Counting the Number of Proofs in the Commutative Lambek Calculus

Hans-Joerg Tiede

Abstract

This paper is concerned with the study of the number of proofs of a sequent in the commutative Lambek calculus. We show that in order to count how many different proofs in $\beta\eta$-normal form a given sequent $\Gamma \vdash \alpha$ has, it suffices to enumerate all the $\Delta \vdash \beta$ which are "minimal", such that $\Gamma \vdash \alpha$ is a substitution instance of $\Delta \vdash \beta$. As a corollary we obtain van Benthem's finiteness theorem for the Lambek calculus, which states that every sequent has finitely many different normal form proofs in the Lambek calculus.

Contents

Dedicated to Johan van Benthem on the occasion of his fiftieth birthday.[1]

1 Introduction

Among Johan van Benthem's many contributions to logic, I would like to single out the long lists of open problems that he has provided in his publications. They have been an important stimulus in my own research and, I am sure, that of many other researchers in logic, linguistics, and computer science. Thus, it is only fitting that this paper addresses one of the open problems in *Language in Action*, namely: "Provide an explicit function computing numbers of non-equivalent readings for sequents in the Lambek calculus."[2]

In this paper, we provide an algorithm to solve this problem using the theory of type assignment. As a corollary to this solution, we obtain another proof of van Benthem's finiteness theorem, which states that any sequent has only finitely many normal form proofs in the Lambek calculus.

Since the connection between the syntax and semantics of natural languages that the commutative Lambek calculus presents is the Curry-Howard isomorphism (see, for example, Girard [et al.], 1988) between proofs and λ-terms (proofs corresponding to syntactic derivations, terms to the meaning of an expression), the question of how many different normal form proofs there are of a given sequent is equivalent to asking how many different meanings an expression has (different meanings being terms none of which is convertible to any other).

Our solution to van Benthem's question relies on technical results due to Hirokawa (1993). Hirokawa proved that any two BCK-terms in β-normal form with the same principal type are identical. This theorem applies to the Lambek calculus and can be used to supply the first stage of an algorithm to count normal form derivations. However, since normal form proofs correspond to terms in $\beta\eta$-normal form, we need to extend the algorithm to those principal types which are the principal types of terms in $\beta\eta$-normal form. It was established by Hirokawa (1991) that minimal BCI types are the principal types of terms in $\beta\eta$-normal form. This theorem also is valid for the Lambek calculus, and we will use it to eliminate those principal types which are not the principal types of terms in $\beta\eta$-normal form. The main contribution of this paper consists of the introduction of principal types into the Lambek calculus and the application of these.

The reader is assumed to have some background in categorial grammar or type theory, preferably both. For the former, the reader should consult van Benthem (1995); for the latter we recommend Hindley's recent book (1997). The reader might also consult some of the review articles in the *Handbook of Logic and Language* (van Benthem & ter Meulen (eds.), 1997).

The following notational conventions will be used throughout this paper: capital roman letters from the beginning of the alphabet $(A, B, C, \ldots)$ will be

[1] I would like to thank Larry Moss and Benjamin Pierce for comments on earlier versions of this paper. All remaining errors are the author's.

[2] Page 351 of the American paperback edition.

used for types consisting of constants only, small roman letters ($a, b, c, \ldots$, but not e or t) will be used for type variables, e and t are reserved for the constants of the Lambek calculus (they correspond to *entity* and *truth value*, respectively), x, y, z will be used for term variables, $M, N, \ldots$ for arbitrary terms, small Greek letters for types consisting only of type variables, capital Greek letters for environments (in type theory) or assumptions (in logic).

2 The Lambek-van Benthem Calculus

Lambek calculi are substructural logics used for the study of natural language syntax and semantics. They are minimal substructural logics, lacking the structural rules of weakening, exchange, and contraction. They usually consist of two implications, / and \, which in absence of exchange do not coincide. The Lambek-van Benthem calculus (LP) is a commutative extension of the Lambek calculus, which collapses the two implications into one: $\rightarrow$. In its natural deduction format, the rules of LP can be presented as follows:

Definition 2.1

$$A \;\; ID$$

$$\frac{A \rightarrow B \quad A}{B} \rightarrow E$$

$$\frac{\begin{array}{c}[A] \\ \vdots \\ B\end{array}}{A \rightarrow B} \rightarrow I$$

In the rule $[\rightarrow I]$, the following side conditions need to be satisfied:

