Interpretations of Transfinite Recursion and Parametric Abstraction in Types

Mariko Yasugi[1)] and Susumu Hayashi[2)]

1) Faculty of Science, Kyoto Sangyo University, Kyoto 603, Japan

2) Department of Applied Mathematics and Informatics,
Ryukoku University, Ohtsu 520-21, Japan

1 Introduction

In our previous work Yasugi[3], the first author defined a system of generalized types, which has transfinite recursion and parametric abstraction among formation rules of types. A system of terms with these types (now called **TRM**) was then introduced. It contains parametric abstraction,lambda-abstraction and bar recursion among formation rules of terms. Our purpose there was to realize the modified realizability interpretation of a constructive system of arithmetic, now called **TRDB**, which contains universal quantification over functions, definition of predicates by transfinite recursion and bar induction. The significance of these systems is explained in Yasugi[3].

Strong normalizability and the Church-Rosser property of these types (called type-forms) as well as of the terms (called term-forms) have been established in Yasugi-Hayashi[4]. There, a map from **TRDB** to **TRM** is defined, by which the modified realizability interpretation of **TRDB** is better presented, and also the computational meaning of **TRDB** is explicated. In this article, we continue to investigate some properties of **TRM**. First, a set model of type-forms is presented, and the term-forms are interpreted as functions in the image sets of type-forms (Theorem 1). Since a map from **TRDB** to **TRM** is defined in Yasugi-Hayashi[4], this means that **TRDB** also

has a set model. Another theme is to translate **TRM** into **TRDB** (Theorem 2). This is the reversal of a result in Yasugi-Hayashi[4].

The article is more or less self-contained, although familiarity with Yasugi[3] will help understanding its background. Howard-Kreisel[1] and Troelstra[2] are useful in learning bar induction and systems of constructive arithmetic.

2 Set-model of type-forms

Type-forms are types with parametric variables. They are briefly described below. We do not elaborate with details, since they are thoroughly discussed in Yasugi[3] and Yasugi-Hayashi[4].

Definition 2.1. Let $\mathcal{I} \equiv (I, <_I)$ be a primitive recursive well-ordered structure (on natural numbers). This will be the basis of the subsequent arguments.
(1) The basic language $\mathcal{BL}$ consists of the following.
(1.1) Constant symbols of the following kind are prepared. Let $\mathcal{C}_n$ be the set of constants representing all the number-theoretic functions of arity n, where $\mathcal{C}_0$ denotes the set of natural numbers. Let $\mathcal{C}$ be the union of $\mathcal{C}_n$ for all n = 0,1,2,3,.... Constant symbols of $\mathcal{BL}$ are those of $\mathcal{C}$.
(1.2) Propositional connectives are $\wedge, \vee$ and $\supset$.
(1.3) Constant symbols denoting some primitive recursive predicates such as $=, <, <_I$ are assumed.
(1.4) Countably many variables of arity n are prepared for each $n = 0, 1, 2, \ldots$. A variable of arity n is supposed to range over $\mathcal{C}_n$.

Now $\mathcal{BL}$-terms and $\mathcal{BL}$-formulas are defined as usual from these symbols. ($A \supset 0 = 1$ will be abbreviated to $\neg A$.) We assume that elementary properties of $\mathcal{I}$ can be expressed in $\mathcal{BL}$.
(2) Type-forms are defined below, based on the language $\mathcal{BL}$.
(2.1) Symbols N and $\mathbf{1}$ are (atomic) type-forms.
(2.2) If α and β are type-forms, then so are $\alpha \to \beta$ (function space), $\{x\}\alpha$ (parametric abstraction), $\pi(\alpha; t)$ (projection) and $cond[A; \alpha, \beta]$ (case definition), where x is a variable of $\mathcal{BL}$, t is a $\mathcal{BL}$-term and A is a $\mathcal{BL}$-formula.
(2.3) $\mathcal{R}[i, t]$ (transfinite recursion) is a type-form, where $\mathcal{R}$ is a special letter denoting recursion operator, i is a $\mathcal{BL}$-term which is supposed to be an element of I and t is a $\mathcal{BL}$-term. ($\mathcal{R}[i, t]$ is characterized by a fixed type-form

η, which is defined without $\mathcal{R}$. See (1.4) of Definition 2.2 below.)
(2.4) $\rho[j <_I i; \mathcal{R}[j,s]]$ (restriction) is a type-form, where j and i are $\mathcal{BL}$-terms which are supposed to be elements of I and s is a $\mathcal{BL}$-term.
(3) A type-form without free variables is called a type.
(4) **Type** will denote the collection of type-forms together with the reduction rules defined in the next definition.

