Archive for Mathematical Logic (2022) 61:317–343
https://doi.org/10.1007/s00153-021-00792-5

Mathematical Logic

Combinatory logic with polymorphic types

William R Stirton[1]

Received: 22 August 2018 / Accepted: 2 July 2021 / Published online: 27 August 2021

Abstract

Sections 1 through 4 define, in the usual inductive style, various classes of object including one which is called the "combinatory terms of polymorphic type". Section 5 defines a reduction relation on these terms. Section 6 shows that the weak normalizability of the combinatory terms of polymorphic type entails the weak normalizability of the lambda terms of polymorphic type. The entailment is not vacuous, because the combinatory terms of polymorphic type are indeed weakly normalizable, as is proven in Sect. 7 using Tait and Girard's computability predicates. The remainder of the paper is devoted to arguing that interesting consequences would follow from the existence of an "ordinally informative" proof, i.e. one which uses transfinite induction over recursive ordinal numbers and otherwise finitary methods, of normalizability for as large a class of the terms as possible.

Keywords Combinatory logic · Polymorphic types · System F · Normalizability

Mathematics Subject Classification 03Fxx

1 Introduction

Sometimes problems about a λ-calculus are most easily solved through consideration of the corresponding combinatory calculus. This is because combinatory terms have, in a way, a simpler structure than λ-terms, being built up from atoms by (in the untyped or simply typed case) the single operation of application rather than the two operations of application on the one hand and λ-binding of variables on the other.

However it is not the case that, for every λ-calculus, an equivalent combinatory calculus has as yet been formulated. While there is indeed an untyped combinatory calculus which is (in a sense that can be made precise) equivalent to the untyped $\lambda\eta$-

✉ William R Stirton
William_Stirton@yahoo.co.uk

[1] Edinburgh Leisure Ltd, Meadowbank Sports Centre, 139 London Road, Edinburgh EH7 6AE, UK

calculus[1] and something similar holds for the simply typed λη-calculus, there is as yet no combinatory calculus that is equivalent to the polymorphically typed λη-calculus.[2] The purpose of the present paper is to close this gap.

A particular example of a problem concerning a combinatory calculus which seems to be technically easier than the corresponding problem for the equivalent λη-calculus will be discussed in Sect. 8 below. Moreover, according to [3] (p. 656), 'a computer handles λ-terms more easily if these are translated into combinatory terms'.[3]

The structure of this paper is as follows. Sections 2 through 5 are fundamental to the rest: they are devoted to defining the concept of a polymorphically typed combinatory term and a reduction relation on these terms. Sections 6 through 11 are devoted to demonstrating that results concerning polymorphically typed combinatory terms will have applications to other areas of logic, above all to the polymorphically typed λ-calculus and to the proof theory of second-order logic. These later sections are to a large extent independent of each other and can be read, apart from the need for a few cross-references, in any order. Readers who are concerned about motivational matters might like to look at the later sections first.

2 Classes of terms, type-assignment systems and equational calculi

The concept of a polymorphically typed λ-term is presumably familiar to most readers, but for the sake of maximum precision one particular definition of this concept will be presented in the present section. This section will also define the common substratum of all the various languages that will be studied in the sequel.

In Definition 2.1 below, the concept of a (polymorphic) *semi-type* is defined simultaneously with the function *VarF* which takes every semi-type to a (possibly empty) set of bindable type-variables. Throughout this paper, '≡' will denote identity between syntactic objects (terms, formulae and types).

Definition 2.1 (i) Each free type-variable $(\alpha, \beta, \gamma, \alpha_0, \alpha_1, \ldots)$ is a semi-type. $\mathrm{VarF}(\alpha) = \{\}$.

(ii) Each bindable type-variable $(\phi, \psi, \omega, \phi_0, \phi_1, \ldots)$ is a semi-type. $\mathrm{VarF}(\phi) = \{\phi\}$.

(iii) If A and B are semi-types, so is $(A \rightarrow B)$. $\mathrm{VarF}(A \rightarrow B) = \mathrm{VarF}(A) \cup VarF(B)$.

(iv) If A is a semi-type, so is $\forall\phi\mathrm{A}$ for every bindable type-variable ϕ. $\mathrm{VarF}(\forall\phi\mathrm{A}) = \mathrm{VarF}(A) - \{\phi\}$.

As usual, outermost parentheses are omitted and $A \rightarrow B \rightarrow C$ should be understood as short for $A \rightarrow (B \rightarrow C)$. Note that in clause (iv) there is no prohibition on $\forall\phi$ occurring in A. To ensure that a bindable variable is bound by at most one quantifier, the binding relation is defined as follows:

[1] A definition of "equivalent" and a proof of the statement are given in section 6E of [4]. The problem of finding a combinatory calculus which is equivalent to the untyped λβ-calculus seems to be considerably harder and will be ignored here. See [18] for discussion.

[2] Other than, perhaps, the attempt made in [20], which is superseded by the present treatment.

[3] Compare a similar remark at the foot of p. 122 of [19].

Definition 2.2 The relation *binds*, which holds between quantifier-occurrences and bindable variable-occurrences, is defined by recursion on lengths of semi-types. Suppose that $\sharp$ is an occurrence of ϕ inside the semi-type A; then

(i) If $\sharp$ is not within the scope of some quantifier $\forall\phi$ occurring inside A, $\sharp$ is bound by the outermost $\forall\phi$ in $\forall\phi A$.

(ii) If $\sharp$ *is* within the scope of some quantifier $\forall\phi$ occurring inside A, then the same quantifier-occurrence binds $\sharp$ within $\forall\phi A$ as binds it within A.

Definition 2.3 A semi-type A is a (sc., polymorphic) *type* iff $\mathrm{VarF}(A) = \{\}$.

Definition 2.4 A semi-type A is a *simple type* iff it is built up from free type-variables by clause (iii) alone of Definition 2.1.

Types can be thought of as formulae of second-order implicational logic, as on p. 345f. of [24]. The notion of *substituting* a type B for a free type-variable occurring in a type A can therefore be defined as is standardly done in textbooks on second-order logic (*loc. cit.*). Changing bindable variables in B is unnecessary, thanks to Definition 2.2.

Definition 2.5 The result of substituting the type B for the free variable α in type A shall be $A(^{\alpha}_{B})$.

Notational conventions As it is usually possible to tell from the context which variable it is for which a substitution is being made, this notation can be simplified. For example, if "$A(\alpha)$" and "$A(B)$" are used in the same context, "$A(B)$" should be read as meaning $A(^{\alpha}_{B})$. This implies that α does not occur in $A(B)$, unless it is in B. Occasionally the notion of substitution will be applied also to semi-types: thus "$A(^{\phi}_{\alpha})$" and "$A(^{\alpha}_{\phi})$" will be used with the obvious meaning. When either of these last two symbols is used, it must be understood that the indicated occurrences of ϕ are not bound by a quantifier in A, i.e. that they are in $VarF(A)$. In the case of "$A(^{\alpha}_{\phi})$", this can be ensured by tacitly changing some of the binding variables in A if necessary.

Every one of the languages studied in this paper includes among its atomic terms a stock of variables. These will occasionally be called *term-variables*, when it is necessary to emphasize that they are distinct from the type-variables introduced by Definition 2.1. For each type A, denumerably many variables of type A are assumed, over which letters like "x^A", "y^A", "x_0^A", "x_1^A", ... range; or (when it is not essential to draw attention to the type) simply "x", "y".... Also that there is a denumerable stock of untyped variables (x, y, etc.). When x and y are variables, $x \equiv y$ only if x and y are either both untyped or both have the same type.

Sometimes, for example in Definition 5.4, it is useful to be able to assume that the variables within each type are ordered as an ω-sequence, so this will be required from the outset.

Definition 2.6 (i) The class Λ of untyped λ-terms is built up from the untyped variables by application and λ-abstraction, as on p. 3 of [9].

(ii) The class CL of untyped combinatory terms is built up from untyped variables and the three combinators **S**, **K** and **I** by application alone.

(iii) The class $\mathfrak{J}$ of *untyped* J-*terms* is built up from the same basis as CL, but employing λ-abstraction as well.

Remark in this paper $\lambda x.x$ and $\lambda y.y$ count as distinct terms unless $x \equiv y$.

In addition to the three classes of untyped terms distinguished by Definition 2.6, this paper will distinguish six classes of typed terms. An inductive definition of a class of typed terms can be presented in the shape of a formal system such as the one in Table 2.1 (cf. p. 16 of [2]). The expression "$\in \Lambda_2(A)$" should be read as "is an Λ_2-term of type A". The first line of Table 2.1 shows the shape of initial statements of this formal system; the other four lines show the rules of inference.

Table 2.1: the formal system Λ_2

(a) $x^A \in \Lambda_2(A)$ for every polymorphic type A
(b) $a \in \Lambda_2(B) \quad \Rightarrow \quad \lambda x^A a \in \Lambda_2(A \to B)$
(c) $a \in \Lambda_2(A \to B),\ b \in \Lambda_2(A) \quad \Rightarrow \quad ab \in \Lambda_2(B)$
(d) $a \in \Lambda_2(A(\alpha)) \quad \Rightarrow \quad \Lambda_{\alpha,\phi} a \in \Lambda_2(\forall\phi A(\phi))$
(e) $a \in \Lambda_2(\forall\phi A(\phi)) \quad \Rightarrow \quad aC \in \Lambda_2(A(C))$ for any type C

In clause (d), it is required that α be in the type of no free variable in a and that ϕ differ from every variable ψ such that $\forall\psi$ occurs in A and has α within its scope.

The rules presented in clauses (b)–(e) of Table 2.1 clearly resemble the rules of a natural deduction calculus and will be called →I, →E, ∀I and ∀E respectively.

Definition 2.7 For any a, a shall be a Λ_2-*term* iff $a \in \Lambda_2(A)$ for some A.

Definition 2.8 (i) $\Lambda_\to$ is the subsystem of Λ_2 in which all types are required to be simple and the only rules of inference are →I and →E.
(ii) For any a, a is a $\Lambda_\to$-*term* iff $a \in \Lambda_\to(A)$ for some simple type A.

Definition 2.9 $\mathfrak{J}_\to$ shall be the formal system defined otherwise like $\Lambda_\to$ but with all statements of the shapes shown in Table 2.2 included among the initial statements.

Table 2.2

(a) $\mathbf{I}_A \in \mathfrak{J}_\to(A \to A)$ for every simple type A
(b) $\mathbf{K}_{AB} \in \mathfrak{J}_\to(A \to B \to A)$ for all simple types A, B
(c) $\mathbf{S}_{ABC} \in \mathfrak{J}_\to((A \to B \to C) \to (A \to B) \to A \to C)$ for all simple types A, B, C

Definition 2.10 (i) $CL_\to$ is the formal system defined otherwise like $\mathfrak{J}_\to$ but with only one rule of inference, namely →E (clause (c) of Table 2.1).
(ii) For any a, a is a $\mathfrak{J}_\to$-term iff $a \in \mathfrak{J}_\to(A)$ for some simple type A.
(iii) For any a, a is a $CL_\to$-*term* iff $a \in CL_\to(A)$ for some simple type A.

Definition 2.11 A function $x^A, a \mapsto [x^A]a$ which, for any term a and variable x^A of $CL_\to$, outputs a new term $[x]a$ is defined by recursion on lengths of terms.

(a) If $a \in CL_\to(B)$ and x^A does not occur in a, then $[x^A]a \equiv \mathbf{K}_{BA}a$;
(b) If $a \equiv x^A$, then $[x^A]a \equiv \mathbf{I}_A$;
(c) If $a \equiv bx^A$ for some b in which x^A does not occur, then $[x^A]a \equiv b$; and

(f) If $a \equiv bc$ for some bc, if a is not as described in any of clauses (a)−(c), if $b \in CL_{\rightarrow}(B \rightarrow C)$ while $c \in CL_{\rightarrow}(B)$, then $[x^A]a \equiv \mathbf{S}_{ABC}[x^A]b[x^A]c$.

Definition 2.12 A function $x, a \mapsto [x]a$ which takes a term a and variable x of CL to a new term $[x]a$ is defined by deleting all references to types from Definition 2.11 and generalizing over CL-terms instead of $CL_{\rightarrow}$-terms.

Remark the clauses of these definitions are labelled (a), (b), (c) and (f) because of tradition. See p. 190 of [4] and cf., for example, Definition 2.18 on p. 26 of [9].