- *A has to have been used in an elimination rule,*
- *there is only one occurrence of A,*
- *there has to be another unwithdrawn assumption besides A.*

We use the Lambek calculus as a formal grammar by distinguishing a start symbol (t). The language generated are the concatenations of words $w_1, \ldots, w_n$ with associated categories $A_1, \ldots, A_n$, such that

$$A_1, \ldots, A_n \vdash t.$$

Example 2.2 *Consider the sentence "Everyone loves someone." Assuming that we make the following association of formulas with the individual words:*[3]

$$\begin{array}{rcl}\text{EVERYONE} & : & (e \rightarrow t) \rightarrow t \\ \text{LOVES} & : & e \rightarrow (e \rightarrow t) \\ \text{SOMEONE} & : & (e \rightarrow t) \rightarrow t\end{array}$$

[3]For the rationale for associating these formulas with these words, see van Benthem (1995).

we can present a derivation of $(e \to t) \to t, e \to (e \to t), (e \to t) \to t \vdash t$*:*

$$\dfrac{(e \to t) \to t \qquad \dfrac{\dfrac{(e \to t) \to t \qquad \dfrac{e \to (e \to t) \qquad [e]}{e \to t} \to E}{t} \to E}{e \to t} \to I}{t} \to E$$

which proves that t *is derivable from the assumptions corresponding to "Everyone loves someone." However, there are a number of different proofs for this.*

LP can be extended to a type assignment calculus by the usual Curry-Howard correspondence. The typing rules then are:

Definition 2.3

$$x : \alpha \;\; ID$$

$$\dfrac{M : \alpha \to \beta \quad N : \alpha}{MN : \beta} \to E$$

$$\dfrac{\begin{array}{c}[x : \alpha] \\ \vdots \\ M : \beta\end{array}}{\lambda x.M : \alpha \to \beta} \to I$$

Note: the same restrictions that were stated for $[\to I]$ *in definition 2.1 need to be satisfied here.*

By a simple inductive argument, we can establish that the λ-terms typeable by these rules satisfy the restriction that every λ-abstraction binds exactly one free variable and that every subterm of every typeable term contains a free variable. This type assignment calculus associates with every proof in the Lambek calculus a λ-term, in particular it associates with every proof in normal form a λ-term in $\beta\eta$-normal form. Thus the number of different proofs of a sequent $A_1, \ldots, A_n \vdash B$ corresponds to

$$\{M \mid x_1 : A_1, \ldots, x_n : A_n \vdash M : B\}\,.$$

The Curry-Howard correspondence is used in linguistics to represent the meaning of an expression. It is a form of the principle of compositionality, since the meaning of every complex expression is derived from the meaning of its parts and the way they are combined. The connection between λ-terms and meanings is illustrated below.

Example 2.4 *For the derivation from the previous example*

$$(e \to t) \to t, e \to (e \to t), (e \to t) \to t \vdash t.$$

there are four distinct readings:

$$\begin{array}{lcl} x:(e\to t)\to t, y:e\to(e\to t), z:(e\to t)\to t & \vdash & x(\lambda w.z(yw)):t \\ x:(e\to t)\to t, y:e\to(e\to t), z:(e\to t)\to t & \vdash & x(\lambda u.z(\lambda w.((yw)u))):t \\ x:(e\to t)\to t, y:e\to(e\to t), z:(e\to t)\to t & \vdash & z(\lambda w.x(yw)):t \\ x:(e\to t)\to t, y:e\to(e\to t), z:(e\to t)\to t & \vdash & z(\lambda u.x(\lambda w.((yw)u))):t \end{array}$$

These in turn correspond to the following logical formulas:

$$\begin{array}{c} \forall v_1\exists v_2(\textsc{love}(v_1)(v_2)) \\ \forall v_1\exists v_2(\textsc{love}(v_2)(v_1)) \\ \exists v_2\forall v_1(\textsc{love}(v_1)(v_2)) \\ \exists v_2\forall v_1(\textsc{love}(v_2)(v_1)) \end{array}$$

as can be verified by performing the following substitution on each term above:

$$\begin{array}{lcl} x & \mapsto & \lambda P.\forall v_1 P v_1 \\ y & \mapsto & \textsc{love} \\ z & \mapsto & \lambda Q.\exists v_2 Q v_2 \end{array}$$

The first and the third are derivable in the directed calculus as well, the second and fourth arise out of the commutativity.

