

Natural Term Logic

Clarence Protin

April 7, 2026

Abstract

In this paper we develop a formal logic called Natural Term Logic (NTL) which is based on the syntactic and logical structure of natural language. NTL can be seen as a refinement of the ideas of Quine's paper 'Variables Explained Away' and the technical concepts introduced by Bealer and Zalta. NTL is more fine-grained than Bealer's first-order intensional logic (BL): there is a many-to-one correspondence ν between NTL terms and closed BL terms as well as a canonical map β which assigns to each BL term a corresponding NTL term. We define a series of reductions on NTL and our main result is that each NTL term T reduces to a unique normal term N . The reductions fall into the structural, predicative and pushing-in categories. Predicative reductions decompose NTL terms so that predication is only applied to a primitive term. Such NTL terms are called prenormal and in general βt assigns a prenormal NTL term to a Bealer term t . We can associate a unique canonical NTL term to a BL term t by first calculating βt and then applying structural and pushing-in reductions until obtaining the unique normal term N such that $\nu N = t$.

(...) *for it is because of the interweaving of the forms with one another that we come to have the logos.*

- Plato, Sophist 259E

1 Introduction

In this paper we develop a formal logic called Natural Term Logic (NTL). NTL aims to faithfully represent key aspects of the logical and grammatical mechanisms of natural language as well as grammatical transformations which preserve meaning. In NTL quantification (multiple generality) is expressed without variables as in 'All Alice's friends know a friend of Bob's'. Key examples of meaning invariance under grammatical transformations are the following:

- (i) 'Alice likes Bob' = 'Bob is liked by Alice'.
- (ii) 'Alice likes Alice' = 'Alice likes herself'.
- (iii) 'Alice knows that Bob sleeps' = 'Bob is known by Alice to sleep'.

Item (i) is an instance of the 'passive voice', item (ii) an instance of the use of a ' reflexive pronoun', and item (iii) an instance of 'passive raising'. The latter involves what is commonly called 'intensional logic' (since expressions representing propositions are subjects of a predicate) but the grammatical transformation would seem to involve a complex kind of property-formation.

NTL expressions are built up inductively from primitive *terms* and *constructors* which take one or more terms and yield a new term. Each term has a *valency* which is a non-negative integer that represents its “degree of saturation”, 0 for propositions (0-ary relations), 1 for properties (unary relations), 2 for binary relations and so forth.

Our approach is quite different from traditional approaches to formalization of natural language because it is not based on variables and their bindings by operators (see e.g. [2, 4]). In fact, in NTL there are no variables nor, in particular, variable binding. NTL differs radically from Schönfinkel and Curry’s Combinatory Logic [3] in that constructors are not terms in their own right. The precursors of NTL are the constructors used in Quine’s paper ‘Variables Explained Away’ [5] for first-order logic and the syntactic and semantic functions used by Bealer in [1] (and also to a certain extent Zalta in his book ‘Axiomatic Metaphysics’ [6]).

There are three families of constructors Υ , Π and Λ which are indexed with finite combinatorial information (see Section 4 for the details). The Υ family can be thought of as permuting the arguments of a term, as in the construction of the passive voice in (i) of ‘Bob is liked by Alice’, where a constructor Υ is applied to the term representing ‘likes’ to yield ‘is liked by’. The Υ constructor represents also a generalized diagonalization of the arguments of a term, and in particular *reflexivization* of a term: applying a certain Υ to ‘likes’ yields the 1-valent term ‘likes oneself’. The Π family in its simplest form expresses ordinary predication, the application of a verb to its subject and object(s): for instance the Subject-Verb-Object construction ‘Alice likes Bob’ corresponds to the $\Pi^{(0,0)}$ application of ‘likes’ to the arguments ‘Alice’ and ‘Bob’. But Π -applications also have more complex forms of term-formation involving ‘embedded predication’. Consider the simple predication ‘Alice knows that Bob sleeps’ which corresponds to applying the Π constructor to ‘knows’, to ‘Alice’ and the 0-valent term (proposition) ‘Bob sleeps’, itself the simple predication of ‘sleeps’ to ‘Bob’. Thus ‘Bob’ might be called an ‘embedded subject’ which we can abstract to transform the proposition ‘Alice knows that Bob sleeps’ into a 1-valent term (property): ‘the property of being known by Alice to sleep’ or more simply ‘known by Alice to sleep’. Our constructor Π can construct directly the property ‘known by Alice to sleep’ from the terms ‘known’, ‘Alice’ and ‘sleeps’. Herein ‘sleeps’ does not enter directly as an argument (as in ‘Alice knows the property “sleeps”’) but as something that contributes through its predicability to an argument to forming a more complex property via the predication of ‘knows’. Having this kind of predication also allows us to express the analogue of variable substitutions (useful for instance in universal quantifier rules) directly in NTL.

Finally, the family Λ includes logical connectives as well as a way of expressing quantification without variables or variable binding. These are completely general and thus can accommodate both classical and non-classical logics.

The examples (i) – (iii) above can be expressed in NTL as follows:

$$(I) \quad \Pi^{(0,0)}[\text{likes}][\text{Alice}][\text{Bob}] = \Pi^{(0,0)}\Upsilon^{((1,2),(\{2\},\{1\}))}[\text{likes}][\text{Bob}][\text{Alice}]$$

$$(II) \quad \Pi^{(0,0)}[\text{likes}][\text{Alice}][\text{Alice}] = \Pi^{(0,0)}\Upsilon^{((1,2),(\{1,2\}))}[\text{likes}][\text{Alice}]$$

$$(III) \quad \Pi^{(0,0)}[\text{knows}][\text{Alice}]\Pi^{(0)}[\text{sleeps}][\text{Bob}] = \Pi^{(0)}\Pi^{(0,1)}[\text{know}][\text{Alice}][\text{sleep}][\text{Bob}]$$

and ‘all Alice’s friends know Bob’ can be expressed as

$$\Lambda_{\forall}^1\Upsilon^{((1,2),(\{1,2\}))}\Lambda_{\rightarrow}[\text{being Alice’s friend}]\Pi^{(1,0)}[\text{know}][\text{Bob}].$$

Each constructor of the three families is indexed with finite combinatorial information. This combinatorial information crucially depends on a theory we call *weaver theory* which can be described roughly as the theory of a certain class of maps between finite linearly ordered sets that can be interpreted geometrically as a generalization of braids in which strands can be merged together.

In the examples (i) – (iii) above not only do we intuitively assign the same meaning to the two sides but also the left-hand side is intuitively taken to be more simple or basic in some sense (the analogue of the ‘normal form’ used in type theory). Formalizing this intuition and setting up the corresponding reduction theory for NTL terms is the main goal of this paper.

