LJQ: A Strongly Focused Calculus for Intuitionistic Logic*

Roy Dyckhoff and Stéphane Lengrand

School of Computer Science,
University of St Andrews,
St Andrews, Fife,
KY16 9SX, Scotland.
{rd,sl}@dcs.st-and.ac.uk

Abstract. LJQ is a focused sequent calculus for intuitionistic logic, with a simple restriction on the first premisss of the usual left introduction rule for implication. We discuss its history (going back to about 1950, or beyond), present the underlying theory and its applications both to terminating proof-search calculi and to call-by-value reduction in lambda calculus.

Keywords: sequent calculus, purification, call-by-value semantics, focused, depth-bounded, guarded logic

1 Introduction

Proof systems for intuitionistic logic close to natural deduction are well-known to be related to computation. For example, ordinary typed λ-calculus, with beta-reduction, is the classic model of computation for typed functional programmes with call-by-name (CBN) semantics; likewise, a system of uniform proofs for Horn logic is a coherent explanation of proof search in pure Prolog, as argued by (e.g.) [23]. The focused calculus **LJT** of Herbelin [17] (with antecedents in work [20], [3] by Joinet et al) is an intuitionistic sequent calculus equivalent to natural deduction (in the sense that its cut-free proofs are in a natural 1-1 correspondence with normal deductions); it also has a well-developed theory of proof-reduction with strong normalisation [17], [10], [12]. It can thus be seen to fulfil both these important roles, in being a basis both for proof search (where the proofs are of interest in themselves [9]) and for functional program evaluation with CBN-semantics. Work by the second author [21], [22] is developing the first of these ideas for a wide range of type theories.

The purpose of the present paper is to consider a different focused calculus **LJQ**, as named by Herbelin [16] and with similar antecedents [20], [3]. We

* Thanks are due to James McKinna and Christian Urban for useful suggestions, to Hugo Herbelin and Sara Negri for stimulating questions, to José Carlos Espírito Santo for his unpublished [14] and to Jörg Hudelmaier for a copy of his thesis [18].

present some aspects of its history and its applications both in structural proof theory and in λ-calculus, with connections in the first instance to automated reasoning and in the second to call-by-value programming language semantics. Fuller details will appear elsewhere.

Vorob'ev [34] (detailing ideas published in 1952) showed (Theorem 3) that, in a minor variant of Gentzen's calculus **LJ** for intuitionistic logic, one may, without losing completeness, restrict instances of the left rule $L\supset$ for implication to those in which, if the antecedent A of the principal formula $A \supset B$ is atomic, then the first premiss is an axiom. Independently, Hudelmaier [18] showed that one could further restrict this rule to those instances where the first premiss was either an axiom or the conclusion of a right rule; the result was proved in his [18] and described in [19] as folklore. The same result is mentioned by Herbelin in [16] as the completeness of a certain calculus **LJQ**, described simply as **LJ** with the last-mentioned restriction.

It is convenient to formalise such restrictions in terms of a sequent calculus **LJQ′** with two forms of sequent; letting Γ range over multisets of formulae, we have the *ordinary* sequent $\Gamma \Rightarrow A$ to express the deducibility of the formula A from assumptions Γ, and the *focused* sequent $\Gamma \rightarrow A$ to impose the restriction that the last step in the deduction is by an axiom or a right rule (i.e. with the succedent formula principal). A natural deduction interpretation is straightforward. Note that the focused sequent $p \vee q \rightarrow p \vee q$ is not derivable; the last step of its derivation can only be a right-introduction step.

The rules of the calculus are then as presented below, in Sect. 2. We use the name **LJQ′** rather than **LJQ** both to indicate the explicit focusing (use of two kinds of sequent) and the extra focusing (in the premisses of right rules for $\vee$ and $\wedge$). In later sections, when we consider a term calculus to represent derivations, we revert to the generic name **LJQ** for this kind of calculus.

For example, the rule $L\supset'$ has, as conclusion and second premiss, ordinary sequents, but as first premiss a focused sequent, capturing the restriction on proofs discussed by Hudelmaier (given that the focused sequents are exactly the axioms or the conclusions of right introduction rules). However, further restrictions are allowed: our right rules for disjunction and conjunction also have focused premisses. This represents a strengthening of Hudelmaier's folklore result.

LJQ as described in [16] originates in linear logic, in work by Danos *et al* [3] without mention of disjunction and conjunction. This in turn goes back to the thesis [20] of Joinet. Focusing itself is a technique pioneered by Andreoli [1] (but one of the points of our paper is a demonstration of its origins in much earlier work).

Such calculi are of interest not just because of the restricted proof search imposed by the focusing but because the completeness of **LJQ** (or of **LJQ′**) has as an easy corollary the completeness of more specialised "depth-bounded" calculi (as devised e.g. by [34], [18], [5]) in which proof search has limited (e.g. linear) depth (a.k.a. "height"); [33] gives a convenient account of the **G4ip** calculus, as it is there called.

These focused calculi are complementary to other focused calculi like Herbelin's **LJT**, as studied in [16], [17], [24], [32], [33], [9], [10], [12].