3 Injectivity of Principal Type Assignment

The previous discussion showed that in order to count how many normal form derivations can be given to a given sequent, it suffices to consider how many $\beta\eta$-normal form terms can be typed in the type assignment calculus from the assumptions corresponding to the sequent.

Departing from the usual treatment of the Lambek calculus, we can consider the type assignment calculus to assign types consisting only of variable types rather than constant types.[4] This change makes it possible to consider the principal types assigned by this calculus.

Definition 3.1 *(Principal Types) α is a principal type of M if for some Γ , $\Gamma \vdash M : \alpha$ and for all β, if for some Δ, $\Delta \vdash M : \beta$, then there is a substitution function σ, such that $\sigma(\alpha) = \beta$.*

Definition 3.2 *(Principal Pairs) (Γ, α) is a principal pair of M if $\Gamma \vdash M : \alpha$ and for all (Δ, β), if $\Delta \vdash M : \beta$, then there is a substitution function σ, such that $\sigma(\Gamma) = \Delta$, $\sigma(\alpha) = \beta$.*

Note that principal types and pairs are unique up to renaming of variables (see Hindley, 1997).

[4] A system having both constant and variable types has been considered by van Benthem (1995). The restriction to a constant-free system is customary in the type theoretical literature in order to fascillitate principal typing and will be made here too.

Example 3.3 *The principal type of*

$$x(\lambda w.z(yw))$$

is any type variable a. Its principal pair is

$$(\{x : (a \to b) \to c, y : a \to d, z : d \to b\}, a).$$

Definition 3.4 *(Two-property): (Γ, α) has the two-property, if every type variable in (Γ, α) occurs exactly twice.*

Proposition 3.5 *(Characterization of principal pairs) $\Gamma \vdash M : \alpha$ is principal, if $\Gamma \vdash M : \alpha$ and (Γ, α) has the two-property.*

Proof. See Hindley (1993), and Hindley and Meredith (1990). ∎

Notice that since the two property is a decidable property of pairs (just count the number of occurrences of any variable) and that derivability is decidable (either by giving a principal type algorithm or by Lambek's cut-elimination theorem for the calculus, cf. van Benthem 1995), $\Gamma \vdash M : \alpha$ is a decidable relation.

Theorem 3.6 *(Injectivity of principal type assignment) If $\Gamma \vdash M : \alpha$ and $\Gamma \vdash N : \alpha$ in the Lambek calculus, and M and N are in β-normal form, then $M = N$.*

Proof. See Hirokawa (1993). ∎

Using the above correspondence, the first stage of our algorithm for computing the number of distinct derivations for some $A_1, \ldots, A_n, B$, such that $A_1, \ldots, A_n \vdash B$ consists of enumerating the principle pairs (Γ, α), such that $\sigma(\Gamma) = x_1 : A_1, \ldots, x_n : A_n$ and $\sigma(\alpha) = B$. The number of possible principle pairs to consider is finite up to renaming of variables, as any such principle pair has to be such that $\Gamma = x_1 : \alpha_1, \ldots, x_n : \alpha_n$, with $length(\alpha_i') \leq length(\alpha_i)$, and $length(\alpha) \leq b$. It should be noted that the number of such principle pairs is finite and that the question whether some (Γ, α) is a principle pair is decidable, using the principal type algorithm (cf. Hindley, 1997).

Example 3.7 *For*

$$(e \to t) \to t, e \to (e \to t), (e \to t) \to t \vdash t,$$

there are six different principal pairs (Γ, α), such that for some substitution σ,

$$\sigma(\Gamma) = x : (e \to t) \to t, y : e \to (e \to t), z : (e \to t) \to t$$

and

$$\sigma(\alpha) = t$$

which are:

$$\begin{array}{rcl}
x:(a\to b)\to c, y:a\to d, z:d\to b & \vdash & x(\lambda w.z(yw)):c \\
x:d\to b, y:a\to d, z:(a\to b)\to c & \vdash & z(\lambda w.x(yw)):c \\
x:(a\to b)\to c, y:d\to(a\to f), z:(d\to f)\to b & \vdash & x(\lambda u.z(\lambda w.((yw)u))):c \\
x:(d\to f)\to b, y:d\to(a\to f), z:(a\to b)\to c & \vdash & z(\lambda u.x(\lambda w((yw)u))):c \\
x:(a\to b)\to c, y:a\to(d\to f), z:(d\to f)\to b & \vdash & x(\lambda w.z(\lambda u.((yw)u))):c \\
x:(d\to f)\to b, y:a\to(d\to f), z:(a\to b)\to c & \vdash & z(\lambda w.x(\lambda u.((yw)u))):c
\end{array}$$

While the first four of these correspond to our desired readings, the last two are just η-expansions of the first and second, respectively. Thus, we have not yet solved the problem.