2 Weaver Theory

2.1 Weavers

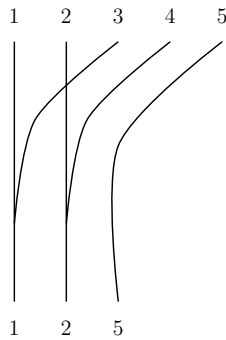
In what follows if A is a sequence we denote its i th element (for i a non-negative integer) by A_i and its *length* by $|A|$. We denote the empty sequence by ϵ and consider $|\epsilon| = 0$. Given a sequence A we denote by $\text{set}(A)$ the set of elements occurring in A . Given a non-repeating sequence A we can also consider A to be $\text{set}(A)$ with the linear order induced by the sequential order of A .

Definition 2.1 A weaver W consists in a pair (X, Y_R) where X is a finite linearly ordered set and Y_R is a linear ordering of the set of equivalence classes of an equivalence relation R on X .

If the equivalence relation is clear from the context we may write Y instead of Y_R . The equivalence relation of a weaver can be defined as $R = \text{set}(Y)$. As illustrated in the example below, one should think of R as embodying the information relating to joining together of threads (generalized diagonalization) while the order on Y captures the permutation of threads.

Example 2.2 Some simple examples of weavers are:

- (i) The *identity weaver* $I = ((x_1, \dots, x_n), (\{x_1\}, \dots, \{x_n\}))$.
- (ii) The *empty weaver* (ϵ, ϵ) .
- (iii) Let $W = ((1, 2, 3, 4, 5), (\{1, 3\}, \{2, 4\}, \{5\}))$. This weaver identifies threads 1 and 3, 2 and 4 into a single strand and leaves 5 untouched. This can be illustrated by the following diagram



Notation 2.3 Given a weaver $W = (X, Y_R)$ we use the notation $W^\circ = X$, $W^\flat = Y_R$ and $W_R = \text{set}(Y)$.

Definition 2.4 We say that a weaver $W = (X, Y)$ has *order* (m, n) if $|X| = m$ and $|Y| = n$.

For instance, the weaver from Example 2.2(iii) has order $(5, 3)$.

Definition 2.5 An equivalence relation R on a set X is called *discrete* if xRy iff $x = y$. A weaver W is called a *permutator* if W_R is the discrete relation on W° .

Definition 2.6 Let $W = (X, Y)$ be a weaver. Then we define the function

$$W^* : \text{set}(X) \rightarrow \text{set}(Y)$$

as follows. Given $x \in X$ we set $W^*(x) = r$ where r is the element in W_R such that $x \in r$.

Let $M = (M_1, \dots, M_n)$ where the M_i are sets of sets. Then we use the notation

$$M^\cup = \left(\bigcup M_1, \dots, \bigcup M_n \right)$$

Analogously we define $N^\cup$ if N is a set (rather than an ordered sequence) of sets of sets.

Definition 2.7 Given a weaver $W = (X, Y)$ and a linearly ordered set A of the same cardinality as X we denote by W^A the weaver (A, Y') (called the *shift* of W relative to A) defined as follows. We set $A_i(W^A)_R A_j$ iff $X_i W_R X_j$ and we define $Y' = (Y'_1, \dots, Y'_n)$ where $Y'_k = \{A_l : X_l \in Y_k\}$.

Definition 2.8 Let $V = (X, Y)$ be a weaver of order (n, m) and $W = (Z, U)$ a weaver of order (m, k) . Let $W' = W^{V^b} = (V^b, U')$. The *composition* of V and W is defined as the weaver $W'V = (X, (U')^\cup)$.

Proposition 2.9 With the notation of the previous definition we have

$$(WV)^* = \cup \circ (W')^* \circ V^*$$

Definition 2.10 Given a weaver $W = (X, Y)$ we denote by $W^\#$ the sequence obtained by substituting each x_i in X by the $r \in W_R$ such that $x_i \in r$.

Example 2.11 Let $W = ((1, 2, 3), (\{1, 3\}, \{2\}))$ then $W^\# = (\{1, 3\}, \{2\}, \{1, 3\})$.

Given two sequences (possibly with repeating elements) X and Y both of length n , we say that X is isomorphic to Y if there is a bijection $f : \text{set}(X) \rightarrow \text{set}(Y)$ such that $Y_i = f(X_i)$ for $i = 1, \dots, n$.

Lemma 2.12 Given a sequence X of length n (possibly with repeated elements) and a non-repeating sequence S such that $\text{set}(S) = \text{set}(X)$, there is a weaver W and a bijection such that $W^\#$ is equivalent to X and W^b is equivalent to S .

Proof. Let $|X| = n$ and consider the sequence $N = (1, 2, \dots, n)$. We define the equivalence relation R on N by iRj iff $X_i = X_j$. Note that we have a bijection $f : S \rightarrow R$ where $f(x) = r$ iff for a $n \in r$ we have that $X_n = x$. Then f determines an order on R by setting $r < s$ iff $f^{-1}(r)$ comes before $f^{-1}(s)$ in S . Let M be R endowed with this linear order. It is then it is easy to check that the weaver (N, M) satisfies the conditions of the lemma.

Notation 2.13 We call the weaver constructed in the proof of Lemma 2.12 the *canonical weaver* associated to X and S and denote it by $\mathcal{W}(X, S)$.

The following will be useful further ahead.

Definition 2.14 Let Z be a non-repeating sequence of length m and W a weaver of order (n, m) . Then we define $W^\#(Z)$ as being obtained by replacing each r in $W^\#$ with Z_i if r occurs in position i in $W^\#$.

The following will be useful in the next section.

Definition 2.15 Let $W = (X, Y)$ and $W' = (X', Y')$ be weavers and assume without loss of generality that X and X' have no common elements. Then we define $W + W' = (X + X', Y + Y')$ which is easily seen to be a weaver with $(W + W')_R = W_R \cup W'_R$.

The finite sum of weavers $\sum_{i=1}^n W_i$ can be defined in a similar way.