Discussion: the formal systems Λ_2, $\Lambda_{\rightarrow}$, $\mathfrak{J}_{\rightarrow}$ and CL$_{\rightarrow}$ are all paradigms of what will here be called "systems *á la Church*" or "Church-style systems". See, for example, p. 16 of [2]. Usually nowadays ([1], [19]), these names are given to systems in which the terms have shapes like $\lambda x^A.x$, in which typed variables occur in binding, untyped in bound, positions.[4] Such systems will, however, play no part in the present paper. It is for this reason that our definition of the set of terms of Λ_2 is taken from [7] (p. 82f.) rather than [1] or [19], while the style of formulation is that used on p. 16 of [2].

While Λ_2 is the polymorphic extension of $\Lambda_{\rightarrow}$, polymorphic extensions of $\mathfrak{J}_{\rightarrow}$ and CL$_{\rightarrow}$ have not yet been defined. The goal of the present paper is to close this gap.

Seven classes of term have been defined in this section: the three untyped classes mentioned in Definition 2.6 and the four typed classes mentioned in the discussion following Definition 2.11. In the following two definitions, the letter "Σ" ranges over these seven classes.

Definition 2.13 A Σ-equation is an equation $a = b$, where $a, b \in \Sigma$ and (if Σ is a typed class) a, b have the same type.

Definition 2.14 For any Σ, a *calculus of* Σ*-terms* is a deductive system having as axioms some nonempty set of Σ-equations and rules of inference expressing general properties of equality, including possibly some form of rule (ξ) appropriate to Σ.

Thus, for example, the familiar untyped calculi $\lambda\beta$ and $\lambda\eta$ are both calculi of Λ-terms.

3 A problem and its solution

The main problem in finding a Church-style system that stands in the same sort of relation to CL$_{\rightarrow}$ as Λ_2 stands in to $\Lambda_{\rightarrow}$ is that it is not easy to see how anything corrresponding to rules →I and ∀I can be derivable within it. The general shape of ∀I must be: from $\Gamma \vdash a_0 : A(\alpha)$ infer $\Gamma \vdash a_1 : \forall\phi A(\phi)$, but what will be the relationship between a_0 and a_1? $a_0 \equiv a_1$ is not possible as, in a Church-style system, each term must determine its type uniquely. The two obvious strategies are:

(1) To introduce a new combinator Θ and stipulate that $a_1 \equiv \Theta a_0$.
(2) To postulate some new operation of term-construction, distinct from application, which, applied to a_0 (alongside possibly other arguments), yields a_1.

[4] On p. 18 of [2], systems of this last kind are called "de Bruijn systems".

Both strategies run into difficulties. The class of combinators Θ envisaged in (1), if applicable to arbitrary terms, would have to inhabit between them all types of the shape $A(\alpha)\to\forall\phi A(\phi)$; but formulae of this shape are not in general admissible in a sound logical calculus. As for (2), it is not clear how a calculus of combinatory terms constructed in accordance with (2) could admit the rule of β-conversion. In the combinatory calculi that have been constructed so far, the fact that the set of terms is closed under the function $x^B, a \mapsto [x^B]a$ or $x, a \mapsto [x]a$ (Definitions 2.11 and 2.12) is what makes β-conversion admissible; but these functions are in general applicable only to terms built up by application alone and would not necessarily be applicable if a new term-building operation were allowed.

Rejection of (2) motivates experimenting with the idea that a_1, a term of type $\forall\phi A(\phi)$, should be built up from atoms without using any term-constructing operations other than application. This problem is a bit like the problem of finding a formal system that generates the first-order logical validities without any quantifier-introducing rules of inference. This problem had been solved at least by the time Quine wrote [13]; however the solution depends on stipulating that the set of logical axioms shall be closed under prefixing of quantifiers. In the present context, this would be like insisting that there must be, not only for every type of the shape $A\to B\to A$ but also for every type of the shape $\forall\phi(A\to B\to A)$ and so on, a typed instance of $\mathbf{K}$ which inhabits that type. Since this contradicts the usual way of thinking about combinators, the solution adopted here will be to postulate, after all, a new term-constructing operation which, applied to a term of type $A(\alpha)$ yields a term of type $\forall\phi A(\phi)$, but to allow this new operation to be applied only to terms which contain no variables. In this way, it does not impede the definition of an operation which generalizes the one defined in 2.11. At the same time, we also admit new combinators with the property that (1) envisages Θ as having, but with the restriction that they can be applied to a term of type A only if α does not occur in A. The detailed working-out of these ideas is found in Sect. 4 below.

4 The formal systems $\mathbf{CL}_2$ and $\mathfrak{J}_2$

The goal of the next few definitions is to define the Church-style system $\mathfrak{J}_2$, which extends $\mathfrak{J}_\to$ and which will be, in a sense to be explored presently, equivalent to Λ_2. The shapes of the initial statements of $\mathfrak{J}_2$ are enumerated in the following Table, which extends Table 2.2. The expression "$\in \mathfrak{J}_2(A)$" should be read as "is a $\mathfrak{J}_2$-term of type A".
Table 4.1

(a) $x^A \in \mathfrak{J}_2(A)$ for every type A
(b) $\mathbf{I}_A \in \mathfrak{J}_2(A\to A)$ for every type A
(c) $\mathbf{K}_{AB} \in \mathfrak{J}_2(A\to B\to A)$ for all types A, B
(d) $\mathbf{S}_{ABC} \in \mathfrak{J}_2((A\to B\to C)\to(A\to B)\to A\to C)$ for all types A, B, C
(e) $\mathbf{U}^{\forall\phi B(\phi)\to B(C)} \in \mathfrak{J}_2(\forall\phi B(\phi)\to B(C))$ under the conditions described below
(f) $\Xi_{\forall\phi(F\to G)} \in \mathfrak{J}_2(\forall\phi(F\to G)\to\forall\phi F\to\forall\phi G)$
(g) $\Theta_{\forall\phi B} \in \mathfrak{J}_2(B\to\forall\phi B)$ for every type B

In line (e) $B(\alpha)$ and C must be types and ϕ a variable such that no quantifier in $B(\alpha)$ that binds ϕ should have α in its scope.

In (f), F, G must be semi-types and ϕ a variable such that $(VarF(F) \cup VarF(G)) - \{\phi\} = \emptyset$.

In (g), the requirement that B be a type (not a semi-type) entails that the outermost quantifier in $\forall\phi B$ does not actually bind anything.

The rules of inference of $\mathfrak{J}_2$ are shown in Table 4.2:

Table 4.2

(a) $a \in \mathfrak{J}_2(B) \quad \Rightarrow \quad \lambda x^A a \in \mathfrak{J}_2(A \to B)$
(b) $a \in \mathfrak{J}_2(A \to B),\ b \in \mathfrak{J}_2(A) \quad \Rightarrow \quad ab \in \mathfrak{J}_2(B)$
(c) $a \in \mathfrak{J}_2(A(\alpha)) \quad \Rightarrow \quad \Lambda_{\alpha,\phi} a \in \mathfrak{J}_2(\forall\phi A(\phi))$
(d) $a \in \mathfrak{J}_2(A(\alpha)) \quad \Rightarrow \quad \Pi_{\alpha,\phi} a \in \mathfrak{J}_2(\forall\phi A(\phi))$

In (c), α must not occur in the type of any term-variable free in a. In (d), a may not contain any free term-variables. In lines (c) and (d), it is required that the variable ϕ be distinct from every ψ that occurs in a quantifier $\forall\psi$ in A having α within its scope.

Remark the difference in function between the prefixes $\Lambda_{\alpha,\phi}$ and $\Pi_{\alpha,\phi}$ will become clear in the next section. It will turn out that whether rule (c) or rule (d) is used in building up a $\mathfrak{J}_2$-term makes a difference to how the term may be reduced.

Definition 4.1 The subsystem of $\mathfrak{J}_2$ in which only clauses (b) and (d) of Table 4.2 are used shall be CL_2.

Definition 4.2 "$\mathfrak{J}_2$-*term*" and "CL$_2$-*term*" shall be defined analogously to "Λ_2-term" etc (see Sect. 2 above); and "$\in CL_2(A)$" analogously to "$\in \Lambda_2(A)$'.

Thus the CL$_2$-terms are precisely the atoms shown in Table 4.1 together with well-formed terms built up from them by the two operations $a, b \mapsto ab$ and $a \mapsto \Pi_{\alpha,\phi} a$.

Definition 4.3 Whenever a and b are CL_2-terms, b is an *α/ϕ-generalization* of a iff:

(i) $b \equiv \Pi_{\alpha,\phi} a$; or
(ii) $b \equiv \Theta_{\forall\phi A} a$; or
(iii) There are terms a_0, a_1, b_0, b_1 and semi-types F, G such that:

(α) b_0 and b_1 are α/ϕ-generalizations of a_0, a_1 respectively; and
(β) $b_0 \in CL_2(\forall\phi(\mathrm{F} \to \mathrm{G}))$ and $b_1 \in CL_2(\forall\phi F)$; and
(γ) a $\equiv a_0 a_1$ and $b \equiv (\Xi_{\forall\phi(F \to G)} b_0) b_1$, abbreviated as usual to $\Xi_{\forall\phi(F \to G)} b_0 b_1$.

Definition 4.4 A free type-variable α is *used as an eigenvariable* in a CL_2-term a iff there are b and ϕ such that $\Pi_{\alpha,\phi} b$ is a subterm of a.

Proposition 4.5 *For any CL_2-term a and any type-variables α and ϕ such that*

(i) *α is not used as an eigenvariable in a; and*
(ii) *$B(\alpha)$ is the type of a; and*
(iii) *α does not occur in the type of any term-variable inside a; and*
(iv) *No occurrence of $\forall\phi$ in B has α within its scope,*

there is an α/ϕ-generalization b of a having type $\forall\phi B(\phi)$.

Proof if a contains no variables, we can take $\Pi_{\alpha,\phi}a$ for b. If α does not occur in $B(\alpha)$, we can take $\Theta_{\forall\phi B}a$ for b. The case where a is a variable with α in its type does not arise. The only remaining possibility, therefore, is that $a \in CL_2 B(\alpha)$ in virtue of line (b) of Table 4.2, which means a is a CL$_2$-term because both its immediate subterms are. Then the theorem is proven by induction on the construction of a and applying clause (iii) of Definition 4.3.

Discussion: if α *is* used as an eigenvariable in a, then there will always be another term, differing from a only in that eigenvariables have been changed, in which α is no longer used as an eigenvariable. Proposition 4.5 means in effect that the third rule in Table 2.1 can be mimicked in CL$_2$. We will express this by saying that ∀I is derivable in both these systems. In general, there will be more than one way of forming an α/ϕ-generalization of a given term. It is possible, however, to define a *function* from terms to their α/ϕ-generalizations by requiring that, whenever clauses (i) and (ii) of Definition 4.3 are applied to suberms of a, the subterms chosen shall be as large as possible. The α/ϕ-generalization of a formed in this way shall be called $\wedge_{\alpha,\phi}a$.

If a is a CL$_2$-term of type C and x^A a CL$_2$-variable, the term $[x^A]a$ is defined by:-

Definition 4.6 (a) If x^A does not occur in a, then $[x^A]a \equiv \mathbf{K}_{CA}a$.
(b) If $a \equiv x^A$, then $[x^A]a \equiv \mathbf{I}_A$.
(c) If $a \equiv bx^A$ for some b in which x^A does not occur, then $[x^A]a \equiv b$.
(f) If $a \equiv bc$ and a is not as described in clause (a), (b) or (c), if $b \in CL_2(B{\rightarrow}C)$ and $c \in CL_2(B)$, then $[x^A]a \equiv \mathbf{S}_{ABC}[x^A]b[x^A]c$.

Proposition 4.7 *Every* CL$_2$*-term has one of the four shapes distinguished in Definition* 4.6.

Proof by Definition 4.1, using the fact that a term of the shape $\Pi_{\alpha,\phi}a$ contains no term-variables.

Corollary *for every CL_2-term a of type C, $[x^A]a$ is defined and has type $A{\rightarrow}C$.*

This corollary may be expressed by saying that rule →I is derivable in CL$_2$.

Remark the formulation of $\mathfrak{J}_2$ implements the ideas outlined informally in the last paragraph of Sect. 3. The "new term-forming operation" mentioned there is precisely the operation which takes b to $\Pi_{\alpha,\phi}b$ when there are no variables in b. The "new combinators" mentioned two sentences later are those introduced in line (g) of Table 4.1. Other new combinators, introduced in line (f) of Table 4.1, are needed in order to make these ideas work.

5 Reduction of *CL*$_2$-terms

In section 6F2 of [4], the relation of *strong reduction* on CL-terms is defined by first embedding the CL-terms in the wider class of $\mathfrak{J}$-terms (see Definition 2.6 above) and then defining strong reduction over the $\mathfrak{J}$-terms. Similarly, a form of strong reduction on CL_2-terms will be defined here by first defining a reduction relation on $\mathfrak{J}_2$-terms and then restricting it to CL_2-terms.

