The Completeness of Heyting First-order Logic

W. W. Tait

Abstract

Restricted to first-order formulas, the rules of inference in the Curry-Howard type theory are equivalent to those of first-order predicate logic as formalized by Heyting, with one exception: ∃-elimination in the Curry-Howard theory, where $\exists x : A.F(x)$ is understood as disjoint union, are the projections, and these do not preserve first-orderedness. This note shows, however, that the Curry-Howard theory is conservative over Heyting's system.

The meaning of the title becomes clear if we take the intuitionistic meaning of the logical constants to be given in Curry-Howard type theory, CH. This is the basic theory of types built up by means of ∀ and ∃, as outlined in [Howard, 1980] and presented in some detail by Martin-Löf, for example in [Martin-Löf, 1998], as part of his intuitionistic theory of types. The quantifier ∀ denotes in this system the operation of taking cartesian products

$$\forall x\!:\!A.F(x) = \Pi_{xA}F(x)$$

and ∃ denotes the operation of taking disjoint unions

$$\exists x\!:\!A.F(x) = \Sigma_{xA}F(x).$$

First-order formulas can be regarded as formulas of CH and the intuitionistic rules of inference are all derivable, as rules of term formation, in CH.[1]

[1] The term of type A corresponding to a proof in HL of A will in general have to contain variables representing the first-order function symbols in HL as well as a variable of type D, representing the domain of individuals. The latter variable represents the implicit assumption of first-order logic that the domain of individuals is non-empty.

But the converse is not true: there are, for example, closed normal terms of CH whose types are first-order formulas, but which are not first-order proofs. This has to do with the rule of existential quantification elimination in CH, which expresses the intended meaning of the existential quantifier as disjoint union:

$$p\!:\!\exists x\!:\!D.F(x) \Rightarrow p1\!:\!D,\ p2\!:\!F(p1)$$

where $t\!:\!A$ means that t is an object of type A or, equivalently, a proof of A. Even when D is the type of individuals and $\exists x : D.F(x)$ is a first-order formula, providing that x occurs in $F(x)$, $F(p1)$ is not a first-order formula. In intuitionistic logic as formalized by Heyting, which we denote by HL, this rule of existential quantifier elimination can be replaced by

$$p\!:\!\exists x\!:\!D.F(x), q\!:\!\forall x\!:\!D[F(x) \to C] \Rightarrow [[p, q]]\!:\!C.$$

The aim of this paper is to show that, in spite of the stronger form of existential quantifier elimination in CH, every first-order formula deducible in CH is also deducible in HL.[2]

In order to treat the full system of HL, we include disjunction among the operations of CH. Disjunction is also a disjoint summing operation, but it cannot be formalized as such in CH without the introduction of further proposition and propositional function constants. For example, with the introduction of the type $\mathbf{N}$ of natural numbers it can be defined by

$$A \vee B := \exists x\!:\!\mathbf{N}[(x = 0 \to A) \wedge (x \neq 0 \to B)]$$

In fact, it would suffice to introduce just the two-element type, together with a propositional function defined on it whose values are the absurd proposition $\mathbf{0}$ for one element and $\neg\mathbf{0}$ for the other; but we will be content here to simply stick to the usual, unsatisfactory, formalization of disjunction.

1 The System CH

The formulas, terms and the relation of definitional equivalence $\equiv$ must be simultaneously defined.

[2]This result was conjectured in [Tait, 1994, p. 59], where it was also remarked that I remembered once having proved it. For the case of CH and HL without disjunction, the present proof (which may or may not be the basis for that memory) has existed, in a slightly defective form, since at least 1997. The defect was corrected for me by Frank Pfenning.

The relation constants in CH are either propositional constants or are constants for propositional functions on A, for some *sentence*, i.e. formula without free variables, A. No further kind of relation symbols are needed. For example, one might want also to consider the case of symbols A, F and G, where A denotes a type, F a propositional function on A and G a propositional function of two variables, such that Gab is defined when $a\!:\!A$ and $b\!:\!Fa$. But we can simply regard G as a propositional function on $\exists x : A.Fx$, since (a, b) is of this type. The atomic formulas are the propositional constants and the expressions Et, where E denotes a propositional function on A and t is a term of type A. Among the propositional constants will be $\mathbf{0}$, denoting the absurd proposition (i.e. the initial or null type). Since we are concerned with pure logic, there will be no constants for objects of atomic type. (Individual and function constants in the language of first-order logic can be represented by free variables.)