2.2 Weavers and Order Partitions

Definition 2.16 Given a finite sequence X (or equivalently, a finite linearly ordered set X with order $<$), an *order partition* P on X is an ordered sequence $(I_1, \dots, I_k)$ where each I_i is either ϵ or a segment of X of the form $[a_i, b_i] = \{x \in X : a_i \leq x \leq b_i\}$. The sequence of the I_i must satisfy the following conditions. Let $(I'_1, \dots, I'_m)$ be the sequence obtained by removing occurrences of ϵ . Then

$$(i) \quad X = \bigcup_{i=1}^k \text{set}(I'_i)$$

$$(ii) \quad a'_1 \text{ is the minimum element of } X, b'_i < a'_{i+1} \text{ for } i < m \text{ and } b'_m \text{ is the maximum element of } X_{<}.$$

Definition 2.17 Let $W = (X, Y)$ be a weaver and $A \subset \text{set}(X)$. Then we define the *restriction* of W to A , denoted by W_A , as the weaver $W_X = (A, Y')$ where A is considered with the induced order from X and for $Y = (Y_1, \dots, Y_n)$ we have that Y' is obtained from $(Y_1 \cap A, \dots, Y_n \cap A)$ by deleting occurrences of $\emptyset$.

Definition 2.18 Given a sequence A , an order partition $P = (I_1, \dots, I_k)$ on A and a weaver W we say that W is *within* P if $W = W_{I_1} + \dots + W_{I_k}$.

The following lemma is an immediate consequence of the previous definition:

Lemma 2.19 Let W be a weaver inside a partition $P = (I_1, \dots, I_k)$ on A . Then W induces an order partition $P' = (J_1, \dots, J_k)$ on W^b .

Definition 2.20 Given a weaver W and a partition P on A we say that W is *without* P if W is not the identity weaver and its restriction W_{I_i} to each I_i of P is the identity weaver.

The main result (and algorithm) of this section is the following.

Theorem 2.21 Let $W = (X, Y)$ be a weaver and P an order partition on X then either W is within P or W can be factored uniquely as $W = W_{out}W_{in}$ where W_{in} is a weaver within P and W_{out} is a weaver without the partition P' induced by W_{in} .

Proof. Let $P = (I_1, \dots, I_n)$. Then we define $W_{in} = \sum_{i=1}^n W_{I_i}$ which is evidently within P . We then construct the weaver $W_{out} = ((W_{in})^b, Z)$ as follows. We take Y and we replace each Y_i with a set $\{r_1, \dots, r_k\}$ where the $r_i \in (W_{in})_R$ and $Y_i = \bigcup_{l=1}^k r_l$. It is then evident that $W = W_{out}W_{in}$. That this factorization is unique is left as an exercise.

2.3 Actions of Number Sequences on Weavers

Definition 2.22 Given a non-negative integer n and a symbol a we define for $n > 0$ $\sqcup_n a = a^1 a^2 \dots a^n$ (considered as a sequence of distinct elements all whose elements are tagged by a) and for $n = 0$ we set $\sqcup_0 = \epsilon$.

Definition 2.23 Let $W = (A, L)$ be a weaver of order (n, m) and S a sequence of non-negative integers of length m . Then we define $W \odot S$ to be the weaver (A', L') obtained as follows. Suppose $L_i = \{c_1, \dots, c_k\}$ and $S_i = p$. Then we replace each c_j in X with $\sqcup_p c_j$. and we replace L_i with $L_i^1 \dots L_i^p$ where $L_i^s = \{c_1^s, \dots, c_k^s\}$. We do this for all the elements of L .

Example 2.24 Let $L = ((1, 2, 3, 4), (\{1, 3\}, \{2, 4\}))$ and $S = (2, 2)$. Then

$$W \odot S = ((1^1, 1^2, 2^1, 2^2, 3^1, 3^2, 4^1, 4^2), (\{1^1, 3^1\}, \{1^2, 3^2\}, \{2^1, 4^1\}, \{2^2, 4^2\}))$$

Definition 2.25 A *selector* is a pair (N, M) of sequences of non-negative integers such that $|N| = |M|$ and $M_i \leq N_i$.

Given a sequence of integers N we denote by ΣN the sum of the elements of N . Given a symbol s we denote by s^n for n a non-negative integer the sequence consisting of s repeated n times. If $n = 0$ we set $s^0 = \epsilon$.

Definition 2.26 Let $N = (n_1, \dots, n_k)$ be a sequence of non-negative integers and S a sequence of length ΣN . Then the *division* of S by N , denoted by S/N , is the unique decomposition

$$S = A_1 + \dots + A_k$$

such that $|A_i| = n_k$. Note that S/N is, strictly speaking, the sequence $(A_1, \dots, A_k)$.

Note that S/N defines an order partition on S .

Definition 2.27 Given a selector (N, S) with $|N| = k$ and a selector (M, T) with $|M| = \Sigma N$, we define the *associator* $S \otimes T$ to be a tuple $(A, B_1, \dots, B_k)$ constructed as follows. Let

$$M/S = \bar{\mu}_1 + \dots + \bar{\mu}_k$$

and

$$T/S = \bar{\tau}_1 + \dots + \bar{\tau}_k$$

Then $B_i = (\bar{\mu}_i + 1^{N_i - S_i}, \bar{\tau}_i + 1^{N_i - S_i})$ for $i = 1, \dots, k$ and

$$A = (\Sigma \bar{\tau}_1 + N_1 - S_1, \dots, \Sigma \bar{\tau}_k + N_k - S_k), (\Sigma \bar{\tau}_1, \dots, \Sigma \bar{\tau}_k)$$

We use the notation $(S \otimes T)^1 = A$ and $(S \otimes T)_i = B_i$.

Example 2.28 Let S be $((3, 3), (2, 1))$ and T be $((3, 2, 3), (2, 2, 1))$. Then $(3, 2, 3)/(2, 1) = (3, 2) + (3)$ and $(2, 2, 1)/(2, 1) = (2, 2) + (1)$. So $(S \otimes T)_1$ and $(S \otimes T)_2$ are the $((3, 2, 1), (2, 2, 1))$ and $((3, 1, 1), (1, 1, 1))$ respectively (we have $3 - 2 = 1$ and $3 - 1 = 2$). It follows that $(S \otimes T)^1$ is $((5, 3), (4, 1))$.

Selectors act on sequences of sequences as follows.

Definition 2.29 Let $S = (N, M)$ be a selector with $|N| = n$ and $A = (B^1, \dots, B^n)$ a sequence of sequences B^i with $|B^i| = N_i$. Let C^i be the subsequence of M_i consisting of the first M_i elements of B^i and let D^i be the 'remainder' of B^i (that is, $B^i = C^i + D^i$). Then we set $S \bullet A = (C^1, \dots, C^n)$ and $A \bullet S = (D^1, \dots, D^n)$.

We also use the notation $A! = B^1 + \dots + B^n$.