The present extended abstract outlines the theory (Sect. 2), presents some variations (Sect. 3), summarises some applications (Sect. 4), presents the calculus with term annotations (Sect. 5) (including a strongly normalising reduction system for **LJQ** and a preservation theorem relating **LJQ** to Moggi's calculus λ_C) and summarises some related work (joint with Delia Kesner: Sect. 6).

2 LJQ′

Basic syntactic conventions are as in [33]; in particular, P is a metavariable for "proposition variables" and Γ indicates a multiset of formulae. The symbols p and q are distinct proposition variables. The rules of **LJQ′** are as given below in Fig. 1.

$$\frac{}{\Gamma, \bot \Rightarrow A}\ L\bot \qquad \frac{\Gamma \to A}{\Gamma \Rightarrow A}\ Der \qquad \frac{}{\Gamma, P \to P}\ Ax$$

$$\frac{\Gamma, A{\supset}B \to A \quad \Gamma, B \Rightarrow C}{\Gamma, A{\supset}B \Rightarrow C}\ L{\supset}' \qquad \frac{A, \Gamma \Rightarrow B}{\Gamma \to A{\supset}B}\ R{\supset}'$$

$$\frac{\Gamma, A \Rightarrow C \quad \Gamma, B \Rightarrow C}{\Gamma, A \vee B \Rightarrow C}\ L{\vee}' \qquad \frac{\Gamma \to A_i}{\Gamma \to A_0 \vee A_1}\ R{\vee}'$$

$$\frac{\Gamma, A, B \Rightarrow C}{\Gamma, A \wedge B \Rightarrow C}\ L{\wedge}' \qquad \frac{\Gamma \to A \quad \Gamma \to B}{\Gamma \to A \wedge B}\ R{\wedge}'$$

Fig. 1. Rules of **LJQ′**

Expressed in terms of our notation, the right rule for conjunction in the calculus **LJQ** of [16] would be

$$\frac{\Gamma \Rightarrow A \quad \Gamma \Rightarrow B}{\Gamma \to A \wedge B}$$

and similarly for disjunction; the definition of "pure" derivations in [18] could be expressed in similar terms. The rule *Der* is named after the dereliction rule in linear logic; the latter rule has (used from conclusion to premiss) a similar effect, enabling a transition between a sequent where a certain formula is optional to one where it is required. The restriction to proposition variables P in Ax has the effect that the natural deduction interpretations of derivations are in long normal form. Use of arbitrary axioms $\Gamma, A \to A$ would give a different notion of derivability, e.g. $p \vee q \to p \vee q$ would be derivable.

A formula is *irreducible* when it is of the form P or $B{\supset}C$. To save space, proofs omit treatment of absurdity, conjunction and disjunction; details will appear in the full paper. Results as stated apply to the full calculus.

Lemma 1. *All sequents of the following form are derivable:*

1. $\Gamma, A, A \supset B \Rightarrow B$;
2. $\Gamma, A \rightarrow A$ *for irreducible* A;
3. $\Gamma, A \Rightarrow A$.

Proof: The three parts are handled by a simultaneous induction on the sizes of $A \supset B$, A and A respectively. Each part is allowed to depend on its predecessor (up to the same size) and on itself and its successors (at smaller sizes). □

The condition "for irreducible A" is needed once absurdity, disjunction and conjunction are included in the language; if we omit them all, then the condition can be omitted.

Weakening rules, some *Inversion* rules and *Contraction* rules are routinely shown to be admissible.

Theorem 1. *The following* Cut *rules are admissible:*

$$\frac{\Gamma \rightarrow A \quad A, \Gamma' \rightarrow B}{\Gamma, \Gamma' \rightarrow B}\ C_1 \qquad \frac{\Gamma \rightarrow A \quad A, \Gamma' \Rightarrow B}{\Gamma, \Gamma' \Rightarrow B}\ C_2 \qquad \frac{\Gamma \Rightarrow A \quad A, \Gamma' \Rightarrow B}{\Gamma, \Gamma' \Rightarrow B}\ C_3$$

Proof: Simultaneous induction on *cut rank* (size of cut formula A, height of first derivation d_1, height of second derivation d_2), with case analysis. □

Note that $p \rightarrow p$ and $p, p \supset q \Rightarrow q$ and $q \rightarrow q$ and $q \Rightarrow q$ are all derivable but that $p, p \supset q \rightarrow q$ is not derivable, hence the rules

$$\frac{\Gamma \rightarrow A \quad A, \Gamma' \Rightarrow B}{\Gamma, \Gamma' \rightarrow B} \qquad \frac{\Gamma \Rightarrow A \quad A, \Gamma' \rightarrow B}{\Gamma, \Gamma' \rightarrow B} \qquad \frac{\Gamma \Rightarrow A \quad A, \Gamma' \Rightarrow B}{\Gamma, \Gamma' \rightarrow B}$$

are not admissible.