For each formula A we assume given an infinite supply of symbols, called the *free variables* of *sign* A. We denote these variables by v_A, or when the sign is given by the context or is irrelevant, by v. For distinct formulas A and B, the free variables of sign A are distinct from those of sign B. Note that A is *not* a part of the syntax of v_A, which is an atomic symbol. (It would suffice to introduce variables v_A only for *normal* formulas A. See below for the definition of normality.) If $F(v_A)$ and A are formulas not containing x, then $\forall x\!:\!A.F(x)$ and $\exists x\!:\!A.F(x)$ are formulas. x is called a *bound variable* and has no type. Its meaning is given only in the context of the quantifier $\forall x : A$ or $\exists x : A$ or the lambda operator $\lambda x : A$ (see below) which binds it. (So, when v_A actually occurs in $F(v_A)$, $F(x)$ is not a formula. Similarly, if v_A actually occurs in the term $t(v_A)$, then $t(x)$ is not a term.) If A and B are formulas, then so is $A \vee B$.

Abbreviations:

$$A \rightarrow B := \forall x\!:\!A.B$$

$$A \wedge B := \exists x\!:\!A.B$$

where x does not occur in B.

$$\neg A := A \rightarrow \mathbf{0}$$

The terms and their types are as follows:

Every variable of sign A is a term of type A.

For **0** we have only the rule of

0-elimination

$$t : \mathbf{0} \Rightarrow N(A, t) : A.$$

$\vee$-introduction

$$t : A \quad \Rightarrow \quad [t, B] : A \vee B,\ [B, t] : B \vee A.$$

$\vee$-elimination If v_A does not occur in C or in the sign of any free variable in $s(v_A)$ and v_B does not occur in C or in the sign of any free variable in $t(v_B)$, then

$$r : A \vee B,\ s(v_A) : C,\ t(v_B) : C \ \Rightarrow\ [r, \lambda x : A.s(x), \lambda x : B.t(x)] : C.$$

$\forall$-introduction If v_A does not occur in the sign of any free variable occurring in $t(v_A)$ or $F(v_A)$, then

$$t(v_A) : F(v_A) \Rightarrow \lambda x : At(x)\ :\ \forall x : A.F(x).$$

$\forall$-elimination

$$f\ :\ \forall x : AF(x), s : A\ \Rightarrow\ fs : F(s).$$

$\exists$-introduction

$$s : A, t : F(s)\ \Rightarrow\ (s, t) :\ \exists x : A.F(x).$$

and, to repeat,

$\exists$-elimination

$$p\ :\ \exists x : AF(x)\ \Rightarrow\ (p1) : A, (p2) : F(p1).$$

$\equiv$, defined below, is an equivalence relation on the set of terms and formulas.

$$t\!:\!A, A \equiv B \Rightarrow t\!:\!B$$

expresses that the types of a term form an equivalence class. It also turns out that

$$s \equiv t, s\!:\!A \;\Rightarrow\; t\!:\!A$$

Terms and formulas that can be obtained from one another by renaming bound variables will be identified.

The *components* of a term are as follows: a variable has no components; t is a component of

$$N(A,t), (s,t), (t,s), [t,A], [t,A], [r,s,t], [r,t,s], [t,r,s], ts, st, t1, t2$$

and $t(v_A)$ is a component of $\lambda x\!:\!A.t(x)$, where v_A is the least variable of sign A not occurring in $\lambda x\!:\!A.t(x)$. (We assume that there is a fixed enumeration of variables.) The *components* of a formula are as follows: atomic formulas have no components, the components of $A \vee B$ are A and B, and the components of $\forall x\!:\!A.F(x)$ and $\exists x\!:\!A.F(x)$ are A and $F(v_A)$, where v_A is the least variable of sign A not occurring in $\forall x\!:\!A.F(x)$. The *subterms* of a term t form the least set M containing t and containing all components of terms in M. A subterm s of a term t is not in general a part of t. Rather, it will correspond to parts $s'(x_1, \ldots, x_n)$ ($n \geq 0$) of t, called *instances* of s in t, where $s = s(v_1, \ldots, v_n)$. It is understood that the x_i are distinct and the v_i are distinct and the v_i do not occur in t. When we wish to speak of an occurrence of a subterm of t as a part of t, we shall refer to it as a *subterm part* of t.