Note. $\mathcal{BL}$ in the present context is stronger than the original one, since it contains constsnts corresponding to all the number-theoretic functions while previously we assumed constants corresponding to those which are primitive recursive in a parameter. It does not affect the validity of the argument that follows, since our interest at this time is a set model of type-forms.

Definition 2.2. (1) Conversion rules of type-forms, say α to β, denoted by $\alpha \Rightarrow \beta$, are the following.
(1.1) $\pi(\{x\}\alpha; t) \Rightarrow \alpha[t/x]$,where $\alpha[t/x]$ represents substitution of t for x in α.
(1.2) $cond[A; \alpha_1, \alpha_2] \Rightarrow \beta$, where β is α_1 if A is a true sentence in the standard interpretation of the symbols, and β is α_2 if A is a false sentence.
(1.3) $\rho[j <_I i; \mathcal{R}[j,s]] \Rightarrow \beta$, where β is $\mathcal{R}[j,s]$ if $j <_I i$ is a true sentence, and it is $\mathbf{1}$ if $j <_I i$ is a false sentence.
(1.4) Let Ξ be a designated symbol which is temporarily regarded as an atomic type, let η be a type-form with Ξ but without $\mathcal{R}$, whose free variables are k and w, where k is of arity 0 and w is of an arbitrary arity. $\eta[i/k, t/w]$ will denote simultaneous substitution of i for k and t for w, and $\mathcal{R}(i)$ will abbreviate $\{j\}\{x\}\rho[j <_I i; \mathcal{R}[j,x]]$.

$$\mathcal{R}[i,t] \Rightarrow \eta[i/k; t/w][\mathcal{R}(i)/\Xi],$$

where $X[Y/\Xi]$ represents the result of replacement of Ξ by Y in X.
(2) α is said to be 1-reduced to β if β is obtained from α by one conversion applied to a subtype of α, with the strategy that conversions (1.3) and (1.4) enjoy priorities; that is, one does not go inside ρ and $\mathcal{R}$. α is said to be reducible to β if there is a sequence of reductions (possibly empty) from α to β.

Note. It is obvious that the reducts of a type-form are again type-forms, and that free variables do not increase by reductions.

Let $\mathcal{F}_n$ denote the collection of all the number-theoretic functions of arity n, where $\mathcal{F}_0$ will denote the set of natural numbers, and let $\mathcal{F}$ be the union of $\mathcal{F}_n$, $n = 0, 1, 2, ...$

We are now to present an interpretation of **Type** in V_θ, the universe of sets of rank below θ, where θ is a limit regular ordinal (hence a cardinal) such that $\mathcal{F} \in V_\theta$ (hence $\mathcal{F}_n \in V_\theta$ for every n). The definition below is carried out in ZFC relativized to V_θ.