The class of $\mathfrak{J}_2$-terms which may be reduced is defined by reference to the class Ir, which comprises precisely the CL_2-terms which may *not* be reduced. First an auxiliary definition:

Definition 5.1 For any $\mathfrak{J}_2$-terms a, b, we say that b is a *head of* a or *occurs in head position within* a iff one of the following holds.

(i) a is an atom and $b \equiv a$; or
(ii) $a \equiv a_0 a_1$ and b is either a itself or a head of a_0; or
(iii) $a \equiv \lambda x^A c$ and b is a head of c; or
(iv) $a \equiv \Lambda_{\alpha,\phi} c$ and b is a head of c; or
(v) $a \equiv \Pi_{\alpha,\phi} c$ and b is a head of c.

A *maximal* head of a is a head of a that is not a proper part of any larger head of a.

The inductive definition of the class Ir itself will be presented in a similar style to the definition of the class of $\mathfrak{J}_2$-terms; here "$\in$ Ir(A)" should be read as "is an irreducible term of type A".

Table 5.1

(a) $x^A \in Ir(A)$ for every typed variable x^A of CL_2
(b) $a \in Ir(B) \quad \Rightarrow \quad [x^A]a \in \mathrm{Ir}(A \to B)$ subject to restrictions (see below)
(c) $a \in Ir(B(\alpha)) \Rightarrow a' \in Ir(\forall\phi B(\phi))$ whenever a' is an α/ϕ-generalization of a
(d) $a \in Ir(A \to B),\ b \in Ir(A) \Rightarrow ab \in Ir(B)$ subject to restrictions (see below)
(e) $a \in Ir(\forall\phi B(\phi)) \Rightarrow \mathbf{U}^{\forall\phi B(\phi) \to B(C)} a \in Ir(B(C))$ subject to restrictions (below)
(f) $a \in Ir(\forall\phi(F \to G)),\ b \in Ir(\forall\phi F) \Rightarrow \Xi_{\forall\phi(F \to G)} ab \in Ir(\forall\phi G)$
(g) $a \in Ir(\forall\phi(F \to G)) \Rightarrow \Xi_{\forall\phi(F \to G)} a \in Ir(\forall\phi F \to \forall\phi G)$

In (b) it is required that at least one of the two following conditions should hold:-

(3) $[x^A]a$ is not formed from a by clause (c) of Definition 4.6 alone; or
(4) a does not have a variable in head position and, if a has a $\mathbf{U}$-term in head position, a has the shape $\mathbf{U}^{\forall\phi B(\phi) \to B(C)} x$.

In (d) it is required that a should have either a variable or a $\mathbf{U}$-term in head position but not be itself a $\mathbf{U}$-term.

In (e) and (g), it is required that a should not be a generalization and, in (f), that at least one of a, b should not be a generalization.

The motivation for these restrictions, and indeed for the way the rules presented in Table 5.1 are formulated, will be discussed in Sect. 9.

Proposition 5.2 *Every irreducible term is a CL_2-term*

Proof if every occurrence of "Ir" in Table 5.1 is replaced with "CL_2", then the resulting rules are all either primitive or admissible rules of CL_2. This can be checked using, in particular, Proposition 4.5 and the Corollary to 4.7.

Definition 5.3 A $\mathfrak{J}_2$-term is *reducible* iff it is not in Ir.

Five classes of $\mathfrak{J}_2$-terms will now be identified as *redexes* and a *contractum* will be associated with every redex. The first three classes of redex correspond to what are called, at p. 223 of [4], "type I (resp. II resp. III) redexes"; but it may be inadvisable in the present context to overwork the word "type". So the five classes will here be called: (I) *weak redexes*; (II) *open redexes*; (III) λ'-*redexes*; (IV) $\wedge$-*redexes*; (V) Λ-*redexes*. Throughout Table 5.2 and Definitions 5.4 to 5.7, it is required that the terms a, b, c be in Ir (a requirement which means that the reduction relation defined here somewhat resembles the *strict* or *standard* reduction of [23]).

The weak redexes comprise all CL$_2$-terms of the shapes displayed in the left-hand column of Table 5.2, while the shapes of the corresponding contracta are on the right. In the last row it is required that a, besides being irreducible, should be an α/ϕ-generalization of $b \in \mathrm{CL}_2(B(\alpha))$.
Table 5.2

Redex	*Contractum*
$\mathbf{S}_{ABC}abc$	$ac(bc)$
$\mathbf{K}_{AB}ab$	a
$\mathbf{I}_{A}a$	a
$\mathbf{U}^{\forall\phi B(\phi)\to B(C)}a$	$b(^{\alpha}_{C})$

The second class of redexes is defined by:

Definition 5.4 (i) An *open* CL_2-term is one of the shape $\mathbf{S}_{ABC}ab$ or $\mathbf{S}_{ABC}a$, where $a, b \in$ Ir.

(ii) Any open[5], reducible CL_2-term a of type $A\to$B is an *open redex*. Its contractum is $\lambda x^A.ax^A$, where x^A is the first term-variable of type A that does not occur in a and is also distinct from every λ-binding variable that has a within its scope.

Definition 5.5 A λ'-*redex* is any $\mathfrak{J}_2$-term of the shape $\lambda x.a$, where $a \in Ir$. Its contractum is $[x]a$.

Definition 5.6 A $\wedge$-*redex* is any CL$_2$-term of the shape $\Xi_{\forall\phi(F\to G)}(\wedge_{\alpha,\phi}a)(\wedge_{\alpha,\phi}b)$, where $ab \notin$ Ir. Its contractum is $\Lambda_{\alpha,\phi}(ab)$.

Remark this reduction rule does for reducible terms of the shape $\wedge_{\alpha,\phi}a$ roughly the same job as Definition 5.4 does for open reducible terms. Despite what the notation might suggest, $\wedge_{\alpha,\phi}a$ will in general have a radically different form from $\Lambda_{\alpha,\phi}a$, since the latter is formed from a by a single term-building operation while the operation which forms $\wedge_{\alpha,\phi}a$ from a may be complicated and, in general, a will not be a subterm of $\wedge_{\alpha,\phi}a$.

[5] Why it is possible to restrict the class of open redexes to terms of the two shapes mentioned in clause (1) is explained at pp. 91–93 of [5].

Definition 5.7 A Λ-*redex* is any $\mathfrak{J}_2$-term of the shape $\Lambda_{\alpha,\phi}a$, where $a \in Ir$. Its contractum is $\wedge_{\alpha,\phi}a$.

Corollaries a redex has the same type as its contractum. No redex has more than one contractum.

Let the maximal head of a $\mathfrak{J}_2$-term a be written as $Xa_1a_2...a_n$, where X is an atom and $a_1, a_2 \ldots a_n$ are arbitrary $\mathfrak{J}_2$-terms. Then we define:

Definition 5.8 A subterm b of a is an *active redex within a* iff one of the following holds.

(i) $a_i \in$ Ir for each i such that $1 \leq i \leq n$ and b is in head position within a and b is a weak redex; or
(ii) $a_i \in$ Ir for each i such that $1 \leq i \leq n$ and $Xa_1a_2...a_n$ is an open redex or a $\wedge$-redex and $b \equiv Xa_1a_2...a_n$; or
(iii) $b \equiv \lambda x^A(Xa_1a_2...a_n)$ for some x^A and b is a λ'-redex; or
(iv) $b \equiv \Lambda_{\alpha,\phi}(Xa_1a_2...a_n)$ for some α, ϕ and b is a Λ-redex; or
(v) $j < n, a_1, a_2, \ldots a_j \in$ Ir, $a_{j+1} \notin$ Ir and b is an active redex in a_{j+1}.

Corollary *No $\mathfrak{J}_2$-term contains more than one active redex.*

Definition 5.9 (i) $a \rhd_1 b$ shall hold iff b is derived from a by replacing the active redex in a with its contractum.
(ii) A *reduction (sequence) for a* shall be any sequence $a, a_1, a_2, \ldots$ with the property that, for a any nonterminal a_i, $a_i \rhd_1 a_{i+1}$.
(iii) $a \rhd b$ shall mean that a and b are, respectively, the first and last elements of some reduction sequence.
(iv) A *complete* reduction sequence is one which either continues ad infinitum or else has a last element which contains no active redex.
(v) The last element (if it exists) of any complete reduction sequence for a shall be the *normal form of a*.

Proposition 5.10 *Suppose $a \in \mathfrak{J}_2(A)$; then the following are all equivalent.*

(i) $a \in Ir(A)$.
(ii) *a contains no active redex.*
(iii) *a contains no redexes.*
(iv) *a is a normal form.*

Proof that (i) implies (iii) is easily seen by induction on the deduction of $a \in \mathrm{Ir}(A)$ (Table 5.1). (iii) implies (ii) because every active redex is a redex. That (ii) implies (i) is proven by induction on the deduction of $a \in \mathfrak{J}_2(A)$.

If $a \equiv \lambda xb$ or $a \equiv \Lambda_{\alpha,\phi}b$, then a is its own active redex if $b \in$ Ir. If $b \notin$ Ir, then by the (contraposition of the) I.H., there is an active redex in b and therefore in a.

If $a \equiv Xa_1a_2...a_n$ and some a_i is reducible, then, again by the contraposition of the I.H., the leftmost such a_i contains the active redex of a. From now on, therefore, we assume that every a_i within $Xa_1a_2...a_n$ is irreducible. If $Xa_1a_2...a_n$ begins with a combinator but has no active redex, then a must have one of the following shapes: $\mathbf{S}_{ABC}$, $\mathbf{K}_{AB}$, $\mathbf{K}_{AB}b$ (with $b \in$ Ir), $\mathbf{I}_A$, $\mathbf{S}_{ABC}b$, $\mathbf{S}_{ABC}bc$. In the first four cases, $a \in$ Ir is then easily shown using Table 5.1. If $a \equiv \mathbf{S}_{ABC}b$ or $a \equiv \mathbf{S}_{ABC}bc$ where $b, c \in \mathrm{Ir}$ and if a is not a redex, then $a \in \mathrm{Ir}$ by contraposing Definition 5.4. Similarly, if $Xa_1a_2...a_n$ begins with a $\mathbf{U}$-term, a variable, a Ξ-term or a Θ-term and contains no active redex, $a \in \mathrm{Ir}$ can be shown using the rules of Table 5.1.

If $a \equiv \Pi_{\alpha,\phi}b$ for some b which has no active redex, then $a \in \mathrm{Ir}$ follows easily from the I.H. and rule (c) of Table 5.1.

(ii) and (iv) are equivalent because, by Definition 5.9, a is a normal form iff there is no active redex in a.

Proposition 5.11 *If a' is the normal form of a and α is not used as an eigenvariable in the construction of a, then $a'\binom{\alpha}{C}$ is the normal form of $a\binom{\alpha}{C}$, for any type C.*

Proof by induction on the length of the reduction sequence leading from a to a'.

6 Relations between the calculi of Λ_2-terms and CL_2-terms

In this section, $M, N, P, Q, M_0, M_1, \ldots$ shall be Λ_2-terms while $a, b, c, d, e, a_0, a_1, \ldots$ shall be $\mathfrak{J}_2$-terms. $M =_{\Lambda_2} N$ shall mean that the equation $M = N$ is provable in the calculus of Λ_2-terms defined on p. 83 of [7], with η-conversion included.

The *H-transformation* defined by the following definition takes each Λ_2-term M to a CL_2-term M_H.

Definition 6.1 (i) If M is a variable, then $M_H \equiv M$.

(ii) If $M \equiv \mathrm{PQ}$, then $M_H \equiv P_H Q_H$.

(iii) If $M \equiv \lambda x^A N$, then $M_H \equiv [x^A](N_H)$ as defined by Definition 4.6.

(iv) If $M \equiv \Lambda_{\alpha,\phi}N$, then $M_H \equiv \wedge_{\alpha,\phi}(N_H)$.

(v) If $M \equiv NC$ for $N \in \Lambda_2(\forall\phi A(\phi))$ and some type C, then $M_H \equiv \mathbf{U}^{\forall\phi A(\phi)\to A(C)}(N_H)$.

Proposition 6.2 *When a CL_2-term b is itself a redex, the active redex within b is b itself.*

Proof immediate from Definition 5.8.