3 Bealer Logic

Bealer logic (BL) is first-order logic with a term-forming operator $\lambda x_1 \dots x_n \phi$ where $x_1, \dots, x_n$ is a (possibly empty) list of distinct variables and ϕ is a formula.

The language of BL consists of a countable collection of variables $x_1, x_2, \dots$, constants $c_1, c_2, \dots$, and atomic predicates $A_1, A_2, \dots$ of various arities.

We assume that there are variables denoted by $\xi_{\{x_1, \dots, x_n\}}$ where the x_i are variables (which may in turn be of this form). The subscript is for book keeping purposes and the variables the appear therein are never bound or participate in substitutions.

Definition 3.1 *Formulas and terms* are defined by mutual recursion as follows.

- Variables and constants are terms.
- If $t_1, \dots, t_n$ are terms then $A(t_1, \dots, t_n)$ is a formula whenever A is an n -ary atomic predicate.
- If ϕ is a formula and $x_1, \dots, x_n$ is a possibly empty list of distinct variables then $\lambda x_1 \dots x_n \phi$ is a term.
- Given a variable x we have that $\lambda x x$ is a term.
- If ϕ and ψ are formulas and x is a variable then $\phi \wedge \psi$, $\neg \phi$ and $\forall x \phi$ are formulas.

Terms of the form $\lambda x_1 \dots x_n \phi$ are called *complex* terms. In this case n is called the *sort* of the term. Sort 1 terms of the form $\lambda x x$ are called pseudo-terms.

We allow the renaming of bound variables and discarding of dummy variables. We can also interchange x and $\lambda x x$. The use of $\lambda x x$ will be explained further ahead.

Let S be a selector and W a weaver. We define the operators π^S , v^S , $\lambda_{\neg}$, $\lambda_{\wedge}$ and $\lambda_{\forall}$ which act of formulas of BL to produce a new formula.

Definition 3.2 Let $p = \lambda x_1 \dots x_n \phi(x_1, \dots, x_n)$ be a complex term and $t_1, \dots, t_n$ be terms such that when not a constant $t_i = \lambda x_1^i \dots x_{m_i}^i \phi_i$ (which we write $\lambda \bar{x}^i \phi_i$ and rename variables to be distinct for different i s) is of sort m_i . If a t_i is a variable then we assume it is $\lambda x x$ and considered of sort 1. Consider a selector S of the form $S = ((m_1, \dots, m_n), (a_1, \dots, a_n))$. Then

$$\pi^S p t_1 \dots t_n = \lambda(S \bullet (\bar{x}^1, \dots, \bar{x}^n))! \phi[t'_1 \dots t'_n / x_1 \dots x_n]$$

where $t'_i = \lambda((\bar{x}^1, \dots, \bar{x}^n) \bullet S)_i \phi_i$ (when t_i is a constant we set $t'_i = t_i$).

Definition 3.3 let W be a weaver of order (n, m) . Then

$$v^W \lambda x_1 \dots x_n \phi = \lambda(W^{(x_1, \dots, x_n)})^b \phi[(W^{(x_1, \dots, x_n)})^\# / x_1 \dots x_n]$$

The expression $\lambda(W^{(x_1, \dots, x_n)})^b$ is to be read as replacing the occurrence of an equivalence class r with the variable ξ_r and the same goes for $(W^{(x_1, \dots, x_n)})^\#$ in the substitution.

Definition 3.4 Let $p = \lambda x_1 \dots x_n \phi(x_1, \dots, x_n)$ and $q = \lambda y_1 \dots y_m \psi(y_1, \dots, y_m)$ be complex terms. Then we define

$$\begin{aligned} \lambda_{\neg}^n p &= \lambda x_1 \dots x_n \neg \phi(x_1, \dots, x_n) \\ \lambda_{\wedge}^{n,m} p q &= \lambda x_1 \dots x_n y_1 \dots y_m \phi(x_1, \dots, x_n) \wedge \psi(y_1, \dots, y_m) \\ \lambda_{\forall}^n p &= \lambda x_2 \dots x_n \forall x_1 \phi(x_1, \dots, x_n). \end{aligned}$$

Remark 3.5 The operator π includes, as a particular case, standard substitution of a variable for a term. Also note that we have a trivial instance of the application $\pi^{(1,1,\dots,1)} t x_1 \dots x_n$ which yields t itself.

Definition 3.6 A *primitive* BL term is a term of the form $\lambda x_1 \dots x_n A(x_1, \dots, x_n)$ where A is an atomic predicate. We also use the notation $[A]$.

We show that any complex term t in BL can be obtained by successive applications of π, σ, λ_* for $(* \in \{\neg, \wedge, \forall\})$ to primitive terms corresponding to the atomic predicates occurring in t .

Example 3.7 The term $\lambda w A(\lambda v B(w, v), \lambda u C(u))$ is obtained from $\pi^{(1,0)} [A][B][C]$.

Definition 3.8 A term of the form $\lambda x_1 \dots x_m A(t_1, \dots, t_n)$ is called an *atomic* term.

4 Natural Term Logic

The language of Natural Term Logic (abbreviated NTL from now on) is defined as follows. We are given a collection of *primitive* terms $A_1, A_2, \dots, A_n, \dots$ including the special term I and a finite collection of constructors $\Gamma_1, \Gamma_2, \dots, \Gamma_n$. Each primitive term A has a *valency* (saturation degree) which is a non-negative integer s . Each constructor Γ has a signature $(s_1, \dots, s_n) \rightarrow s_{n+1}$ where the s_i are non-negative integers.

Definition 4.1 *Terms* in NTL are defined as follows

- A primitive term is a term.
- If Γ is a constructor with signature $(s_1, \dots, s_n) \rightarrow s_{n+1}$ and T_i are terms with valencies s_i for $i = 1, \dots, n$ then $\Gamma T_1 \dots T_n$ is a term of valency s_{n+1} .
- Nothing else is a term.

We sometimes indicate that a term T has valency s by writing $T^{(s)}$. should be introduced before The special term I has valency 1.

In this paper we assume the following relationship between the language of BL and that of NTL. For every n -ary atomic predicate symbol A of BL there is a unique corresponding primitive term $A^{(n)}$ of NTL and vice-versa. We can also assume that NTL has a set of primitive propositions (primitive terms of valency 0) which correspond to constants in BL. We will also work only with the following set of constructors:

$$\Pi^S, \Upsilon^W, \Lambda_{\wedge}^{(n,m)}, \Lambda_{\neg}^n, \Lambda_{\vee}^n$$

where S is a selector and W is a weaver (considered modulo shifting). We have that Π^S has signature $(a_1, \dots, a_n) \rightarrow \Sigma b$ for $S = (a, b)$, Υ^W has signature $(n) \rightarrow m$ for W of order (n, m) , $\Lambda_{\wedge}^{(n,m)}$ has signature $(n, m) \rightarrow n + m$, $\Lambda_{\neg}^n$ has signature $(n) \rightarrow n$ and $\Lambda_{\vee}^n$ signature $(n) \rightarrow n - 1$.