Definition 2.3. Let Var be the collection of variables of $\mathcal{BL}$, and let ρ be a map:$Var \to \mathcal{F}$, where a variable of arity n is assigned a function in $\mathcal{F}_n$. $[\![\alpha]\!]_\rho$, to be defined below, will express the interpretation of α in V_θ under the assignment ρ. Let ι denote a designated set in V_θ (different from the elements of $\mathcal{F}$).
(1) For any $\mathcal{BL}$-term t, $[\![t]\!]_\rho$ is defined to be the natural interpretation of t under assignment ρ. In order to spare notational complications, we write t^* for $[\![t]\!]_\rho$.
(2) $[\![N]\!]_\rho = \mathcal{F}_0$ (the set of natural numbers); $[\![\mathbf{1}]\!]_\rho = \{\iota\}$ (the singleton ι); $[\![\alpha \to \beta]\!]_\rho = [\![\beta]\!]_\rho^{[\![\alpha]\!]_\rho}$ (function space); $[\![\{x\}\alpha]\!]_\rho = \prod_{v \in \mathcal{F}_n} [\![\alpha]\!]_{\rho[v/x]}$ (Cartesian product) if x is a variable of arity n; $[\![\pi(\alpha; t)]\!]_\rho = \{v | \exists f \in [\![\alpha]\!]_\rho (v = f't^*)\}$ (Here $a'b$ is defined to be c if there is a unique set c such that $\langle b, c\rangle \in a$, and it is ι otherwise.); $[\![cond[A; \alpha, \beta]]\!]_\rho = \{a \in [\![\alpha]\!]_\rho \cup [\![\beta]\!]_\rho | (A^* \supset a \in [\![\alpha]\!]_\rho) \wedge (\neg A^* \supset a \in [\![\beta]\!]_\rho)\}$ (A^* is a set-theoretic formula obtained from A by interpreting its terms under ρ.).
(3) To interpret $\mathcal{R}$, we proceed as follows. Let η be the type-form (without $\mathcal{R}$) as in (1.4) of Definition 2.2. Put $\rho(\Xi) = z$ for a set variable z.Then $[\![\eta]\!]_\rho$ (denoted by $g(z)$) can be defined according to (1) and (2) above.We can define a subset h of V_θ such that $h'c = g(c)$ for any set c in V_θ. It is a fact that the principle of transfinite recursion holds for h along I^*.That is,there is a set F in V_θ such that F is a unique function over I^* satisfying $\forall i \in I^*[F'i = h'(F''i)]$. Now put $[\![\mathcal{R}[i, t]]\!]_\rho = (F'i^*)'t^*$. ($F''i$ is F with domain restricted to the elements below i.)

$$[\![\rho[j <_I i; \mathcal{R}[j, s]]\!]_\rho$$
$$= \{a \in (F'j^*)'s^* \cup \{\iota\}; (j^* <_I i^* \supset a \in (F'f^*)'s^*) \wedge (\neg j^* <_I i^* \supset a = \iota)\}.$$

Note. The apparently unnatural interpretation of π and ρ is to guarantee

that interpretation sets be nonempty, and only for this purpose. (See the proposition below.)

Proposition 2.1. (1) $[\![\alpha[t/x]]\!]_\rho = [\![\alpha]\!]_{\rho[t^*/x]}$.
(2) $[\![\alpha]\!]_\rho \in V_\theta$.
(3) $[\![\alpha]\!]_\rho$ is nonempty.
(4) If $\alpha \Rightarrow \beta$, then $[\![\alpha]\!]_\rho = [\![\beta]\!]_\rho$. That is,$[\![\pi(\{x\}\alpha; t)]\!]_\rho = [\![\alpha[t/x]]\!]_\rho$;
$[\![cond[A; \alpha, \beta]]\!]_\rho = [\![\alpha]\!]_\rho$ if A is a true sentence, and $= [\![\beta]\!]_\rho$ if A is a false sentence; similarly for $\rho[j <_I i; \mathcal{R}[j, s]]$;

$$[\![\mathcal{R}[i,t]]\!]_\rho = ((h'(F''i^*))'t^* = [\![\eta[\mathcal{R}(i)/\Xi]]\!]_{\rho'} = [\![\eta[i/k; t/w][\mathcal{R}(i)/\Xi]]\!]_\rho,$$

where $\rho' = \rho[i^*/k, t^*/w]$.
(5) If α is reduced to β, then $[\![\alpha]\!]_\rho$=$[\![\beta]\!]_\rho$.
(6) An element of $[\![\alpha]\!]_\rho$ can be regarded as a function if ι is regarded as a function of arity 0.
(7) If α is a type (a type-form without free parameters), then its interpretation is independent of assignment ρ. We write it as $[\![\alpha]\!]$.

The proof is carried out in ZFC. It is a routine work, using transfinite induction along I^* and induction on the construction of type-forms.

3 Term-forms and objects

The entity of a type is the collection of terms having that type. Similarly, the entity of a set is the collection of its elements. The system of term-forms (**TRM**) in Yasugi[3] gives a substance to **Type**, and it is the purpose of this section to interpret each term-form with a function belonging to the image set of its type-form.