Corollary 1. *The following rules are admissible:*

$$\frac{\Gamma \Rightarrow A \quad A, \Gamma' \rightarrow B}{\Gamma, \Gamma' \Rightarrow B}\ C_4 \qquad \frac{\Gamma \rightarrow A \quad A, \Gamma' \rightarrow B}{\Gamma, \Gamma' \Rightarrow B}\ C_5$$

Proof: Using Der. □

Corollary 2. *The following rules are admissible:*

$$\frac{\Gamma, A \Rightarrow B}{\Gamma \Rightarrow A \supset B}\ R\supset \qquad \frac{\Gamma, A \supset B \Rightarrow A \quad \Gamma, B \Rightarrow C}{\Gamma, A \supset B \Rightarrow C}\ L\supset$$

Proof: The first is derivable using $R\supset'$ and Der. The second can be achieved, using Lemma 1 for the premiss $A, A \supset B \Rightarrow B$, as

$$\dfrac{\dfrac{\Gamma, A \supset B \Rightarrow A \quad \dfrac{\dfrac{\ldots}{A, A \supset B \Rightarrow B} \quad \Gamma, B \Rightarrow C}{A, A \supset B, \Gamma \Rightarrow C}\ C_3}{\Gamma, A \supset B, \Gamma, A \supset B \Rightarrow C}\ C_3}{\Gamma, A \supset B \Rightarrow C}\ Contr$$

□

It follows from Corollary 2 and Lemma 1 (3) that this calculus $\mathbf{LJQ'}$ is as strong as **G3ip**. Since a derivation therein becomes a **G3ip** derivation if we ignore the distinction between the two kinds of sequent (and remove instances of Der), the two calculi are equivalent.

3 Variations

Several variations (and combinations of the variations) on the above are possible.

The first is to include the principal formula $A \supset B$ in the antecedent of the second premiss of $L\supset'$. This is preferable when we come to consider a term calculus, in Sect. 5 below; from the point of view of derivability it makes no difference.

The second removes the focusing from the premisses of the rules $R\wedge$ and $R\vee$ (this gives us the calculus **LJQ** of Herbelin [16]). Completeness of the calculus so modified is an immediate corollary of the completeness of **LJQ′**, since the focused versions (as we have presented them) are derivable using the unfocused versions and the *Der* rule.

The third is a multi-succedent version **LJQ*** (a variant of this appears in [16], page 78). We use two kinds of sequent as before; but this time, because of the need for a multiple succedent, we have a semi-colon to separate the focused formula (the *stoup*) from the rest of the succedent. The rules of **LJQ*** are as in Fig. 2 (− indicates an empty multi-set).

$$\frac{}{\Gamma, \bot \Rightarrow \Delta}\ L\bot^* \qquad \frac{\Gamma \rightarrow A; \Delta}{\Gamma \Rightarrow A, \Delta}\ Der^* \qquad \frac{}{\Gamma, P \rightarrow P; \Delta}\ Ax^*$$

$$\frac{\Gamma, A \supset B \rightarrow A; - \quad \Gamma, B \Rightarrow \Delta}{\Gamma, A \supset B \Rightarrow \Delta}\ L\supset^* \qquad \frac{A, \Gamma \Rightarrow B}{\Gamma \rightarrow A \supset B; \Delta}\ R\supset^*$$

$$\frac{\Gamma, A \Rightarrow \Delta \quad \Gamma, B \Rightarrow \Delta}{\Gamma, A \vee B \Rightarrow \Delta}\ L\vee^* \qquad \frac{\Gamma \Rightarrow A, B, \Delta}{\Gamma \rightarrow A \vee B; \Delta}\ R\vee^*$$

$$\frac{\Gamma, A, B \Rightarrow \Delta}{\Gamma, A \wedge B \Rightarrow \Delta}\ L\wedge^* \qquad \frac{\Gamma \rightarrow A; \Delta \quad \Gamma \rightarrow B; \Delta}{\Gamma \rightarrow A \wedge B; \Delta}\ R\wedge^*$$

Fig. 2. Rules of the multi-succedent calculus **LJQ***

The crucial *Cut* rules are

$$\frac{\Gamma \rightarrow A; \Delta \quad A, \Gamma' \rightarrow B; \Delta'}{\Gamma, \Gamma' \rightarrow B; \Delta, \Delta'}\ C_1 \qquad \frac{\Gamma \rightarrow A; \Delta \quad A, \Gamma' \Rightarrow \Delta'}{\Gamma, \Gamma' \Rightarrow \Delta, \Delta'}\ C_2$$

$$\frac{\Gamma \Rightarrow \Delta, A \quad A, \Gamma' \Rightarrow \Delta'}{\Gamma, \Gamma' \Rightarrow \Delta, \Delta'}\ C_3$$

and these are admissible by a routine argument similar to that already given.

4 Applications

4.1 Completeness of G4ip

The calculus **G4ip** was introduced by Hudelmaier [18], who gives a completeness proof that is essentially the following. For brevity we omit consideration of absurdity, conjunction and disjunction. Formulae are *weighted* as follows: $w(P) = 0$ and $w(A \supset B) = 1 + w(A) + w(B)$. Sequents $\Gamma \Rightarrow A$ are then ordered using the multi-set ordering (on the multiset Γ, A); in effect, this allows us to refer to the *weight* of a sequent. The rules are as follows.

$$\frac{}{P, \Gamma \Rightarrow P}\ Ax. \qquad\qquad \frac{A, \Gamma \Rightarrow B}{\Gamma \Rightarrow A \supset B}\ R\supset$$

$$\frac{P, B, \Gamma \Rightarrow E}{P, P \supset B, \Gamma \Rightarrow E}\ L0\supset \qquad \frac{D \supset B, C, \Gamma \Rightarrow D \quad B, \Gamma \Rightarrow E}{(C \supset D) \supset B, \Gamma \Rightarrow E}\ L\supset\supset$$

Note that every inference has as its conclusion a sequent with greater weight than each premiss; so root-first proof search is terminating, in a depth bounded by the weight of the sequent being proved.