4 Minimal Types

As was noted above, if we only enumerate the principal pairs (Γ, α), such that for a derivable sequent in the Lambek calculus $\alpha_1, \ldots, \alpha_n \vdash \beta$ there exists a substitution σ, such that $\sigma(\Gamma) = x_1 : \alpha_1, \ldots, x_n : \alpha_n$ and $\sigma(\beta) = \alpha$, then we are only enumerating the λ-terms in β-normal form corresponding to their respective proofs. However, normal form proofs correspond to λ-terms in $\beta\eta$-normal form. Thus we need to consider only principal pairs of λ-terms in $\beta\eta$-normal form. The search for such principal pairs has led to the notion of minimal types.

Definition 4.1 *A principal type α is minimal, if, for any principal type β such that there is a substitution σ_1, such that $\sigma_1(\beta) = \alpha$, there is a substitution σ_2, such that $\sigma_2(\alpha) = \beta$.*

The notion of minimal type is extended to pairs in the obvious fashion, i.e. a principal pair (Γ, α) is a minimal pair if for all principal pairs (Δ, β), if for some substitution σ_1, $\sigma_1(\Delta) = \Gamma$ and $\sigma_1(\beta) = \alpha$, then there exists a substitution σ_2, such that $\sigma_2(\Gamma) = \Delta$ and $\sigma_2(\alpha) = \beta$. Minimal types can also be defined by considering a relation $<$ on types, such that $\alpha < \beta$ if there exists a substitution σ, such that $\sigma(\alpha) = \beta$. The relation is then extended to pairs in the obvious way. A minimal pair then is a principal pair (Γ, α) such that for all principal pairs (Δ, β), if $(\Delta, \beta) < (\Gamma, \alpha)$, then $(\Gamma, \alpha) < (\Delta, \beta)$.

The following theorem sums up the relationship between minimal types and λ-terms in $\beta\eta$-normal form.

Theorem 4.2 *A principal pair (Γ, α) is minimal iff $\Gamma \vdash M : \alpha$ for some M in $\beta\eta$-normal form.*

Proof. See Hirokawa (1991). ∎

Using this property of minimal pairs, the algorithm for counting the number of proofs is extended as follows:[5] Given a derivable sequence $A_1, \ldots, A_n \vdash B$,

[5] A similar technique is used by Hirokawa & Komori (1993) for BCK formulas.

we enumerate the principal pairs (Γ, α), such that for some substitution σ, $\sigma(\Gamma) = x_1 : A_1, \ldots, x_n : A_n$, and $\sigma(\alpha) = B$. Let us denote the set of these principal pairs for $A_1, \ldots, A_n \vdash B$ by $\Pi_{A_1,\ldots,A_n \vdash B}$. Given these, we eliminate those principal pairs in $\Pi_{A_1,\ldots,A_n \vdash B}$ that are not minimal, i.e. we eliminate each $(\Gamma, \alpha) \in \Pi_{A_1,\ldots,A_n \vdash B}$ such that for some $(\Delta, \beta) \in \Pi_{A_1,\ldots,A_n \vdash B}$, such that for some substitution σ_1, $\sigma_1(\Delta) = \Gamma$ and $\sigma_1(\beta) = \alpha$. This suffices as we do not consider alphabetic variants of principal pairs. This gives us the set of principal pairs that are minimal, corresponding to the number of normal form terms, corresponding in turn to the number of distinct normal form derivations.

Example 4.3 *To complete our example, we had arrived at the following principal pairs for*

$$(e \to t) \to t, e \to (e \to t), (e \to t) \to t \vdash t,$$

$$\begin{aligned}
x : (a \to b) \to c, y : a \to d, z : d \to b &\vdash x(\lambda w.z(yw)) : c \\
x : d \to b, y : a \to d, z : (a \to b) \to c &\vdash z(\lambda w.x(yw)) : c \\
x : (a \to b) \to c, y : d \to (a \to f), z : (d \to f) \to b &\vdash x(\lambda u.z(\lambda w.((yw)u))) : c \\
x : (d \to f) \to b, y : d \to (a \to f), z : (a \to b) \to c &\vdash z(\lambda u.x(\lambda w.((yw)u))) : c \\
x : (a \to b) \to c, y : a \to (d \to f), z : (d \to f) \to b &\vdash x(\lambda w.z(\lambda u.(yw)u))) : c \\
x : (d \to f) \to b, y : a \to (d \to f), z : (a \to b) \to c &\vdash z(\lambda w.x(\lambda u.((yw)u))) : c
\end{aligned}$$

