $(I_{\mathcal{E}})$ rule [3, Figure 3.7] also translate correctly; below I give two examples.

$$\begin{aligned}(\mathsf{let}\ M\ \mathsf{be}\ *\ \mathsf{in}\ N)P &\stackrel{\text{def}}{=} (\mathsf{deunit}_\alpha^{\phi\multimap\psi}(M(\mathsf{unit}_\alpha^{\phi\multimap\psi}(N))))P\\ &\leadsto_c \mathsf{deunit}_\alpha^{\psi}(M(\mathsf{unit}_\alpha^{\psi}(NP)))\\ &\stackrel{\text{def}}{=} \mathsf{let}\ M\ \mathsf{be}\ *\ \mathsf{in}\ (NP)\end{aligned}$$

$$\begin{aligned}\mathsf{derelict}(\mathsf{let}\ M\ \mathsf{be}\ *\ \mathsf{in}\ N) &\stackrel{\text{def}}{=} \mathsf{derelict}(\mathsf{deunit}_\alpha^{!\phi}(M(\mathsf{unit}_\alpha^{!\phi}(N))))\\ &\leadsto_c \mathsf{deunit}_\alpha^{\phi}(M(\mathsf{unit}_\alpha^{\phi}(\mathsf{derelict}(N))))\\ &\stackrel{\text{def}}{=} \mathsf{let}\ M\ \mathsf{be}\ *\ \mathsf{in}\ \mathsf{derelict}(N)\end{aligned}$$

## 5 A Continuation-Passing Interpretation

In §3 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,

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

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

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

Here $\kappa$ is to be thought of as a *continuation variable.* 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 $\Sigma$ is now a multiset of continuation variables.) The $\bot_{\mathcal{E}}$ rule can similarly be rewritten as

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

In standard work in continuation-passing, e.g. [6], the non-local behaviour of evaluation is reflected by writing the reduction rules in context. Thus closed terms are evaluated in a context of the current environment. For this formulation there is an additional context which contains a multiset of labelled terms (the continuations). For example given a term

$$\rhd M\colon \phi^\bullet, \kappa_1\colon \sigma_1\multimap\bot, \ldots, \kappa_n\colon \sigma_n\multimap\bot,$$

we need a multiset of continuations $\mathcal{E} = [M_1, \ldots, M_n]$. Evaluation is then written as

$$\mathcal{E}[\![M]\!]E \Rightarrow M'$$

where $E$ is the current environment. The important evaluation rules are then

$$\mathcal{E} \dagger [\kappa\colon N][\![\kappa M]\!]E \Rightarrow \mathcal{E}[\![NM]\!]E,$$

$$\mathcal{E}[\![\mathsf{throw}^\kappa(M)]\!]E \Rightarrow \mathcal{E} \dagger [\kappa\colon E][\![M]\!]\mathsf{id};$$

where $\mathsf{id}$ is the identity continuation and $\mathcal{E} \dagger [k: N]$ denotes the extension of the continuation multiset $\mathcal{E}$ with $k: N$. Thus *Throw* captures the current environment and places it in the continuation multiset, labelled with $\kappa$. The *Catch* catches a continuation [4] from the multiset and replaces the continuation variable with the caught term.

The rather confusing formulation of the *Promotion* rule from §4 becomes slightly clearer with this continuation interpretation. The rule is rewritten as

$$\frac{\begin{array}{ll} \Gamma_1 \triangleright M_1 : !\phi_1^\bullet, \Sigma_1 & \Delta_1 \triangleright P_1 : !((!\sigma_1 \multimap \bot) \multimap \bot)^\bullet, \Upsilon_1 \\ \Gamma_n \triangleright M_n : !\phi_n^\bullet, \Sigma_n & \Delta_m \triangleright P_m : !((!\sigma_m \multimap \bot) \multimap \bot)^\bullet, \Upsilon_m \\ \multicolumn{2}{c}{x_1 : !\phi_1, \ldots, x_n : !\phi_n \triangleright N : \psi^\bullet, \kappa_1 : (!\sigma_1 \multimap \bot) \multimap \bot, \ldots, \kappa_m : (!\sigma_m \multimap \bot) \multimap \bot} \end{array}}{\vec{\Gamma}, \vec{\Delta} \triangleright \mathsf{promote}\ \vec{M} | \vec{P}\ \mathsf{for}\ \vec{x} | \vec{\kappa}\ \mathsf{in}\ N : !\psi^\bullet, \vec{\Sigma}, \vec{\Upsilon}} \ \textit{Promotion}.$$

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.

Of course this interpretation applies to **CL** in a similar way. In comparison to other works where authors have used continuation-passing work to explain classical logic (e.g. [1]), 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 [7] 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.

## 6 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 $\lambda$-calculus. I hope to have at least shed some light on its relationship with more traditional treatments of classical logic in natural deduction. I would claim that the resulting programming language is of more use than one based on proof nets. As mentioned earlier, proof nets rely on equivalent datatypes 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 classical linear $\lambda$-calculus there are explicit coercion terms.

In particular I would promote the computational interpretation suggested in §5 for both the linear and non-linear calculus. Although tentative, it promises a new programming language facility: multiple-continuation-passing, which unlike most proposals has an exact correspondence with a proof theory. This alone makes it worthy of further study.

[4] Linearity guarantees that the continuation exists.

A semantic study would also be desirable. Ong has proposed a categorical semantics and a class of game-theoretic models for **CL** based on Parigot's system. It would be interesting to see if a similar extension of linear categories [4] would produce some sort of $\star$-autonomous category.

## 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 $\pi$-calculus interpretation still remains work in progress). I should also like to thank Nick Benton for his continued support and critical advice. Nick Benton and Valeria de Paiva offered useful criticisms of an earlier draft of this paper.

## References

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

[2] P.N. Benton, G.M. Bierman, V.C.V. de Paiva, and J.M.E. Hyland. A term calculus for intuitionistic linear logic. In M. Bezem and J.F. Groote, editors, *Proceedings of First International Conference on Typed λ-calculi and applications*, volume 664 of *Lecture Notes in Computer Science*, pages 75–90, 1993.

[3] 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.

[4] 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.

[5] G.M. Bierman. A classical linear λ-calculus. Cambridge Computer Laboratory Technical Report, to appear, 1996.

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

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

[8] J.-Y. Girard, Y. Lafont, and P. Taylor. *Proofs and Types*, volume 7 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, 1989.

[9] C.-H.L. Ong. A semantic view of classical proofs: type-theoretic, categorical and denotational characterizations. To appear in LICS'96, December 1995.

[10] 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.

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

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

[13] D.J. Shoesmith and T.J. Smiley. *Multiple-Conclusion Logic.* Cambridge University Press, 1978.