We now define a canonical process ν to convert an NTL term T into a closed BL term νT .

Definition 4.2 We define ν by induction on the structure of T .

- $\nu A^{(n)} := \lambda x_1 \dots x_n A(x_1, \dots, x_n)$
- $\nu I = \lambda x. x$
- $\nu \Gamma T_1 \dots T_n := \gamma(\nu T_1) \dots (\nu T_n)$,
where, on the left-hand side Γ, γ corresponds to associating the constructor with the corresponding BL operator: Π^S becomes π^S , Υ^W becomes ν^W , $\Lambda_{\wedge}^{(n,m)}$ becomes $\lambda_{\wedge}^{(n,m)}$ and so forth.

The following is clear from the definition of ν .

Proposition 4.3 *To every term T in NTL we can associate canonically a closed term νT of BL in such a way that for a primitive term $A^{(n)}$ we have that $\nu A^{(n)} := \lambda x_1 \dots x_n A(x_1, \dots, x_n)$.*

Definition 4.4 Given a closed BL term t , its *Bealer decomposition* βt is a process which associates t to an NTL term βt defined according to the following clauses:

- If t is a primitive term $[A]$, a constant, or a variable, then $\beta t = A$.

- Let t be a non-primitive atomic term $\lambda x_1 \dots x_m A(t_1, \dots, t_n)$ where the t_i have sort c_i . Let V_i be the sequence of free variables in t_i ordered according to the order they appear in $x_1, \dots, x_m$. Consider the sequence $V = V_1 + \dots + V_n$. Let $W = \mathcal{W}(V, x_1, \dots, x_n)$. Consider $(W^{(x_1, \dots, x_n)})^\# = U_1 + \dots + U_n$ where $|U_i| = |V_i|$ for $i = 1, \dots, n$ and where each equivalence class r has been replaced by ξ_r .

Then we define:

$$\beta t = \Upsilon^W \Pi^{((|c_1|+|V_1|, \dots, |c_n|+|V_n|), (|V_1|, \dots, |V_n|))} [A] (\beta \lambda U_1 t_1 [U_1/V_1]) \dots (\beta \lambda V_n t_n [U_n/V_n]),$$

By $\lambda U_k t_k$ is understood that we place the variables U_k right after λ in t_k (if not a constant). If U_k is empty then this expression is just t_k .

In the case in which t_i is a variable x we have that $d_i = (x)$ and $\lambda d_i t_i [x/x]$ is $\lambda x x$ which is identified with x .

- If t is $\lambda x_1 \dots x_n \phi \wedge \psi$. Let V_1 be the free variables in ϕ ordered as they occur in $x_1 \dots x_n$ and V_2 the free variables in ψ ordered as they occur in $x_1 \dots x_n$. Let $V = V_1 + V_2$ and $W = \mathcal{W}(F, x_1 \dots x_n)$. Write $(W^{(x_1 \dots x_n)})^\# = U_1 + U_2$ where $|U_1| = |V_1|$ and $|U_2| = |V_2|$ and the equivalence classes r are replaced by ξ_r . Then we set

$$\beta t = \Upsilon^W \Lambda_{\wedge}^{(|V_1|, |V_2|)} (\beta \lambda V_1 \phi [V_1/U_1]) (\beta \lambda V_2 \psi [V_2/U_2])$$

- If t is $\lambda x_1 \dots x_n \neg \phi$, then $\beta t = \Lambda_{\neg}^n (\beta \lambda x_1 \dots x_n \phi)$.
- If t is of the form $\lambda x_1 \dots x_n \forall y \phi(y)$, then $\beta t = \Lambda_{\forall}^{n+1} (\beta \lambda y x_1 \dots x_n \phi)$.

Theorem 4.5 βt is well-defined NTL term and we have $\nu \beta t = t$.

Proof. By a straightforward induction on the structure of t .

Corollary 4.6 A closed complex term t of BL is always the result of the application of some composition of the syntactic operators π, σ, λ_* for $(* \in \{\neg, \wedge, \forall\})$ to $[A_1], \dots, [A_k]$ and possibly some variables and constants where $A_1, \dots, A_k$ are the atomic predicates occurring in t .

Note that in general the decomposition of a BL term into compositions of the syntactic operators is not unique. Consider $t = \lambda y x A(\lambda B(x, y))$. Then this term can be decomposed either in the form $\Upsilon^W \Pi^S[A][B]$ or $\Pi^T[A] \Upsilon^V[B]$.

Remark 4.7 Bealer in [1] used a semantics which can be described as a semantic version of a variant of NTL. Terms in BL are interpreted via a semantic version of β . Since β produces terms with Π -applications to atomic heads only, it is non-trivial (in the context of Bealer's theory) to show that the interpretation via β of $t(x)[s/x]$ corresponds to $\Pi^{((1),(1))}(\beta t)(\beta s)$.

4.1 Reductions

We will define a series of reductions for NTL. The central idea is to reduce the complexity of the head of a Π application and to organize Υ applications in a canonical way. Reductions will allow us to define the concepts of normalization and normal form for NTL terms. The goal is to show that T_1 and T_2 reduce to the same normal term N iff $\nu T_1 = \nu T_2$.

Reductions will be divided into

- Structural reduction: the merging of successive Υ -applications

- Predicative reductions, involving the removing of complex terms from the head of Π -applications and the simplification of trivial applications to series of I s.
- Pushing-in reductions.

We introduce the reductions progressively while also checking their ‘soundness’, that is, if the associated BL term remains the same. This will be used later on.

The structural reductions follow the rule:

$$(C_{\Upsilon}) \quad \Upsilon^{W_1} \Upsilon^{W_2} X \rightsquigarrow \Upsilon^{W_1 W_2} X$$

Predicative reductions have as goal to progressively simplify the head T of a Π application to the case in which the head is a primitive term. Thus we have reductions for each of the types of complex term T .

Let us start with the case in which T begins with Π . We postulate reductions of the form:

$$(R_{\Pi}) \quad \Pi^A(\Pi^B T T_1 \dots T_n) S_1 \dots S_m \rightsquigarrow \Pi^{(B \otimes A)^1} T(\Pi^{(B \otimes A)_1} T_1 \overline{S'}^1) \dots (\Pi^{(B \otimes A)_n} T_n \overline{S'}^n)$$

where for selectors $A = (A^1, A^2)$ and $B = (B^1, B^2)$ $S_1 \dots S_m / B^2 = \overline{S'}^1 + \dots + \overline{S'}^n$ and $\overline{S'}^i = \overline{S'}^i I^{(B_i^1 - B_i^2)}$.

