Chapter 5

System F

5.1 System F

In system F, we have type variables X, Y, etc. and the universal quantifier $\forall$ binding type variables. So the BNF generating the types of system F is given as follows:

$$A \quad ::= \quad X \quad | \quad A \Rightarrow A \quad | \quad \forall X.\, A.$$

In contrast with simple types, we do not have to have atomic types to start the generation of types, since we have type variables. The universal quantifier $\forall$ binds the type variable that follows immediately. So, as in lambda abstraction, we do not distinguish a type $\forall X.A$ from its α-equivalent one $\forall Y.\, A[Y/X]$.

According to the new type constructor $\forall$, we extend the terms by allowing abstraction $\Lambda X.\, M$ by a type variable and application M_A of types A:

$$M \quad ::= \quad x \quad | \quad MM \quad | \quad \lambda x^A.\, M \quad | \quad M_A \quad | \quad \Lambda X.\, M.$$

Let Δ be a sequence of type variables $X_1, X_2, \ldots, X_m$. We write $\Delta \vdash A\ \mathsf{Type}$ if A is a type with free type variables included in $\{X_1, X_2, \ldots, X_n\}$. The typing judgment of system F has the shape $\Gamma \vdash_\Delta M : B$ where Δ is a sequence of type variables and Γ a sequence $x_1 : A_1, x_2 : A_2, \ldots, x_n : A_n$ of the typing of term variables.

Table 5.1.1 *Typing judgements of system F*

(Id) $\quad \Gamma,\ x : A \ \vdash_\Delta \ x : A$

$$\text{(Wk)}\ \frac{\Gamma\ \vdash_\Delta\ M : B}{\Gamma,\ x : A\ \vdash_\Delta\ M : B} \qquad \text{(Wk}')\ \frac{\Gamma\ \vdash_\Delta\ M : B}{\Gamma\ \vdash_{\Delta,X}\ M : B}$$

$$\text{(}\Rightarrow\text{I)}\ \frac{\Gamma,\ x : A\ \vdash_\Delta\ M : B}{\Gamma\ \vdash_\Delta\ \lambda x^A.\, M : A \Rightarrow B} \qquad \text{(}\Rightarrow\text{E)}\ \frac{\Gamma\ \vdash_\Delta\ M : A \Rightarrow B \qquad \Gamma\ \vdash_\Delta\ N : A}{\Gamma\ \vdash_\Delta\ MN : B}$$

$$\text{(}\forall\text{I)}\ \frac{\Gamma\ \vdash_{\Delta,X}\ M : A}{\Gamma\ \vdash_\Delta\ \Lambda X.\, M : \forall X.\, A} \qquad \text{(}\forall\text{E)}\ \frac{\Gamma\ \vdash_\Delta\ M : \forall X.\, A \qquad \Delta \vdash B\ \mathsf{Type}}{\Gamma\ \vdash_\Delta\ M_B : A[B/X]}$$

if X is not free in types in Γ

(TBD).

5.2 Encoding of types

The type constructs of system F are only exponentiation $\Rightarrow$ and the quantifier $\forall$ over type variables. In the early twentieth century, Russell already pointed out that disjunction can be encoded by implication and universal quantification over propositions. This observation was given a firm basis afterwards by Prawitz, who showed that all logical connectives are encoded by implication and universal quantification in intuitionistic second order logic. The idea is applicable also to higher order type theory, and we use the encodings given in Tab. 5.2.1 for various type constructs. In the table, we give also some abbreviations for terms. For example, the encoding of $A \times B$ is associated with terms π, π' and $\langle \cdot, \cdot \rangle$.

Table 5.2.1 *Encoding of various constructs*

$\bot\ =\ \forall X.\, X$

$1\ =\ \forall X.\, X \Rightarrow X$

Term:Type	*Definition*
$* : 1$	$\Lambda X\, \lambda x^X.\, x$

$A \times B \;=\; \forall X.\,(A \Rightarrow B \Rightarrow X) \Rightarrow X$

Term:Type	*Definition*
$\pi : (A \times B) \Rightarrow A$	$\lambda x^{A\times B}.\, x_A(\lambda y^A \lambda z^B.\, y)$
$\pi' : (A \times B) \Rightarrow B$	$\lambda x^{A\times B}.\, x_B(\lambda y^A \lambda z^B.\, z)$
$\langle M, N\rangle : A \times B$ for $M : A$ and $N : B$	$\Lambda X \lambda y^{A\Rightarrow B\Rightarrow X}.\, yMN$

$A + B \;=\; \forall X.\,(A \Rightarrow X) \Rightarrow (B \Rightarrow X) \Rightarrow X$

Term:Type	*Definition*
$\iota : A \Rightarrow (A + B)$	$\lambda x^A \Lambda X \lambda y^{A\Rightarrow X} \lambda z^{B\Rightarrow X}.\, yx$
$\iota' : B \Rightarrow (A + B)$	$\lambda x^B \Lambda X \lambda y^{A\Rightarrow X} \lambda z^{B\Rightarrow X}.\, zx$

$\exists X.\, A(X) \;=\; \forall Y\,(\forall X.A(X) \Rightarrow Y) \Rightarrow Y$

Term:Type	*Definition*
$I_C : A(C) \Rightarrow \exists X.\, A(X)$	$\lambda x^{A(C)} \Lambda X \lambda y^{\forall X.A(X)\Rightarrow Y}.\, y_C x$

$\mathbb{N} \;=\; \forall X.\, X \Rightarrow (X \Rightarrow X) \Rightarrow X$

Term:Type	*Definition*
$0 : \mathbb{N}$	$\Lambda X \lambda x^X \lambda y^{X\Rightarrow X}.\, x$
$\mathsf{S}M : \mathbb{N}$ for $M : \mathbb{N}$	$\Lambda X \lambda x^X \lambda y^{X\Rightarrow X}.\, y(M_X xy)$

$\mathbb{L}_{\mathbb{N}} \;=\; \forall X.\, X \Rightarrow (\mathbb{N} \Rightarrow X \Rightarrow X) \Rightarrow X$

Term:Type	*Definition*
$[\,] : \mathbb{L}_{\mathbb{N}}$	$\Lambda X \lambda x^X \lambda y^{\mathbb{N}\Rightarrow X\Rightarrow X}.\, x$
$[L|M] : \mathbb{L}_{\mathbb{N}}$ for $M : \mathbb{N},\ N : \mathbb{L}_{\mathbb{N}}$	$\Lambda X \lambda x^X \lambda y^{\mathbb{N}\Rightarrow X\Rightarrow X}.\, yL(M_X xy)$

5.2.2 Lemma
The following reductions hold for terms of appropriate types by the β-rule only:

(i) $\pi\langle M, N\rangle \to_\beta M$ *and* $\pi'\langle M, N\rangle \to_\beta N$.

(ii) $(\iota K)_C MN \to_\beta MK$ *and* $(\iota' L)_C MN \to_\beta NL$.

(iii) $(I_C K)_B M \to_\beta M_C K$
(iv) $0_A NK \to_\beta N$, *and moreover* $(\mathsf{S}M)_A NK \to_\beta K(M_A NK)$.
(v) $[]_A NK \to_\beta N$, *and moreover* $[L|M]_A NK =_{\beta\eta} KL(M_A NK)$. $\square$

We call an object in a category a *weak* limit if the object satisfies the same universal condition as that of limits except that the uniqueness of the morphism to the limit is omitted. Similarly weak colimits are defined. We may regard system F as a category where terms of type $A \Rightarrow B$ modulo $\beta\eta$-equality are considered to be morphisms from A to B. The composition of $M : A \Rightarrow B$ and $N : B \Rightarrow C$ is given by $\lambda x^A . N(Mx)$.

