Novi Sad J. Math.
Vol. 29, No. 2 (1999), 209–220
Proc. VIII Int. Conf.
"Algebra & Logic" (Novi Sad, 1998)

NATURAL DEDUCTION AND SEQUENT TYPED LAMBDA CALCULUS

Silvia Ghilezan
Faculty of Engineering, University of Novi Sad
Trg Dositeja Obradovića 6, 21000 Novi Sad, Yugoslavia
e-mail: *gsilvia@uns.ns.ac.yu*

Abstract

Two different formulations of the simply typed lambda calculus: the natural deduction and the sequent system, are considered. An analogue of cut elimination is proved for the sequent lambda calculus.

AMS Math. Subject Classification (1991): 03B40, 03F05
Key words and phrases: cut elimination, normalization, lambda calculus

1. Introduction

Natural deduction and sequent systems of the simply typed lambda calculus are the subject of our investigation. The usual way of giving the simply typed lambda calculus $\lambda\!\to$ is with explicit typing (Church version) and with implicit typing (Curry version). Both ways can be seen as *natural deduction systems*, since they consist of elimination and introduction rules for implication. There is a *sequent system* formulation of the simply typed lambda calculus $L\lambda\!\to$ given in Pottinger [6]. It is based on the symmetry of introducing implication both on the left-hand and right-hand side of the turnstile in the sequent. These two systems $\lambda\!\to$ and $L\lambda\!\to$ are used in Pottinger [6] in order to study the correspondence between normalization procedures in the natural deduction system of intuitionistic propositional logic, and the cut-elimination procedures in the sequent system of intuitionistic propositional logic. The main new feature in $L\lambda\!\rightarrow$ is an analogue of the cut rule which involves, in addition to usual cuts of types (formulae), the cuts of terms as well, which are actually substitutions of lambda terms. In the sequel we call this rule *term-cut rule* as opposed to the cut rule which is a logical rule.

The problem of *inhabitation* in a type system is whether there exists a term of a given type in the system considered. By the Curry–Howard correspondence types inhabited in the simply typed lambda calculus coincide with the formulae provable in the implicational fragment of intuitionistic logic. In that sense, terms encode proofs of formulae corresponding to their types. This correspondence can be extended to intuitionistic logic and simply typed lambda calculus with pairing, projections and constants for disjunction and negation.

Strictly speaking, the Curry–Howard correspondence holds only for natural deduction systems of logic and lambda calculus, since only then it is not just a one-to-one correspondence between provable formulae and inhabited types (*formulae-as-types*), but is a one-to-one correspondence between derivations and lambda terms (*proofs-as-terms*). The latter does not hold for sequent systems of logic and lambda calculus since one term can encode different proofs of the same sequent. In this sense the axiomatic system of logic corresponds to the combinatory logic, but they are out of the scope of our study.

Section 2 is an overview of the natural deduction system $\lambda\!\rightarrow$ and the sequent system $L\lambda\!\rightarrow$ of the simply typed lambda calculus and their equivalence. In Section 3, beside the well known Curry–Howard correspondence between natural deduction systems of logic and lambda calculus, the Curry–Howard correspondence between the sequent systems of logic and the lambda calculus is investigated. The difference between these two links is pointed out. In Section 4, term-cut elimination for sequent simply typed lambda calculus is proved, which is an analogue of cut elimination in logic. Section 5 is a discussion about the related work.

2. Natural deduction and sequent system of simply typed lambda calculus

Natural deduction systems consist of elimination and introduction rules for logical connectives. Untyped lambda calculus can be regarded as a natural deduction system, where the application and abstraction correspond to elimination and introduction rules, respectively. Let $V = \{x, y, z, x_1, \ldots\}$ be a denumerable set of variables.
The axiom-scheme is

$$(ax) \quad x;$$

the elimination rule is

$$(app) \quad \frac{M \quad N}{MN};$$

and the introduction rule is

$$(abs) \quad \frac{M}{(\lambda x.M)}.$$

Let us recall some basic notions of the simply typed lambda calculus of the well known $\lambda\!\rightarrow$ and of its less familiar formulation $L\lambda\!\rightarrow$.

