Technological University, Eindhoven
Department of Mathematics
Notitie 22, mei 1972.

AUT27

The closure theorem in λ-typed λ-calculus

by R.P. Nederpelt.

1. Synopsis

The aim of this paper is to prove a theorem (the closure theorem) in lambda-typed lambda-calculus which states that the set of so-called legal expressions is closed under β-reduction. The proof of this theorem will be given in detail.

The theory of lambda-typed lambda-calculus itself will be described in a nutshell, with a minimum of comment. We shall only give definitions necessary for the main theorem and we shall describe roughly some indispensable notions of the theory. Moreover we shall list a number of lemma's needed in the main theorem. These lemma's, which are theorems from lambda-typed lambda-calculus, will be given without proof. The theory of lambda-typed lambda-calculus will be described amply in [1].

2. The language of λ-typed λ-calculus

2.1 The *alphabet* of the language consists of an infinite set of *variables*, a *basic constant* (τ), a number of *improper symbols* (the brackets '[', ']', '{', '}' and the comma), and some symbols denoting functions and relations.

2.2 *Expressions* of the language are defined as follows:

(1) A variable is an expression; τ is an expression.

(2) If A and B are expressions and x is a variable, then $[x,A]B$ and $\{A\}B$ are expressions.

The expression $[x,A]B$ is equivalent to λxB in λ-calculus, where A is the *type* of x. The expression $\{A\}B$ is equivalent to BA in λ-calculus.

An expression A can be a *subexpression* of an expression B (denoted as $A \subset B$). A recursive definition of subexpression is:

(1) $A \subset A$.

(2) $C \subset A$ or $C \subset B \Rightarrow C \subset [x,A]B$.

(3) $C \subset A$ or $C \subset B \Rightarrow C \subset \{A\}B$.

2.3 Parts of expressions of the form $[x,A]$ or $\{A\}$ are called *abstractors* or *applicators* respectively. In meta-language we use the capitals A, B, C,... for expressions, Q for strings of abstractors, and P for strings consisting of abstractors and applicators. (A string is the result of juxtaposition; a string may be empty.) For variables we use in meta-language the lower-case letters x, y, z,...

Juxtaposition of letters in meta-language means juxtaposition of the objects they represent. If two meta-letters A and B represent the same object then we put $A \equiv B$; if not, then $A \not\equiv B$.

All meta-symbols can be primed or indexed.

2.4 Variables occurring in an expression immediately following an opening bracket '[' are called *binding*. All other variables occurring in an expression are called *non-binding*. The binding x in the expression $[x,A]B$ *binds* all those non-binding x's in B which are not yet bound by another binding x. Those non-binding x's are then called *bound* (by the binding x in the abstractor $[x,A]$). All non-binding variables which are not bound are *free*.

2.5 A *bound expression* is an expression which does not contain free variables. A *closed expression* is a bound expression in which all binding variables are different. The set of all closed expressions is denoted by Γ. We only use bound expressions which are closed in order to avoid misunder andings as to the binding of variables. This has also its advantages while manipulating expressions. For the theory this is not restriction at all.

2.6 An equivalent definition for closed expressions is the following recursive one (where we write BV(A) for the set of binding variables in A):

(1) $\tau \in \Gamma$.

(2) If $QA \in \Gamma$ and if x does not occur in QA then $Q[x,A]x \in \Gamma$.

(3) If QA and $QB \in \Gamma$ and if $BV(A) \cap BV(B) = \phi$ then $Q\{A\}B \in \Gamma$.

(4) If QA and $Q\tau \in \Gamma$ and if x does not occur in QA then $Q[x,A]\tau \in \Gamma$.

(5) If QA and $Qy \in \Gamma$, if x does not occur in QA and if $x \not\equiv y$, then $Q[x,A]y \in \Gamma$.

By means of this definition we can construct every closed expression in a unique way.

2.7 Let x and y be variables. We say "replace x with y in A" if we want that every x occurring in A is replaced by a y. We define the operator Ref working on an expression A to have the following effect: each variable, which occurs as a binding variable in A, has to be replaced with a 'fresh', never having been used variable. For example, if y_1 and y_2 have not yet been used as variables, and $A \equiv [x_1,\tau][x_2,x_1]\{x_2\}x_3$ then Ref A can be $[y_1,\tau][y_2,y_1]\{y_2\}x_3$.

By defining an order in the set of all variables and by prescribing that Ref uses the 'fresh' variables in this order we can procure that Ref is a uni-valued function, so the expression Ref A is uniquely defined at the spot where we introduce it. But we agree that this same particular expression is also meant when we use 'Ref A' <u>after</u> this spot of introduction, as long as we use it in the same context.

If we need other "fresh variables", in the same expression A, then we use an index with 'Ref': Ref A uses the first set of fresh variables, $\mathrm{Ref}_1 A$ another one.