We first outline the definition of term-forms for the reader's convenience.

Definition 3.1. (1) If either α is reduced to β or β is reduced to α, then α and β are said to be related. In this section, we restrict ourselves to regular type-forms, where a type-form is said to be regular if it does not contain any subtype of the form $\pi(\{x\}\alpha; t)$. (Since this is immediately reduced to the

related type-form $\alpha[t/x]$, such a restriction is not an essential one.)
(2) Term-forms are defined below. $\phi : \alpha$ will express that term-form ϕ is of type-form α. If α and β are related, then we let $\phi : \alpha$ imply $\phi : \beta$.
(2.1) For each n, $n = 0, 1, 2, 3, ...$, and for each β, X_n^β (the n-th variable of type-form β) is an atomic term-form.
(2.2) $*$ is an atomic term-form and $*$:**1**.
(2.3) A $\mathcal{BL}$-term t of arity n ($n = 0, 1, 2, 3, ...$) is a term-form and $t : \{y_1\}...\{y_n\}N$,where $y_1, ..., y_n$ are of arity 0.
(2.4) If $\phi : \gamma$, then $\lambda x.\phi : \{x\}\gamma$. If $\phi : \gamma$, then $\lambda X_n^\beta.\phi : \beta \to \gamma$, where the condition $\Gamma(\phi, \beta)$ below is assumed.
$\Gamma(\phi, \beta)$ For each X_n^β, where β and δ are related, any free parameter x in δ occurring free in ϕ is not bound by λx in ϕ.
(2.5) If $\phi : \{x\}\gamma$, then $app(\phi; t) : \gamma[t/x]$. If $\phi : \beta \to \gamma$ and $\psi : \beta$, then $app(\phi; \psi) : \gamma$.

We abbreviate these as ϕt and $\phi\psi$ respectively.

(2.6) If $\phi : \beta$ and $\psi : \gamma$, then $cond[A; \phi, \psi] : cond[A; \beta, \gamma]$.
(2.7) If $\phi : \mathcal{R}[j, s]$, then $\sigma[j <_I i; \phi] : \rho[j <_I i; \mathcal{R}[j, s]]$.
(3) A term-form which has no free parameters is called a term.
(4) Conversion rules of term-forms are the following.
(4.1) $(\lambda x.\phi)t \Rightarrow \phi[t/x]$.
(4.2) $(\lambda X_n^\beta.\phi)\psi \Rightarrow \phi[\psi/X_n^\delta]$, where β and δ are related and X_n^δ is free in ϕ. (ψ is substituted for X_n^δ for any such X_n^δ.)
(4.3) $cond[A; \phi, \psi] \Rightarrow \chi$, where χ is ϕ if A is a true sentence and χ is ψ if A is a false sentence.
(4.4) $\sigma[j <_I i; \phi] \Rightarrow \chi$, where χ is ϕ if $j <_I i$ is a true sentence and χ is $*$ if $j <_I i$ is a false sentence.
(5) We can define "ϕ 1-reduces to ψ" and "ϕ is reducible to ψ" similarly to Definition 2.1, with the strategy that σ has priority of reduction.
(6) The collection of term-forms together with conversion rules will be denoted by **Term**.

Corollary. If $\phi : \alpha$ and ϕ is reducible to ψ, then $\psi : \alpha$.

Definition 3.2. Let ρ be as in Definition 2.3. We extend ρ to an assignment to term variables. That is, $\rho(X_n^\beta)$ is an element of $[\![\beta]\!]_\rho$. With such an extended assignment ρ, we define an interpretation of term-forms under assignment ρ. The interpretation of ϕ under ρ is denoted by $[\![\phi]\!]_\rho$. It