Terms are usual untyped lambda terms. We use $M, N, P, \ldots$ as schematic letters for terms. *Types* are implicational propositional formulae, where $\rightarrow$ is the only type forming operator. We use $\varphi, \sigma, \tau, \ldots$ as schematic letters for types.

The expression $M : \sigma$, called *statement*, where M is a term and σ is a type, links terms and types. M is the *subject* and σ is the *predicate* of the statement $M : \sigma$. If x is a variable, then $x : \tau$ is a *basic statement.* A *context (basis)* is a set of basic statements, with different term variables. Contexts are denoted by $\Gamma, \Delta, \ldots$ and Γ, Δ denotes the set-theoretic union of Γ and Δ. *Freedom* of variables is defined in the usual way. *Substitution* $M[N/x]$ is defined to be the term obtained from the term M by replacing every free occurrence of the variable x in M by the term N, provided that there is no free variable y in N such that x falls within a subterm of M of the form $\lambda y.M_1$.

The standard way of defining the simply typed lambda calculus $\lambda\!\rightarrow$ is by a natural deduction system of elimination and introduction rules. This refers to both versions, to the Church version and the Curry version of the simply typed lambda calculus. There is:

the axiom-scheme

$$(ax) \quad \Gamma, x:\sigma \vdash x:\sigma;$$

the elimination rule

$$(\to E) \quad \frac{\Gamma \vdash M:\sigma \to \tau \quad \Delta \vdash N:\sigma}{\Gamma, \Delta \vdash MN:\tau};$$

and the introduction rule

$$(\to I) \quad \frac{\Gamma, x:\sigma \vdash M:\tau}{\Gamma \vdash (\lambda x.M):\sigma \to \tau}.$$

This is the Curry version of the simply typed lambda calculus. The Church version is also a natural deduction system, which will not be discussed in detail here.

The *sequent simply typed lambda calculus* $L\lambda{\to}$ is given by:
the same axiom-scheme

$$(ax) \quad \Gamma, x:\sigma \vdash x:\sigma;$$

the left introduction rule

$$(\to L) \quad \frac{\Gamma \vdash N:\tau \quad \Delta, x:\rho \vdash M:\sigma}{\Delta, y:\tau \to \rho, \Gamma \vdash M[yN/x]:\sigma}, \text{ (y is fresh for Γ and Δ)};$$

the right introduction rule

$$(\to R) \quad \frac{\Gamma, x:\sigma \vdash M:\tau}{\Gamma \vdash (\lambda x.M):\sigma \to \tau};$$

and the term-cut rule

$$(term - cut) \quad \frac{\Delta \vdash N:\sigma \quad \Gamma, x:\sigma \vdash M:\tau}{\Gamma, \Delta \vdash M[N/x]:\tau}.$$

As we may notice the term-cut rule involves, beside usual cuts of types, cuts of terms as well, which are actually substitutions of terms.

Remark. Untyped lambda calculus is a natural deduction system, as we noticed above. $\lambda{\to}$ is a complete natural deduction system since both terms and types are introduced as natural deduction systems. In spite of introducing types in $L\lambda{\to}$ in a sequent system manner, terms inherit their natural deduction origin. Therefore $L\lambda\!\rightarrow$ is a mixture of the natural deduction term system and the sequent type system.

Let $\Gamma \vdash_N M : \sigma$ and $\Gamma \vdash_L M : \sigma$ denote the derivability of the statement $\Gamma \vdash M : \sigma$ in $\lambda\!\rightarrow$ and $L\lambda\!\rightarrow$, respectively. The following theorem shows the equivalence of the two formulations mentioned, in the sense that the derivability of a statement in one system implies its derivability in the other as well, and the other way round.

Theorem 1. *For every context* Γ, *term* M *and type* σ

$$\Gamma \vdash_N M : \sigma \text{ if and only if } \Gamma \vdash_L M : \sigma.$$

Proof. Both directions are proved by induction on the derivation.