Proposition 6.3 *If $M_H \equiv \mathbf{S}_{ABC}ab$, then $M =_{\Lambda_2} \lambda z^A.Nz^A(Pz^A)$ for some N, P such that $N_H \equiv a$, $P_H \equiv b$.*

Proof assume first that M contains no η-redexes. Then inspection of the right-hand terms of Definition 6.1 shows that M must be either $\lambda z^A.L$ or L_0L_1 for some L, L_0, L_1. In the former case, since $M_H \equiv \mathbf{S}_{ABC}ab$, L must be K_0K_1 for some K_0, K_1. So $M \equiv \lambda z^A(K_0K_1) =_{\Lambda_2} \lambda z^A.((\lambda v^A K_0)z^A((\lambda v^A K_1)z^A))$ where $(\lambda v^A.K_0)_H \equiv a$ and $(\lambda v^A.K_1)_H \equiv b$.

If $M \equiv L_0L_1$, for some L_0, L_1, then $L_{0H} \equiv \mathbf{S}_{ABC}a$ and $L_{1H} \equiv b$ and either $L_0 \equiv K_0K_1$ for some K_0, K_1 with $K_{0H} \equiv \mathbf{S}_{ABC}$ and $K_{1H} \equiv a$ or $L_0 \equiv \lambda y^{A\to B}K$ for some K. In the latter case, inspection of Definition 6.1 shows that $(\lambda y^{A\to B}K)_H \equiv \mathbf{S}_{ABC}a$ is possible only if $K \equiv \lambda z^A.J(y^{A\to B}z^A)$ for some J in which $y^{A\to B}$ does not occur. As before, $\lambda y^{A\to B}z^{A}.J(y^{A\to B}z^{A}) =_{\Lambda_2} \lambda y^{A\to B}z^{A}.(\lambda v^{A}J)z^{A}(y^{A\to B}z^{A})$ where $(\lambda v^{A}J)_{H} \equiv a$. Then $(\lambda y^{A\to B}K)L_{1} =_{\Lambda_2} \lambda z^{A}.(\lambda v^{A}J)z^{A}(L_{1}z^{A})$, where $(L_{1})_{H} \equiv b$.

If $L_{0} \equiv K_{0}K_{1}$, then $K_{0H} \equiv \mathbf{S}_{ABC}$ while $K_{1H} \equiv a$. In this case, inspection of Definition 6.1 shows that $K_{0} \equiv \lambda x^{A\to B\to C}y^{A\to B}z^{A}.x^{A\to B\to C}z^{A}(y^{A\to B}z^{A})$ is a necessary condition of $K_{0H} \equiv \mathbf{S}_{ABC}$. So $L_{0}L_{1} \equiv K_{0}K_{1}L_{1} =_{\Lambda_2} \lambda z^{A}(K_{1}z^{A}(L_{1}z^{A}))$, where $K_{1H} \equiv a$ and $L_{1H} \equiv b$.

If M contains η-redexes and M' is the term obtained from M by contracting them all, then by Definitions 4.6 and 6.1 $M_{H} \equiv M'_{H}$ and the foregoing argument applies.

Proposition 6.4 *If $M_{H} \equiv \mathbf{S}_{ABC}abc$, then $M =_{\Lambda_2} (\lambda z^{A}.Nz^{A}(Pz^{A}))Q$ for some N, P, Q such that $N_{H} \equiv a$, $P_{H} \equiv b$ and $Q_{H} \equiv c$.*

Proof as before, the general case can be reduced to the case where M contains no η-redexes. Then inspection of the right-hand terms of Definition 6.1 shows that M can only have the shape $M_{0}Q$, where $Q_{H} \equiv c$ and $M_{0H} \equiv \mathbf{S}_{ABC}ab$. Now apply Proposition 6.3.

Proposition 6.5 *If $e \equiv M_{H}$ is a weak redex which contracts to e', then $M =_{\Lambda_2} R$ for some R such that $R_{H} \equiv e'$.*

Proof e must have one of the four shapes shown in Table 5.2. If $e \equiv \mathbf{S}_{ABC}abc$, then by Proposition 6.4, $M =_{\Lambda_2} NQ(PQ)$ for some N, P, Q such that $(NQ(PQ))_{H} \equiv ac(bc)$. If $e \equiv \mathbf{K}_{AB}ab$ then either $M \equiv (\lambda v^{B}M_{1})M_{2}$ where $M_{1H} \equiv a$ and $M_{2H} \equiv b$ or $M \equiv M_{0}M_{1}M_{2}$, where $M_{0H} \equiv \mathbf{K}_{AB}$, $M_{1H} \equiv a$ and $M_{2H} \equiv b$. In either case, $M =_{\Lambda_2} M_{1}$ and $M_{1H} \equiv a \equiv e'$. If $e \equiv \mathbf{I}_{A}a$ then $M \equiv (\lambda v^{A}.v^{A})N$, where $N_{H} \equiv a \equiv e'$. Finally, if $e \equiv \mathbf{U}^{\forall\phi A(\phi)\to A(C)} \wedge_{\alpha,\phi} a$ then $M \equiv (\Lambda_{\alpha,\phi}N)C$ for some N having type $A(\alpha)$ such that $N_{H} \equiv a$. Then $M =_{\Lambda_2} N\binom{\alpha}{C}$ and $N\binom{\alpha}{C}_{H} \equiv a\binom{\alpha}{C}$.

Proposition 6.6 *If $M_{H} \equiv \wedge_{\alpha,\phi}a$, then $M =_{\Lambda_2} \Lambda_{\alpha,\phi}N$ for some N such that $N_{H} \equiv a$.*

Proof by Definition 4.3, $\wedge_{\alpha,\phi}a$ will have one of the following shapes: $\Pi_{\alpha,\phi}a$, $\Theta_{\forall\phi B}a$, $\Xi_{\forall\phi(F\to G)}bc$. In the first and third cases, the stronger proposition $M \equiv \Lambda_{\alpha,\phi}N$ holds. For $\Pi_{\alpha,\phi}a$ can only be the H-transform of $\Lambda_{\alpha,\phi}N$ for some N such that $N_{H} \equiv a$. In the third case, it might seem that $\wedge_{\alpha,\phi}a$ could also be the H-transform of $M_{0}M_{1}$, where, for example, $M_{0H} \equiv \Xi_{\forall\phi(F\to G)}b$ and $M_{1H} \equiv b$. However there is no Λ_2-term of which $\Xi_{\forall\phi(F\to G)}b$ is the H-transform, just as if there is none of which $\Xi_{\forall\phi(F\to G)}$ is. An atom of this last shape could only be obtained by the H-transformation of Definition 6.1 if an application of clause (iv) yielded $\Xi_{\forall\phi(F\to G)}xy$ and this were followed by two applications of clause (iii); but an application of clause (iv) can only yield terms of the shape $\Xi_{\forall\phi(F\to G)}bc$ where b and c are themselves generalizations, which means they cannot be variables.

If $\wedge_{\alpha,\phi}a \equiv \Theta_{\forall\phi B}a$, then a can indeed be a variable and M can be an applicative term $M_{0}M_{1}$. In this case, $M_{0} \equiv \lambda x^{B}\Lambda_{\alpha,\phi}.x^{B}$ so that $M_{0H} \equiv [x^{B}]\Theta_{\forall\phi B}x^{B} \equiv \Theta_{\forall\phi B}$, while $M_{1H} \equiv a$. But then $M_{0}M_{1}$ β-contracts to $\Lambda_{\alpha,\phi}M_{1}$ where $M_{1H} \equiv a$.

Definition 6.7 If b is the active redex in a, then $d(b, a)$, the *depth of b within a*, is defined to be:

(i) 0 if b is in head position within a;
(ii) $d(b, e)+1$ if $a \equiv ce$ and b occurs inside e.

Proposition 6.8 *If $M_H \equiv a$ and b is the normal form of a, then there is an N such that $M =_{\Lambda_2} N$ and $N_H \equiv b$.*

Proof as a is an H-transform, by Definition 6.1 a will be a CL$_2$-term. Moreover, Ξ-terms will occur in a only at the head of generalizations: that is, in a context $\Xi_{\forall\phi(F\rightarrow G)}cd$, where c and d are themselves generalizations.

The proposition is trivial if there are no redexes in a. Assume then that there are redexes in a and let n be the number of terms following a in the reduction sequence which leads from a to b, let p be the depth of the active redex within a and let q be the number of times the operation $c \mapsto \Pi_{\alpha,\phi}c$ is used in the construction of a. The proposition is now proven by transfinite induction on $\omega^n + p + q$.

Only composite terms can be reduced, so a has either the shape $\Pi_{\alpha,\phi}a_0$ or the shape a_0a_1. In the former case, the proposition easily follows from the I.H., so we assume now that $a \equiv a_0a_1$.

Case 1 a is $ce_1e_2...e_m$, where $0 \leq m$ and the active redex in a is c, which is a weak redex. If $M_H \equiv ce_1e_2...e_m$, then by Definition 6.1 $M \equiv PQ_1Q_2...Q_m$ for some $P, Q_1, Q_2, \dots Q_m$ such that $P_H \equiv c$ etc. By Proposition 6.5, there is a P' such that $PQ_1Q_2...Q_m =_{\Lambda_2} P'Q_1Q_2...Q_m$ and $P'_H \equiv c'$, where c' is the contractum of c. By the hypothesis of the induction on n, there is a term N such that N_H is the normal form of $c'e_1e_2...e_m$ and $P'Q_1Q_2...Q_m =_{\Lambda_2} N$. The conclusion follows using the transitivity of =.

Case 2 a is $\mathbf{S}_{ABC}c_1...c_i$ where $i \in \{1, 2\}$ and $c_1, \dots c_i \in$ Ir. If $i = 2$, then a reduces in two steps to $\lambda x^A.c_1x^A(c_2x^A)$. By Proposition 6.3 there are terms N, P such that $M =_{\Lambda_2} \lambda z^A.Nz^A(Pz^A)$ and $N_H \equiv c_1$, $P_H \equiv c_2$. By the stipulations of Sect. 5, a has a normal form only if $c_1x^A(c_2x^A)$ does. By Definition 6.1, $(Nx^A(Px^A))_H \equiv c_1x^A(c_2x^A)$. Since the normal form of $c_1x^A(c_2x^A)$ is reached in $\leq n-3$ steps, the I.H. of the induction on n entails there is a term Q such that $Nx^A(Px^A) =_{\Lambda_2} Q$ and Q_H is the normal form of $c_1x^A(c_2x^A)$. By rule (ξ), $\lambda x^A.Nx^A(Px^A) =_{\Lambda_2} \lambda x^A.Q$ while, by Definition 6.1, $(\lambda x^A.Q)_H \equiv [x^A]Q_H$ and, by the stipulations of Sect. 5, $[x^A]Q_H$ is the normal form of a. It follows that $\lambda x^A.Q$ is the term required to make the proposition true.

If $i = 1$, $M \equiv \lambda v^{A\rightarrow B}K$ for some K such that $K_H \equiv \mathbf{S}_{ABC}c_1v^{A\rightarrow B}$ for some $v^{A\rightarrow B}$ not in c_1. By the argument of the last paragraph, there is then a Q such that $K =_{\Lambda_2} \lambda x^A.Q$ and $(\lambda x^A.Q)_H \equiv e$ for some e which is the normal form of $\mathbf{S}_{ABC}c_1v^{A\rightarrow B}$. In that case, $[v^{A\rightarrow B}]e$ is the normal form of $\mathbf{S}_{ABC}c_1$ and $(\lambda v^{A\rightarrow B}x^A.Q)_H \equiv [v^{A\rightarrow B}]e$.

Case 3 a is a $\wedge$-redex, say $\wedge_{\alpha,\phi}b$. By Definition 5.6 and Proposition 6.2, $a \rhd_1 \Lambda_{\alpha,\phi}b$. Since a is a redex it follows from Definition 5.6 that b is reducible. There is therefore a reduction of $n-2$ steps taking b to its normal form b'. By Proposition 6.6, $M \equiv \Lambda_{\alpha,\phi}P$ for some P such that $P_H \equiv b$, so by the I.H. on n there is a Q such that $P =_{\Lambda_2} Q$ and $Q_H \equiv b'$. By rule (ξ) for type-abstraction, $M \equiv \Lambda_{\alpha,\phi}P =_{\Lambda_2} \Lambda_{\alpha,\phi}Q$, so $\Lambda_{\alpha,\phi}Q$ is the term that makes the proposition true in this case.