Then the preceding lemma shows that the encoded types provide weak (co)limits. For example, $A + B$ is a weak coproduct. In fact, if we let $[M, N]$ for $M : A \Rightarrow C$ and $N : B \Rightarrow C$ be defined by $\lambda x^{A+B} . x_C MN$, then the phrase (ii) implies $[M, N] \circ \iota = M$ and $[M, N] \circ \iota' = N$ by $\beta\eta$-equality. The uniqueness fails since $[\iota, \iota'] \neq_{\beta\eta} id_{A+B}$.

Remark: Later we will show that, if we consider parametricity, then $A+B$ turns out to be a coproduct, and similarly for other encoded types. At the same time, we will explain the reason the encoded types take those forms.

Chapter 6

Term Extraction

One of the main themes towards the theory of programming language is to provide the theory for the justification of the programme codes. We want to have an assurance that the programmes we code are correct. The Curry-Howard isomorphism gives an answer to the problem from the perspective of mathematical logic. This principle asserts that one can read off a programme from a formalized proof. The proof verifies some mathematical formula and the associated programme turns out to satisfy the property stated by the formula.

In this chapter, we discuss to extract a term of system F from a proof given in second order logic with the axioms of arithmetic and those for other inductively defined sets. It is known that the second order arithmetic is fairly strong and most theories of mathematics can be formalized and proved in this system. So, in comparison with the simplicity of the syntax, the expressiveness of system F is surprisingly strong.

6.1 Martin-Löf's productions and second order logic

In chapter 2, we gave the system NJ of natural deduction for intuitionistic first order predicate logic. If we want to prove anything interesting, we definitely need more primitives than just logical connectives. First we give the inference rules for the equality predicate. Then the inductively defined structures, such as natural numbers, are introduced by Martin-Löf's production rules. Finally we extend our logic to second order.

We add the equality predicate $=$. Accordingly we include new rules in natural deduction as in Tab. 6.1.1. There $A(x)$ is an arbitrary formula

Table 6.1.1 *The rules for equality predicate*

$$\frac{}{x = x}\ (=\text{Ref}) \qquad \frac{\begin{array}{cc}\vdots & \vdots \\ A(x) & x = y\end{array}}{A(y)}\ (=\text{Sub}_1) \qquad \frac{\vdots}{\dfrac{x = y}{t(x) = t(y)}}\ (=\text{Sub}_2)$$

Table 6.1.2 *Production for* $\mathbb{N}$

$$\frac{}{\mathbb{N}(0)}\ (\mathbb{N}\text{-I}_1) \qquad \frac{\vdots}{\dfrac{\mathbb{N}(t)}{\mathbb{N}(\mathsf{S}t)}}\ (\mathbb{N}\text{-I}_2) \qquad \frac{\begin{array}{ccc}\vdots & \vdots & \cancel{A(x)} \\ \mathbb{N}(t) & A(0) & \vdots \\ & & A(\mathsf{S}x)\end{array}}{A(t)}\ (\mathbb{N}\text{-E})^{\dagger}$$

† x is a fresh variable.

and t is an arbitrary term where x designates some (not necessarily all) occurrences of x in the formula A and in the term t. The symmetricity and transitivity of the equality predicate is derivable. For symmetricity $(x = y) \to (y = x)$, let $A(\cdot)$ be $(\cdot = x)$. Since $A(x) = (x = x)$ is derived by (=-Ref), the rule (=-Sub$_1$) gives $A(y)$, i.e., $y = x$ from $x = y$. The transitivity is proved similarly.

Martin-Löf's production is a formalization of inductive data types in the form of inference rules of natural deduction. In place of presenting general rules, we give two simple examples we are interested in.

We introduce a new unary relation symbol $\mathbb{N}(\cdot)$. The intended meaning of $\mathbb{N}(t)$ is that "t is a natural number". Martin-Löf's production for natural numbers consists of three new inference rules concerning the predicate $\mathbb{N}$ in Tab. 6.1.2. The first two are introduction rules and the last is the elimination rule. The introduction rules correspond to that the set of natural numbers n is generated by the syntax $n ::= 0 \,|\, \mathsf{S}n$ where $\mathsf{S}n$ is the successor of n. The elimination rule is the induction principle over natural numbers: If we have $A(0)$ and $\forall x.\, A(x) \to A(\mathsf{S}x)$ then $A(t)$ holds for every natural number t.

The next example is the production for the predicate $\mathbb{L}_{\mathbb{N}}(\cdot)$ of finite lists of natural numbers in Tab. 6.1.3, where t and u are arbitrary terms. The finite lists l of natural numbers is generated by the syntax

Table 6.1.3 *Production for* $\mathbb{L}_{\mathbb{N}}$

$$\frac{}{\mathbb{L}_{\mathbb{N}}([\,])}\ (\mathbb{L}_{\mathbb{N}}\text{-I}_1) \qquad \frac{\mathbb{N}(t) \quad \mathbb{L}_{\mathbb{N}}(u)}{\mathbb{L}_{\mathbb{N}}([t|u])}\ (\mathbb{L}_{\mathbb{N}}\text{-I}_2) \qquad \frac{\mathbb{L}_{\mathbb{N}}(u) \quad A([\,]) \quad \cancel{\mathbb{N}(x)}\ \cancel{A(y)} \;\ddots\; A([x|y])}{A(u)}\ (\mathbb{L}_{\mathbb{N}}\text{-E})^{\dagger}$$

† x, y are fresh variables.

$$l \quad ::= \quad [\,] \quad | \quad [n|l]$$

where n ranges over the set of natural numbers. So we have two introduction rules corresponding to the right hand sides. The third rule is the induction principle over finite lists of natural numbers.

So far we presented the system of natural deduction for first order logic. We want to extend the system to second order logic, where the formulas are allowed to have set variables. Let X etc. denote set variables. If t is a term and X is a set variable, then $t \in X$ is a formula. We write this $X(t)$ occasionally. Moreover, the formulas may have quantifications over set variables as $\forall X.\, A$ or $\exists X.\, A$. For reader's convenience, we give the syntax of formulas A of second order logic with equality augmented by production rules for $\mathbb{N}$ and $\mathbb{L}_{\mathbb{N}}$:

$$\begin{array}{rcl} A & ::= & t \in X \ \mid\ t = t \ \mid\ \bot \\ & & \mid\ A \,\&\, A \ \mid\ A \vee A \ \mid\ A \to A \\ & & \mid\ \forall x.\, A \ \mid\ \exists x.\, A \ \mid\ \forall X.\, A \ \mid\ \exists X.\, A \\ & & \mid\ \mathbb{N}(t) \ \mid\ \mathbb{L}_{\mathbb{N}}(t) \end{array}$$

where t ranges over terms, x over first order variables and X over set variables. If $B(x)$ is a formula with a chosen free variable x that may or may not occur in B, we can substitute $B(\cdot)$ for a free set variable X in a formula $A(X)$ by simply changing $t \in X$ by $B(t)$. The inference rules for second order quantifiers in natural deduction is given in Tab. 6.1.4, where A and B are arbitrary formulas and $A(B)$ is the substitution of $B(\cdot)$ for X in A.

Let $\mathrm{NJ}^2_{\mathbb{N},\mathbb{L}_{\mathbb{N}}}$ denote the natural deduction for intuitionistic second order predicate logic with equality augmented by Martin-Löf's production rules for natural numbers $\mathbb{N}$ and the finite lists $\mathbb{L}_{\mathbb{N}}$ of natural numbers.