For convenience we divide the corresponding reduction for Υ -applications into two cases, the general case following from the canonical decomposition of a weaver into the composition of a plectum and permutator.

Assume that T is a Υ^W application with W a plectum. Then we postulate:

$$(R_{\Upsilon}) \quad \Pi^{(A,B)}(\Upsilon^W T) S_1 \dots S_n \rightsquigarrow \Upsilon^{W \odot B} \Pi^{(W^\sharp A, L^\sharp B)} T W^\sharp(S_1 \dots S_n)$$

When the head T is an $\Lambda_{\neg}^n$ application we postulate:

$$(R_{\neg}) \quad \Pi^{(A,B)}(\Lambda_{\neg}^n T) S_1 \dots S_n \rightsquigarrow \Lambda_{\neg}^{\Sigma B}(\Pi^{(A,B)} T S_1 \dots S_n).$$

The case in which T is an $\Lambda_{\wedge}^{n,m}$ application is as follows. We postulate

$$(R_{\wedge}) \quad \Pi^{(A,B)}(\Lambda_{\wedge}^{n,m} T^{(n)} S^{(m)}) T_1 \dots T_{n+m} \rightsquigarrow \Lambda_{\wedge}^{\Sigma A_2, \Sigma B_2}(\Pi^{(A_1, B_1)} T T_1 \dots T_n)(\Pi^{(A_2, B_2)} S T_{n+1} \dots T_m)$$

Where for the selector $(A, B) \mid A \mid = n+m$. Here $A/(n, m) = A_1 + A_2$ and $B/(n, m) = B_1 + B_2$. For $\Lambda_{\vee}^n$ applications we postulate:

$$(R_{\vee}) \quad \Pi^{(A,B)}(\Lambda_{\vee}^{n+1} T^{(n+1)}) T_1 \dots T_n \rightsquigarrow \Lambda_{\vee}^{1+\Sigma B}(\Pi^{(1+A, 1+B)} T I T_1 \dots T_n)$$

The final predicative reduction is:

$$(R_I) \quad \Pi^{(1,1,\dots,1)} T I I \dots I \rightsquigarrow T$$

Suppose we have a term of the form $\Upsilon^W \Pi^A T S_1 \dots S_n$. By Theorem 2.21, W can be decomposed as $W_{out} W_{in}$ for W_{in} within the order partition $(1, 2, \dots, \Sigma S)/S$. Consider the decomposition induced by the partition: $W_{in} = \Sigma_i W_i$ with W_i of order (o_i, p_i) Then we postulate the reduction

$$(P_{\Pi}) \quad \Upsilon^W \Pi^{(A,B)} T S_1 \dots S_n \rightsquigarrow \Upsilon^{W_{out}} \Pi^{(A',B')} T (\Upsilon^{W_1+1^{A_1-B_1}} S_1) \dots (\Upsilon^{W_n+1^{A_n-B_n}} S_n)$$

where $A' = (A_1 - (o_1 - p_1), \dots, A_n - (o_n - p_n))$, $B' = (B_1 - (o_1 - p_1), \dots, B_n - (o_n - p_n))$.

Let W be a weaver of order (n, m) decomposed as $W_{out} W_{in}$ for the partition $(1, 2, \dots, n + m)/(n, m)$ with corresponding decomposition $W_{in} = W_1 + W_2$ of orders (n, o) and (m, p) . Then we postulate:

$$(P_{\wedge}) \quad \Upsilon^W \Lambda_{\wedge}^{n,m} T S \rightsquigarrow \Upsilon^{W_{out}} \Lambda_{\wedge}^{o,p} (\Upsilon^{W_1} S) (\Upsilon^{W_2} T).$$

The case of $\Upsilon^W \Lambda_{\neg}^n T$ for W of order (n, m) is:

$$(P_{\neg}) \quad \Upsilon^W \Lambda_{\neg}^n T \rightsquigarrow \Lambda_{\neg}^m \Upsilon^W T.$$

Finally for $\Upsilon^W \Lambda_{\vee}^n T$ we postulate:

$$(P_{\vee}) \quad \Upsilon^W \Lambda_{\vee}^n T \rightsquigarrow \Lambda_{\vee}^{m+1} \Upsilon^{1+W} T$$

where W has order $(n - 1, m)$.

Lemma 4.8 *The reduction C_{Υ} is sound.*

Proof. This amounts to showing that

$$v^W v^V \lambda X \phi(X) = v^{WV} \lambda X \phi(X)$$

for V of order (n, m) , W of order (m, k) and $|X| = n$. Assume V and W have been shifted so that $V^{\circ} = X$ and $W^{\circ} = V^{\flat}$. Then this amounts to showing that

$$[V^{\sharp}/X][W^{\sharp}/V^{\flat}] = [(VW)^{\sharp}/X]$$

and

$$(WV)^{\flat} = (W^{V^{\flat}})^{\flat} = W^{\flat}$$

Lemma 4.9 *The reduction R_{Π} is sound.*

Proof. Consider the term

$$\Pi^{(A,B)} (\Pi^{(C,D)} T T_1 \dots T_n) S_1 \dots S_m$$

with $|C| = n$ and $\Sigma D = m$. Assume

$$\nu(T) = \lambda X \phi$$

$$\nu(T_i) = \lambda Y^i \psi_i$$

$$\nu(S_j) = \lambda Z^j \rho_j$$

where X, Y^i and Z^j are sequences of distinct variables with $|Y^i| = C_i$ and $|Z^j| = A_j$.