Proposition 1 (Completeness of G4ip).

1. *If $\Gamma \rightarrow E$ is derivable in* **LJQ′**, *then $\Gamma \Rightarrow E$ is derivable in* **G4ip**.
2. *If $\Gamma \Rightarrow E$ is derivable in* **LJQ′**, *then $\Gamma \Rightarrow E$ is derivable in* **G4ip**.

Proof: By simultaneous induction on the sequent weight, using case analysis on the last step of the derivation. For (1), the last step is either an axiom (in which case we are done) or an $R\supset'$ inference, where the inductive hypothesis (2) can be used. For (2), the last step is either a dereliction, in which case (1) (for the same weight) is used, or an $L\supset'$ inference with principal formula $A \supset B$. In the latter case, if A is an atom P, then the first premiss is an axiom, with P in Γ; the inductive hypothesis (2) applied to the second premiss followed by an $L0\supset$-inference provides the required **G4ip** derivation. Otherwise, with $A = C \supset D$ and $\Gamma = (C \supset D) \supset B, \Gamma'$, the premisses are $\Gamma', (C \supset D) \supset B \rightarrow C \supset D$ and $\Gamma', B \Rightarrow E$. The inductive hypothesis (2) provides a **G4ip** derivation of $\Gamma', B \Rightarrow E$. The first premiss must be the conclusion from $\Gamma', (C \supset D) \supset B, C \Rightarrow D$, whose derivability in **LJQ′** easily implies that of the less weighty sequent $\Gamma', D \supset B, C \Rightarrow D$. The induction hypothesis (2) provides a **G4ip** derivation of this, and an $L\supset\supset$ inference provides a **G4ip** derivation of $\Gamma', (C \supset D) \supset B \Rightarrow E$. □

An almost identical argument, using **LJQ*** from Section 3, demonstrates the completeness of the multi-succedent version of **G4ip** in [5].

4.2 Completeness of Dragalin's GHPC

Dragalin [4] presented a multi-succedent sequent calculus **GHPC** for intuitionistic predicate logic, with the feature that the first premiss of the left rule for implication was single-succedent. This feature also appears in **LJQ*** in Section 3. An easy argument based on the completeness of the calculus **LJQ*** shows the completeness of **GHPC**; every inference (except by *Der*) in **LJQ*** becomes an inference in **GHPC**, and *Der* can be simulated by *Weakening* in **GHPC**.

4.3 Calculi for (Intuitionistic) Guarded Logic

Guarded first-order classical logic is of interest for its ability to interpret modal logics. The "guarded" restriction on formulae is that universal quantifiers are allowed only in the form $\forall \mathbf{x}(P \supset A)$, where $\mathbf{x}$ is a list of variables, P is an atom, A is a formula with $FV(A) \subseteq FV(P)$ and all the variables bound by the quantifier are free in P, i.e. *guarded* by the atom P; there is a similar restriction on existential quantifiers. No function symbols are allowed. In such a situation, the free variables in P (and hence in A) are a combination of those in $\mathbf{x}$ and possibly others. So, we indicate by $P\mathbf{xy}$ (resp. $A\mathbf{xy}$) an atom (resp. formula) all of whose free variables are in $\mathbf{x}, \mathbf{y}$ and by $P\mathbf{zy}$ (resp. $A\mathbf{zy}$) the result of substituting $\mathbf{z}$ for $\mathbf{x}$ therein. The notation $(\forall \mathbf{x}{:}P\mathbf{xy})A\mathbf{xy}$ then abbreviates $\forall \mathbf{x}(P\mathbf{xy} \supset A\mathbf{xy})$.

It is of interest to see whether this specialised form of quantification leads to a specialised inference rule. We treat this in the intuitionistic case; full details of this (treating also the existential quantifier) and the classical case are given in [11], including cut-admissibility proofs. The relevant inference rules are

$$\frac{P\mathbf{zy}, \Gamma, (\forall \mathbf{x}{:}P\mathbf{xy})A\mathbf{xy}, A\mathbf{zy} \Rightarrow B}{P\mathbf{zy}, \Gamma, (\forall \mathbf{x}{:}P\mathbf{xy})A\mathbf{xy} \Rightarrow B}\ L\forall' \qquad \frac{\Gamma, P\mathbf{zy} \Rightarrow A\mathbf{zy}}{\Gamma \Rightarrow (\forall \mathbf{x}{:}P\mathbf{xy})A\mathbf{xy}}\ R\forall'$$

where the variables $\mathbf{z}$ are *fresh* (i.e. disjoint from the free variables of $\Gamma, A\mathbf{xy}$) in the $R\forall'$ rule. We regard the atom $P\mathbf{zy}$ in the conclusion of the $L\forall'$ rule as a *key* that *unlocks* the guard on $A\mathbf{xy}$.