Case 4 by Definition 5.8, the only remaining possibility is that $a \equiv Xd_1d_2...d_m$ for some atom X and the active redex in a is inside d_k. In this case, there are various possible shapes that M may have but, by Propositions 6.3, 6.4 and 6.6, M will have a component Q_k such that $Q_{kH} \equiv d_k$. By Definition 5.8, the reduction of a must begin by repeatedly contracting the active redex in d_k until its normal form d' is reached, at which point either we have reached the normal form of a or else there is an active redex in $Xd_1d_2...d'...d_m$. In the latter case, since less than n steps are needed in order to reach d', by the I.H. of the induction on n there is a Λ_2-term Q' such that $Q_k =_{\Lambda_2} Q'$ and $Q'_H \equiv d'$, from which follows that M can be converted to a term M' such that $M'_H \equiv Xd_1d_2...d'...d_m$ by converting the component Q_k to Q' while, by another application of the I.H. on n, M' can in turn be converted to a term N such that N_H is the normal form of $Xd_1d_2...d'...d_m$ and therefore of a.

If n reduction steps are needed in order to reduce d_k to its normal form, then use is made of the I.H. on the depth of the active redexes, since this is smaller for d_k than it is for a.

Proposition 6.9 *If $M_H \in Ir$ and there are no η-redexes in M, then there are no β-redexes in M either.*

Proof by induction on the construction of M.

If $M \equiv M_0M_1$, then $M_H \equiv M_{0H}M_{1H}$. Because $M_{0H}M_{1H}$ is in Ir, it follows from the definition of Ir (Table 5.1) together with Proposition 5.10 that M_{0H} and M_{1H} contain no redexes. It follows by the I.H. that M_0 and M_1 contain no redexes of Λ_2. Nor can M_0M_1 be a β-redex because, if it were, either M_0 would be an η-redex or $M_{0H}M_{1H}$ would be a weak redex and therefore not in Ir.

If $M \equiv \lambda v^A N$ for some v^A and N, then by Definition 6.1 M_H is $\mathbf{S}_{ABC}ab$, $\mathbf{K}_{BA}a$ or $\mathbf{I}_A$ for some a, b, B, C (there would be other possibilities if M were an η-redex but it is not). So one of the following must hold: $N \equiv N_0N_1$, $N_H \equiv a$ or $N \equiv v^A$. In the second and third cases, N clearly contains no β-redex (using the I.H. in the case where $N_H \equiv a$). By definition of Ir, $\mathbf{S}_{ABC}ab$ is in Ir only if N_H is. By the I.H., N_H is in Ir only if N contains no β-redex.

If $M \equiv \Lambda_{\alpha,\phi}N$ then $M_H \equiv \wedge_{\alpha,\phi}N_H$. By the definition of Ir (Table 5.1; especially row (c) and the condition appended to (f)) $\wedge_{\alpha,\phi}N_H$ is in Ir only if N_H is. By the I.H., there are no β-redexes in N.

If $M \equiv NC$ for some type C, then by Definition 6.1 $M_H \equiv \mathbf{U}^{\forall\phi B(\phi)\to B(C)}N_H$ for some B and ϕ. Since this is in Ir, so is N_H and therefore, by the I.H., N contains no β-redexes.

Proposition 6.10 *If $M_H \in Ir$, there are no type-redexes in M.*

Proof by induction on the construction of M.

If $M \equiv NC$ for some type C, then by Definition 6.1 $M_H \equiv \mathbf{U}^{\forall\phi B(\phi)\to B(C)}N_H$ for some B and ϕ. Since this term is in Ir it is, by Proposition 5.10, not a redex and therefore N_H is not a generalization. By Definition 6.1, it follows that N is not formed by Λ-abstraction.

That no subterm of M is a type-redex is shown, both in this case and in general, using the definition of Ir and the I.H., just as in the proof of 6.9.

Theorem 1 *If Σ is a class of CL_2-terms such that every term in Σ has a normal form, so does every Λ_2-term M such that M_H is in Σ.*

Proof if $M_H \in \Sigma$, by Proposition 6.8 there is an N such that $M =_{\Lambda_2} N$ and N_H is the normal form of M_H. Let all η-redexes in N be contracted and let N' be the result. By Definition 4.6, $N_H \equiv N'_H$. By Propositions 6.9 and 6.10, N' contains no redexes.

7 Normalizability of CL$_2$-terms

Theorem 1 shows (or, at least, so it will be argued in Sect. 8) that interesting consequences follow from the assumption that all CL$_2$-terms are normalizable. The theorem will, admittedly, be of limited interest if there are CL$_2$-terms which are *not* normalizable. The purpose of this section is to apply the method invented by Girard ([6]) and used in [1,7,12,19,24] to show that there are no such terms.

Definition 7.1 The *rank* of a type A, henceforth $\rho(A)$, is defined to be:-

(i) 0 if A is a type-variable.
(ii) $\max\{\rho(B), \rho(C)\} + 1$ if $A \equiv B \to C$.
(iii) $\rho(B(\alpha)) + 1$ if $A \equiv \forall\phi B(\phi)$.

Definition 7.2 For each type A, a subset S of $CL_2(A)$ is *regular* iff:

(i) Every term in S has a normal form.
(ii) S contains every term of type A that can be built up by starting with either a variable or a term of the shape $\Xi_{\forall\phi(F\to G)}ab$ which is irreducible and not a generalization and applying the two operations:

(α) $a \mapsto ab_1b_2...b_n$ for some normalizable CL_2-terms $b_1, b_2, \ldots b_n$.
(β) $a \mapsto Xa$, for some **U**-term X.

(iii) If $a' \in S$ and $a \rhd a'$, then $a \in S$.
(iv) If $a \in S$ and $a \rhd a'$, then $a' \in S$.

Definition 7.3 A *valuation* (sc., of the free type-variables) is a function ξ with the property that, for every α, there is an A such that $\xi(\alpha)$ is a regular subset of $CL_2(A)$.

The letters $\xi, \xi', \xi'', \ldots$ will range over valuations.

Definition 7.4 For each type A, A^ξ is the type defined by:

(i) If $A \equiv \alpha$, then A^ξ is the type of the elements of $\xi(\alpha)$.
(ii) If $A \equiv B \to C$, then $A^\xi \equiv B^\xi \to C^\xi$.
(iii) If $A \equiv \forall\phi(B\binom{\alpha}{\phi})$, then $A^\xi \equiv \forall\phi(B^{\xi'}\binom{\alpha}{\phi})$, where ξ' assigns an arbitrary regular set of type α to α and otherwise agrees with ξ.

Corollary *if $\xi(\alpha)$ is a set of type C, then $(A(\alpha))^\xi \equiv A^\xi(C)$.*

Definition 7.5 For any CL_2-term a and any valuation ξ, the term a^ξ shall be constructed as follows:

(i) If a is an atom of type A described on some row of Table 4.1, a^ξ shall be the atom of type A^ξ on the same row.

(ii) If $a \equiv a_0 a_1$, then a^ξ shall be $a_0^\xi a_1^\xi$

(iii) If $a \equiv \Pi_{\alpha,\phi} a_0$, then a^ξ shall be $\Pi_{\alpha,\phi}(a_0^{\xi'})$, where ξ' takes α to a set of type α but otherwise agrees with ξ.

Corollary *if* $a \in CL_2(A)$ *then* $a^\xi \in CL_2(A^\xi)$.

The next definition defines the sequence of sets $\mathrm{Comp}_{\xi,A}$, read as "the set of terms that are computable relative to A and ξ", for each ξ and A.[6] The definition is by recursion on $\rho(A)$. In clause (iii), the type $\forall\phi B(\phi)$ shall be $(\forall\phi A(\phi))^\xi$.

Definition 7.6 (i) If $A \equiv \alpha$, then $\mathrm{Comp}_{\xi,A} = \xi(\alpha)$.

(ii) $\mathrm{Comp}_{\xi,A\to B} = \{a$: if $b \in \mathrm{Comp}_{\xi,A}$ then $ab \in \mathrm{Comp}_{\xi,B}\}$.

(iii) $\mathrm{Comp}_{\xi,\forall\phi A(\phi)} = \{a$: for each type C, $\mathbf{U}^{\forall\phi B(\phi)\to B(C)} a \in \mathrm{Comp}_{\xi',A(\alpha)}$ for every valuation ξ' which assigns a regular set of type C to α and otherwise agrees with $\xi\}$.

Proposition 7.7 *For each* A *and* ξ, *Comp*$_{\xi,A}$ *is a subset of* $CL_2(A^\xi)$.

Proof by induction on $\rho(A)$. If $A \equiv B\to C$, then by clause (ii) of Definition 7.6, $\mathrm{Comp}_{\xi,A}$ is a set of terms each of which, applied to a term in $\mathrm{Comp}_{\xi,B}$ yields a term in $\mathrm{Comp}_{\xi,C}$. Applying the I.H., this is equivalent to saying $\mathrm{Comp}_{\xi,A}$ is a set of terms of type $B^\xi \to C^\xi$. Suppose A is a quantified type and $A^\xi \equiv \forall\phi B(\phi)$. Then the elements of $\mathrm{Comp}_{\xi,A}$ must have type $\forall\phi B(\phi)$ in order for $\mathbf{U}^{\forall\phi B(\phi)\to B(C)} a$ to be well-formed.

Definition 7.8 For any term b of type B and any A and ξ such that $A^\xi \equiv B$, a term c is an (A, ξ)-*supplementation* of b iff one of the following conditions is fulfilled:

(i) A is a type-variable and $c \equiv b$.

(ii) $A \equiv A_0 \to A_1$ and c is an (A_1, ξ)-supplementation of bd for some d in $\mathrm{Comp}_{A_0,\xi}$.

(iii) $A \equiv \forall\phi D(\phi)$ for some D, which means that B is also a quantified type, say $\forall\phi E(\phi)$. Moreover, there is a C such that $\xi(\alpha)$ is of type C, $(D(\alpha))^\xi \equiv E(C)$ and c is a $(D(\alpha), \xi)$-supplementation of $\mathbf{U}^{\forall\phi E(\phi)\to E(C)} b$.

It follows easily from this definition that an (A, ξ)-supplementation of a term in $\mathrm{CL}_2(A^\xi)$ will have the type of the terms in $\xi(\beta)$, for some β. Further work, though, is needed to show that it will also be *in* $\xi(\beta)$ for some β.

Proposition 7.9 *For any term* b *of type* B *and any* A *and* ξ *such that* $A^\xi \equiv B$, *if every* (A, ξ)-*supplementation of* b *is in* $\xi(\alpha)$ *for some* α *occurring in* A, *then* $b \in$ *Comp*$_{\xi,A}$.

Proof by induction on $\rho(A)$. If A is a type-variable, the proposition is obvious. Suppose $A \equiv A_0 \to A_1$ and let d be in Comp_{ξ,A_0}. By the I.H., if every (A_1, ξ)-supplementation of bd is in $\mathrm{Comp}_{\xi,\alpha}$ for some α occurring in A_1, then bd is in Comp_{ξ,A_1}. By Definitions 7.6 and 7.8, this is equivalent to the proposition to be proven.

To deal with the case where $A \equiv \forall\phi D(\phi)$, note that the last sentence of clause (iii) of Definition 7.8 is equivalent (using the corollary to Definition 7.4) to the following:

[6] If the normalizability proof presented in this section appears complicated, this is largely to be put down to the fact that the type of terms in $\mathrm{Comp}_{\xi,A}$ is A^ξ, which is not, in general, the same as A. This is due to the fact that the terms in $\xi(\alpha)$ may have *any* type (not necessarily α); and this in turn seems unavoidable in view of the fact that valuations were introduced in order to deal with instantiations of universal quantifiers ranging over all types. Valuable discussions of the heuristic thinking behind Girard's proof can be found in [7,12] ("Appendix B") and [23] (Sect. 7.9).

There are a C and a ξ' such that ξ' takes α to a set of type C and otherwise agrees with ξ; and c is a $(D(\alpha), \xi')$-supplementation of $\mathbf{U}^{\forall\phi E(\phi)\to E(C)}b$.

Suppose now $A \equiv \forall\phi D(\phi)$ while $A^{\xi} \equiv \forall\phi E(\phi)$ and every $(\forall\phi D(\phi), \xi)$-supplementation of b is in $\xi(\beta)$ for some β occurring in A. That is, by Definition 7.8, for every C and ξ' such that ξ' takes α to a set of type C and otherwise agrees with ξ, every $(D(\alpha), \xi')$-supplementation of b is in $\xi(\beta)$ for some β occurring in A. Since α does not occur in A, β is distinct from α and "$\xi(\beta)$" may here be replaced with "$\xi'(\beta)$". Then by the I.H. $\mathbf{U}^{\forall\phi E(\phi)\to E(C)}b$ is in $\mathrm{Comp}_{\xi', D(\alpha)}$ for every such ξ'. But this is precisely the condition for b to be in $\mathrm{Comp}_{\xi, \forall\phi D(\phi)}$.