Table 6.1.4 *Inference rules for second order quantifiers*

$$\frac{\begin{array}{c}\vdots\\ A(X)\end{array}}{\forall X.\, A(X)}\ (\forall^2\text{-I})^\dagger \qquad \frac{\begin{array}{c}\vdots\\ \forall X.\, A(X)\end{array}}{A(B)}\ (\forall^2\text{-E})$$

† X is not a free variable in any of the hypotheses.

$$\frac{\begin{array}{c}\vdots\\ A(B)\end{array}}{\exists X.\, A(X)}\ (\exists^2\text{-I}) \qquad \frac{\begin{array}{c}\vdots\\ \exists X.\, A(X)\end{array} \quad \begin{array}{c}\not{A(X)}\\ \vdots\\ B\end{array}}{B}\ (\exists^2\text{-E})^\ddagger$$

‡ X is not free in B or the hypotheses of the derivation of B.

6.2 First order erasure

The fundamental idea of intuitionistic logic is that every proof has a computational content. It is so, especially if we have a proof of a formula of the shape $\forall x \in \mathbb{N}\, \exists y \in \mathbb{N}.\, A(x, y)$ where $\forall x \in \mathbb{N} \cdots$ abbreviates $\forall x.\, \mathbb{N}(x) \to \cdots$ and $\exists x \in \mathbb{N} \cdots$ abbreviates $\exists x.\, \mathbb{N}(x)\ \&\ \cdots$. We want to extract a term of type $\mathbb{N} \Rightarrow \mathbb{N}$ in system F from a $\mathrm{NJ}^2_{\mathbb{N},\mathbb{L}_\mathbb{N}}$-proof of the formula. There are several ways to do this. In this section, we provide the *first order erasure*, which is the simplest method. Indeed, the term extraction is achieved simply by erasing all first-order parts from the derivation.

We associate a type $\mathtt{t}A$ to each formula A of system $\mathrm{NJ}^2_{\mathbb{N},\mathbb{L}_\mathbb{N}}$ in such a way that, if A has free set variables $X_1, \ldots, X_n$, then $\mathtt{t}A$ has free type variables $X_1, \ldots, X_n$. The definition of $\mathtt{t}A$ is given in Tab. 6.2.1. It amounts to eliminating all first-order parts from formulas. We note especially that an atomic formula $t = u$ is translated into 1. This means we do not expect any computational information from the equality predicate.

To each $\mathrm{NJ}^2_{\mathbb{N},\mathbb{L}_\mathbb{N}}$-derivation of a formula B from (bags of) hypotheses $A_1, \ldots, A_n$, we associate a term M of type $\mathtt{t}B$ under the environment $x_1 : \mathtt{t}A_1, \ldots, x_n : \mathtt{t}A_n$ in system F. We write

$$\begin{array}{l}\Gamma\\ \vdots M\\ B\end{array}$$

Table 6.2.1 *Definition of* $\mathtt{t}A$

formula A	*type* $\mathtt{t}A$
$t \in X$	X
$t = u$	1
$\bot$	$\bot$
$A \,\&\, B$	$\mathtt{t}A \times \mathtt{t}B$
$A \vee B$	$\mathtt{t}A + \mathtt{t}B$
$A \to B$	$\mathtt{t}A \Rightarrow \mathtt{t}B$

$\forall x.\, A$	$\mathtt{t}A$
$\exists x.\, A$	$\mathtt{t}A$
$\forall X.\, A$	$\forall X.\, \mathtt{t}A$
$\exists X.\, A$	$\exists X.\, \mathtt{t}A$
$\mathbb{N}(t)$	$\mathbb{N}$
$\mathbb{L}_{\mathbb{N}}(t)$	$\mathbb{L}_{\mathbb{N}}$

if M is the term assigned to the proof $\Gamma \vdash B$ (we may omit Γ). In case we need emphasize a specific variable, we write

$$\begin{array}{l} A \\ \vdots{\scriptstyle M[x]} \\ B \end{array}$$

where $x : \mathtt{t}A$ is the variable associated to the bag of the formula A. For explanation, let us consider the ($\to$I)-rule of natural deduction. Suppose that the term $M = M[x]$ of type $\mathtt{t}B$ is assigned to the proof $\Gamma, A \vdash B$. Then we assign $\lambda x^{\mathtt{t}A}.\, M$ to the proof $\Gamma \vdash A \to B$. We denote this argument by the figure

$$\dfrac{\begin{array}{l} \not{A} \\ \vdots{\scriptstyle M[x]} \\ B \end{array}}{A \to B} \qquad \rightsquigarrow \qquad \lambda x^{\mathtt{t}A}.\, M.$$

The assignment of terms to the inference rules of system $\mathrm{NJ}^2_{\mathbb{N},\mathbb{L}_{\mathbb{N}}}$ is given in Tab. 6.2.2.

Table 6.2.2 *Assignment of terms to* $\mathrm{NJ}^2_{\mathbb{N},\mathbb{L}_{\mathbb{N}}}$*-proofs*

(Hyp)	($\to$I)	($\to$E)	(&I)
A	$\not{A}$	$\vdots{\scriptstyle M}$ $\quad \vdots{\scriptstyle N}$	$\vdots{\scriptstyle M}$ $\quad \vdots{\scriptstyle N}$
$\rightsquigarrow\ x$	$\vdots{\scriptstyle M[x]}$	$\dfrac{A \to B \quad A}{B}$	$\dfrac{A \quad B}{A \,\&\, B}$
	$\dfrac{B}{A \to B}$	$\rightsquigarrow\ MN$	$\rightsquigarrow\ \langle M, N \rangle$
	$\rightsquigarrow\ \lambda x^{\mathtt{t}A}.\, M$		

($\&E^1$)

$$\frac{\overset{\vdots M}{A \mathbin{\dot\&} B}}{A}$$

$\rightsquigarrow\ \pi M$

($\&E^2$)

$$\frac{\overset{\vdots M}{A \mathbin{\dot\&} B}}{B}$$

$\rightsquigarrow\ \pi' M$

($\vee I^1$)

$$\frac{\overset{\vdots M}{\dot A}}{A \vee B}$$

$\rightsquigarrow\ \iota M$

($\vee I^2$)

$$\frac{\overset{\vdots M}{\dot B}}{A \vee B}$$

$\rightsquigarrow\ \iota' M$

($\vee E$)

$$\frac{\overset{\vdots K}{\dot A \vee B} \qquad \overset{\vdots M[x]}{\dot C}\ \not{A} \qquad \overset{\vdots N[y]}{\dot C}\ \not{B}}{C}$$

$\rightsquigarrow\ K_{\mathtt{t}C}(\lambda x^{\mathtt{t}A}.\, M)(\lambda y^{\mathtt{t}A}.\, N)$

($\forall I$)

$$\frac{\overset{\vdots M}{\dot A(x)}}{\forall x.\, A(x)}$$

$\rightsquigarrow\ M$

($\forall E$)

$$\frac{\overset{\vdots M}{\forall x.\, \dot A(x)}}{A(t)}$$

$\rightsquigarrow\ M$

($\exists I$)

$$\frac{\overset{\vdots M}{\dot A(t)}}{\exists x.\, A(x)}$$

$\rightsquigarrow\ M$

($\exists E$)

$$\frac{\exists x.\, \overset{\vdots K}{\dot A(x)} \qquad \overset{\vdots M[x]}{\dot C}\ \not{A(y)}}{C}$$