($\Rightarrow$) The interesting case is when the last rule applied in the natural deduction system is ($\rightarrow E$)

$$\frac{\Gamma \vdash_N P : \rho \rightarrow \tau \quad \Delta \vdash_N Q : \rho}{\Gamma, \Delta \vdash_N PQ : \tau},$$

then we have the following derivation in the sequent system

$$\frac{\Gamma \vdash_L P : \rho \rightarrow \tau \qquad \frac{\Delta \vdash_L Q : \rho \quad x : \tau \vdash_L x : \tau}{\Delta, y : \rho \rightarrow \tau \vdash_L yQ : \tau}}{\Gamma, \Delta \vdash_L PQ : \tau}.$$

($\Leftarrow$) If the last step in the sequent system derivation is ($\rightarrow L$)

$$\frac{\Gamma \vdash_L P : \tau \quad \Delta, x : \rho \vdash_L Q : \sigma}{\Delta, y : \tau \rightarrow \rho, \Gamma \vdash_L Q[yP/x] : \sigma},$$

then we have the following in the natural deduction system

$$\frac{\Gamma \vdash_N P : \tau \quad y : \tau \rightarrow \rho \vdash_N y : \tau \rightarrow \rho}{\Gamma, y : \tau \rightarrow \rho \vdash_N yP : \rho} \; (\rightarrow E)$$

and $\Delta, y : \tau \rightarrow \rho, \Gamma \vdash_N Q[yP/x] : \sigma$ is obtained from $\Delta, x : \rho \vdash_N Q : \sigma$ by substitution. $\square$

3. Curry–Howard N-correspondence and L-correspondence

The two different formulations of the simply typed lambda calculus, $\lambda\!\rightarrow$ and $L\lambda\!\rightarrow$ can be regarded as two different formulations of intuitionistic logic, natural deduction and sequent system, with encodings of natural deduction and sequent derivations, respectively, denoted explicitly by lambda terms.

The *Curry–Howard correspondence* establishes the relation between two completely natural deduction systems: $\lambda\rightarrow$ and intuitionistic logic. We shall call it the *Curry–Howard N-correspondence* The explicit definition of the Curry–Howard N-correspondence can be found for example in Girard et al. [2].

1. To the deduction α corresponds $x : \alpha \vdash_N x : \alpha$.

2. To the deduction
$$\begin{array}{c}[\sigma]\\ \vdots \\ \dfrac{\tau}{\sigma\rightarrow\tau}\end{array}$$
corresponds $\Gamma \vdash_N \lambda x.M : \sigma \rightarrow \tau$, if $\Gamma, x : \sigma \vdash_N M : \tau$ corresponds to the deduction of τ from σ.

3. To the deduction
$$\frac{\sigma \quad \sigma\rightarrow\tau}{\tau}\ (MP)$$
corresponds $\Gamma, \Delta \vdash_N MN : \tau$, if $\Gamma \vdash_N M : \sigma \rightarrow \tau$ corresponds to the deduction of $\sigma \rightarrow \tau$ and $\Delta \vdash_N N : \sigma$ corresponds to the deduction of σ.

The Curry–Howard N-correspondence is based on pasting together the steps in a logical deduction with the steps in the construction of the encoding lambda term. The following classical result is due to Curry and Howard [4].

Theorem 2. *Let σ be a type. There is a term M and a context*

$$\Gamma = \{x_1 : \sigma_1, \ldots x_n : \sigma_n\}$$

such that

$\Gamma \vdash_N M : \sigma$ iff σ is provable (intuitionistically) from the assumptions $\sigma_1, \ldots, \sigma_n$.

The problem of inhabitation in a type system is whether there exists a term of a given type in the system considered. By the Curry–Howard N-correspondence, types inhabited in the simply typed lambda calculus coincide with formulae provable in the implicational fragment of intuitionistic logic. Apparently, it is a *formulae-as-types* correspondence. On the other hand, it is a *proofs-as-terms* correspondence as well, since terms encode directly proofs of formulae corresponding to their types, i.e., each step in the logical deduction changes the term. In this case terms are treated as "good book-keeping devices"(Pottinger [6]) of proofs, whereas the type of the term obtained is exactly the formula proved.

The situation changes with sequent systems since terms remain book-keeping devices, but not so neat anymore. The *Curry–Howard L-correspondence* between the sequent calculus formulation of intuitionistic logic and the simply typed lambda calculus can be given in the following way.