2.8 The transformation *α-reduction* (denoted by '$\geq_\alpha$') is the transitive transformation generated by:
"If $A \in \Gamma$, if y is a variable not occurring in A, and if B is the expression obtained from A by replacing the variable x by y in A, then $A \geq_\alpha B$."

Since we did not require x to occur in A it follows that α-reduction is reflexive. Obviously α-reduction is symmetric, too. So α-reduction generates an equivalence-relation. It is also clear that $A \geq_\alpha$ Ref A.

An α-reduction with one variable or none renamed as described between the apostrophes of the definition is called a *one-step* α-reduction and denoted as $A \geq'_\alpha B$.

We state the following theorems:
(i) If $A \in \Gamma$ and $A \geq_\alpha B$ then $B \in \Gamma$,
(ii) corollary: if $A \in \Gamma$ then Ref $A \in \Gamma$.

2.9 Substitution of A for x in B is denoted by $(x := A)B$ and defined as follows:
(1) $(x := A)x \equiv \mathrm{Ref}\ A$; $(x := A)y \equiv y$ if $y \not\equiv x$; $(x := A)\tau \equiv \tau$
(2) $(x := A)[x,B]C \equiv [x,B]C$; $(x := A)[y,B]C \equiv [y,(x := A)B](x := A)C$ if $y \not\equiv x$
(3) $(x := A)\{B\}C \equiv \{(x := A)B\}(x := A)C$.

For convenience we also define $(x := A)P$ for a string P of abstractors and applicators:

If $(x := A)P\tau \equiv P'\tau$, then $(x := A)P \equiv P'$.

$A \in \Gamma$, $B \in \Gamma \Rightarrow (x := A)B \in \Gamma$

2.10 The relation *β-reduction* (denoted by '$\geq_\beta$') is the reflexive and transitive relation generated by:

(1) If $Q\{A\}[x,B]C \in \Gamma$ then $Q\{A\}[x,B]C \geq_\beta Q(x := A)C$.
(2) If QC and $Q\{A\}C \in \Gamma$ and if $QC \geq_\beta QD$ then $Q\{A\}C \geq_\beta Q\{A\}D$.
(3) If QA and $Q[x,A]C \in \Gamma$ and if $QA \geq_\beta QB$ then $Q[x,A]C \geq_\beta Q[x,B]C$.
(4) If QA and $Q\{A\}C \in \Gamma$ and if $QA \geq_\beta QB$ then $Q\{A\}C \geq_\beta Q\{B\}C$.

We recall that Q denotes a string of abstractors.

We call rule (2) to (4) the *monotony-rules* of β-reduction.

A β-reduction $A \geq_\beta B$ obtained by one application of rule (1) followed by a number of applications of the monotony-rules is called a *one-step* β-reduction and denoted as $A \geq'_\beta B$.

If A reduces to B by a sequence of α- and β-reductions then we write $A \geq B$.

We state the following theorems:

(i) If $A \in \Gamma$ and $A \geq_\beta B$ then $B \in \Gamma$.
(ii) If QC and $Q[x,A]C \in \Gamma$ and if $QC \geq_\beta QD$ then $Q[x,A]C \geq_\beta Q[x,A]D$.
(This last theorem is apparently missing in the set of monotony-rules for β-reduction; it turns out to be a consequence of the rules of β-reduction.)

2.11 Let A and $B \in \Gamma$. We define an equivalence between A and B ($A \sim B$) as follows:

$A \sim B$ if there exists a D such that $A \geq D$ and $B \geq D$.

The transitivity of this equivalence is implied by the Church-Rosser theorem, which holds also for our special λ-calculus. (We can also prove Church-Rosser directly; cf. [2]).

2.12 If $A \in \Gamma$ and $A \equiv P_1[x,B]P_2x$, then we define $\text{Typ}\, A \equiv P_1[x,B]P_2 \text{ Ref } B$. Analogously, if $\text{Typ}^n A \equiv P_1[x,B]P_2x$ for some $n \geq 1$ then $\text{Typ}^{n+1} A \equiv P_1[x,B]P_2 \text{ Ref } B$. We define $\text{Typ}^0 A \equiv A$.

The following theorem holds:

(i) If $A \in \Gamma$ and Typ A is defined then $\text{Typ}\, A \in \Gamma$.

2.13 For each expression in Γ we define a number called class (Cl):
(1) If $A \equiv P\tau$ then $\text{Cl}\, A = 1$;
(2) If $A \equiv Px$ then $\text{Cl}\, A = \text{Cl}(\text{Typ}\, A) + 1$.

The theorem holds:

(i) If $A \in \Gamma$ then $\text{Typ}^{\text{Cl}A-1} A \equiv P\tau$.

This "highest possible power" of Typ has a special significance. We abbreviate $\text{Typ}^{\text{Cl}A-1} A$ as $\text{Typ}^* A$.