will be defined so that if $\phi : \alpha$, then $[\![\phi]\!]_\rho \in [\![\alpha]\!]_\rho$, $[\![\phi[t/x]]\!]_\rho = [\![\phi]\!]_{\rho[t*/x]}$ and $[\![\phi[\psi/X]]\!]_\rho = [\![\phi]\!]_{\rho[[\![\psi]\!]_\rho/X]}$. Recall that $[\![\alpha]\!]_\rho$ is nonempty and that $[\![\alpha]\!]_\rho = [\![\beta]\!]_\rho$ if α and β are related. (See (3) and (5) of Proposition 2.1.)
(1) $[\![*]\!]_\rho = \iota; [\![t]\!]_\rho = t^*; [\![X_n^\beta]\!]_\rho = \rho(X_n^\beta)$.
(2) $[\![\lambda x.\phi]\!]_\rho = \langle [\![\phi]\!]_{\rho[f/x]} | f \in \mathcal{F}_n \rangle$ if x is of arity n; $[\![\lambda X_n^\beta.\phi]\!]_\rho = \Psi$, where $\Psi' W = [\![\phi]\!]_{\rho[W/X_n^\beta]}$ for all $W \in [\![\beta]\!]_\rho$.
(3) $[\![\phi t]\!]_\rho = [\![\phi]\!]'_\rho t^*; [\![\phi\psi]\!]_\rho = [\![\phi]\!]'_\rho [\![\psi]\!]_\rho$.
(4) $[\![cond[A; \phi, \psi]]\!]_\rho = a$, where a is defined as follows. Let

$$D = \{b | (A \supset b = [\![\phi]\!]_\rho) \wedge ((\neg A \supset b = [\![\psi]\!]_\rho)\}.$$

We can prove that there is a unique element in D. Then a is that unique element, and hence a satisfies that

$$(A \supset a = [\![\phi]\!]_\rho) \wedge (\neg A \supset a = [\![\psi]\!]_\rho).$$

$[\![\sigma[j <_I i; \phi]]\!]_\rho = a$, where a satisfies

$$(j <_I i \supset a = [\![\phi]\!]_\rho) \wedge (\neg j <_I i \supset a = \iota).$$

We can easily show the following.

Proposition 3.1. (1) The map $[\![\phi]\!]_\rho$ is well-defined, satisfying the desired conditions.
(2) If ϕ is reducible to ψ, then $[\![\phi]\!]_\rho = [\![\psi]\!]_\rho$ (equal elements).
(3) If ϕ is a closed term (a term-form without free occurrences of parameters and variables), then the interpretation of ϕ is independent of assignment ρ. We write it as $[\![\phi]\!]$.

Summing up the preceding results, we can state the conclusion.

Theorem 1. There is a natural set-theoretic interpretation of types and terms, denoted by $[\![\ ]\!]$ such that, if ϕ is a term of type α, then $[\![\phi]\!]$ is a function belonging to the set $[\![\alpha]\!]$.

We can extend **Term** to a wider class of terms, still preserving the property in Theorem 1 above. We give an example of such.

Definition 3.3 (1) To **Term** above, we add the following.
(1.1) A functional constant μ of type $\{f\}N$, where f is a unary function variable. (μ represents a modulus of finiteness functional of the order $<_I$.)
(1.2) Bar recursion. Let $\mathbf{b}$ denote a functional evaluating a barred tree. If $\phi : \{z\}\gamma$ and $\psi : \{z\}(\{s\}[z{*}s/z] \to \gamma)$, then $\mathcal{B}[\mathbf{b}; \phi, \psi, m, f] : \gamma[f''m/z]$,where m and f are $\mathcal{BL}-$terms of arities respectively 0 and 1. The conversion rule for $\mathcal{B}$ is as follows.

$$\mathcal{B}[\mathbf{b}; \phi, \psi, m, f] \Rightarrow app(\phi; f''m),$$

if $\mathbf{b}f \leq m$ is a true sentence with $\mathbf{b}$ added to $\mathcal{BL}$, and

$$\mathcal{B}[\mathbf{b}; \phi, \psi, m, f] \Rightarrow \mathcal{B}\{m+1\},$$

if $\mathbf{b}f > m$ is a true sentence, where $\mathcal{B}\{m+1\}$ abbreviates

$$app(app(\psi; f''m); \lambda k \mathcal{B}[\mathbf{b}; \phi, \psi, m+1, (f''m) * k]).$$

Reductions of thus extended term-forms can be defined similarly as before with the added strategy that $\mathcal{B}$ has priority.
(2) $[\![\mu]\!]_\rho = \mu; [\![\mathcal{B}[\mathbf{b}; \phi, \psi, m, f]]\!]_\rho = S$, where S is obtained from the interpretaions of ϕ, ψ,m, f by an application of (informal) bar recursion.