Proposition 7.10 *(i) For every ξ and A, every element of $Comp_{\xi,A}$ has a normal form. (ii) For every term-variable x^A and every ξ, $(x^A)^{\xi} \in Comp_{\xi,A}$.*

Proof the two halves are proven simultaneously by induction on $\rho(A)$. If A is a type-variable α, then $(x^A)^{\xi} \in \mathrm{Comp}_{\xi,A}$ by clause (ii) of Definition 7.2 while every element of $\mathrm{Comp}_{\xi,A}$ has a normal form by clause (i) of the same definition.

Suppose $\rho(A) > 0$. Then an (A, ξ)-supplementation of $(x^A)^{\xi}$ is built up by starting with $(x^A)^{\xi}$ and applying the two operations $a \mapsto ab_1b_2...b_n$ for computable $b_1, b_2, \ldots b_n$ and $a \mapsto Xa$, for some $\mathbf{U}$-term X. The terms $b_1, b_2, \ldots b_n$ used in the first operation all have normal forms by part (i) of the I.H., so every (A, ξ)-supplementation of $(x^A)^{\xi}$ is in $\xi(\beta)$ for some β by clause (ii) of Definition 7.2, which is sufficient by Proposition 7.9 for $(x^A)^{\xi}$ to be in $\mathrm{Comp}_{\xi,A}$.

Suppose $a \in \mathrm{Comp}_{\xi,A}$ and $A \equiv A_0 \to A_1$. Any variable x^B, where $B \equiv A_0^{\xi}$, is in Comp_{ξ,A_0} by part (ii) of the I.H. By Definition 7.6 therefore, ax^B is in Comp_{ξ,A_1} and therefore has a normal form, say a', by part (i) of the I.H. So $[x^B]a'$ is irreducible by clause (b) of Table 5.1 and is also, by Sect. 5, a term to which a reduces.

Finally if $a \in \mathrm{Comp}_{\xi,A}$, if $A \equiv \forall\phi B(\phi)$ and if $B^{\xi}(\alpha) \equiv D(\alpha)$, then by clause (iii) of Definition 7.6, $\mathbf{U}^{\forall\phi D(\phi)\to D(\alpha)}a \in \mathrm{Comp}_{\xi', B(\alpha)}$ where ξ' is a valuation which takes α to a set of type α and otherwise agrees with ξ. $B(\alpha)$ has a lower rank than $\forall\phi B(\phi)$, so by part (i) of the I.H. $\mathbf{U}^{\forall\phi D(\phi)\to D(\alpha)}a$ has a normal form, which can only be the case if $\mathbf{U}^{\forall\phi D(\phi)\to D(\alpha)}a'$, where a' is the normal form of a, is a term of its reduction sequence.

Proposition 7.11 *Every (A, ξ)-supplementation of a term b in $Comp_{\xi,A}$ belongs to $\xi(\beta)$ for some β which occurs in A.*

Proof by induction on $\rho(A)$. If A is a type-variable α, then $\xi(\alpha) = \mathrm{Comp}_{\xi,A}$. Suppose $A^{\xi} \equiv B$. If $A \equiv A_0 \to A_1$ then $B \equiv B_1 \to B_1$, where $B_i \equiv A_i^{\xi}$ for i = 1,2. Let c be an arbitrary term in Comp_{ξ,A_0}; then by the I.H. every (A_1, ξ)-supplementation of bc is in $\xi(\beta)$ for some β; but by Definition 7.8 this is all that is required. If $A \equiv \forall\phi D(\phi)$ then $B \equiv \forall\phi E(\phi)$, for some $E(\alpha)$ such that $D(\alpha) \equiv E(\alpha)^{\xi'}$, for any valuation ξ' which takes α to a set of terms of type α and otherwise agrees with ξ. Suppose that ξ takes $D(\alpha)$ to $E(C)$; then, by the I.H., every $(D(\alpha), \xi)$-supplementation of $\mathbf{U}^{\forall\phi E(\phi)\to E(C)}b$ is in $\xi(\beta)$ for some β; but then by Definition 7.8 so is every (A, ξ)-supplementation of b.

Proposition 7.12 *$Comp_{\xi,A}$ is a regular set for every ξ and A.*

Proof by Proposition 7.10, $\mathrm{Comp}_{\xi,A}$ has property (i) of the four properties mentioned in Definition 7.2.

That it has property (ii) will be shown by induction on $\rho(A)$. The basis is obvious. Let $A \equiv A_0 \rightarrow A_1$ and a be a term of type A^ξ that is built up in the manner described in clause (ii) of Definition 7.2. Let b be a normalizable term of type A_0^ξ; now by the I.H. Comp_{ξ,A_1} is a regular set and therefore contains ab, since that term is of the shape described in clause (ii) of Definition 7.2. If this is so for every normalizable b, then *a fortiori* (by Proposition 7.10) it is so for every b that is in Comp_{ξ,A_0}. But the latter is the very condition required by Definition 7.6 for a to be in $\mathrm{Comp}_{\xi,A}$. A fairly similar argument deals with the case where A is a quantified type.

That $\mathrm{Comp}_{\xi,A}$ has properties (iii) and (iv) of the four properties mentioned in Definition 7.2 can be shown by considering an arbitrary (A, ξ)-supplementation of a term in $\mathrm{Comp}_{\xi,A}$ and applying Propositions 7.9 and 7.11.

Corollary *if a is in $Comp_{\xi,A}$, then so is every term to which a reduces.*

Proposition 7.13 *$Comp_{\xi,A(D)} = Comp_{\xi',A(\alpha)}$ for every ξ' such that $\xi'(\alpha) = Comp_{\xi,D}$ and ξ' otherwise agrees with ξ.*

Proof by induction on $\rho(A(\alpha))$ (cf. lemma 11.3.6 in [24]).

Proposition 7.14 *If $a \in CL_2(A)$ is an atomic term, then $a^\xi \in Comp_{\xi,A}$ for every ξ.*

Proof if a is a variable, this is Proposition 7.10. If a is an atomic constant, six cases are distinguished according to which kind of constant a is.

(Case 1) $a \equiv$ $\mathbf{S}_{CDE}$. By Definition 7.6 and Proposition 7.9, it suffices to show that, whenever c^ξ, d^ξ, e^ξ belong to $\mathrm{Comp}_{\xi,C\rightarrow D\rightarrow E}$, $\mathrm{Comp}_{\xi,C\rightarrow D}$, $\mathrm{Comp}_{\xi,C}$ respectively and $f(\mathbf{S}_{CDE}^\xi c^\xi d^\xi e^\xi)$ is an (A, ξ)-supplementation of $\mathbf{S}_{CDE}^\xi$, then $f(\mathbf{S}_{CDE}^\xi c^\xi d^\xi e^\xi) \in \xi(\beta)$ for some β.

By Proposition 7.10, each of c^ξ, d^ξ, e^ξ has a normal form; let c'^ξ, d'^ξ, e'^ξ be the respective normal forms of c^ξ, d^ξ, e^ξ. By the corollary to Proposition 7.12, c'^ξ, d'^ξ, e'^ξ belong to $\mathrm{Comp}_{\xi,C\rightarrow D\rightarrow E}$, $\mathrm{Comp}_{\xi,C\rightarrow D}$ and $\mathrm{Comp}_{\xi,C}$ respectively. By clause (ii) of Definition 7.6, $c'^\xi e'^\xi (d'^\xi e'^\xi) \in \mathrm{Comp}_{\xi,E}$ and therefore (by Proposition 7.11) there is a β such that $f(c'^\xi e'^\xi (d'^\xi e'^\xi)) \in \xi(\beta)$, where f is as in the last paragraph. By clause (iii) of Definition 7.2, $f(\mathbf{S}_{CDE}^\xi c^\xi d^\xi e^\xi) \in \xi(\beta)$ follows.

Cases 2 and 3, i.e. where a is a typed instance of **K** or **I** are treated likewise.

Case 4 $a \equiv$ $\mathbf{U}^{\forall\phi B(\phi)\rightarrow B(C)}$. Let $(\forall\phi B(\phi)\rightarrow B(C))^\xi \equiv \forall\phi D(\phi)\rightarrow D(E)$ and let b be in $\mathrm{Comp}_{\xi,\forall\phi B(\phi)}$. By clause (iii) of Definition 7.6, $\mathbf{U}^{\forall\phi D(\phi)\rightarrow D(E)} b$ is in $\mathrm{Comp}_{\xi',B(\alpha)}$ for every ξ' that takes α to a set of type E and otherwise agrees with ξ, including the particular ξ' that takes α to $\mathrm{Comp}_{\xi,C}$. By Proposition 7.13, $\mathbf{U}^{\forall\phi D(\phi)\rightarrow D(E)} b \in \mathrm{Comp}_{\xi,B(C)}$, from which the conclusion follows by clause (ii) of Definition 7.6.

Case 5 $a \equiv$ $\Xi_{\forall\phi(F\rightarrow G)}$. Let $D(\alpha)$ and $E(\alpha)$ be types such that $\forall\phi D(\phi) \equiv \forall\phi F$ and $\forall\phi E(\phi) \equiv \forall\phi G$. Let b, c belong to $\mathrm{Comp}_{\xi,\forall\phi(F\rightarrow G)}$, $\mathrm{Comp}_{\xi,\forall\phi F}$ respectively and let $f(\mathbf{U}^{\forall\phi(D^\xi(\phi)\rightarrow E^\xi(\phi))\rightarrow D^\xi(C)\rightarrow E^\xi(C)}(\Xi_{\forall\phi(F\rightarrow G)}^\xi bc))$ be an (A, ξ)-supplementation of $\Xi^{\xi}_{\forall\phi(F\to G)}$. Under these assumptions, it suffices to show this last term belongs to $\xi(\beta)$ for some β.

The assumptions on b and c entail, by Proposition 7.10, that b and c have normal forms; let the latter be b' and c' respectively.

Subcase 5a in both of the reduction sequences leading from b to b' and from c to c', there are α/ϕ-generalizations; let $\wedge_{\alpha,\phi}e_0$ and $\wedge_{\alpha,\phi}e_1$ be the first such generalizations which appear in the two respective reduction sequences.

The computability of b and c also entails, by clause (iii) of Definition 7.6, that $\mathbf{U}^{\forall\phi(D^{\xi}(\phi)\to E^{\xi}(\phi))\to D^{\xi}(\alpha)\to E^{\xi}(\alpha)}b(\mathbf{U}^{\forall\phi(D^{\xi}(\phi)\to D^{\xi}(\alpha)}c)$ is in $\mathrm{Comp}_{\xi',E(\alpha)}$ for every ξ' that takes α to a set of type α. This term therefore has a terminating complete reduction that proceeds through $\mathbf{U}^{\forall\phi(D^{\xi}(\phi)\to E^{\xi}(\phi))\to D^{\xi}(\alpha)\to E^{\xi}(\alpha)}(\wedge_{\alpha,\phi}e'_0)(\mathbf{U}^{\forall\phi(D^{\xi}(\phi)\to D^{\xi}(\alpha)}(\wedge_{\alpha,\phi}e'_1))$, where e'_0 and e'_1 are the respective normal forms of e_0 and e_1. Let its normal form, therefore, be e'.

By Sect. 5, the term $f(\mathbf{U}^{\forall\phi(D^{\xi}(\phi)\to E^{\xi}(\phi))\to D^{\xi}(C)\to E^{\xi}(C)}(\Xi^{\xi}_{\forall\phi(F\to G)}bc))$ is reduced by first replacing the component $\Xi^{\xi}_{\forall\phi(F\to G)}bc$ with successive reducts as follows:

$$\begin{aligned}
&\Xi^{\xi}_{\forall\phi(F\to G)}bc \rhd \Xi^{\xi}_{\forall\phi(F\to G)}(\wedge_{\alpha,\phi}e_0)c\\
&\quad\rhd \Xi^{\xi}_{\forall\phi(F\to G)}(\wedge_{\alpha,\phi}e'_0)c\\
&\quad\rhd \Xi^{\xi}_{\forall\phi(F\to G)}(\wedge_{\alpha,\phi}e'_0)(\wedge_{\alpha,\phi}e'_1)\\
&\quad\rhd \Lambda_{\alpha,\phi}(e'_0e'_1) \quad \text{by Definitions 5.6 and 5.8, assuming } e'_0e'_1 \notin \mathrm{Ir}\\
&\quad\rhd \Lambda_{\alpha,\phi}(e') \qquad \text{because } e'_0e'_1 \rhd e' \text{ (see above)}\\
&\quad\rhd \wedge_{\alpha,\phi}(e')
\end{aligned}$$

The term $f(\mathbf{U}^{\forall\phi(D^{\xi}(\phi)\to E^{\xi}(\phi))\to D^{\xi}(C)\to E^{\xi}(C)}(\Xi^{\xi}_{\forall\phi(F\to G)}bc))$ therefore reduces to $f(e'\binom{\alpha}{C})$.