2.14 Let $A \in \Gamma$. We define a norm ν on A by recursion with respect to the subexpressions of A:

(1) If $\tau \subset A$ then $\nu(\tau) \equiv \tau$.

(2) If $x \subset A$ and x is bound by the binding x in the abstractor $[x,B]$ then $\nu(x) \equiv \nu(B)$.

(3) If $[x,B]C \subset A$ then $\nu([x,B]C) \equiv [x,\nu(B)]\nu(C)$.

(4) If $\{B\}C \subset A$ then $\nu(\{B\}C) \equiv \{\nu(B)\}\nu(C)$.

We can prove that $\nu(A)$ is well-defined for every $A \in \Gamma$. It is obvious that $\nu(A)$ contains no bound variables. The most important property of this norm is stated in 3.4 (XIII).

We state the theorem:

If $A \in \Gamma$ then $\nu(A) \in \Gamma$; if $A \geq_\alpha B$ then $\nu(A) \geq_\alpha \nu(B)$.

2.15 In the alternative definition of closed expression, given in 2.6, we find:

"(3) If QA and $QB \in \Gamma$ and if $BV(A) \cap BV(B) = \emptyset$ then $Q\{A\}B \in \Gamma$". The common interpretation of $\{A\}B$ is: the function B applied to the argument A. Clearly we put in Γ no restrictions on the argument A with respect to the domain of B. In a subset of Γ, called the set of *legal expressions*, abbreviated as Λ, we want the argument of a function to "match" with the domain of that function.

As definition of legal expressions we use the definitions of 2.6 (1), (2), (4) and (5), where we read Λ instead of Γ, and a new (3) which reads:

(3) If QA and $QB \in \Lambda$, *if* $Typ^* QB \geq Q[y,K]L$ *and* $Typ\ QA \geq QK$ *for certain* K *and* L, and if $BV(A) \cap BV(B) = \emptyset$ then $Q\{A\}B \in \Lambda$.

The condition printed in italics is called the *applicability condition*. We can weaken this applicability condition: see 3.5 (XIV).

3. A list of lemma's

3.1 A lemma concerning Γ:

I. $Q[x,A]B \in \Gamma$, $x \notin B \Rightarrow QB \in \Gamma$.

3.2 Lemma's concerning reduction and equivalence:

II. QC, $QPC \in \Gamma$, $QC \geq QD \Rightarrow QPC \geq QPD$.

III. $Q[x,A]B \in \Gamma$, $Q[x,A]B \geq Q[x,C]D \Rightarrow QA \geq QC$.

IV. PA, $P[x,B]\text{Ref } D \in \Gamma \Rightarrow P \text{ Ref}(x := A)D \geq_\alpha P(x := A)\text{Ref}_1 D$.

3.3 Lemma's concerning Typ and Cl:

N.B. If we write $\mathrm{Typ}^n A$ then we assume that $\mathrm{Typ}^n A$ is defined (i.e. $n \leq \mathrm{Cl}\ A - 1$); we call such a value of n *valid*.

V. $A \in \Gamma,\ A \geq_\alpha B \Rightarrow \mathrm{Typ}^n A \geq_\alpha \mathrm{Typ}^n B$.

VI. $PB,\ PP'B \in \Gamma \Rightarrow \mathrm{Cl}\ PB = \mathrm{Cl}\ PP'B$.

VII. $A \in \Gamma,\ A \geq_\alpha B \Rightarrow \mathrm{Cl}\ A = \mathrm{Cl}\ B$.

VIII. $PA \in \Gamma \Rightarrow \mathrm{Typ}^n PA \equiv PA'$.

IX. $PA,\ PP'A \in \Gamma$: $\mathrm{Typ}^n PA \geq_\alpha PA' \Leftrightarrow \mathrm{Typ}^n PP'A \geq_\alpha PP'A'$.

X. $QA,\ QPA \in \Gamma,\ \mathrm{Typ}^n QA \geq QB \Rightarrow \mathrm{Typ}^n QPA \geq QPB$.

XI. $QA,\ QB,\ QPA,\ QPB \in \Gamma,\ \mathrm{Typ}^n QA \sim \mathrm{Typ}^n QB \Rightarrow \mathrm{Typ}^n QPA \sim \mathrm{Typ}^n QPB$.

3.4 Lemma's concerning the norm ν:

XII. $A \in \Gamma,\ B \subset A \Rightarrow \nu(B)$ shorter than $\nu(A)$.

XIII. $A \in \Gamma \Rightarrow \nu(A) \geq_\alpha \nu(\mathrm{Typ}^n A)$.

3.5 Lemma's concerning Λ:

XIV. $QA,\ QB \in \Lambda,\ \mathrm{Typ}^* QB \sim Q[y,K]L$ and $\mathrm{Typ}\ QA \sim QK$ for certain K and L, and $BV(A) \cap BV(B) = \emptyset \Rightarrow Q\{A\}B \in \Lambda$.