$\rightsquigarrow\ M[K]$

($\forall^2$-I)

$$\frac{\overset{\vdots M}{\dot A(X)}}{\forall X.\, A(X)}$$

$\rightsquigarrow\ \Lambda X.\, M$

($\forall^2$-E)

$$\frac{\overset{\vdots M}{\forall X.\, \dot A(X)}}{A(B)}$$

$\rightsquigarrow\ M_{\mathtt{t}B}$

($\exists^2$-I)

$$\frac{\overset{\vdots M}{\dot A(B)}}{\exists X.\, A(X)}$$

$\rightsquigarrow\ I_B M$

($\exists^2$-E)

$$\frac{\exists X.\, \overset{\vdots K}{\dot A(X)} \qquad \overset{\vdots M[x]}{\dot B}\ \not{A(X)}}{B}$$

$\rightsquigarrow\ K_{\mathtt{t}B}(\Lambda X \lambda x^{\mathtt{t}A}.\, M)$

($\perp E$)

$$\frac{\overset{\vdots M}{\dot\perp}}{A}$$

$\rightsquigarrow\ M_{\mathtt{t}A}$

(=Ref)

$$\frac{}{x = x}$$

$\rightsquigarrow\ *$

($=\mathrm{Sub}_1$)

$$\frac{\overset{\vdots M}{\dot A(x)} \qquad \overset{\vdots N}{\dot{x = y}}}{A(y)}$$

$\rightsquigarrow\ M$

($=\mathrm{Sub}_2$)

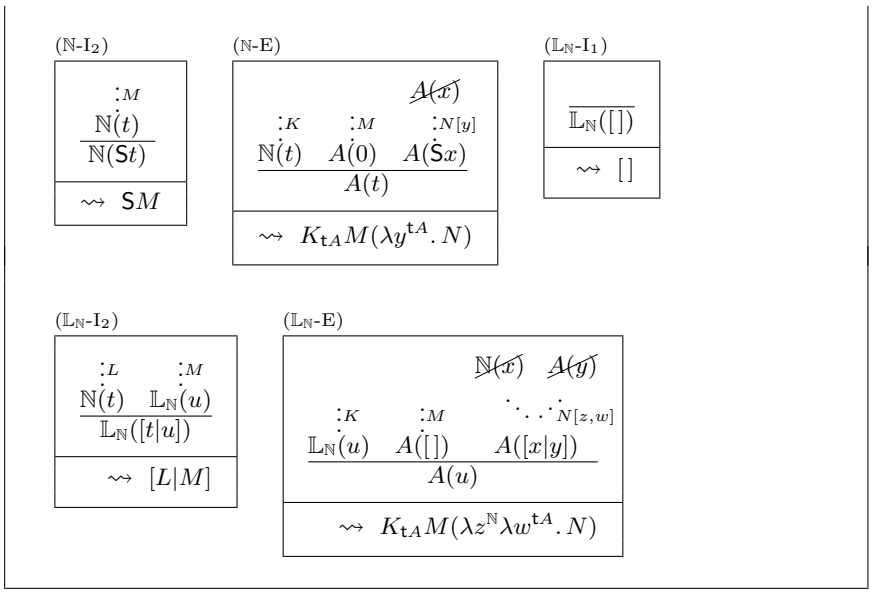
$$\frac{\overset{\vdots M}{\dot{x = y}}}{t(x) = t(y)}$$

$\rightsquigarrow\ M$

($\mathbb{N}$-I_1)

$$\frac{}{\mathbb{N}(0)}$$

$\rightsquigarrow\ 0$



Remark: The encodings of types in Tab. 5.2.1 mimic the elimination rules of the corresponding connectives. Only the encoding of $A \times B$ does not follow this principle. We can explain the encoding of $A \times B$ by Kan's theorem, as given in a chapter of parametricity.

If we assign a term to a proof of the formula $\forall x \in \mathbb{N}\, \exists y \in \mathbb{N}.\, A(x, y)$ by the translations of Tab. 6.2.2, then we have a closed term M' of type $\mathbb{N} \Rightarrow (\mathbb{N} \times \mathsf{t}A)$. Hence we can extract a closed term of type $\mathbb{N} \Rightarrow \mathbb{N}$ by $M = \lambda x^{\mathbb{N}}.\, \pi(M'x)$. Similarly, for example, we can extract a term of type $\mathbb{L}_{\mathbb{N}} \Rightarrow \mathbb{N}$ from a proof of a formula of the shape $\forall x \in \mathbb{L}_{\mathbb{N}}\, \exists y \in \mathbb{N}.\, A(x, y)$.

6.3 Realisability

We must show that the terms extracted by the first-order erasure are correct in some sense. We achieve this by the realisability argument. Here, a formula A of $\mathrm{NJ}^2_{\mathbb{N},\mathbb{L}_{\mathbb{N}}}$ is realised by a term M of system F, denoted by $M \vDash A$.

Convention: In order to avoid the confusion, we denote types of system F by lower greek letters σ etc. in this and the following sections.

Let $\mathcal{U}$ be a first-order structure for constants $0, [\,]$, function symbols $\mathsf{S}, [\cdot|\cdot]$. The standard model is the collection of all natural numbers and all finite lists of natural numbers, but we do not exclude other structures. The equality symbol $=$ is interpreted as the real equality. The crux is the interpretation of the symbol $\in$, which involves in set variables.

In models of classical logic, the range of set variables is a collection of subsets of the structure $\mathcal{U}$. So $t \in X$ is true iff the interpretation $[\![t]\!]$ is a member of the subset $[\![X]\!]$. This is not sufficient for intuitionistic logic, since we are required to have a computational reasoning if we claim that $t \in X$ is true. Therefore we consider subsets of $\mathcal{U}$ endowed with realisers witnessing how each member of $\mathcal{U}$ belongs to the subset:

6.3.1 Definition
A *realised set* is a pair $R = (\sigma, S)$ of a closed type σ of system F and a family $S = \{S_a\}_{a \in \mathcal{U}}$ where S_a is a set of closed terms of type σ.

A realised set may be regarded as a subset $\{a \in \mathcal{U} \mid S_a \neq \emptyset\}$, endowed with a family of realisers S_a to each a in this subset. Let $\mathcal{A}$ be the collection of all realised sets. We will interpret set variables as realised sets. Suppose that $A[\overline{X}, \overline{x}]$ is a formula with free variables among set variables $\overline{X} = X_1, \ldots, X_m$ and individual variables $\overline{x} = x_1, \ldots, x_n$. If we assign realised sets R_i to X_i and members a_j of $\mathcal{U}$ to x_j, then we write $A[\overline{R}, \overline{a}]$ as if the variables were substituted for.

6.3.2 Definition
Let $A[\overline{R}, \overline{a}]$ be a formula of $\mathrm{NJ}^2_{\mathbb{N},\mathbb{L}_\mathbb{N}}$ under the assignment of $R_i = (\sigma_i, S_i)$ to X_i and the assignment of a_j to x_j.

A term M *realises* the formula $A[\overline{R}, \overline{a}]$ if M is a closed term of type $\mathtt{t}A[\overline{\sigma}]$ satisfying $M \vDash A[\overline{R}, \overline{a}]$, this relation $\vDash$ defined as the smallest relation subject to the conditions in Tab. 6.3.3.

Table 6.3.3 *Definition of* $M \vDash A[\overline{R}, \overline{a}]$

(o)	$N \vDash A[\overline{R}, \overline{a}]$	if $M \vDash A[\overline{R}, \overline{a}]$ and $N \to M$ by β-reduction for a closed term N.
(i)	$* \vDash A(\overline{a})$	if $A(\overline{a})$ is true in $\mathcal{U}$, for every atomic formula $A(\overline{x})$ except $\bot$ and those in (ii) through (iv).
(ii)	$M \vDash (t \in R)$	if M is a member of S_a where a is the interpretation of t and $R = (\sigma, S)$.