The first of these may be considered to be the composition of a standard $L\forall$ rule and a $L\supset$ rule, as in

$$\dfrac{\dfrac{\dfrac{}{P\mathbf{zy}, \Gamma, (\forall \mathbf{x}{:}P\mathbf{xy})A\mathbf{xy}, Pzy \supset A\mathbf{zy} \Rightarrow P\mathbf{zy}}\ Ax \qquad P\mathbf{zy}, \Gamma, (\forall \mathbf{x}{:}P\mathbf{xy})A\mathbf{xy}, A\mathbf{zy} \Rightarrow B}{P\mathbf{zy}, \Gamma, (\forall \mathbf{x}{:}P\mathbf{xy})A\mathbf{xy}, P\mathbf{zy} \supset A\mathbf{zy} \Rightarrow B}\ L\supset}{P\mathbf{zy}, \Gamma, (\forall \mathbf{x}{:}P\mathbf{xy})A\mathbf{xy} \Rightarrow B}\ L\forall$$

with the same restriction as in **LJQ** that the first premiss of the $L\supset$ inference have its succedent principal. Since this succedent is (by the guarded restriction) an atom $P\mathbf{zy}$, that means it must occur in the antecedent, as indicated. Thus, the **LJQ** restriction occurs also in this context, of intuitionistic guarded logic.

4.4 Negri's Conservativity Theorem

Negri [26] showed conservativity of the intuitionistic propositional theory of apartness over the theory of equality defined as the negation of apartness. The first complete proof used the calculus **G3ip** as basic; this was simplified in [27] once the completeness of the calculus **G4ip** (extended with rules for apartness) was demonstrated (in [8]). The use of **G4ip** was explained in [27] as "allowing a better control on derivations". In retrospect, it appears[1] that the use of the **LJQ** calculus would have sufficed.

5 LJQ with Terms

In this section we describe **LJQ** as the typing system of a term syntax, which we then use to establish a connection between **LJQ** and the call-by-value λ-calculus λ_C of Moggi [25]. For brevity, we consider in this section implication only, and the main distinction between **LJQ** and **LJQ**' can therefore be ignored; so, hereafter we just use the name **LJQ**.

5.1 A Term Calculus for LJQ

This term syntax is described as follows:

$$\begin{array}{ll} V, V' & ::= x \mid \lambda x.M \mid C_1(V, x.V') \\ M, N, P & ::= \uparrow V \mid x(V, y.N) \mid C_2(V, x.N) \mid C_3(M, x.N) \end{array}$$

The terms $C_i(-, -.-)$ are explicit substitutions, to be distinguished from the meta-notation $M\{x = N\}$ standing for "M with x replaced by N". Binding occurrences of variables are those immediately followed by ".". A term without any occurrence of a C_i is said to be *cut-free*. *Values* are cut-free terms of the form V.

$$\frac{}{\Gamma, x : A \rightarrow x : A}\ Ax \qquad \frac{\Gamma \rightarrow V : A}{\Gamma \Rightarrow\ \uparrow V : A}\ Der$$

$$\frac{\Gamma, x : A \Rightarrow M : B}{\Gamma \rightarrow \lambda x.M : A \supset B}\ R\supset' \qquad \frac{\Gamma, x : A \supset B \rightarrow V : A \quad \Gamma, x : A \supset B, y : B \Rightarrow N : C}{\Gamma, x : A \supset B \Rightarrow x(V, y.N) : C}\ L\supset'$$

$$\frac{\Gamma \rightarrow V : A \quad \Gamma, x : A \rightarrow V' : B}{\Gamma \rightarrow C_1(V, x.V') : B}\ C_1 \qquad \frac{\Gamma \rightarrow V : A \quad \Gamma, x : A \Rightarrow N : B}{\Gamma \Rightarrow C_2(V, x.N) : B}\ C_2$$

$$\frac{\Gamma \Rightarrow M : A \quad \Gamma, x : A \Rightarrow N : B}{\Gamma \Rightarrow C_3(M, x.N) : B}\ C_3$$

Fig. 3. **LJQ** with terms

[1] Personal communication from Sara Negri (Summer 2004)

The typing rules, shown in Fig. 3, are naturally derived from Fig. 1. Note that the rules Ax and $R\supset'$ with focused conclusions are those that type values. There are three changes, all more appropriate for the consideration of proof-terms.

The first change allows Ax to have an arbitrary formula as principal; by Lemma 1 this is acceptable in the implicational case.

The second change is that $L\supset'$ now allows the use of the formula $A\supset B$ in the proof of its second premiss, thus widening the space of proofs(-terms), as in Sect. 3. For instance, when establishing a connection with λ-calculus, this enables the proper representation of Church numerals; otherwise they would all (except 0) be mapped to the same proof-term.

The third change is that we include the cut rules as primitive; in contrast to those earlier, they are context-sharing (i.e. additive) rather than context-splitting (i.e. multiplicative or context-independent). This removes the need to formulate the admissibility of *Contraction* separately from the admissibility of cuts, the former being an easy sub-case of the latter.