XV. $A \in \Lambda,\ A \geq_\alpha B \Rightarrow B \in \Lambda$.

XVI. $A \in \Lambda \Rightarrow \mathrm{Typ}^n A \in \Lambda$.

XVII. $QA,\ QB \in \Lambda,\ Q[x,A]B \in \Gamma \Rightarrow Q[x,A]B \in \Lambda$.

XVIII. $Q\{A\}B \in \Lambda \Rightarrow QA,\ QB \in \Lambda$.

XIX. $Q[x,A]B \in \Lambda \Rightarrow QA \in \Lambda$.

XX. $Q[x,A]B \in \Lambda,\ QB \in \Gamma \Rightarrow QB \in \Lambda$.

XXI. $Q[x,A]C,\ QB \in \Lambda,\ \mathrm{Cl}\ QA = \mathrm{Cl}\ QB,\ \mathrm{Typ}^n QA \sim \mathrm{Typ}^n QB$ for all valid $n \Rightarrow Q[x,B]C \in \Lambda$.

XXII. $Q\{A\}[x,B]C,\ Q[x,B]D \in \Lambda \Rightarrow Q\{A\}[x,B]D \in \Lambda$.

4. The closure theorem

4.1 Definition: We call two expressions E and F *co-legal* if (1) $E \in \Lambda$, $F \in \Lambda$, (2) $\mathrm{Typ}^k E \sim \mathrm{Typ}^k F$ for all valid k and (3) $\mathrm{Cl}(E) = \mathrm{Cl}(F)$.

<u>Lemma A</u>: If E and F are co-legal, $E \geq_\alpha E'$ and $F \geq_\alpha F'$, then E' and F' are co-legal.

<u>Proof</u>: By Lemma (XV): $E' \in \Lambda$ and $F' \in \Lambda$.
By (V): $\mathrm{Typ}^k E \geq_\alpha \mathrm{Typ}^k E'$ and $\mathrm{Typ}^k F \geq_\alpha \mathrm{Typ}^k F'$. So $\mathrm{Typ}^k E' \sim \mathrm{Typ}^k F'$.
By (VII): $\mathrm{Cl}\ E = \mathrm{Cl}\ E'$ and $\mathrm{Cl}\ F = \mathrm{Cl}\ F'$, hence $\mathrm{Cl}\ E' = \mathrm{Cl}\ F'$.

<u>Note</u>: Co-legality is an equivalence relation.

4.2

<u>Lemma B</u>: Let E and F be co-legal and $E \geq_\beta F$. If $E' \geq_\beta F'$ is a direct consequence of $E \geq_\beta F$ according to one of the monotony rules for β-reduction (i.e. rule (2), (3), (4) of 2.10), and if $E' \in \Lambda$ then E' and F' are co-legal.

<u>Proof</u>: *case 1:* $E \geq_\beta F$ is $QC \geq_\beta QD$, $E' \equiv Q\{A\}C \in \Lambda$ and $E' \geq_\beta F' \equiv Q\{A\}D$.

(1) Since $Q\{A\}C \in \Lambda$: $QA \in \Lambda$, $\mathrm{Typ}^* QC \geq Q[x,M]N$ and $\mathrm{Typ}\ QA \geq QM$. If $\mathrm{Typ}^* QC \equiv \mathrm{Typ}^p QC$, then $\mathrm{Typ}^* QD \equiv \mathrm{Typ}^p QD$ since $\mathrm{Cl}(QC) = \mathrm{Cl}(QD)$. Hence $\mathrm{Typ}^* QC \sim \mathrm{Typ}^* QD$, so $\mathrm{Typ}^* QD \sim Q[x,M]N$. From (XIV): $Q\{A\}D \in \Lambda$.

(2) (XI) implies: $\mathrm{Typ}^k Q\{A\}C \sim \mathrm{Typ}^k Q\{A\}D$ for all valid k.

(3) Moreover: $\mathrm{Cl}\ Q\{A\}C = \mathrm{Cl}\ QC = \mathrm{Cl}\ QD = \mathrm{Cl}\ Q\{A\}D$ by (VI).

case 2: $E \geq_\beta F$ is $QA \geq_\beta QB$, $E' \equiv Q[x,A]C \in \Lambda$ and $E' \geq_\beta F' \equiv Q[x,B]C$. We prove co-legality for $Q[x,A]C$ and $Q[x,B]C$.

(1) From (XXI): $Q[x,B]C \in \Lambda$.