(iii)	$0 \vDash \mathbb{N}(0)$.	
	$\mathsf{S}M \vDash \mathbb{N}(\mathsf{S}t)$	if $M \vDash \mathbb{N}(t)$ holds.
(iv)	$[\,] \vDash \mathbb{L}_{\mathbb{N}}([\,])$.	
	$[K\vert M] \vDash \mathbb{L}_{\mathbb{N}}([t\vert l])$	if both $K \vDash \mathbb{N}(t)$ and $M \vDash \mathbb{L}_{\mathbb{N}}(l)$ hold.
(v)	$M \vDash A \,\&\, B$	if both $\pi M \vDash A$ and $\pi' M \vDash B$ hold.
(vi)	$\iota M \vDash A \vee B$	if $M \vDash A$ holds.
	$\iota' N \vDash A \vee B$	if $N \vDash B$ holds.
(vii)	$M \vDash A \to B$	if it holds that $MN \vDash B$ for every N such that $N \vDash A$.
(viii)	$M \vDash \forall x.\, A(x)$	if, for every $a \in \mathcal{U}$, it holds that $M \vDash A(a)$.
(ix)	$M \vDash \exists x.\, A(x)$	if there is $a \in \mathcal{U}$ such that $M \vDash A(a)$.
(x)	$M \vDash \forall X.\, A(X)$	if, for every realised set $R = (\sigma, S)$ in $\mathcal{A}$, it holds that $M_\sigma \vDash A(R)$.
(xi)	$I_\sigma M \vDash \exists X.\, A(X)$	if $M \vDash A(R)$ for some realised set $R = (\sigma, S)$ in $\mathcal{A}$

Remark: (1) The definition of $M \vDash A[\overline{R}, \overline{a}]$ is by induction on the construction of A. In the clauses for $\mathbb{N}$ and $\mathbb{L}_{\mathbb{N}}$, inner inductions on the construction of terms M are used.

(2) In the clause (i), a formula is true in $\mathcal{U}$ if it is so in the sense of the usual validity in a structure. The involved atomic formulas are only the equality $t = u$ at this moment. We have $* \vDash t = u$ iff the interpretations of t and u are equal. Afterwards, we will deal with the case we have additional relation symbols (e.g., $t < u$).

(3) It never happens that $M \vDash \bot$.

(4) It is not the restriction that, in the clause (o), we enforce the reduction to be the β-rule only. The argument below is not affected if we change it by $\beta\eta$-reduction or $\beta\eta$-equality. We note that the proof of Thm. 6.3.9 uses only the β-reduction .

6.3.4 Definition

Let $B(x)$ be a formula in free variables among $\{x\}$ only, but possibly with parameters by realised sets in $\mathcal{A}$ and members of $\mathcal{U}$.

The realised set R_B is the pair of $\mathtt{t}B$ and the family $S_B = \{(S_B)_a\}_{a \in \mathcal{U}}$ where $(S_B)_a$ is the set of all terms M satisfying $M \vDash B(a)$.

6.3.5 Lemma
Let $A(X)$ and $B(x)$ be a formulas where free variables are those displayed only, but possibly with parameters by realised sets in $\mathcal{A}$ and members of $\mathcal{U}$.

We have $M \vDash A(R_B)$ iff $M \vDash A(B)$ for every closed M term of type $\mathsf{t}(A(B))$.

(*Proof*) Easy by induction on the construction of A. □

Remark: We have put $\mathcal{A}$ as the collection of all realised sets for simplicity so far. It suffices, however, that $\mathcal{A}$ is closed under the formation of R_B, for the arguments throughout this section.

6.3.6 Definition
A *(realisability) model* is a pair $(\mathcal{U}, \mathcal{A})$ where $\mathcal{U}$ is a structure interpreting the first order language (except $\in$) and $\mathcal{A}$ is a collection of realised sets closed under the construction R_B. The *standard model* has the set of all natural numbers and all finite lists of natural numbers as $\mathcal{U}$, and has all realised sets as $\mathcal{A}$.

Proof of Correctness

In the rest of this section, we fix an arbitrary model $(\mathcal{U}, \mathcal{A})$, and consider the realisability relation $\vDash$ based on it.

6.3.7 Lemma
Let $t[\overline{x}]$ and $u[\overline{x}]$ be terms in the language of $\mathrm{NJ}^2_{\mathbb{N},\mathbb{L}_\mathbb{N}}$.

If $t[\overline{a}] = u[\overline{a}]$ in $\mathcal{U}$ and $M \vDash A(t[\overline{a}])$, then $M \vDash A(u[\overline{a}])$ for all formulas $A(x)$.

(*Proof*) In the clause (iii) of Tab. 6.3.3, we must regard $\mathsf{S}t$ as any term that has the same interpretation as $\mathsf{S}t$, not as an expression starting with symbol S. Likewise for clause (iv). Otherwise the proof is obvious. □

We let n denote the interpretation of $n = \mathsf{S}(\cdots\mathsf{S}(\mathsf{S}0)\cdots)$ (n copies of S) in the structure $\mathcal{U}$. We also let $[a_1, \ldots, a_{n-1}, a_n]$ denote the interpretation of finite lists $[a_1|\cdots[a_{n-1}|a_n]\cdots]$ in $\mathcal{U}$. We use the same abbreviations for (encoded) terms of system F.

6.3.8 Lemma

(i) *If $K \vDash \mathbb{N}(t)$, then t is interpreted as some numeral n in $\mathcal{U}$ and $K \to_\beta n$.*

(ii) *If $K \vDash \mathbb{L}_\mathbb{N}(t)$, then t is interpreted as some $[a_1, \ldots, a_n]$ in $\mathcal{U}$, and $K \to_\beta [N_1, \ldots, N_n]$ for some N_i such that $N_i \vDash \mathbb{N}(a_i)$ for $i = 1, \ldots, n$.*

(*Proof*) Note that $K \vDash \mathbb{N}(t)$ happens only by rules (o) or (iii) in Tab. 6.3.3. We argue by induction on the number of application of these rules. Likewise for $\mathbb{L}_\mathbb{N}$. $\square$

The following theorem shows that every formula provable in system $\mathrm{NJ}^2_{\mathbb{N},\mathbb{L}_\mathbb{N}}$ is realisable by the term extracted by the first order erasure.

6.3.9 Theorem

Suppose that $A_1, \ldots, A_p \vdash B$ is derivable in $\mathrm{NJ}^2_{\mathbb{N},\mathbb{L}_\mathbb{N}}$ where $A_1, \ldots, A_n$ and B are formulas in free variables among $\overline{X} = X_1, \ldots, X_m$ and $\overline{x} = x_1, \ldots, x_n$. Let $M[\overline{X}, \overline{z}]$ be the term extracted from the proof by first order erasure under the environment $z_i : (\mathtt{t}A_i)[\overline{X}]$ $(i = 1, \ldots, p)$.

Then, for all assignments of $R_i = (\sigma_i, S_i)$ to X_i and all assignments of a_i to x_i, and for all terms $L_1, \ldots, L_p$,

$$L_1 \vDash A_1[\overline{R}, \overline{a}] \ , \ \ldots \ , \ L_p \vDash A_p[\overline{R}, \overline{a}] \quad \Rightarrow \quad M[\overline{\sigma}, \overline{L}] \vDash B[\overline{R}, \overline{a}].$$

(*Proof*) The proof is by induction on the derivation of $A_1, \ldots, A_n \vdash B$. To simplify the description, we suppress the substitutions $[\overline{\sigma}, \overline{L}]$ and $[\overline{R}, \overline{a}]$. We start with interesting cases. Suppose the last rule of the derivation is ($\mathbb{N}$-E). The induction hypotheses are $K \vDash \mathbb{N}(t)$, $M \vDash A(0)$, and that, for all $a \in \mathcal{U}$ and all J such that $J \vDash A(a)$, it holds that $N[J] \vDash A(\mathsf{S}a)$. The claim is to show

$$(K_{\mathtt{t}A} M(\lambda y^{\mathtt{t}A}. N)) \quad \vDash \quad A(t).$$

By Lemma 6.3.8 (i), $t = n$ for some n in $\mathcal{U}$, and $K \to_\beta n$. We prove the claim by induction on n. If $n = 0$, then the left hand side of the claim β-reduces to M. Hence the second induction hypothesis is exactly the claim. If $n = \mathsf{S}n'$, the left hand side β-reduces to $N[J]$ where $J = n'_{\mathtt{t}A} M(\lambda y^{\mathtt{t}A}. N))$. By inner induction hypothesis, $J \vDash A(n')$. So the third induction hypothesis implies the claim.