The proposition below can be proved by using the bar induction.

Proposition 3.2. The map extended as above is well-defined, satisfying the desired properties.

4 From type-forms to admissible formulas

In Yasugi[3] and Yasugi-Hayashi[4], we defined a map from "admissible" formulas to type-forms. Here we will show that a reverse map is possible.

Definition 4.1. (1) The language $\mathcal{L}$ is the language of Heyting arithmetic augmented by $\forall f$ of function variables, constant symbols of $\mathcal{BL}$ and H a predicate constant (for definition by transfinite recursion along I).

$\mathcal{L}$-formulas are defined as usual, where $H(i,t)$ is an atomic formula. Here i is a $\mathcal{BL}$-term of arity 0 which is supposed to denote an element of I and t is a $\mathcal{BL}$-term. An $\mathcal{L}$-formula is said to be admissible if H does not occur in the scope of $\exists$ and $\vee$. (Details are seen in Yasugi[4].) **Formula** will denote the collection of admissible formulas.
(2) A type-form $\pi(\alpha;t)$ where α is *not* of the form $\{x\}\beta$ is said to be irrelevant. In the following, **Type**$'$ will denote the family of type-forms which are *not* irrelevant.

The map $form$ from **Type**$'$ to **Formula** is defined as follows.
(2.1) $form(N) \equiv \exists x(x = x)$, where x is of arity 0; $form(\mathbf{1}) \equiv 0 = 1$.
(2.2) $form(\alpha \to \beta) \equiv form(\alpha) \supset form(\beta); form(\{x\}\alpha) \equiv \forall x form(\alpha)$; $form(\pi(\{x\}\alpha;t)) \equiv form(\alpha)[t/x]$; $form(cond[A;\alpha,\beta]) \equiv (A \supset form(\alpha)) \wedge (\neg A \supset form(\beta))$.
(2.3) $form(\mathcal{R}[i,t]) \equiv H(i,t)$; $form(\rho[j <_I i;\mathcal{R}[j,s]) \equiv (j <_I i \supset H(j,s)) \wedge (\neg j <_I i \supset 0 = 1)$.

Proposition 4.1. (1) α and $form(\alpha)$ have the same free variables.
(2) $form(\alpha[t/x]) = form(\alpha)[t/x]$.

Note. (1) The map from **Formula** to **Type**$'$ defined in Yasugi-Hayashi[4] and the reverse map $form$ are not mutual inverses, since the former *crushes*.
(2) The restriction of **Type** to **Type**$'$ (non-irrelevant type-forms) does not reduce the significance of our treatment, since an irrelevant type-form is either reduced to a non-irrelevant one or else it has no computational meaning.

5 From term-forms to proofs

The set of term-forms in whose formations only non-irrelevant type-forms are involved will be denoted by **Term**$'$. (See Definitions 3.1 and 4.1.) Here we define a map from **Term**$'$ to proofs. In what follows, the arguments goes through consistently with such restriction.

Definition 5.1. System **S** is Heyting arithmetic based on language $\mathcal{L}$, as defined above, formulated in natural deduction, augmented by $\forall - introduction$ and $\forall - elimination$ on function variables as well as the rules of transfinite

recursion listed below. (We write $\langle A; B\rangle$ and $\langle P; B\rangle$ to denote respectively an inference which derives B from formula A and an inference to drive B from the conclusion of proof P.)

$$H - introduction \qquad \langle G(i,t,H[i]); H(i,t)\rangle$$
$$H - elimination \qquad \langle H(i,t); G(i,t,H[i])\rangle$$

$G(i,t,H[i])$ is an admissible formula (specified for **S**) and $H[i]$ is an abstract (a meta-expression) of the form $\{j,b\}((j <_I i \supset H(j,b)) \wedge (\neg j <_I i \supset 0 = 1))$. (See, for example, Troelstra[2] for arithmetic systems based on **NJ**. **S** is a subsystem of system **TRDB** in Yasugi-Hayashi[4].)