(2) We now prove the
"<u>lemma</u>: If $Q[x,A]C \in \Lambda$ then $\mathrm{Typ}^k Q[x,A]C \sim \mathrm{Typ}^k Q[x,B]C$ (for all valid k) and $\mathrm{Cl}\ Q[x,A]C = \mathrm{Cl}\ Q[x,B]C$"
by induction on $\mathrm{Cl}\ Q[x,A]C$. If $\mathrm{Cl}\ Q[x,A]C = 1$ then this is trivial. So assume $\mathrm{Typ}^k Q[x,A]H \sim \mathrm{Typ}^k Q[x,B]H$ (for all valid k) and $\mathrm{Cl}\ Q[x,A]H = \mathrm{Cl}\ Q[x,B]H$ for all H such that $Q[x,A]H \in \Lambda$ and $\mathrm{Cl}\ Q[x,A]H < \mathrm{Cl}\ Q[x,A]C$.

subcase 2.1: Assume $Q[x,A]C \equiv Q[x,A]Py$ where $y \not\equiv x$. Then the final y's in $Q[x,A]C$ and $Q[x,B]C$ are bound by binding y's of the same abstractors $[y,E]$, so $\mathrm{Typ}\ Q[x,A]C \equiv Q[x,A]P\ \mathrm{Ref}_1 E \in \Lambda$ (by (XVI)) and $\mathrm{Typ}\ Q[x,B]C \ \square\ Q[x,B]P\ \mathrm{Ref}_2 E$ and $\mathrm{Cl}\ Q[x,A]P\ \mathrm{Ref}_1 E = \mathrm{Cl}\ Q[x,A]C - 1$.

So by induction: $\mathrm{Typ}^\ell Q[x,A]P\ \mathrm{Ref}_1 E \sim \mathrm{Typ}^\ell Q[x,B]P\ \mathrm{Ref}_1 E \geq_\alpha \mathrm{Typ}^\ell Q[x,B]P\ \mathrm{Ref}_2 E$ for all valid $\ell \geq 0$. Hence $\mathrm{Typ}^k Q[x,A]C \sim \mathrm{Typ}^k Q[x,B]C$ for all valid $k \geq 1$.

(by (V))

But for $k = 0$ the last equivalence also holds. By induction, too:
$\mathrm{Cl}\ Q[x,A]P\ \mathrm{Ref}_1 E = \mathrm{Cl}\ Q[x,B]P\ \mathrm{Ref}_1 E = \mathrm{Cl}\ Q[x,B]P\ \mathrm{Ref}_2 E$ (the last equality by (VII)), so also $\mathrm{Cl}\ Q[x,A]C = \mathrm{Cl}\ Q[x,B]C$.

subcase 2.2: Assume $Q[x,A]C \equiv Q[x,A]Px$. Then $\mathrm{Typ}\ Q[x,A]C \equiv Q[x,A]P\ \mathrm{Ref}\ A \in \Lambda$ (by XVI)) and $\mathrm{Typ}\ Q[x,B]C \equiv Q[x,B]P\ \mathrm{Ref}\ B$. Now $\mathrm{Cl}\ Q[x,A]P\ \mathrm{Ref}\ A = \mathrm{Cl}\ Q[x,A]C - 1$, so by induction: $\mathrm{Typ}^{\ell} Q[x,A]P\ \mathrm{Ref}\ A \sim \mathrm{Typ}^{\ell} Q[x,B]P\ \mathrm{Ref}\ A$ for all valid $\ell \geq 0$. By lemma A: $Q\ \mathrm{Ref}\ A$ and $Q\ \mathrm{Ref}\ B$ are co-legal, xo $\mathrm{Typ}^{\ell} Q\ \mathrm{Ref}\ A \sim \mathrm{Typ}^{\ell} Q\ \mathrm{Ref}\ B$.

(XI) then implies that $\mathrm{Typ}^{\ell} Q[x,B]P\ \mathrm{Ref}\ A \sim \mathrm{Typ}^{\ell} Q[x,B]P\ \mathrm{Ref}\ B$ for all valid ℓ. It follows that $\mathrm{Typ}^{k} Q[x,A]C \sim \mathrm{Typ}^{k} Q[x,B]C$ for all valid $k \geq 1$. For $k = 0$ the last equivalence also holds.

Moreover: $\mathrm{Cl}\ Q[x,A]P\ \mathrm{Ref}\ A = \mathrm{Cl}\ Q\ \mathrm{Ref}\ A$ (by (VI)) $= \mathrm{Cl}\ Q\ \mathrm{Ref}\ B$ (by the co-legality of $Q\ \mathrm{Ref}\ A$ and $Q\ \mathrm{Ref}\ B$) $= \mathrm{Cl}\ Q[x,B]P\ \mathrm{Ref}\ B$ (by (VI) again), so also $\mathrm{Cl}\ Q[x,A]C = \mathrm{Cl}\ Q[x,B]C$.

case 3: $E \geq_\beta F$ is $QA \geq_\beta QB$, $E' \equiv Q\{A\}C \in \Lambda$ and $E' \geq_\beta F \equiv Q\{B\}C$.
We prove co-legality for $Q\{A\}C$ and $Q\{B\}C$.