The rule ($\mathbb{L}_\mathbb{N}$-E) is handled similarly. The case of the rule (=-Sub$_1$) is derived from Lemma 6.3.7. The rule ($\bot$-E) is handled by the remark (3) after Def. 6.3.2.

The remainder follows a uniform pattern. We show the case of the existential quantifier over set variables. For the introduction rule, the induction hypothesis $M \vDash A(B)$ implies $I_{\mathtt{t}B} M \vDash \exists X.\, A(X)$ as required, since we can replace $A(B)$ by $A(R_B)$ (Lemma 6.3.5). For the elimination rule, the induction hypotheses are $K \vDash \exists X.\, A(X)$ and that, for all $R = (\sigma, S)$ in $\mathcal{A}$ and all term J such that $J \vDash A(R)$, it holds that $M[J] \vDash B$. The claim is $K_{\mathtt{t}B}(\Lambda X \lambda x^{\mathtt{t}A[X]}.\, M[x]) \vDash B$. By definition in Tab. 6.3.3, K reduces to $I_\sigma J$ for some $R = (\sigma, S)$ in $\mathcal{A}$ and some term $J : \mathtt{t}A[\sigma]$ such that $J \vDash A(R)$. Hence Lemma 5.2.2 implies that $K_{\mathtt{t}B}(\Lambda X \lambda x^{\mathtt{t}A[X]}.\, M)$ reduces to $M[J]$. So the induction hypothesis on M implies the claim. □

6.3.10 Definition
A *true* sentence of $\mathrm{NJ}^2_{\mathbb{N},\mathbb{L}_\mathbb{N}}$ is a sentence (i.e., a closed formula) A such that there is a term M of system F satisfying $M \vDash A$. For general formulas, we call them true formulas if their universal closures are true sentences.

6.3.11 Corollary
Suppose that $\forall x \in \mathbb{N}\, \exists y \in \mathbb{N}.\, A(x, y)$ *is provable in* $\mathrm{NJ}^2_{\mathbb{N},\mathbb{L}_\mathbb{N}}$.

Then there is a closed term $K : \mathbb{N} \Rightarrow \mathbb{N}$ *satisfying the following: For every numeral* m*, there is a numeral* n *such that* Km *reduces to* n *by* β*-reduction and* $A(m, n)$ *is true.*

(*Proof*) We extract a term M of type $\mathbb{N} \Rightarrow (\mathbb{N} \times \mathtt{t}A)$ from the derivation. By definition, for every numeral m, there is a member a of $\mathcal{U}$ such that $\pi(Mm) \vDash \mathbb{N}(a)$ and $\pi'(Mm) \vDash A(m, a)$. By Lemma 6.3.8, there is a numeral n such that $\pi(Mm) \to_\beta n$ and a equals n in $\mathcal{U}$. Let us put $K = \lambda x^{\mathbb{N}}.\, \pi(Mx)$. Of course, this corollary holds if we replace $\mathbb{N}$ by $\mathbb{L}_\mathbb{N}$. □

In particular, if A is a first order formula composed of atomic formulas $t = u$ or $P(t_1, \ldots, t_n)$ for some additional relation symbol P as handled in the next section, then the formula $A(m, n)$ is true in the first-order structure $\mathcal{U}$ in the usual sense of model theory.

Remark: (1) We note Cor. 6.3.11 shows also that term Km enjoys the weak normalization property.

(2) By the standard diagonalization argument, Cor. 6.3.11 implies that we cannot prove Thm. 6.3.9 in second order Peano arithmetic.

6.4 Harrop formulas

In this section, we show that we can add new axioms keeping Thm. 6.3.9 valid. So far the symbols we have had are those of natural numbers, $0, \mathsf{S}$, those of finite lists, $[\,], [\cdot|\cdot]$, and the equality predicate $=$, with which we can virtually do nothing. So we consider any expansion of the language with new constants, new function symbols and new relations symbols. The allowed axioms are those formulas which are true and have no effect on realisers.

First of all we extend $\mathtt{t}A$ for new relation symbols. If $A = P(t_1, \ldots, t_n)$ where P is an additional relation symbol, then we put $\mathtt{t}A = 1$.

6.4.1 Definition
A *Harrop formula* is a formula obtained as one of the following: (i) Formulas $t = u$ and $P(t_1, \ldots, t_n)$ for additional relation symbols P are Harrop formulas; (ii) If A and B are Harrop, then $A \,\&\, B$ is Harrop; (iii) If A is Harrop, then both $\forall x.\, A$ and $\exists x.\, A$ are Harrop; (iv) If A is Harrop, then $C \to A$ is Harrop for any formula C.

Remark: The Harrop formulas are those formulas A for which $\mathtt{t}A$ is a terminal object, if the encoded type 1 is regarded as a terminal object in a cartesian closed category. Note that $1 \times 1 \cong 1$ and $1^C \cong 1$. We regard Harrop formulas as those which have no computational information. Conventionally, $\bot$ may be added to Harrop formulas but we omit this since $\mathtt{t}\bot = \bot$ has no closed terms.

To each Harrop formula A, we associate a closed term s_A of type $\mathtt{t}A$ as follows: If A is atomic, then s_A equals $*$. If the formula is conjunction $A \,\&\, B$, then $s_{A\&B}$ is defined by $\langle s_A, s_B \rangle$. If the formula is implication $C \to A$, then $s_{C\to A}$ is defined by $\lambda x^{\mathtt{t}C}.\, s_A$. Moreover $s_{\forall x.A} = s_{\exists x.A} = s_A$.

6.4.2 Lemma
For each Harrop formula A, the following two are equivalent: (i) *A is true, and* (ii) $s_A \vDash A$. $\square$

Remark: For Harrop formulas A, their truth follows the usual definition in the theory of models of classical logic. For example, a Harrop formula $C \to A$ is true iff the truth of C implies that of A.

6.4.3 Definition
An *axiom* is a true Harrop formula. Of course, the truth depends on the model $(\mathcal{U}, \mathcal{A})$ we bear in mind.

For example, in the standard model, $(Sx < Sy) \to (x < y)$ is an axiom if we add a new relation symbol $<$ and interpret it standardly. Oppositely, $(x < y) \vee (x = y) \vee (y < x)$ is not an axiom whereas the formula is true in the usual sense of validity in the models of classical logic.

Remark: Most importantly, one of Peano's axioms, $0 \neq 1$, is not an axiom since it is not Harrop. We deal with a possibility to have non-Harrop axioms in a later section.