Let

$$(O^1, \dots, O^n) = (C, D) \bullet (Y^1, \dots, Y^n), \quad (P^1, \dots, P^n) = (Y^1, \dots, Y^n) \bullet (C, D)$$

$$(Q^1, \dots, Q^m) = (A, B) \bullet (Z^1, \dots, Z^m), \quad (R^1, \dots, R^m) = (Z^1, \dots, Z^m) \bullet (A, B)$$

We have thus $Y^i = O^i + P^i$ and $Z^j = Q^j + R^j$. Now consider

$$\begin{aligned} (Q^1, \dots, Q^m)/D &= (U^1, \dots, U^n) \\ (R^1, \dots, R^m)/D &= (V^1, \dots, V^n) \\ (\rho_1, \dots, \rho_m)/D &= (\Gamma^1, \dots, \Gamma^n) \\ (Z^1, \dots, Z^m)/D &= (W^1, \dots, W^n) \end{aligned}$$

Now we have¹

$$\nu(\Pi^{(C,D)} TT_1 \dots T_n) = \lambda O^1 \dots O^n \phi[\lambda P^i \psi_i / X_i]_{i=1}^n$$

hence $\nu(\Pi^{(A,B)}(\Pi^{(C,D)} TT_1 \dots T_n) S_1 \dots S_m)$ is:

$$\lambda Q^1 \dots Q^m \phi[\lambda P^i \psi_i / X_i]_{i=1}^n [\lambda V_1^j \Gamma_1^j \dots \lambda V_{D_j}^j \Gamma_{D_j}^j / O_1^j \dots O_{D_j}^j]_{j=1}^n$$

But inspecting the substitution $[\lambda P^i \psi_i / X_i]_{i=1}^n [\lambda V_1^j \Gamma_1^j \dots \lambda V_{D_j}^j \Gamma_{D_j}^j / O_1^j \dots O_{D_j}^j]_{j=1}^n$ reveals it to be the same as the substitution

$$[[\lambda P^i \psi_i / X_i] [\lambda V_1^i \Gamma_1^i \dots \lambda V_{D_i}^i \Gamma_{D_i}^i / O_1^i \dots O_{D_i}^i]]_{i=1}^n$$

Let $t^i = \lambda U^i P^i \psi_i [\lambda V_1^i \Gamma_1^i \dots \lambda V_{D_i}^i \Gamma_{D_i}^i / O_1^i \dots O_{D_i}^i]$ for $i = 1, \dots, n$.

Then it is easy to check that

$$\nu(\Pi^{(A,B)}(\Pi^{(C,D)} TT_1 \dots T_n) S_1 \dots S_m) = \pi^{((C,D) \otimes (A,B))^1} (\lambda X \phi) t^1 \dots t^n$$

and that²

$$t^i = \pi^{((C,D) \otimes (A,B))_i} (\lambda Y^i \psi_i) (\lambda W_1^i \Gamma_1^i) \dots (\lambda W_{D_i}^i \Gamma_{D_i}^i) [\lambda x x]^{C_i - D_i}$$

and so the result follows.

Lemma 4.10 *The reduction R_Υ is sound.*

Proof. Let $\nu T = \lambda X \phi$ and $\nu S = \lambda Y^i \psi_i$ for $i = 1, \dots, n$. Let L be a weaver of order (m, n) . Write $Y^i = \leftarrow Y^i + \rightarrow Y^i$ where $|\leftarrow Y^i| = B_i$ and $|\rightarrow Y^i| = A_i - B_i$. Then

$$\begin{aligned} \nu(\Pi^{(A,B)}(\Upsilon^L T) S_1 \dots S_n) &= \pi^{(A,B)} (v^L \lambda X \phi) \lambda Y^1 \psi_1 \dots \lambda Y^n \psi_n \\ &= \pi^{A,B} \lambda (L^X)^\flat \phi[(L^X)^\sharp / X] \lambda Y^1 \psi_1 \dots \lambda Y^n \psi_n = \lambda \leftarrow Y^1 \dots \leftarrow Y^n \phi[(L^X)^\sharp / X] [\lambda \rightarrow Y^1 \psi_1 \dots \lambda \rightarrow Y^n \psi_n / (L^X)^\flat] \end{aligned}$$

But the composition of substitutions $[(L^X)^\sharp / X] [\lambda \rightarrow Y^1 \psi_1 \dots \lambda \rightarrow Y^n \psi_n / (L^X)^\flat]$ is evidently equivalent to the substitution³

$$[L^\sharp (\lambda \rightarrow Y^1 \psi_1 \dots \lambda \rightarrow Y^n \psi_n) / X]$$

Also observe that

$$\pi^{(L^\sharp A, L^\sharp B)} (\lambda X \phi) L^\sharp (\lambda Y^1 \psi_1 \dots \lambda Y^n \psi_n) = \lambda L^\sharp (\leftarrow Y^1 \dots \leftarrow Y^n) \phi[L^\sharp (\lambda \rightarrow Y^1 \psi_1 \dots \lambda \rightarrow Y^n \psi_n) / X]$$

Hence we get that

$$\nu(\Pi^{(A,B)}(\Upsilon^L T) S_1 \dots S_n) = v^{L \odot B} \pi^{(L^\sharp A, L^\sharp B)} (\lambda X \phi) L^\sharp (\lambda Y^1 \psi_1 \dots \lambda Y^n \psi_n) = \nu(\Upsilon^{L \odot B} \Pi^{(L^\sharp A, L^\sharp B)} T L^\sharp (S_1 \dots S_n))$$

¹We use the notation $[a_i/b_i]_{i=1}^n = [a_1 \dots a_n / b_1 \dots b_n]$ for simultaneous substitutions.

²In the formula below $[\lambda x x]^{C_i - D_i}$ denotes $C_i - D_i$ repeated occurrences of the argument $\lambda x x$.

³When we apply $L^\sharp$ to a sequence of λ -terms we always assume that for each pair of repeated terms the bound variables have been renamed so that they are all distinct in the resulting sequence.

Lemma 4.11 *The reductions R_Σ , $R_\wedge$, $R_\neg$, $R_\vee$ and R_I are sound.*

Proof. The proof that R_Σ is sound is similar to the case of R_Δ but simpler. The soundness of the other rules is straightforward. We show as an example the soundness of $R_\neg$. Let A be the selector (N, S) , $t = \lambda x_1 \dots x_n T(x_1, \dots, x_n)$ and $s_i = \lambda \bar{a}^i \bar{b}^i B_i(\bar{a}^i, \bar{b}^i)$ for $i = 1, \dots, n$ and $|\bar{a}^i| = S_i$ and $|\bar{b}^i| = N_i - S_i$.

Then a direct calculation yields

$$\nu(\Pi^A(\Lambda_\neg^n t) s_1 \dots s_n) = \lambda \bar{a}^1 \dots \bar{a}^n \neg A(x_1, \dots, x_n) [x_i / \lambda \bar{b}^i B_i(\bar{a}^i, \bar{b}^i)]_{i=1}^n$$

which is evidently equal to

$$\nu(\Lambda_\neg^{\Sigma S}(\Pi^A t s_1 \dots s_n))$$

Lemma 4.12 *The pushing-in reductions are sound.*

Proof. This follows from a straightforward computation.