(1) Since $Q\{A\}C \in \Lambda$: $QA \in \Lambda$, $\mathrm{Typ}^{*}QC \geq Q[x,M]N$ and $\mathrm{Typ}\ QA \geq QM$. Also $\mathrm{Typ}\ QA \sim \mathrm{Typ}\ QB$, so $\mathrm{Typ}\ QB \sim QM$. From (XIV): $Q\{B\}C \in \Lambda$.

(2) Assume $\mathrm{Typ}^{k}QC \equiv QC'$ (cf. (VIII)). Then $\mathrm{Typ}^{k}Q\{A\}C \geq_\alpha Q\{A\}C'$(by (IX)) $\in \Lambda$ (by (XVI)) and $\mathrm{Typ}^{k}Q\{B\}C \geq_\alpha Q\{B\}C'$ (by (IX)).
From $QA \geq QB$ follows $Q\{A\}C' \geq Q\{B\}C'$, so $\mathrm{Typ}^{k}Q\{A\}C \geq \mathrm{Typ}^{k}Q\{B\}C$.
Moreover $\mathrm{Cl}\ Q\{A\}C = \mathrm{Cl}\ QC = \mathrm{Cl}\ Q\{B\}C$ by (VI).

4.3 Main theorem: If $K \in \Lambda$ and $K \geq_\beta L$ then K and L are co-legal.

Proof: We regard the theorem as a proposition about K and refer to it as Prop (K). We prove the theorem by induction on the length of $\nu(K)$. If $K \equiv \tau$ then the theorem is trivial. So assume that Prop (H) is true for all H with $\nu(H)$ shorter than $\nu(K)$ (induction hypothesis 1). Under this assumption we first prove:

central lemma: If $K \equiv Q\{A\}[x,B]C \geq_\beta L \equiv Q\ (x := A)C$ and $\mathrm{Cl}\ K = \mathrm{Cl}\ L$ then $L \in \Lambda$.

proof of the central lemma: Assume $C \ \square\ [y_1,C_1] \ldots [y_k,C_k]P's$, where $P' \not\equiv [z,E]F$ and $s \equiv \tau$, $s \equiv y$ ($\not\equiv x$) or $s \equiv x$.

case 1: $P' \equiv \emptyset$, $k = 0$. Then $C \equiv s$ and $L \ \square\ Qs$ if $s \not\equiv x$ or $L \equiv Q\ \mathrm{Ref}\ A$ if $s \equiv x$. From $K \in \Lambda$ follows by (XVIII): $QA \in \Lambda$, so also

$Q\,\mathrm{Ref}\,A \in \Lambda$ (by (XV)). Also from $K \in \Lambda$ follows by (XVIII): $Q[x,B]s \in \Lambda$, so if $s \not\equiv x$ then (I) and (XX) imply $Qs \in \Lambda$. Hence always $L \in \Lambda$.

case 2: $P' \equiv \emptyset$, $k > 0$. Now $K \equiv Q\{A\}[x,B][y_1,C_1] \ldots [y_k,C_k]s$
$L \equiv Q[y_1,(x := A)C_1] \ldots [y_k,(x := A)C_k](x := A)s$. From $K \in \Lambda$ follows by (XVIII): $Q[x,B]C \in \Lambda$, so by (XIX): $Q[x,B][y_1,C_1] \ldots [y_{k-1},C_{k-1}]C_k \in \Lambda$.

From (XXII) follows that also $Q\{A\}[x,B][y_1,C_1] \ldots [y_{k-1},C_{k-1}]C_k \in \Lambda$. Since the last expression has a shorter ν than K by (XII) we can apply ind. hyp. 1 to find:

$$Q[y_1,(x := A)C_1] \ldots [y_{k-1},(x := A)C_{k-1}](x := A)C_k \in \Lambda. \qquad (1)$$

subcase 2.1: If $s \equiv y_k$ then $L \equiv$
$Q[y_1,(x := A)C_1] \ldots [y_{k-1},(x := A)C_{k-1}][y_k,(x := A)C_k]y_k \in \Lambda$.

subcase 2.2: Let $s \not\equiv y_k$. $Q[x,B]C \in \Lambda$, so if $s \not\equiv y_k$ then (I) and (XX) imply that $Q[x,B][y_1,C_1] \ldots [y_{k-1},C_{k-1}]s \in \Lambda$, so from (XXII): $Q\{A\}[x,B][y_1,C_1] \ldots [y_{k-1},C_{k-1}]s \in \Lambda$. The last expression has a shorter ν than K by (XII), so $Q[y_1,(x := A)C_1] \ldots [y_{k-1},(x := A)C_{k-1}](x := A)s \in \Lambda$ (2)
Now (1) and (2) imply that $L \in \Lambda$ (by (XVII)).