On the other hand, $\mathbf{U}^{\forall\phi(D^{\xi}(\phi)\to E^{\xi}(\phi))\to D^{\xi}(C)\to E^{\xi}(C)}b(\mathbf{U}^{\forall\phi(D^{\xi}(\phi)\to D^{\xi}(C)}c)$ is in $\mathrm{Comp}_{\xi,D(\alpha)}$ in consequence of the assumptions $b \in \mathrm{Comp}_{\xi,\forall\phi(F\to G)}$ and $c \in \mathrm{Comp}_{\xi,\forall\phi F}$, so the term $f(\mathbf{U}^{\forall\phi(D^{\xi}(\phi)\to E^{\xi}(\phi))\to D^{\xi}(C)\to E^{\xi}(C)}b(\mathbf{U}^{\forall\phi(D^{\xi}(\phi)\to D^{\xi}(C)}c))$, being a $(D(\alpha),\xi)$-supplementation of it, is in $\xi(\beta)$ for some β by Proposition 7.11. By Sect. 5, it reduces to $f(e'_0\binom{\alpha}{C}e'_1\binom{\alpha}{C})$, which, by Proposition 5.11, also reduces to $f(e'\binom{\alpha}{C})$. By clause (iv) of Definition 7.2, this is in $\xi(\beta)$ for some β, hence by clause (iii) of the same Definition so is any term that reduces to $f(e'\binom{\alpha}{C})$.

Subcase 5b $\Xi^{\xi}_{\forall\phi(F\to G)}b'c'$ is not an α/ϕ-generalization. Then, by clause (ii) of Definition 7.2, $f(\mathbf{U}^{\forall\phi(D^{\xi}(\phi)\to E^{\xi}(\phi))\to D^{\xi}(C)\to E^{\xi}(C)}(\Xi^{\xi}_{\forall\phi(F\to G)}b'c'))$, being irreducible, belongs to $\xi(\beta)$ for every β. By clause (iii), so does $f(\mathbf{U}^{\forall\phi(D^{\xi}(\phi)\to E^{\xi}(\phi))\to D^{\xi}(C)\to E^{\xi}(C)}(\Xi^{\xi}_{\forall\phi(F\to G)}bc))$.

Case 6 $a \equiv \Theta_{\forall\phi B}$. An arbitrary $(\forall\phi B,\xi)$-supplementation of this has the shape $f(\mathbf{U}^{\forall\phi(B^{\xi}(\phi)\to B^{\xi}(C)}(\Theta^{\xi}_{\forall\phi B}b))$, where $b \in \mathrm{Comp}_{\xi,B}$ and therefore has a normal form b'. Moreover, b' is also the normal form of $\mathbf{U}^{\forall\phi(B^{\xi}(\phi)\to B^{\xi}(C)}(\Theta^{\xi}_{\forall\phi B}b)$, since, as we saw in Sect. 4, the outermost quantifier in $\forall\phi B$ does not actually bind anything. So $f(b')$ is a well-formed term. Moreover, by the corollary to Proposition 7.12 together with Proposition 7.11, $f(b')$ is in $\xi(\beta)$ for some β. By clause (iii) of Definition 7.2, so is $f(\mathbf{U}^{\forall\phi(B^{\xi}(\phi)\to B^{\xi}(C)}(\Theta^{\xi}_{\forall\phi B}b))$.

Proposition 7.15 *For every term a of $CL_2(A)$ and every ξ, a^ξ is in $Comp_{\xi,A}$.*

Proof by induction on the construction of terms. If a is formed by application, then the conclusion follows from the I.H. by clause (ii) of Definition 7.6. Suppose $a \equiv \Pi_{\alpha,\phi}b$ and let b have type $B(\alpha)$. Then an arbitrary $(\forall\phi B(\phi), \xi)$-supplementation of a has the shape $f(\mathbf{U}^{\forall\phi(B^\xi(\phi)\to B^\xi(C)}(\Pi_{\alpha,\phi}b)^\xi)$. By the I.H.,

$$b^\xi \in Comp_{\xi,B(\alpha)} \text{ for every } \xi. \tag{5}$$

Let ξ now be fixed and let D be such that $D^\xi \equiv C$. Then, instantiating (5), $b^{\xi'} \in \mathrm{Comp}_{\xi',B(\alpha)}$ for any valuation ξ' that takes α to $\mathrm{Comp}_{\xi,D}$ and otherwise agrees with ξ. So by Proposition 7.13 $b^{\xi'} \in \mathrm{Comp}_{\xi,B(D)}$ and therefore $f(b^{\xi'})$ is in $\xi(\beta)$ for some β. $b^{\xi'}$ is the same term as $(b(^{\alpha}_{D}))^\xi$ and $f(\mathbf{U}^{\forall\phi(B^\xi(\phi)\to B^\xi(C)}(\Pi_{\alpha,\phi}b)^\xi)$ reduces to $f((b'(^{\alpha}_{D}))^\xi)$, where b' is the normal form of b. However, by clauses (iii) and (iv) of Definition 7.2, any term that reduces to $f((b'(^{\alpha}_{D}))^\xi)$ is in $\xi(\beta)$ for some β.

Theorem 2 *Every CL_2-term has a normal form.*

Proof by Propositions 7.10 and 7.15.

8 The importance of Theorem 1 and the possible utility of CL$_2$-terms

Theorems 1 and 2 together entail that every Λ_2-term has a normal form. It may be objected that this has already been repeatedly proven ([1,6,7,12,19,24]). The answer to this objection is that, for some purposes, a proof of Theorem 2 like the one given here will not be optimal. There are advantages in being able to produce normalizability proofs that can be formalized in primitive recursive arithmetic (PRA) augmented only by transfinite induction (applied to predicates in the language of PRA) over some initial segment of the recursive ordinal numbers. The predicates "$\in \mathrm{Comp}_{\xi,A}$", used in Sect. 7, are of course very far from being formalizable in the language of PRA.

In the terminology used by proof theorists, a normalizability proof of the kind envisaged here is called "ordinally informative".[7]

Definition 8.1 The n-*th polymorphic Church numeral* is the Λ_2-term $\Lambda_{\alpha,\phi}\lambda y^{\alpha\to\alpha}\lambda x^\alpha \underbrace{(y^{\alpha\to\alpha}(y^{\alpha\to\alpha}...(y^{\alpha\to\alpha}}_{n} x^\alpha)...))$, abbreviated henceforth to c_n.

Definition 8.2 For any m-place function f of natural numbers, f is *represented* by the Λ_2-term M iff, for every m-tuple $c_{n_1}, c_{n_2}, \ldots, c_{n_m}$ of polymorphic Church numerals[8]

$$Mc_{n_1}c_{n_2}...c_{n_m} \quad =_{\Lambda_2} \quad c_{f(n_1,n_2,...,n_m)}$$

[7] For a rough definition of "ordinally informative", see [14]. For the present purpose, "terms" has to be read in place of "proofs"

[8] This definition is taken from p. 283 of [19].

Definition 8.3 A class Φ of numerical functions is *represented* by a class Σ of terms of Λ_2 iff each function in Φ is represented by some term in Σ.

Up till now, ordinally informative proofs of normalizability have been developed much less for Λ_2-terms than for the terms of Gödel's **T**,[9] which can however be embedded in Λ_2 ([19], p. 283). A number of writers have exploited this work to prove theorems along the following lines: if all the terms in Σ can be shown to have normal forms by means of transfinite induction up to some ordinal number α and Σ represents the class Φ, then each function in Φ can be defined by ordinal recursion up to some ordinal below α. See for example [11,17,22].

Ordinal recursion has been felt to be a natural and illuminating way of defining subrecursive classes ([15], chapter 3) and convenient for proving the identity of classes that have been defined in different ways (*ibid.*, chapter 4). So ordinally informative proofs of normalizability of classes of Λ_2-terms can be expected to show not only that the terms have normal forms but also that the functions they represent can be characterized in a way that is intelligible to people who are not acquainted with Λ_2 and arguably more illuminating even to people who are.

There is another advantage which at least some ordinally informative proofs of normalizability have over proofs of the kind presented in Sect. 7. If it should prove possible to define a function $\mathcal{F}$ which takes terms in class Σ to recursive ordinal numbers in such a way that:

(6) If M reduces to N, then $\mathcal{F}(M) > \mathcal{F}(N)$.

then this very easily yields not only a proof that the terms of Σ all have normal forms but also a definition by ordinal recursion of a function which outputs an upper bound to the number of reduction steps needed to reach the normal form.[10]

Theorem 1 above entails that an ordinally informative proof of the normalizability of some class Σ of CL_2-terms is also an ordinally informative proof of the normalizability of the set of Λ_2-terms that have H-images in Σ. It remains to discuss why there might be any advantages in trying to achieve such a result for CL_2 first rather than tackling Λ_2 directly. To some extent, the answer has to be that the best way to tackle Λ_2 is not yet known but it makes sense for researchers to arm themselves in advance with any tool which might potentially be useful. But one point can be made already. The most obvious form for the definition of an ordinal assignment function $\mathcal{F}$ to take is for it to follow the construction of terms: to each operation in the construction of terms (term-application, λ-abstraction, type-abstraction and type-application) shall correspond a function taking the $\mathcal{F}$-images of immediate subterms of M to $\mathcal{F}(M)$ itself. As CL_2-terms are built up by fewer and simpler operations than Λ_2-terms—only term-application and a restricted form of type-abstraction denoted by $\Pi_{\alpha,\phi}$—it is a reasonable expectation that the invention of a suitable ordinal assignment function for CL_2-terms will be a simpler matter than for Λ_2-terms.

Not all of the ordinal assignment functions in the proof-theoretic literature do follow this obvious pattern, but some do, notably in [10,16] (pp. 105–112). The normalizability proof in [10] is for **T** formulated using $\Lambda_{\rightarrow}$-terms, in [16] it is for **T** formulated

[9] For a history of **T**, see Sect. 10.5 of [19] or Sect. 1.6 of [23]; Sect. 2.2.35 of [23] surveys proofs of normalizability published before 1973. Important contributions made since that date are [11,22,25].

[10] See for example Sect. 6 of [20].

using $\mathrm{CL}_{\rightarrow}$-terms. The assignment in [10] is more complicated in two respects: firstly what is assigned to each term in [16] is a finite sequence of ordinal numbers whereas in [10] it is a finite sequence of *functions* of ordinal numbers; in the second place in [10] the finite sequence assigned to a term is calculated by means of two operations, corresponding to term-application and λ-application respectively, whereas in [16] only one operation is needed, corresponding to application alone.

9 Irreducible terms; connections to second-order logic

Any deduction formed using rules $(a)-(e)$ of Table 5.1 can be transformed into a proof in the natural deduction calculus $\rightarrow\forall^2\mathbf{Nip}^2$ (described in [24], p. 345), in the following way: every statement $a \in \mathrm{Ir}_2(A)$ should be transformed into a sequent that has, on the left, the types of the variables in a and, on the right, just the formula A. This transformation will be called the *statements-to-sequents transformation*. Under it, the rules $(a)-(e)$ of Table 5.1 become rules of $\rightarrow\forall^2\mathbf{Nip}^2$. It will be permitted, however, that in translating from a statement to a sequent, formulae may be inserted on the left of the sequent which are not necessarily types of free variables in the subject of the statement.

Provided that the given deduction is constructed in accordance with the restrictions stated after Table 5.1, it is possible to see that its image under the statements-to-sequents transformation will actually be a *normal* proof in $\rightarrow\forall^2\mathbf{Nip}^2$.

If the term a in a particular application of rule (b) of Table 5.1 satisfies the suffixed condition (3), then $[x^A]a$ has a combinator in head position. If a satisfies condition (4) but not (3), then either $[x^A]a$ is a **U**-term or $[x^A]a$ has in head position a combinator or a Θ-term. There are no other possibilities. It might seem at first that $[x^A]a$ could have a Ξ-term in head position, but this is so only if a has the shape $\Xi_{\forall\phi(F\rightarrow G)}bx^{\forall\phi F}$ or $\Xi_{\forall\phi(F\rightarrow G)}x^{\forall\phi(F\rightarrow G)}$. However, by Definition 4.3, no term of any such shape is a generalization and it was stipulated that rules (f) and (g) have not been used in the deduction.