Theorem 4.13 *The reductions of NTL are sound*

Proof. The result follows immediately from Lemmas 4.8–4.12.

4.2 Normalization

We use the notation $T \rightsquigarrow^* S$ to indicate that S is obtained from T by multiple applications of reductions to subterms of T

Definition 4.14 An NTL term is called *prenormal* if in all subterms of the form $\Pi T T_1 \dots T_n$ we have that T is a primitive term and there are no subterms of the form $\Pi T I \dots I$.

Note that applying only predicative reductions we always arrive in finitely many steps at a prenormal term. So prenormal terms do not allow further predicative reductions.

Remark 4.15 Given a closed BL term t it is clear that βt is prenormal but it need not be normal.

Definition 4.16 A term of the form $\Upsilon^W T$ where T starts with Π or Λ is called *pushed-in* if the pushing-in reductions cannot be applied.

Definition 4.17 A prenormal NTL term is called *normal* if it is pushed-in and the structural reductions cannot be applied.

It is evident that any sequence of reduction rules applied to subterms of a term T must eventually terminate. Furthermore, the soundness of the reductions permits us conclude directly that:

Lemma 4.18 *Given any NTL term T any sequence of applications of the reduction rules eventually terminates in a normal term N , $T \rightsquigarrow^* N$, and we have $\nu T = \nu N$.*

The main result of this paper is that such a normal term N is unique. To show this we need the following lemma:

Lemma 4.19 *Let N be a normal term. Then we have $\beta \nu N = N$.*

Proof. By induction on the structure of the normal term N . If N is primitive the result is clear. Assume that $N = \Upsilon^W \Pi^S AB_1 \dots B_n$. Then A must be primitive and W is without the partition defined by the selector S . Then we have

$$\beta \nu \Upsilon^W \Pi^S AB_1 \dots B_n = \beta \nu^W \pi^S(\nu A)(\nu B_1) \dots (\nu B_n)$$

Since W is without $S = (N, M)$ we have that ν^W preserves the order of the first M_i variables in νB_i when they appear in $\pi^S(\nu A)(\nu B_1) \dots (\nu B_n)$. By the definition of β and the uniqueness of a weaver without a partition we then have

$$\beta \nu^W \pi^S(\nu A)(\nu B_1) \dots (\nu B_n) = \Upsilon^W \Pi^S(\beta \nu A) \beta t'_1 \dots \beta t'_n$$

but $t'_i = \nu B_i$ because W is without S and so does not permute the first M_i variables after the λ . Hence the result follows by induction. The same argument applies to show the case in which $N = \Upsilon^W \Lambda_\lambda^{(n,m)} AB$. The other cases are simpler and straightforward.

Remark 4.20 We illustrate the main step of the proof above through a concrete example. We take the case of a normal term $N = \Upsilon^W \Pi^S AB_1 B_2$ in which $\nu(A) = \lambda xy A(x, y)$, $\nu(B_1) = \lambda x_1 x_2 B_1(x_2, x_1)$ and $\nu(B_2) = \lambda y_1 y_2 y_2 B_2(y_3, y_2, y_1)$, $S = ((2, 3), (2, 2))$ and $W = ((1, 2, 3, 4), (\{1, 3\}, \{2, 4\}))$ (which is clearly without the partition $((1, 2), (3, 4))$, and hence restricts to the identity - and so preserves the order - on the intervals of the partition⁴). Then a direct calculation of $\nu(N)$ yields

$$\nu^W \pi^S(\nu A)(\nu B_1)(\nu B_2) = \lambda \xi_{\{1,3\}} \xi_{\{2,4\}} A(\lambda B_1(\xi_{\{2,4\}}, \xi_{\{1,3\}}), \lambda y_3 B_2(y_3, \xi_{\{2,4\}}, \xi_{\{1,3\}}))$$

Now consider the application of β to the above term. By the definition of β the weaver W' for the resulting Υ constructor is obtained as $\mathcal{W}(\xi_{\{1,3\}} \xi_{\{2,4\}} \xi_{\{1,3\}} \xi_{\{2,4\}}, \xi_{\{1,3\}} \xi_{\{2,4\}})$ which is equal to W . The selector obtained from the definition of β is clearly equal to S and the terms that appear in the scope of β after the head argument of Π result from $\nu(B_1)$ and $\nu(B_2)$ by renaming bound variables. Thus, since we can easily check that $\beta \nu B_1 = B_1$, $\beta \nu B_2 = B_2$ and $\beta \nu A = A$ we obtain that

$$\beta \nu \Upsilon^W \Pi^S AB_1 B_2 = N$$

We can now prove the main result of this paper

Theorem 4.21 *Let T be NTL term. Then there is a unique normal term N such that $T \rightsquigarrow^* N$.*

Proof. Assume that we had normal terms N_1 and N_2 such that $T \rightsquigarrow^* N_1$ and $T \rightsquigarrow N_2$. Then $\nu T = \nu N_1$ and $\nu T = \nu N_2$ and thus $\nu N_1 = \nu N_2$. Since N_1 and N_2 are normal we may apply the lemma to obtain $\beta \nu N_1 = N_1$ and $\beta \nu N_2 = N_2$. But since $\nu N_1 = \nu N_2$ the result follows.

A consequence of the theorem is that we have an alternative method of computing the normal term of a NTL term T . Instead of applying reductions directly, we first compute the pre-normal term $\beta \nu T$ and then we need only apply structural and pushing-in reductions to obtain the unique normal term N . Also, given a closed BL term t we can define a variant β^* of β as $\beta^* t = N$ where N is the normal term associated to βt .

⁴and thus ν^W when restricted to $x_1 x_2$ and $y_1 y_2$ preserves the corresponding order of the variables so that these sequences are restored when β is calculated.

References

- [1] George Bealer. *Quality and Concept*. Oxford University Press, New York, 1982.
- [2] Alonzo Church. A formulation of the simple theory of types. *J. Symbolic Logic*, 5:56–68, 1940.
- [3] Haskell Brooks Curry, Robert Feys, and William Craig. *Combinatory logic Vol. 1 / Haskell B. Curry [and] Robert Feys ; with two sections by William Craig*. Studies in logic and the foundations of mathematics. North-Holland Publishing Company, Amsterdam, 1958.
- [4] Richard Montague. Universal grammar. *Theoria*, 36:373–398, 1970.
- [5] Willard V. Quine. Variables explained away. *Proceedings of the American Philosophical Society*, 104(3):343–347, 1960.
- [6] Edward N. Zalta. *Abstract objects: an introduction to axiomatic metaphysics*. Synthese library 160. Reidel, Dordrecht u.a, 1983.