case 3: $P' \equiv \{E\}P''$. Abbreviate $[y_1,C_1] \ldots [y_k,C_k] \equiv Q'$ and $[y_1,(x := A)C_1] \ldots [y_k,(x := A)C_k] \equiv (x := A)Q'$. Then $K \equiv Q\{A\}[x,B]Q'\{E\}P''s \geq L \equiv Q(x := A)Q'\{(x := A)E\}(x := A)(P''s)$. Since $K \in \Lambda$: $Q[x,B]Q'\{E\}P''s \in \Lambda$, so also $Q[x,B]Q'E \in \Lambda$, $Q[x,B]Q'P''s \in \Lambda$, $\mathrm{Typ}^* Q[x,B]Q'P''s \geq Q[x,B]Q'[z,M']N'$ and $\mathrm{Typ}\, Q[x,B]Q'E \geq Q[x,B]Q'M'$.

By (XXII): $Q\{A\}[x,B]Q'P''s$ and $Q\{A\}[x,B]Q'E \in \Lambda$. Both ~~having~~ have a shorter ν than K by (XII), so $Q(x := A)Q'(x := A)(P''s)$ and $Q(x := A)Q'(x := A)E \in \Lambda$. Moreover by (VI) and (X): $\mathrm{Typ}^* Q\{A\}[x,B]Q'P''s \geq Q\{A\}[x,B]Q'[z,M']N' \geq Q(x := A)Q'[z,(x := A)M'](x := A)N'$ and $\mathrm{Typ}\, Q\{A\}[x,B]Q'E \geq Q\{A\}[x,B]Q'M' \geq Q(x := A)Q'(x := A)M'$. Combining these results with ind. hyp. 1 we get:
$Q(x := A)Q'(x := A)M' \sim \mathrm{Typ}\, Q\{A\}[x,B]Q'E \sim \mathrm{Typ}\, Q(x := A)Q'(x := A)E$ and
$Q(x := A)Q'[z,(x := A)M'](x := A)N' \sim \mathrm{Typ}^* Q\{A\}[x,B]Q'P''s \sim \mathrm{Typ}^* Q(x := A)Q'(x := A)(P''s)$. (In the proof of) ~~For~~ this last equivalence we ~~need~~ use the fact that $\mathrm{Cl}\,K = \mathrm{Cl}\,L$, so $\mathrm{Cl}\,Q\{A\}[x,B]Q'P''s = \mathrm{Cl}\,Q(x := A)Q'(x := A)(P''s)$ by (VI).

It follows that $Q(x := A)Q'\{(x := A)E\}(x := A)(P''s) \equiv L \in \Lambda$ by (XIV).
End of the proof of the central lemma.

We shall now prove the main theorem (still assuming ind. hyp. 1):

case 1: $K \geq_\beta L$ is $Q\{A\}[x,B]C \geq_\beta Q(x := A)C$.
We proceed by induction on Cl K. If Cl K = 1 then Cl L = 1 so $L \in \Lambda$ by the central lemma. Moreover $\text{Typ}^k K \sim \text{Typ}^k L$ since k can only be zero. So in that case K and L are co-legal.

So assume (ind. hyp. 2): <u>K' and L' are co-legal if Cl(K') < Cl(K) and if $K' \geq L'$ is $Q'\{A'\}[x,B']C' \geq_\beta Q'(x := A')C'$</u>.

subcase 1.1: Let $K \equiv Q\{A\}[x,B]Py$, where $y \not\equiv x$. Then $L \equiv Q((x := A)P)y$. If the final y in K is bound by the binding y in the abstractor [y,D], then $\text{Typ } K \equiv Q\{A\}[x,B]P \text{ Ref } D$. Also: the final y in L is bound by the binding y in the abstractor [y,(x := A)D], where the substitutor (x := A) is perhaps superfluous, so $\text{Typ } L \equiv Q((x := A)P) \text{ Ref } (x := A)D \geq_\alpha Q((x := A)P)(x := A) \text{ Ref}_1 D$ by (IV). Hence $\text{Typ } K \geq_\alpha Q\{A\}[x,B]P \text{ Ref}_1 D \geq_\beta Q((x := A)P)(x := A)\text{Ref}_1 D \geq_\alpha \text{Typ } L$.

Since Cl Typ K = Cl K - 1 it follows from ind. hyp. 2 and lemma A that Typ K and Typ L are co-legal. So $\text{Typ}^k K \sim \text{Typ}^k L$ for all $k \geq 1$, but this equivalence also holds for k = 0. Moreover Cl Typ K = Cl Typ L, also Cl K = Cl L. But the central lemma then implies that $L \in \Lambda$. So K and L are co-legal.

subcase 1.2: Let $K \equiv Q\{A\}[x,B]Px$. Then $L \equiv Q((x := A)P) \text{ Ref } A$ and $\text{Typ } K \equiv Q\{A\}[x,B]P \text{ Ref } B$. Since $K \in \Lambda$: $QA \in \Lambda$, $Q[x,B]Px \in \Lambda$, $\text{Typ}^* Q[x,B]Px \geq Q[x,M]L$ (so $QB \geq QM$ by (III) and (VIII)) and $\text{Typ } QA \geq QM$.