1. To the derivation of the sequent $\alpha \vdash \alpha$ corresponds $x : \alpha \vdash_L x : \alpha$

2. To the derivation of the sequent in the conclusion

$$\frac{\Gamma' \vdash \tau \quad \Delta', \rho \vdash \sigma}{\Delta', \tau \to \rho, \Gamma' \vdash \sigma} \ (\to L)$$

   corresponds

$$\Delta, y : \tau \to \rho, \Gamma \vdash_L M[yN/x] : \sigma,$$

   if $\Gamma \vdash_L N : \tau$ and $\Delta, x : \rho \vdash_L M : \sigma$ correspond to the premisses and if y is fresh for Γ and Δ . (The contexts Γ and Δ are obtained from Γ' and Δ', respectively, by creating basic statements $x : \varphi$ from every formula of Γ' and Δ', taking into account that all term variables of a context have to be different.)

3. To the derivation of the sequent in the conclusion

$$\frac{\Gamma', \sigma \vdash \tau}{\Gamma' \vdash \sigma \to \tau} \ (\to R)$$

   corresponds $\Gamma \vdash_L (\lambda x.M) : \sigma \to \tau$, if $\Gamma, x : \sigma \vdash_L M : \tau$ corresponds to the derivation of the sequent in the premiss. (The context Γ is obtained from Γ' as in the previous case.)

4. To the derivation of the sequent in the conclusion

$$\frac{\Delta' \vdash \sigma \quad \Gamma', \sigma \vdash \tau}{\Gamma', \Delta' \vdash \tau}$$

corresponds $\Gamma, \Delta \vdash_L M[N/x] : \tau$, if $\Delta \vdash_L N : \sigma$ and $\Gamma, x : \sigma \vdash_L M : \tau$ correspond to the derivations of the premisses, respectively.

The previous theorem holds for the L-correspondence as well, in the sense that the inhabited types coincide with provable formulae.

Theorem 3. *Let σ be a type. There is a term M and a context*

$$\Gamma = \{x_1 : \sigma_1, \ldots x_n : \sigma_n\}$$

such that

$\Gamma \vdash_L M : \sigma$ *iff σ is provable (intuitionistically) from the assumptions* $\sigma_1, \ldots, \sigma_n$.

Hence the L-correspondence preserves formulae-as-types, but it does not preserve proofs-as-terms. For example, $x : \alpha \vdash x : \alpha$ corresponds to both proofs

$$\alpha \vdash \alpha \quad \text{and} \quad \frac{\alpha \vdash \alpha \quad \alpha \vdash \alpha}{\alpha \vdash \alpha}.$$

One of the reasons for this can be found in the application of the term-cut rule of $L\lambda\rightarrow$. Obviously, there are cases of the term-cut rule application which do not change the term from the premiss. These cases are when the term N is a variable, say x, i.e., the left premiss in the (term-cut) rule is $x : \sigma \vdash x : \sigma$. Let us recall that, in logic, (cut)-applications in which the left premiss is of the form $\sigma \vdash \sigma$ are called *trivial*. The other reason is the interchangeability of the rules $(\rightarrow R)$ and $(term - cut)$ which does not offend the structure of the term.

Therefore, the Curry–Howard L-correspondence pastes together the steps in a logical derivation with the steps in the construction of the encoding lambda term ignoring:

- trivial term-cut applications,
- the order in which abstractions in $(\rightarrow R)$ and substitutions in $(term - cut)$ are applied.

- the order in which substitutions in $(\rightarrow L)$ and substitutions in (*term–cut*) are applied.
- the order in which abstractions in $(\rightarrow R)$ and substitutions in $(\rightarrow L)$ are applied.

This mismatch between proofs in the sequent system of intuitionistic logic and their encoding terms in $L\lambda\!\rightarrow$ arises from the fact that the considered logic is a sequent system, whereas the term part of $L\lambda\!\rightarrow$ which encodes the proofs is natural deduction, as we noticed in the previous remark. Nevertheless, the L-correspondence is one-to-one between provable formulae in the sequent system of logic and inhabited types in $L\lambda\!\rightarrow$ since the type part of $L\lambda\!\rightarrow$ is a sequent system as well.