(B)	$C_3(\uparrow \lambda x.M, y.y(V, z.P))$	$\longrightarrow$	$C_3(C_3(\uparrow V, x.M), z.P)$ if $y \notin FV(V) \cup FV(P)$
	$C_3(\uparrow x, y.N)$	$\longrightarrow$	$N\{y = x\}$
	$C_3(M, y.\uparrow y)$	$\longrightarrow$	M
	$C_3(z(V, y.P), x.N)$	$\longrightarrow$	$z(V, y.C_3(P, x.N))$
	$C_3(C_3(\uparrow V', y.y(V, z.P)), x.N)$	$\longrightarrow$	$C_3(\uparrow V', y.y(V, z.C_3(P, x.N)))$
			if $y \notin FV(V) \cup FV(P)$
	$C_3(C_3(M, y.P), x.N)$	$\longrightarrow$	$C_3(M, y.C_3(P, x.N))$
			if the redex is not one of the previous rule
	$C_3(\uparrow \lambda y.M, x.N)$	$\longrightarrow$	$C_2(\lambda y.M, x.N)$
			if N is not an x-covalue (see below)
	$C_1(V, x.x)$	$\longrightarrow$	V
	$C_1(V, x.y)$	$\longrightarrow$	y
	$C_1(V, x.\lambda y.M)$	$\longrightarrow$	$\lambda y.C_2(V, x.M)$
	$C_2(V, x.\uparrow V')$	$\longrightarrow$	$\uparrow C_1(V, x.V')$
	$C_2(V, x.x(V', z.P))$	$\longrightarrow$	$C_3(\uparrow V, x.x(C_1(V, x.V'), z.C_2(V, x.P)))$
	$C_2(V, x.x'(V', z.P))$	$\longrightarrow$	$x'(C_1(V', x.V), z.C_2(V, x.P))$
	$C_2(V, x.(C_3(M, y.P)))$	$\longrightarrow$	$C_3(C_2(V, x.M), y.C_2(V, x.P))$
η	$\lambda x.y(x, z.z)$	$\longrightarrow$	y

N is an *x-covalue* iff $N = \uparrow x$ or N is of the form $x(V, z.P)$ with $x \notin FV(V) \cup FV(P)$

Fig. 4. LJQ-reductions

The reduction rules for the calculus are shown in Fig. 4. This reduction system has the following properties:

1. It reduces any term that is not cut-free;
2. It satisfies the *Subject Reduction* property;
3. It is confluent;
4. It is *Strongly Normalising*;
5. A fortiori, it is *Weakly Normalising*.

As a corollary of 1, 2 and 5, we have the admissibility of *Cut*. It is interesting to see in the proof of *Subject Reduction* how these reductions transform the proof derivations and to compare them to those used in the proof of Theorem 1—details will be in the full paper. Apart from the differences between the inference rules already mentioned, there are also differences between the proof-transformations. The reduction system here is more subtle, because we are now interested not only in its weak normalisation but also in its strong normalisation and its connection with call-by-value λ-calculus.

The main reduction rule (B), breaking a cut on an implication into cuts on its direct sub-formulae, is now done with C_3 rather than with C_2. The reason is that we use C_3 to encode each β-redex of λ-calculus and C_2 to simulate the evaluation of its substitutions. Just as in λ-calculus, where substitutions can be pushed through β-redexes, so may C_2 be pushed through C_3, by use of the penultimate rule of Fig. 4 (which is not needed if the only concern is cut-admissibility).

Similarly, the last rule, (η), which has nothing to do with cut-elimination, is needed to account for η-conversion in (call-by-value) λ-calculus. It is interesting to see its meaning in proof theory: it generates an axiom on an implication, given a proof on the same sequent built from axioms on each of its direct sub-formulae, and then left and right introductions of the implication. In fact, recursive application of the reverse transform is precisely what is used to prove that one can safely restrict **LJQ** (in the implicational case) to atomic axioms.

5.2 Connection with Call-by-Value λ-Calculus

We will now be precise about what we call *CBV λ-calculus*. In [30], Plotkin introduces λ_V, a calculus whose terms are exactly those of Church's λ-calculus and whose reduction rule, called β_V, is merely β-reduction restricted to the case where the argument is a *value*, i.e. a variable (typed by an axiom) or an abstraction (typed by implication introduction).

However, the equational theory produced by β_V-conversion is shown [30] to be incomplete with respect to some canonical call-by-value semantics called *Continuation Passing Style*. Therefore, λ_V was later extended [25] to λ_C with a $\mathsf{let}\ .\ =\ \ldots\ \mathsf{in}\ \ldots$-construct (like our *Cut*-constructs for **LJQ**) and additional reduction rules; [31] shows, in effect, that the equational theory matches the CBV-semantics. Terms of λ_C are defined as follows:

$$M, N, P ::= x \mid \lambda x.M \mid M\ N \mid \mathsf{let}\ x = M\ \mathsf{in}\ N$$

We use V as a meta-variable ranging only over values. The reduction rules of λ_C are as follows:

$(\lambda x.M)\ V$	$\longrightarrow$	$M\{x = V\}$	
$\mathsf{let}\ x = V\ \mathsf{in}\ M$	$\longrightarrow$	$M\{x = V\}$	
$M\ N$	$\longrightarrow$	$\mathsf{let}\ x = M\ \mathsf{in}\ (x\ N)$	(M not a value)
$V\ N$	$\longrightarrow$	$\mathsf{let}\ y = N\ \mathsf{in}\ (V\ y)$	(N not a value)
$\mathsf{let}\ x = M\ \mathsf{in}\ x$	$\longrightarrow$	M	
$\mathsf{let}\ y = (\mathsf{let}\ x = M\ \mathsf{in}\ N)\ \mathsf{in}\ P$	$\longrightarrow$	$\mathsf{let}\ x = M\ \mathsf{in}\ (\mathsf{let}\ y = N\ \mathsf{in}\ P)$	

The reduction η_V can usefully be added: $\lambda x.(V\,x) \longrightarrow_{\eta_V} V \quad \text{if } x \notin FV(V)$
In the presence of β_V, the following rule has the same effect:

$$\lambda x.(y\,x) \longrightarrow_{\eta_V} y \qquad\qquad \text{if } x \neq y$$

We define the translation $.^\flat$ from **LJQ**-terms to λ_C by induction on the structure of terms:

$x^\flat$	$= x$
$(\lambda x.M)^\flat$	$= \lambda x.M^\flat$
$(\uparrow V)^\flat$	$= V^\flat$
$(x(V, y.M))^\flat$	$= \mathsf{let}\ y = x\ V^\flat\ \mathsf{in}\ M^\flat$
$(C_3(N, x.M))^\flat$	$= \mathsf{let}\ x = N^\flat\ \mathsf{in}\ M^\flat$
$(C_2(V, x.M))^\flat$	$= M^\flat\{x = V^\flat\}$
$(C_1(V, x.V'))^\flat$	$= V'^\flat\{x = V^\flat\}$

We define the translation $.^\sharp$ from λ_C to **LJQ** by a similar induction, using an auxiliary translation $.^\natural$ from values to values (a measure shows that the definitions are well-founded).

$x^\natural$	$= x$
$(\lambda x.M)^\natural$	$= \lambda x.M^\sharp$
$V^\sharp$	$= \uparrow V^\natural$
$(\mathsf{let}\ y = x\ V\ \mathsf{in}\ P)^\sharp$	$= x(V^\natural, y.P^\sharp)$
$(\mathsf{let}\ y = (\lambda x.M)\ V\ \mathsf{in}\ P)^\sharp$	$= C_3(\lambda x.M^\sharp, z.z(V^\natural, y.P^\sharp))$
$(\mathsf{let}\ z = V\ N\ \mathsf{in}\ P)^\sharp$	$= (\mathsf{let}\ y = N\ \mathsf{in}\ (\mathsf{let}\ z = V\ y\ \mathsf{in}\ P))^\sharp$ if N is not a value
$(\mathsf{let}\ z = M\ N\ \mathsf{in}\ P)^\sharp$	$= (\mathsf{let}\ x = M\ \mathsf{in}\ (\mathsf{let}\ z = x\ N\ \mathsf{in}\ P))^\sharp$ if M is not a value
$(\mathsf{let}\ z = (\mathsf{let}\ x = M\ \mathsf{in}\ N)\ \mathsf{in}\ P)^\sharp$	$= (\mathsf{let}\ x = M\ \mathsf{in}\ (\mathsf{let}\ z = N\ \mathsf{in}\ P))^\sharp$
$(\mathsf{let}\ y = V\ \mathsf{in}\ P)^\sharp$	$= C_3(V^\sharp, y.P^\sharp)$
$(M\ N)^\sharp$	$= (\mathsf{let}\ y = M\ N\ \mathsf{in}\ y)^\sharp$

Notice that if M is a C_1/C_2-free term of **LJQ**, $M^{\flat\sharp} = M$ and that for any term M of λ_C, $M \longleftrightarrow^* M^{\sharp\flat}$. Now we can state (using $\longrightarrow^*$ for the reflexive transitive closure of $\longrightarrow$, etc) the following:

Theorem 2 (Preservation Theorem).

1. *For any terms M and N of λ_C, if $M \longrightarrow N$ then $M^\sharp \longrightarrow^* N^\sharp$.*
2. *For any terms M and N of* **LJQ***, $M \longleftrightarrow^* N$ iff $M^\flat \longleftrightarrow^* N^\flat$.*

Hence, if a term M of **LJQ** is given the CBV-semantics of $M^\flat$, **LJQ** inherits from λ_C a semantics that captures exactly its equational theory.

Ongoing work includes refining the connection above and generalising it to a framework that would also account for the call-by-name discipline, by using a calculus introduced by Espírito Santo [14].

6 G4ip with Terms

Bringing some of the above ideas together, we can regard **G4ip** itself as the typing system for a term calculus. The associated reduction system for cuts also has the strong normalisation property. Details are in [6]. The main point of interest is the avoidance of auxiliary operations (corresponding to admissibility lemmas) in favour of uses of instances of the explicit substitution operation.

7 Conclusion

We have presented, proved complete and shown some applications of a strongly focused calculus $\mathbf{LJQ}'$, incorporating and extending the restrictions on derivations explicit in the calculus **LJQ** of Herbelin [16], implicit in the work on "purification" in Hudelmaier [18] and with early traces in the work of Vorob'ev [34]. These applications range from sequent calculi for automated proof search to CBV-semantics of λ-calculus.