Since also $Q \text{ Ref } B \geq QM$: $\text{Typ } K \geq Q\{A\}[x,B]PM$ (by (II)) $\geq$ $Q((x := A)P)(x := A)M \equiv Q((x := A)P)M$. The last identity holds because x does not occur in B, and consequently not in M. Put $\text{Typ } Q \text{ Ref } A \equiv QD$, then also $QD \geq QM$ by (V), so $\text{Typ } L \geq_\alpha Q((x := A)P)D$ (by (IX)) $\geq Q((x := A)P)M$ by (II).

Now $\nu(Q \text{ Ref } B)$ is shorter than $\nu(K)$ by (XII) and $\nu(Q \text{ Ref } A) = \nu(QD)$ (by (XIII)) which is shorter than $\nu(K)$, so by ind. hyp. 1:
$\text{Typ}^\ell Q \text{ Ref } B \sim \text{Typ}^\ell QM \sim \text{Typ}^\ell QD$.

We shall now derive from this that $\text{Typ}^\ell Q\{A\}[x,B]P \text{ Ref } B \sim \text{Typ}^\ell Q((x := A)P)D$, so $\text{Typ}^\ell K \sim \text{Typ}^\ell L$ for $\ell \geq 1$. We do this as follows: From $\text{Typ}^\ell Q \text{ Ref } B \sim \text{Typ}^\ell QM$ and (XI) follows: $\text{Typ}^\ell Q\{A\}[x,B]P \text{ Ref } B \sim \text{Typ}^\ell Q\{A\}[x,B]PM$. (1)
From $\text{Typ}^\ell QM \sim \text{Typ}^\ell QD$ and (XI) follows: $\text{Typ}^\ell Q((x := A)P)M \sim \text{Typ}^\ell Q((x := A)P)D$. (2)

If we can now manage to show that $\text{Typ}^\ell Q\{A\}[x,B]PM \sim \text{Typ}^\ell Q((x := A)P)M$ then we are ready.

Let $\text{Typ}^\ell QM \equiv QM'$ (cf. (VIII)), then $\text{Typ}^\ell Q\{A\}[x,B]PM \geq_\alpha Q\{A\}[x,B]PM'$ and $\text{Typ}^\ell Q((x := A)P)M \geq_\alpha Q((x := A)P)M'$ (by (IX). Hence $\text{Typ}^\ell Q\{A\}[x,B]PM \geq$ $\text{Typ}^\ell Q((x := A)P)M$ (because x does not occur in M'). From this, (1) and (2) follows:

$\mathrm{Typ}^{\ell}Q\{A\}[x,B]P$ Ref $B \sim \mathrm{Typ}^{\ell}Q((x := A)P)D$ for all valid $\ell \geq 0$, so $\mathrm{Typ}^k K \sim \mathrm{Typ}^k L$ for all valid $k \geq 1$, but also for $k = 0$.

Since $\nu(QB)$ is shorter than $\nu(K)$ and $QB \geq QM$ it follows from ind. hyp. I and (VII) that Cl QB = Cl QM. Also, since ν(Typ QA) = $\nu(QA)$ is shorter than $\nu(K)$ and Typ QA $\geq$ QM: Cl Typ QA = Cl QB. So also Cl Typ Q Ref A = Cl QD = Cl Q Ref B by (VII)

By this and (VI): Cl Typ K = Cl Q Ref B = Cl QD = Cl Typ L, so Cl K = Cl L. From the central lemma then follows that $L \in \Lambda$, so K and L are co-legal.

case 2: Assume $K \geq_\beta L$ in one step as a direct consequence of $K' \geq_\beta L'$ and one of the monotony-rules. Then always $\nu(K') < \nu(K)$ so by ind. hyp. I: K' and L' are co-legal. Lemma B then implies that K and L are co-legal.

case 3: Assume $K \geq_\beta L$ in more steps, then this reduction can be split up in one-step reductions $K \equiv K_1 \geq_\beta K_2 \geq_\beta \ldots \geq_\beta K_p \equiv L$. Then induction on p shows that K_i and K_{i+1} are co-legal for all $1 \leq i < p$, so by the transitivity of co-legality: K and L are co-legal.

Corollary: Closure theorem: If $K \in \Lambda$ and $K \geq_\beta L$ then $L \in \Lambda$.

5. References

[1] R.P. Nederpelt, Lambda-typed lambda-calculus, in preparation

[2] R.P. Nederpelt, Strong normalisation in a λ-calculus with λ-expressions as types, Technological University, Eindhoven, Internal Report, Dept. of Math., Notitie 18 (27 April 1972).