$\langle A;; B\rangle$ and $\langle P;; B\rangle$ will denote a succession of inferences to obtain B respectively from A and P .

Proposition 5.1. If type-form α is reducible to β, then there is a proof of $form(\beta)$ from assumption $form(\alpha)$ in **S**, and vice versa.

Proof. Consider conversion rules according to Definition 2.2, referring to Definitions 4.1 and 5.1.
(1.2) Suppose, for example, A is a true sentence. Then A is a theorem of **S**. Since

$$form(cond[A;\alpha,\beta]) \equiv (A \supset form(\alpha)) \wedge (\neg A \supset form(\beta)),$$

we can easily deduce $form(\alpha)$ from $form(cond[A;\alpha,\beta])$. Conversely, this follows from $form(\alpha)$ and assumption $\neg A$.
(1.4) Immediate from the rules on H.

Definition 5.2. Suppose $\phi \in \mathbf{Term}'$, $\alpha \in \mathbf{Type}'$ and $\phi : \alpha$. (See Definition 3.1. Recall that α is not unique.) We define $prf(\phi;\alpha)$ as a proof in **S** which proves $form(\alpha)$ (possibly with live assumptions) when $\phi : \alpha$. (By virtue of Proposition 5.1, it suffices to define $prf(\phi;\alpha)$ for a "typical" type-form α which ϕ belongs to.)
(1) $prf(X_n^\beta;\beta) \equiv [n; form(\beta)]$ (the assumption $form(\beta)$ with label n);
$prf(t;N) \equiv \langle t = t; \exists x(x = x)\rangle$; $prf(*;\mathbf{1}) \equiv [0; 0 = 1]$ (the assumption $0 = 1$ with label 0).

(2) Suppose t is of $n$ary type, say ν. Then

$$prf(t;\nu) \equiv \langle ty_1 \ldots y_n = ty_1 \ldots y_n;;\forall y_1 \ldots \forall y_n \exists x(x = x)\rangle.$$

(3) $prf(\lambda x.\phi;\{x\}\gamma) \equiv \langle prf(\phi;\gamma);\forall x form(\gamma)\rangle$ (by $\forall - introduction$)
(4) $prf(\lambda X_n^\alpha.\phi;\alpha \to \beta)$ is defined as follows. Let P be $prf(\phi;\beta)$, and let δ be related to α, where X_n^δ occurs free in ϕ. Let Q be constructed ¿from P by adding to it the proof of $form(\delta)$ from assumption $[n; form(\alpha)]$ for each such δ. (See Proposition 5.1.) Then the desired proof is $\langle Q; form(\alpha) \supset form(\beta)\rangle$, where the inference is $\supset - introduction$.
(5) Suppose $\phi : \{x\}\alpha$ and $P \equiv prf(\phi;\{x\}\alpha)$. P proves $form(\{x\}\alpha)$, which is $\forall x F$, where $form(\alpha) \equiv F$. Now define $prf(app(\phi;t);\alpha[t/x])$ to be $\langle P; F[t/x]\rangle$ by $\forall - elimination$. This makes sense since

$$F[t/x] \equiv form(\alpha)[t/x] \equiv form(\alpha[t/x])$$

by Proposition 4.1. (Notice that $app(\phi;t)$ can be formed only when $\phi : \{x\}\alpha$ for some α. See (2.5) of Definition 3.1.) $prf(app(\phi;\psi);\gamma)$ is defined as

$$\langle prf(\psi;\beta), prf(\phi;\beta \to \gamma); form(\gamma)\rangle$$

by $\supset - elimination$.
(6) $prf(cond[A;\phi,\psi];cond[A;\alpha,\beta])$ is defined successively as follows, by $\supset - introduction$ and $\wedge - introduction$. From $prf(\phi;\alpha)$ and $prf(\psi;\beta)$, we obtain $A \supset form(\alpha)$ and $\neg A \supset form(\beta)$ respectively. From these we deduce $(A \supset form(\alpha)) \wedge (\neg A \supset form(\beta))$. We can add μ and $\mathcal{B}$ (See Definition 3.3) to prf-constructions.