6.4.4 Proposition
Suppose that $A_1, \ldots, A_p \vdash B$ *is derivable in* $\mathrm{NJ}^2_{\mathbb{N},\mathbb{L}_\mathbb{N}}$ *using additional axioms.*

Then we can still extract a term $M[\overline{X}, \overline{z}]$ *of type* $\mathtt{t}B$ *where* $z_i : \mathtt{t}A_i$ *for* $i = 1, \ldots, p$*, and Thm. 6.3.9 remains to hold for this term* M*.*

(*Proof*) We assign terms s_A to axioms A and otherwise follow Tab. 6.2.2 to extract a term M. The proof of Thm. 6.3.9 is valid, since axioms are true, i.e., have realisers. □

Remark: In the proof of Prop. 6.4.4, there is no using that axioms are Harrop. Indeed, we can employ non-Harrop formulas as axioms if we can detect their realisers. But this corresponds to detecting proofs of the formulas in $\mathrm{NJ}^2_{\mathbb{N},\mathbb{L}_\mathbb{N}}$, so the merit to add such axioms are narrow. For Harrop formulas, in contrast, we need not find realisers by ourselves, if we are sure that the formulas are true (Lemma 6.4.2).

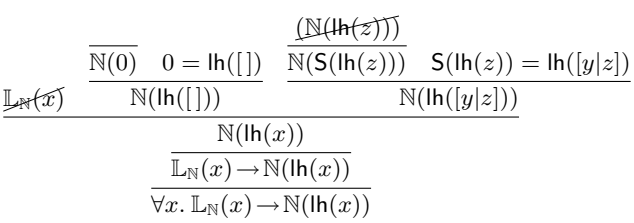
6.5 Examples

Length of finite lists

We extract a term lh of type $\mathbb{L}_\mathbb{N} \Rightarrow \mathbb{N}$ computing the length of finite lists. We add a unary function symbol $\mathsf{lh}(\cdot)$ and two axioms

$$\begin{array}{lcl} \mathsf{lh}([\,]) & = & 0 \\ \mathsf{lh}([y|z]) & = & \mathsf{S}(\mathsf{lh}(z)). \end{array}$$

Figure 6.5.1 *The totality of* lh



We prove the totality $\forall x \in \mathbb{L}_{\mathbb{N}}\, \exists y \in \mathbb{N}.\, \mathsf{lh}(x) = y$, or equivalently $\forall x.\, \mathbb{L}_{\mathbb{N}}(x) \to \mathbb{N}(\mathsf{lh}(x))$. The derivation in $\mathrm{NJ}^2_{\mathbb{N},\mathbb{L}_{\mathbb{N}}}$ is given in Fig. 6.5.1. We omitted the proofs of symmetricity of the equality obtaining $0 = \mathsf{lh}([\,])$ from an axiom $\mathsf{lh}([\,]) = 0$ etc., since they do not affect the extracted term. Applying the first order erasure, we extract the term $\mathsf{lh} = \lambda x^{\mathbb{L}_{\mathbb{N}}}.\, x_{\mathbb{N}} 0 (\lambda y^{\mathbb{N}} \lambda z^{\mathbb{N}}.\, \mathsf{S} z)$.

Append function

We extract a term append of type $\mathbb{L}_{\mathbb{N}} \Rightarrow \mathbb{L}_{\mathbb{N}} \Rightarrow \mathbb{L}_{\mathbb{N}}$ concatenating two finite lists. A familiar program by PROLOG is

```
append([],Y,Y).
append([X|Y],Z,[X|W]) :- append(Y,Z,W).
```

This corresponds to adding a 3-place relation symbol $\mathsf{append}(\cdot,\cdot,\cdot)$ and two axioms $\mathsf{append}([\,], y, y)$ and $\mathsf{append}(y, z, w) \to \mathsf{append}([x|y], z, [x|w])$. The latter may be given in the form of inference rule

$$\frac{\mathsf{append}(y, z, w)}{\mathsf{append}([x|y], z, [x|w])}.$$

We want to prove the totality $\forall x, y \in \mathbb{L}_{\mathbb{N}}\, \exists z \in \mathbb{L}_{\mathbb{N}}.\, \mathsf{append}(x, y, z)$. The proof is given in Fig. 6.5.2. The extracted term from the proof is given as follows: The subproof Π_1 yields $\langle y, * \rangle$ of type $\mathbb{L}_{\mathbb{N}} \times 1$, and the subproof Π_2 yields $\langle [u|z], * \rangle$ where $u : \mathbb{N}$ and $z : \mathbb{L}_{\mathbb{N}} \times 1$. Hence the extracted term of type $\mathbb{L}_{\mathbb{N}} \Rightarrow \mathbb{L}_{\mathbb{N}} \Rightarrow \mathbb{L}_{\mathbb{N}}$ is

$$\lambda x^{\mathbb{L}_{\mathbb{N}}} \lambda y^{\mathbb{L}_{\mathbb{N}}}.\, \pi(x_{\mathbb{L}_{\mathbb{N}} \times 1} \langle y, * \rangle (\lambda u^{\mathbb{N}} \lambda z^{\mathbb{L}_{\mathbb{N}} \times 1}.\, \langle [u|z], * \rangle)).$$

However, we can simplify the extracted term by ignoring the conjunct of append in the formula $A(x, y, z) = \mathbb{L}_{\mathbb{N}}(x)\ \&\ \mathsf{append}(x, y, z)$ throughout the

Figure 6.5.2 *The totality of* append

Let $A(x,y,z)$ be $\mathbb{L}_{\mathbb{N}}(z)$ & $\mathsf{append}(x,y,z)$ in

$$\Pi_1 \;=\; \dfrac{\dfrac{\mathbb{L}_{\mathbb{N}}(y) \quad \overline{\mathsf{append}([\,],y,y)}}{A([\,],y,y)}}{\exists z.\, A([\,],y,z)}$$

$$\Pi_2 \;=\; \dfrac{\exists z.\, A(v,y,z) \qquad \dfrac{\dfrac{\mathbb{N}(u) \quad \dfrac{\cancel{A(v,y,w)}}{\mathbb{L}_{\mathbb{N}}(w)}}{\mathbb{L}_{\mathbb{N}}([u|w])} \qquad \dfrac{\cancel{A(v,y,w)}}{\mathsf{append}(v,y,w)}}{\mathsf{append}([u|v],y,[u|w])}}{A([u|v],y,[u|w])}}{\exists z.\, A([u|v],y,z)}$$

Then the totality is proved as the following:

$$\dfrac{\mathbb{L}_{\mathbb{N}}(x) \quad \exists z.\, A([\,],y,z) \qquad \dfrac{\cancel{\mathbb{L}_{\mathbb{N}}(y)} \quad \cancel{\mathbb{N}(u)} \quad \cancel{\exists z.\, A(v,y,z)}}{\exists z.\, A([u|v],y,z)}}{\exists z.\, \mathbb{L}_{\mathbb{N}}(z)\ \&\ \mathsf{append}(x,y,z)}$$

$$\vdots\ (\to\text{-I) and } (\forall\text{-I) rules}$$

$$\forall x,y.\, \mathbb{L}_{\mathbb{N}}(x) \to \mathbb{L}_{\mathbb{N}}(y) \to \exists z.\, \mathbb{L}_{\mathbb{N}}(z)\ \&\ \mathsf{append}(x,y,z)$$

derivation, since this part contains no computational information. Then the term append is defined as

$$\mathsf{append} \;=\; \lambda x^{\mathbb{L}_{\mathbb{N}}} \lambda y^{\mathbb{L}_{\mathbb{N}}}.\; x_{\mathbb{L}_{\mathbb{N}}} y (\lambda u^{\mathbb{N}} \lambda z^{\mathbb{L}_{\mathbb{N}}}.\, [u|z]).$$

6.6 A-translation

One of Peano's axioms, $0 \neq 1$, is not admissible as an axiom in the sense of Def. 6.4.3, since it is not Harrop. In fact, $\mathsf{t}(0 \neq 1) = 1 \to \bot$ has no closed terms, so Prop. 6.4.4 is not applicable. We show that we can add certain formulas in negated form, such as $0 \neq 1$, and still extract terms from proofs. The vehicle is Friedman's A-translation that is used to get

rid of the axioms in negated form so that the argument in earlier sections are applied. Later we show A-translation can be used also to extract terms from proof of Π^0_2-formulas in *classical* logic.