4. Term-cut elimination

We shall show for sequent lambda calculus $L\lambda\!\rightarrow$ an analogue of the cut elimination property in the usual manner (see Lambek [5]) by showing that for each statement derivable by the application of the (term-cut) rule there is a corresponding statement with the same predicate (type) derivable without it. The addition that we have here is that the subject (term) of the latter statement is the normal form of the subject (term) of the former statement. We say 'an analogue', since usually in proving cut elimination we are concerned with formulae (types) only and we do not care formally about the changes in the proofs (terms). In $L\lambda\!\rightarrow$ the proofs are encoded directly by lambda terms, so these changes, which have not been taken into account up to now, will become explicit.

For the sake of simplicity, derivability in $L\lambda\!\rightarrow\ \Gamma \vdash_L M : \sigma$ will be denoted by $\Gamma \vdash M : \sigma$, there is no place for confusion since $\lambda\!\rightarrow$ is not mentioned in the rest of the section.

In order to prove term-cut elimination we need the following usual notions:

- $d(\varphi)$, the number of occurrences of the connective $\rightarrow$ in φ;
- if $\Gamma, x : \varphi$ is a context, then $d(\Gamma, x : \varphi) = d(\Gamma) + d(\varphi)$;
- the *degree* of (term-cut)

$$\frac{\Delta \vdash N : \varphi \quad \Gamma, x : \varphi \vdash M : \sigma}{\Gamma, \Delta \vdash M[N/x] : \sigma}$$

is the total number of occurrences of the connective $\to$ in the types of Γ and Δ and in φ and σ, i.e. it is $d(\Gamma) + d(\varphi) + d(\Delta) + d(\sigma)$;

- the *main connectives* are the connectives appearing in the formula φ which is eliminated by (term-cut), and φ is called the *term-cut-formula*;

- a derivation is called *term-cut-free* if its inductive construction involves no applications of (term-cut).

We need the notion of the normal lambda term. For that reason let us recall the notion of reduction in the lambda calculus. The axiom of $\beta-$reduction is $(\lambda x.M)N \to M[N/x]$. A term is a *normal form* if there are no more $\beta-$reductions to be performed, i.e., if it does not contain any subterm of the form $(\lambda x.M)N$. It is well known that the general shape of a normal form is $\lambda x_1 \dots x_k.yN_1 \dots N_l$, where y may, but need not, be one of the variables x_i, $i \in \{1, \dots, k\}$ and moreover the terms N_j, $j \in \{1, \dots, l\}$ are normal forms.

First, let us notice that the term in the conclusion of any cut-free derivation is a normal form.

Lemma 1. *If there is a term-cut-free derivation of* $\Gamma \vdash M : \sigma$*, then* M *is a normal form.*

Proof. Easy by induction on a cut-free derivation of $\Gamma \vdash M : \sigma$. The obvious cases are when the last rule applied is the (ax) and $(\to R)$. The case $(\to L)$ also ensures that if the terms N and M from its premisses are normal forms, then the term $M[yN/x]$ created in the conclusion by the substitution of yN for each free occurrence of the variable x in M is a normal form, since yN is a normal form and this substitution cannot create new reductions. □

Now, we can prove the cut elimination property.

Theorem 4. (Term-cut elimination) *Every derivation of a statement* $\Gamma \vdash M : \sigma$ *is associated with a term-cut-free derivation of a corresponding statement* $\Gamma \vdash N : \sigma$*, where* N *is the normal form of* M*.*

Proof. It suffices to prove, as usual, that every application of (term-cut) whose premisses are term-cut-free may be replaced by one or more (term-cut) applications of smaller degree, provided that the final term in the replaced derivation is obtained from the original term by $\beta-$reduction. Then by iterating this property and applying Lemma 1 we obtain the term-cut elimination theorem. The uniqueness of the normal form N follows by the well-known Church-Rosser property.

Proof by induction on the degree of (term-cut). If the rules applied before the application of (term-cut) do not introduce a main connective, then the statement is obvious.