Definition 5.3. (1) System **TRDB** is **S** augmented by the following.
(1.1) Axiom $WF(\mu;I)$, which is a formula expressing that μ is a modulus of finiteness functional for $<_I$.
(1.2) Bar induction BI $\langle \forall z(R(z) \supset B(z)), \forall z(\forall k B(z * k) \supset B(z)); B(t)\rangle$
Here, $z * k = \langle z_1, ..., z_l, k\rangle$ if $z = \langle z_1, ..., z_l\rangle$, t is a $\mathcal{BL}$-term of arity 0 expressing a sequence number, $B(z)$ is an admissible formula and $R(z)$ is free of quantifiers and H. It is also assumed that $\forall f \exists n R(f''n)$ and $\forall u \forall v((\text{“}v \text{ is an extension of } u\text{”} \wedge R(u)) \supset R(v))$.

(2) $prf(\mu;\nu) \equiv WF(\mu;I)$, where ν is the type of μ.
(3) Write **Proof** for the collection of proofs, either of **S** or of **TRDB**.
(4) Suppose $\phi : \{z\}\gamma$, $\psi : \{z\}(\{s\}\gamma[z * s/z] \to \gamma)(\equiv \beta)$, $P \equiv prf(\phi;\{z\}\gamma)$ proving $\forall z C(z)$ and $Q \equiv prf(\psi;\beta)$ proving $\forall z(\forall s C(z * s) \supset C(z))$. Then $prf(\mathcal{B}[\mathbf{b};\phi,\psi,m,f];\gamma[t/x])$ is defined as follows. Let P' be a proof of the formula $\forall z(R(z) \supset C(z))$ obtained from P . Then $C(t)$ can be obtained from P' and Q by BI. Let the entire proof be the desired one.

Note. (1) The relation between the bar induction and bar recursion is discussed in Yasugi-Hayashi[4] in detail.
(2) A map from **Proof** to **Term**$'$ is investigated in Yasugi-Hayashi[4]. This and prf are not mutual inverses.

Proposition 5.2. (1) $prf(\phi;\alpha)$ is a proof of a sentence if and only if α is a type (a closed type-form).
(2) If ϕ does not contain $*$ and free occurrences of variables, then $prf(\phi;\alpha)$ is a proof of $form(\alpha)$ without live assumptions.

Proposition 5.3. If $\phi : \alpha$ and $\phi \Rightarrow \psi$, then either (i) $prf(\phi;\alpha)$ is reduced to $prf(\psi;\alpha)$, (ii) $prf(\psi;\alpha)$ is a part of $prf(\phi;\alpha)$ or (iii) $prf(\psi;\alpha)$ is obtained from a part of $prf(\phi;\alpha)$ with eliminations. (We do not elaborate with reductions of proofs. See, for example, Troelstra[2] for reductions in a system formulated in natural deduction.)

Proof. (i) applies to (4.1) and (4.2) of Definition 3.1, (ii) applies to (4.3) and (4.4), and (iii) applies to bar recursion (Definition 3.3).

Summing up the results in Sections 4 and 5, we can state the following.

Theorem 2. There is a pair of maps from **Type**$'$ to **Formula** and from **Term**$'$ to **Proof**, say $< form, prf >$, such that, if $\phi : \alpha$, then $prf(\phi;\alpha)$ is a proof of $form(\alpha)$.

References

[1] W.A. Howard and G. Kreisel, Transfinite induction and bar induction of types zero and one, and the role of continuity in intuitionistic analysis, J. Symbolic Logic 31(1966) 325-358.
[2] A. S. Troelstra, Mathematical investigations of intuitionistic arithmetic and analysis, Lecture Notes in Math. 344(Springer-Verlag, Berlin, 1973).
[3] M. Yasugi, Hyper-principle and the functional structure of ordinal diagrams, Comment. Math. Univ. St. Pauli 34(1985) 227-263(the opening part); 35(1986) 1-37(the concluding part).
[4] M. Yasugi and S. Hayashi, A functional system with transfinitely defined types and bar recursion, manuscript.