References

1. J.-M. Andreoli. *Logic programming with focusing proofs in linear logic*, J. Logic & Computation **2**, 297–347, 1992.
2. B. Cooper, J. Truss. "Sets and Proofs", (Proceedings of Logic Colloquium 97), Cambridge University Press, 1999.
3. V. Danos, J.-B. Joinet, H. Schellinx. *LKQ and LKT: sequent calculi for second order logic based upon dual linear decompositions of classical implication*, in [15], pp 211–224.
4. A. G. Dragalin. "Mathematical Intuitionism", Translations of Mathematical Monographs, **67**, Amer. Math. Soc., Providence, Rhode Island, 1988.
5. R. Dyckhoff. *Contraction-free sequent calculi for intuitionistic logic*, J. Symbolic Logic **57**, 795–807, 1992.
6. R. Dyckhoff, S. Lengrand, D. Kesner. *Strong cut-elimination systems for Hudelmaiers depth-bounded sequent calculus for implicational logic*, submitted, 2006.
7. R. Dyckhoff, S. Negri. *Admissibility of structural rules for contraction-free systems of intuitionistic logic*, J. Symbolic Logic **65**, 1499–1518, 2000.
8. R. Dyckhoff, S. Negri. *Admissibility of structural rules for extensions of contraction-free sequent calculi*, Logic J. of the IGPL **9**, 573–580, 2001
9. R. Dyckhoff, L. Pinto. *Proof search in constructive logics*, in [2], 53–65.
10. R. Dyckhoff, L. Pinto. *Cut-elimination and a permutation-free sequent calculus for intuitionistic logic*, Studia Logica **60**, 107–118, 1998.
11. R. Dyckhoff, A. Simpson. *Proof theory of guarded logics*, MS, 2004.
12. R. Dyckhoff, C. Urban. *Strong normalization of Herbelin's explicit substitution calculus with substitution propagation*, J. Logic & Comput. **13**, 689–706, 2003.
13. R. Dyckhoff. *Variations on a theme of Hudelmaier*, MS, 2006.
14. J. Espírito Santo. *Unity in structural proof theory and structural extensions of the λ-calculus*, MS available from `http://www.math.uminho.pt/~jes/Publications.htm`, Jul 2005.

15. J.-Y. Girard, Y. Lafont, L. Regnier (eds). “Advances in Linear Logic”, LMS Lecture Note Series **222**, Cambridge University Press, 1995.
16. H. Herbelin. *Séquents qu’on calcule*, Thèse de Doctorat, Université Paris 7, 1995.
17. H. Herbelin. *A λ-calculus structure isomorphic to sequent calculus structure*, in [29], 67–75.
18. J. Hudelmaier. *Bounds for cut-elimination in intuitionistic propositional logic*, PhD thesis, Tübingen, 1989.
19. J. Hudelmaier. *An $O(n\ log(n))$ decision procedure for intuitionistic propositional logic*, J. of Logic & Computation **3**, 63–75, 1993.
20. J.-B. Joinet. *Étude de la normalisation du calcul des séquents classique à travers la logique linéaire*, Thèse de Doctorat, Université Paris 7, 1993.
21. S. Lengrand. *Normalisation and equivalence in proof theory and type theory*, Draft PhD thesis (Paris 7 & St Andrews), 2006.
22. S. Lengrand, R. Dyckhoff. *Type theory in sequent calculus*, submitted, 2006.
23. D. Miller, G. Nadathur, F. Pfenning, A. Scedrov. *Uniform proofs as a foundation for logic programming*, Ann. Pure Appl. Log. **91**, 125–157, 1991.
24. G. Mints. *Normal forms for sequent derivations*, in [28], 469–492.
25. E. Moggi. *Computational lambda-calculus and monads*, Report ECS-LFCS-88-66, University of Edinburgh, 1988.
26. S. Negri. *Sequent calculus proof theory of intuitionistic apartness and order relations*, Arch. Math. Logic **38**, 521–547, 1999.
27. S. Negri. *Conservativity of apartness over equality, revisited*, Research Report CS/99/4, University of St Andrews, 6 pp, 1999.
28. P. Odifreddi (ed). “Kreiseliana”, A. K. Peters, Wellesley, Mass., 1996.
29. L. Pacholski, J. Tiuryn (eds). “Proceedings of Computer Science Logic 1994”, LNCS **933**, Springer, 1995.
30. G. Plotkin. *Call-by-name, call-by-value and the lambda-calculus*, Theoret. Comput. Sci. **1**, 125–159, 1975.
31. A. Sabry, M. Felleisen. *Reasoning about programs in continuation-passing style*, Lisp and Symb. Comput. **6**, 289–360, 1993.
32. A. S. Troelstra, *Marginalia on sequent calculi*, Studia Logica **62**, 291–303, 1999.
33. A. S. Troelstra, H. Schwichtenberg. “Basic Proof Theory”, Cambridge University Press, 2nd ed., 2000.
34. N. N. Vorob’ev. *A new algorithm for derivability in the constructive propositional calculus*, Amer. Math. Soc. Translations, ser. 2, vol. 94, 37–71, 1970.