Now we show that we can eliminate the last two pairs, since the following substitution

$$\begin{aligned}
a &\mapsto a \\
b &\mapsto b \\
c &\mapsto c \\
d &\mapsto (d \to f)
\end{aligned}$$

shows that the fifth and the sixth are substitution instances of the first and the second, respectively. Therefore, we can eliminate the fifth and sixth pair, leaving us with the four readings as desired.

Corollary 4.4 *Any sequent has a finite number of β-η-normal form proofs in the Lambek calculus.*

Proof. Since the number of $\beta\eta$-normal form proofs in the Lambek calculus of any sequent $\alpha_1, \ldots, \alpha_n \vdash \beta$ equals the number of minimal pairs (Γ, γ) such that $\alpha_1, \ldots, \alpha_n \vdash \beta$ is a substitution instance of (Γ, γ), the number of different proofs of a sequent is finite, since there can be only finitely many such pairs up to renaming of variables. ∎

5 Conclusion

The above algorithm shows that the introduction of principal and minimal types leads to interesting results about the Lambek calculus. It is worth noting that the above procedure can also be used to generate every normal form proof of a derivable sequent. This is due to the fact that in the process of enumerating the minimal pairs, we can use the principal type algorithm to construct the principal derivation of the normal form term. If we perform a substitution at every step of the derivation, we obtain normal form proof in the Lambek calculus.

Finally, it should be noted that this procedure to count normal form proofs can be adapted to the non-commutative Lambek calculus.[6] In order to do this, it is necessary to use a bi-directional λ-calculus, as studied in Wansing (1993), to which one adds two versions of application in addition to the two versions of abstraction. It can then be shown that the principal pairs of this calculus also have the two-property and that the principal pairs of terms in $\beta\eta$-normal form coincide with the minimal pairs.[7]

References

[1] van Benthem, Johan (1995), *Language in Action*. Cambridge, MA: MIT Press.

[2] van Benthem, Johan and Alice ter Meulen (eds.) (1997), *The Handbook of Logic & Language*. Cambridge, MA: MIT Press.

[3] Girard, Jean-Yves; Yves Lafont and P. Taylor (1988), *Proofs and Types*. Cambridge: Cambridge University Press.

[4] Hindley, J. Roger (1993), "BCK and BCI logics, condensed detachment and the 2-property," in: *Notre Dame Journal of Formal Logic* 34 (1993): 231-250.

[5] Hindley, J. Roger (1997), *Basic Simple Type Theory*. Cambridge: Cambridge University Press.

[6] Hindley J. Roger & David Meredith (1990), "Principal Type-Schemes and Condensed Detachment," in: Journal of Symbolic Logic 55(1): 90-105.

[7] Hirokawa, Sachio (1991), "Principal Type-Schemes of BCI-Lambda-Terms," in: T. Ito & A.R. Mayer (eds) (1991), *Theoretical Aspects of Computer Software*. Berlin: Springer Verlag: 633-650.

[8] Hirokawa, Sachio (1993), "Principal types of BCK-lambda-terms," in: *Theoretical Computer Science* 107: 253-276.

[6] A more complete treatment of counting the normal form proofs of sequents in the non-commutative Lambek calculus is included in my forthcoming dissertation.

[7] The proof of the latter assertion employs a category theoretic result proved by Lambek (1968), which is known as the "coherence theorem for residuated categories." It states that any two normal form proofs in the Lambek calculus of a sequent that has the two property are identical.

[9] Komori, Y. & Sachio Hirokawa (1993), "The number of proofs for a BCK formula," in: *Journal of Symbolic Logic* 58: 626-628.

[10] Lambek, Joachim (1958), "The Mathematics of Sentence Structure," in: *American Mathematical Monthly* 65: 154-169.

[11] Lambek, Joachim (1968), "Deductive Systems and Categories I," in: *Mathematical Systems Theory* 2(4): 287-318.

[12] Wansing, Heinrich (1993), *The Logic of Information Structures*. Berlin: Springer Verlag.