On the other hand, the conditions on rule (d) in Table 5.1 were that, in the left premiss $a \in Ir(A\rightarrow B)$ of an application of this rule, a had to have either a variable or a **U**-term in head position, but could not be itself a **U**-term. This means that a conclusion of rule (b) cannot be a left premiss of an application of rule (d). In the terminology used to talk about natural deduction calculi this is expressed by saying that the main premiss of an application of →-elimination may not be the conclusion of an application of →-introduction.

The argument of the last two paragraphs may be summarized by saying that, in a deduction using the rules of Table 5.1, or in its image under the statements-to-sequents transformation, the conclusion of an introduction inference may not be the sole premiss of a ∀E-inference nor the main premiss of a →E-inference. When the only connectives present are ∀ and → this entails ([24], p. 179) that the deduction is normal.

This equivalence can now be used to show that the set of sequents provable in $\rightarrow\forall^2\mathbf{Nip}^2$ is just the image under the statements-to-sequents transformation of the set of statements provable in CL_2.

Proposition 9.1 *If a sequent is provable in $\rightarrow\forall^2$**Nip**^{2}, it is the image under the statements-to-sequents transformation of a statement provable in CL$_2$.*

Proof every initial sequent is the image of a statement of the shape $x^A \in \mathrm{CL}_2(A)$. The rules $\forall$I (second-order) and $\rightarrow$I are admissible by Propositions 4.5 and 4.7 respectively. The rule $\rightarrow$E is a primitive rule of CL$_2$. As for $\forall$E: if $a \in \mathrm{CL}_2(\forall\phi A(\phi))$ is deducible in CL$_2$, then, in virtue of the initial statements and rules of CL$_2$, so is $\mathbf{U}^{\forall\phi A(\phi)\rightarrow A(C)}a \in A(C)$; while, if the statements-to-sequents transformation takes $a \in \mathrm{CL}_2(\forall\phi A(\phi))$ to $\Gamma \vdash \forall\phi A(\phi)$, it also takes $\mathbf{U}^{\forall\phi A(\phi)\rightarrow A(C)}a \in A(C)$ to $\Gamma \vdash A(C)$.

Proposition 9.2 *If a sequent $\Gamma \vdash A$ is the image under the statements-to-sequents transformation of a statement $a \in CL_2(A)$, provable in CL$_2$, then $\Gamma \vdash A$ is also provable in $\rightarrow\forall^2$**Nip**^{2}.*

Proof by Theorem 2, a has a normal form a' which also has type A. By Proposition 5.10 and the first corollary following Definition 5.7, there is then a deduction $\mathfrak{D}$ of $a' \in \mathrm{Ir}(A)$ in Ir. Taking advantage, if necessary, of the right to add extra formulae on the left (see above), $\Gamma \vdash A$ is also the image under the statement-to-sequents transformation of $a' \in \mathrm{Ir}(A)$. Suppose, first, that the deduction $\mathfrak{D}$ uses only rules $(a)-(e)$. Then, applying the statements-to-sequents transformation to the whole of $\mathfrak{D}$, we obtain, as the foregoing discussion showed, a proof in $\rightarrow\forall^2$**Nip**^{2} (and, indeed, a normal proof). Even if $\mathfrak{D}$ makes use of rules (f) and (g), the images of these rules under the statement-to-sequents transformation are rules admissible in $\rightarrow\forall^2$**Nip**^{2}.

The proof of Proposition 9.1 introduces in effect a mapping which takes derivations in $\rightarrow\forall^2$**Nip**^{2} to CL$_2$-terms. The image of a derivation $\mathfrak{D}$ under this mapping will be called *the term which encodes* $\mathfrak{D}$. It is used in the next theorem.

Theorem 3 *If a class Σ of CL_2-terms contains normalizable terms only and the derivation $\mathfrak{D}$ is encoded by a term in Σ, then the conclusion of $\mathfrak{D}$ is also the conclusion of some normal derivation.*

Proof if a CL_2-term c is formed by the method described in the proof of Proposition 9.1, it will contain no subterms which begin with a Ξ-term but which are not generalizations. Moreover, a term of the shape $\Xi_{\forall\phi(F\rightarrow G)}ab$ where a and b are generalizations does not lose that shape under reduction of terms, so if c has the normal form c' and a deduction of $c' \in \mathrm{Ir}(A)$ is formed, the deduction will make no use of rules (f) and (g) and therefore, by the discussion at the beginning of this section, the image of c' under the statements-to-sequents transformation will be a normal derivation in $\rightarrow\forall^2$**Nip**^{2}. This suffices to establish the theorem.

It is hoped (see the last section) that future proofs of normalizability will correlate, for some proper subsets Σ of the set of all CL$_2$-terms, each such Σ with an initial case of transfinite induction which suffices to prove the normalizability of all the terms in Σ. Just as Theorem 1 showed that, for each such Σ, a proof of the normalizability of all terms in Σ entails the normalizability of a certain related class of Λ_2-terms, so theorem 3 shows that the normalizability of terms in Σ entails the normalizability of derivations encoded by terms in Σ. It is shown, further, in [24] (especially pp. 346 and 351f.), that normalizability of $\rightarrow\forall^2$**Nip**^{2} entails normalizability of intuitionist second-order logic generally.

Although the discussion at the beginning of this section showed that applying the statements-to-sequents transformation to a deduction in Ir using rules (a)–(e) only yields a proof in $\to\forall^2$**Nip**^{2}, this does not apply to deductions in CL$_2$, which in shape resemble axiomatic deductions of second order logic rather than natural deduction ones.

Section 5.3 of [19] indicates how CL$_\to$-terms can be read as encoding axiomatic deductions in intuitionist implicational logic. A popular way (perhaps first used in [13], pp. 80–85) of axiomatizing quantificational logic (first or second order), in such a way that no specifically quantificational rules of inference are needed but only *modus ponens*, is to stipulate that every generalization of an axiom is also an axiom and to adopt, as specifically quantificational axioms, all formulae of the following shapes:

$$\begin{aligned}&\forall\phi A(\phi) \to A(C)\\ &A \to \forall\phi A \quad \text{where } \phi \text{ does not occur free in } A\\ &\forall\phi(F \to G) \to \forall\phi F \to \forall\phi G \quad \text{where } F, G \text{ are as in Table 4.1}\end{aligned}$$

But these three schemata are just the schemata exemplified by the types of **U**-terms, Θ-terms and Ξ-terms respectively (Table 4.1). Closure of the set of axioms under generalization is obtained by using the function $a \mapsto \Pi_{\alpha,\phi}a$ (Table 4.2).

10 Further possible applications

The Church-style and de Bruijn-style systems mentioned in Sect. 2 above are often (e.g. [1], Sect. 3) contrasted with *Curry-style* systems, also called *theories of type-assignment*. Section 4.1 of [1] defines a theory of this kind, which assigns polymorphic types to Λ-terms. This can easily be turned into a theory which assigns polymorphic types to CL-terms (i.e., *untyped* combinatory terms), simply by reading the variables which, in the formulation of the theory, range over Λ-terms as ranging over their H-transforms instead (here "H-transform" refers not to the H-transformation of Definition 6.1 but to a simpler H-transformation which takes Λ into CL: see p. 212 of [4]). The theory of type-assignment formed in this way shall be called *Curry-style CL$_2$*.

It is obvious that from a term a in CL$_2(A)$, a unique deduction in CL$_2$ of $a \in$ CL$_2(A)$ can be reconstructed. Thus a can be read as encoding a deduction of the fact that a itself has the type it does have. Somewhat similarly, a CL$_2$-term of type A can be read as encoding a deduction in Curry-style CL$_2$ that a certain CL-term can be assigned the type A. How it can be so read should be clear from a reading of Sect. 5A of [8], where $\Lambda_\to$-terms are used to encode deductions in a Curry-style theory that assigns simple types to those Λ-terms that can be formed from $\Lambda_\to$-terms by erasing their types.

Normalizability moreover is carried over from $\Lambda_\to$-terms to their erasures ([8], Sect. 5C). Something similar has been done with CL$_2$-terms (see [21] for a first attempt), but the notion of erasure that needs to be defined is more complicated and things do not work as neatly as they do in the case of $\Lambda_\to$-terms. In any event, it has to be admitted that not many people will necessarily wish to master the machinery of the present paper for the sake of this application alone.

11 Conclusion

An ordinally informative proof of the normalizability of a class of CL_2-terms will yield an ordinally informative proof of the normalizability both of a certain class of Λ_2-terms and of a certain fragment of second-order intuitionist logic. There is some reason, moreover, to expect that such proofs will be easier to achieve by concentrating on CL_2-terms than by tackling either Λ_2-terms or second-order logic directly. It is hoped that future work will vindicate this claim.

References

1. Barendregt, H.P.: Lambda Calculi with types. In: Abramsky, S., et al. (eds.) Handbook of Logic in Computer Science, vol. 2, pp. 117–309. Oxford University Press, Oxford (1992)
2. Barendregt, H.P., Dekkers, W., Statman, R.: Lambda Calculus with Types. Cambridge University Press, Cambridge (2013)
3. Bunder, M.W.: Some improvements to Turner's algorithm for bracket abstraction. J. Symb. Logic **55**, 656–669 (1990)
4. Curry, H.B., Feys, R.: Combinatory Logic, vol. I. North-Holland, Amsterdam (1958)
5. Curry, H.B., Hindley, J.R., Seldin, J.P.: Combinatory Logic, vol. II. North-Holland, Amsterdam (1972)
6. Girard, J-Y.: Une Extension de l'Interpretation de Gödel a l'Analyse. In: J. E. Fenstad (ed.): Proceedings of the Second Scandinavian Logic Symposium, , pp. 63–92. North-Holland, Amsterdam and London (1971)
7. Girard, J.-Y., et al.: Proofs and Types. CUP, Cambridge (1989)
8. Hindley, J.R.: Basic Simple Type Theory. CUP, Cambridge (1997)
9. Hindley, J.R., Seldin, J.P.: Lambda Calculus and Combinators: An Introduction. CUP, Cambridge (2008)
10. Howard, W.A.: Assignment of ordinals to terms for primitive recursive functionals of finite type. In: Kino, A., Myhill, J., Vesley, R.E. (eds.) Intuitionism and Proof Theory, pp. 443–458. North-Holland, Amsterdam (1970)
11. Howard, W.A.: Ordinal analysis of terms of finite type. J. Symb. Logic **45**, 493–504 (1980)
12. Prawitz, D.: Ideas and results of proof theory. In: Fenstad, J.E. (ed.) Proceedings of the Second Scandinavian Logic Symposium, pp. 235–307. North-Holland, Amsterdam (1971)
13. Quine, W.V.O.: Mathematical Logic, 2nd edn. Harvard University Press, Cambridge (1951)
14. Rathjen, M.: Recent advances in ordinal analysis: Π^1_2 and related systems. Bull. Symb. Logic **1**, 468–485 (1995)
15. Rose, H.E.: Subrecursion: Functions and Hierarchies. Clarendon, Oxford (1984)
16. Schütte, K.: Proof Theory. Springer, Berlin (1977). (**Translated by J. N. Crossley**)
17. Schwichtenberg, H.: Elimination of higher type levels from definitions of primitive recursive functionals by means of transfinite recursion. In: Rose, H.E., Shepherdson, J.C. (eds.) Logic Colloquium -73, pp. 279–303. North-Holland, Amsterdam (1975)
18. Seldin, J.P.: The search for a reduction in combinatory logic equivalent to $\lambda\beta$-reduction. Theoret. Comput. Sci. **412**, 4905–4918 (2011)
19. Sørensen, M.H.B., Urzyczyn, P.: Lectures on the Curry–Howard Isomorphism. Elsevier, Amsterdam (2006)
20. Stirton, W.R.: How to assign ordinal numbers to combinatory terms with polymorphic types. Arch. Math. Logic **51**, 475–501 (2012)
21. Stirton, W.R.: A decidable theory of type-assignment. Arch. Math. Logic **52**, 631–658 (2013)
22. Terlouw, J.: Reduction of higher type levels by means of an ordinal analysis of finite terms. Ann. Pure Appl. Logic **28**, 73–102 (1985)
23. Troelstra, A.S. (ed.): Metamathematical Investigation of Intuitionistic Arithmetic and Analysis (Lecture Notes in Mathematics 344). Springer, Berlin (1973)
24. Troelstra, A.S., Schwichtenberg, H.: Basic Proof Theory, 2nd edn. Cambridge University Press, Cambridge (2000)

25. Weiermann, A.: How is it that infinitary methods can be applied to finitary mathematics? Goedel's T: a case study. J. Symb. Logic **63**, 1348–1370 (1998)