The interesting case is when the rules applied before (term-cut) introduce a main connective: $(\to R)$ and $(\to L)$. The following (term-cut) application

$$\frac{\dfrac{\Delta, y:\varphi_1 \vdash P:\varphi_2}{\Delta \vdash \lambda y.P:\varphi_1 \to \varphi_2} \qquad \dfrac{\Gamma, z:\varphi_2 \vdash M:\sigma \quad \Gamma' \vdash_{\leq} Q:\varphi_1}{\Gamma, x:\varphi_1 \to \varphi_2, \Gamma' \vdash M[xQ/z]:\sigma}}{\Gamma, \Delta, \Gamma' \vdash M[xQ/z][\lambda y.P/x]:\sigma}$$

of degree $d(\Gamma, \Gamma') + d(\varphi_1 \to \varphi_2) + d(\Delta) + d(\sigma)$ can be replaced by

$$\frac{\Gamma' \vdash Q:\varphi_1 \quad \dfrac{\Delta, y:\varphi_1 \vdash P:\varphi_2 \quad \Gamma, z:\varphi_2 \vdash M:\sigma}{\Gamma, \Delta, y:\varphi_1 \vdash M[P/z]:\sigma}}{\Gamma, \Delta, \Gamma' \vdash M[P/z][Q/y]:\sigma}$$

two applications of (term-cut) of smaller degree $d(\Gamma) + d(\varphi_2) + d(\Delta, y : \varphi_1) + d(\sigma)$ and $d(\Gamma, \Delta) + d(\varphi_1) + d(\Gamma') + d(\sigma)$.

Let us notice that the final term in the original derivation and in its replacement are not the same. The variable x, in the original derivation, is fresh for the contexts Γ and Γ' and, therefore, x is not a free variable of the terms M and Q. According to this

$$M[xQ/z][\lambda y.P/x] \equiv M[(\lambda y.P)Q/z]$$

in the original derivation. The variable y in the replaced derivation is not a free variable of M, but it is free for P. Hence

$$M[P/z][(Q/y] \equiv M[P[Q/y]/z]$$

and the latter is obtained from the original term $M[(\lambda y.P)Q/z]$ by parallel β-reduction, $M[(\lambda y.P)Q/z] \to_\beta M[P[Q/y]/z]$. □

5. Discussion

Pure lambda calculus is a natural deduction system. The calculus $\lambda\!\to$ is a natural deduction term and a natural deduction type system and it completely matches the natural deduction system of intuitionistic logic (formulae-as-types, proofs-as-terms). The calculus $L\lambda\rightarrow$ is a natural deduction term and a sequent type system and it partly matches the sequent system of intuitionistic logic (formulae-as-types). The calculus given in Herbelin [3] is a sequent term and a sequent type system and it completely matches the sequent system of intuitionistic logic. The price that is paid for moving from the natural deduction character which is immanent for the lambda term system is in the substitution, which becomes explicit and requires a huge formal system. We gave a comparison of $\lambda\rightarrow$ and $L\lambda\rightarrow$ since they are usually used as tools for some proof theoretical studies, such as the relation between normalization and cut elimination in logic. A comprehensive list of references on this subject is given in Barendregt and Ghilezan [1].

Aknowledgement. The author is grateful to Mirjana Borisavljević for helpful remarks.

References

[1] Barendregt, H. P. and Ghilezan, S., Lambda terms for natural deduction, sequent calculus and cut elimination, J. Functional Prog. (to appear).

[2] Girard, J.-Y., Lafont, Y. and Taylor, P., Proofs and Types, Cambridge University Press, 1989.

[3] Herbelin, H., A lambda calculus structure isomorphic to Gentzen-style sequent calculus structure, Lecture Notes Comput. Sci., Vol. 933, pp. 61–75, Springer Verlag, 1995.

[4] Howard, W. A., The formulae-as-types notion of construction, in: eds. J. P. Seldin and J. R. Hindley, "To H. B. Curry, Essays on Combinatory Logic, Lambda Calculus and Formalism", pp. 479–490, Academic Press, 1980.

[5] Lambek, J., Logic without structural rules (Another look at cut elimination), in: eds. K. Došen and P. Schroeder-Heister, "Substructural Logics", pp. 179–206, Oxford University Press, 1993.

[6] Pottinger, G., Normalization as a homomorphic image of cut-elimination, Ann. Math. Logic 12 (1977), 323–357.

Received October 27, 1999.