A-translation

6.6.1 Definition
Let A be an arbitrary formula.

The *A-translation* B_A of a formula B is obtained by replacing all atomic formulas P (including $\bot$, $\mathbb{N}$, etc.) occurring in B by disjunction $P \vee A$.

Remark: The bound variables are renamed if necessary, in order to avoid accidental capture of free variables in A.

6.6.2 Lemma
(i) *$A \to B_A$ is intuitionistically provable for every formula B.*
(ii) *Suppose that A is composed of conjunction and disjunction from atomic formulas. Then $(\neg A)_A$, i.e., $A_A \to A$ is intuitionistically provable.*

(*Proof*) (i) is easy. For (ii), let A be in disjunctive normal form $\bigvee_i \bigwedge_j P_{ij}$ with atomic formulas P_{ij}. Since $(\bigwedge_j P_{ij})_A = \bigwedge_j (P_{ij} \vee A)$ is provably equivalent to $(\bigwedge_j P_{ij}) \vee (A \mathbin{\&} C)$ for some formula C, we can derive A from $(\bigwedge_j P_{ij})_A$. Hence A_A implies A. □

6.6.3 Proposition
In system $\mathrm{NJ}^2_{\mathbb{N},\mathbb{L}_\mathbb{N}}$, if we have a derivation of $C_1, \ldots, C_n \vdash B$, then we have also a derivation of $(C_1)_A, \ldots, (C_n)_A \vdash B_A$ for any formula A.

(*Proof*) We show that the A-translations of all inference rules of $\mathrm{NJ}^2_{\mathbb{N},\mathbb{L}_\mathbb{N}}$ are derivable in $\mathrm{NJ}^2_{\mathbb{N},\mathbb{L}_\mathbb{N}}$. The proof is a use of Lemma 6.6.2 (i). □

Axioms in negated form

In order to apply the A-translation, the axioms by Harrop formulas are too strong, so we restrict the axioms to Horn clauses. In practice, they are sufficient. The axioms in the example of section 6.5 are all Horn clauses. Moreover the axioms of Peano arithmetic except the induction principle, which we already have, are Horn clauses, too.

6.6.4 Definition
A *Horn clause* is a formula of the form $P_1(\bar{t}) \,\&\, \cdots \,\&\, P_n(\bar{t}) \to Q(\bar{t})$ where $P_1, \ldots, P_n$ and Q are atomic formulas that are not $\mathbb{N}(t)$ or $\mathbb{L}_{\mathbb{N}}(t)$.

Note that $\bot$ is allowed as an atomic formula. If Q is $\bot$, then the Horn clause is the negation $\neg(P_1(\bar{t}) \,\&\, \cdots \,\&\, P_n(\bar{t}))$.

Suppose that A is a formula that is not true. For every Horn clause D that is not in negated form (i.e., Q is not $\bot$), D is true iff its A-translation D_A is true. (Why?)

6.6.5 Theorem
We assume axioms given by Horn clauses that are true. Let $D(x, y)$ be a Horn clause.

If we have a derivation of $\forall x \in \mathbb{N}\, \exists y \in \mathbb{N}.\, D(x, y)$ from these axioms, then there is a closed term K of type $\mathbb{N} \Rightarrow \mathbb{N}$ such that, for every numeral m, the term Km β-reduces to some numeral n and $D(m, n)$ is true.

(*Proof*) Suppose that $\neg C_1, \ldots, \neg C_n$ are axioms in negated form that are used in the derivation. We let A be $C_1 \vee \cdots \vee C_n$ and apply the A-translation. Let us note that A is not true. By Prop. 6.4.4, we have a derivation of $\forall x.\, (\mathbb{N}(x) \vee A) \to \exists y.\, (\mathbb{N}(y) \vee A) \;\&\; D_A(x, y)$ from the hypotheses of the A-translations of axioms. We can prove the translation $(\neg C_i)_A$ of axioms in negated form without using added axioms by Lemma 6.6.2 (ii), since $(\neg C_1)_A \,\&\, \cdots \,\&\, (\neg C_n)_A$ is equivalent to $(\neg A)_A$. For other axioms B, the A-translation B_A is derived from B.

So we can apply Prop. 6.4.4 to the proof of $\forall x.\, (\mathbb{N}(x) \vee A) \to \exists y.\, (\mathbb{N}(y) \vee A) \;\&\; D_A(x, y)$ from axioms B that are not in negated form, and we obtain a closed term M of type $(\mathbb{N} + \mathtt{t}A) \to ((\mathbb{N} + \mathtt{t}A) \times \mathtt{t}(B_A))$. Hence $N = \pi(M(\iota m))$ is a realiser of $\mathbb{N}(n) \vee A$ for some numeral n such that $D_A(m, n)$ is true. By definition of $\vDash$, either $N \to_\beta \iota n$ for some $n \vDash \mathbb{N}(n)$ or $N \to_\beta \iota' N_1$ for some $N_1 \vDash A$. But the latter is impossible since A is not true. By the same reason, the truth of $D_A(m, n)$ implies the truth of $D(m, n)$. So if we put $K = \lambda x^{\mathbb{N}}.\, \pi(M(\iota x))_{\mathbb{N}}(\lambda y^{\mathbb{N}}.\, y)(\lambda z^{\mathtt{t}A}.\, 0)$, the assertion of the theorem holds. $\square$

For example, $0 \neq 1$ and $\neg(x < y \;\&\; y \leq x)$ are axioms that are allowed to be included.

Remark: The conditions in the theorem above are not the weakest. All we require for the axioms B not in negated form is that the truth of B implies

the truth of B_A. What we need for $D(x, y)$ is that the truth of D_A implies the truth of D. So the following lemma shows that the formulas given there can replace these. As for the axioms in negated form, we require $(\neg C_i)_A$ is provable (without using $\neg C_i$) where $A = C_1 \vee \cdots \vee C_n$.

6.6.6 Lemma

Suppose that D is a Harrop formula where the clause (iv) of Def. 6.4.1 is restricted by the condition that, in the implication $C \to A$, also C must be Harrop.

For every formula A that is not true, the formula D is true iff its A-translation D_A is true.

(*Proof*) Left to the reader. □

Markov's rule

(TBD).

6.6.7 Exercise

Give a formal proof of $\mathbb{N}(n) \to \mathbb{N}(\mathsf{S}n)$ in natural deduction, where we assume that the predicate $\mathbb{N}(n)$ is defined as $\forall X.\, X(0) \to (\forall x.\, X(x) \to X(\mathsf{S}x)) \to X(n)$. Show that the extracted lambda term is $\lambda n\, \Lambda X \lambda z\, \lambda y.\, y(n_X\, z y)$.

(*Answer*) Let us put $P = (\forall x.\, X(x) \to X(\mathsf{S}x))$. We have the following formal proof

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\forall X.\, X(0) \to P \to X(n)}{X(0) \to P \to X(n)} \quad \not{X(0)}}{P \to X(n)} \quad \not{P}}{X(n)}}{X(\mathsf{S}n)}}{P \to X(\mathsf{S}n)}}{X(0) \to P \to X(\mathsf{S}n)}}{\forall X.\, X(0) \to P \to X(\mathsf{S}n)}$$

The hypothesis not discharged is exactly $\mathbb{N}(n)$ and the conclusion $\mathbb{N}(\mathsf{S}n)$. So the discharging the hypothesis, we complete the proof in NJ.