The Conversion Rules define the left-hand term by means of the right-hand term.

$$N(\mathbf{0}, s) \ CONV \ s$$

$$N(\forall x\!:\!A.F(x), s)t \ CONV \ N(F(t), s)$$

$$N(\exists x\!:\!A.f(x), s)1 \ CONV \ N(A, s)$$

$$N(\exists x\!:\!A.F(x), s)2 \ CONV \ N(F(N(A, s)), s)$$

$$[[r, B], \lambda x\!:\!A.s(x), \lambda x\!:\!B.t(x)] \ CONV \ s(r)$$

$$[[A, r], \lambda x\!:\!A.s(x), \lambda x\!:\!B.t(x)] \ CONV \ t(r)$$

$$[r, \lambda x\!:\!A.s(x), \lambda x\!:\!B.t(x)]u \ CONV [r, \lambda x\!:\!A.(s(x)u), \lambda x\!:\!B.(t(x)u)]$$

where u is a term or is 1 or 2.

$$[\lambda x\!:\!A.t(x)]s\ CONV\ t(s)$$

$$(s,t)1\ CONV\ s \qquad (s,t)2\ CONV\ t$$

We say that s *initially* converts to t

$$s\ i-CONV\ t$$

iff $s = qr_1 \cdots r_n$, $t = pr_1 \cdots r_n$, $0\ eqn$ and $q\ CONV\ p$.

A term is called a *matrix* iff it is not a variable and its only proper subterms are variables. Every term other than a variable is of the form $t(s_1, \ldots, s_n)$ where $t(v_1, \ldots, v_n)$ is a matrix, unique except for the naming of the free variables.

Definition The relations $s\ n-RED\ t$ (s *n-reduces to* t) $s\ RED\ t$ (s *reduces to* t) between terms is defined for $n > 0$ by

- $s\ 1-RED\ s$.
- If $s\ i-CONV\ t$, then $s\ 1-RED\ t$.
- If $t(v_1, \ldots v_n)$ is a matrix and $r_i\ 1-RED\ s_i\ (i = 1, \ldots n), then$ $t(r_1, \ldots r_n)\ 1-RED\ t(s_1, \ldots, s_n)$.
- If $r1-REDs$ and $sn-REDt$, then $rn+1-REDt$.
- $s\ n-RED\ t$ for some n iff $s\ RED\ t$.

$$s > t$$

will mean that $p\ RED\ t$ for some p obtained by replacing an occurrence of a subterm q of s by r, where $q\ CONV\ r$. (So $s > t$ implies $s\ RED\ t$, but because of the first clause in the definition of RED, the converse does not always hold.

The two fundamental theorems concerning CH are:

CHURCH-ROSSER THEOREM

$$r \ RED \ s, t \ \Rightarrow \ \text{there is a } u \text{ such that } s, t \ RED \ u$$

WELLFOUNDEDNESS THEOREM *Every sequence*

$$t_0 \ > \ t_1 \ > \ t_2 \ > \cdots$$

is finite.

COROLLARY Every term reduces to a unique normal term (i.e. a term which cannot be strictly reduced)—its *normal form.*

Now we complete our definition of the terms and their types by defining

$$s \equiv t \iff \text{there is a } u \text{ such that } s, t \ RED \ u$$

$$A \equiv B \iff \text{they are built up in the same way from pairwise} \equiv \text{terms.}$$

$\equiv$ is obviously decidable equivalence relation between both terms and formulas. *From now on, we will speak of* the type *of a term t of CH, meaning the unique normal type of t.*

The *simple* terms of CH are variables or of one of the forms

$$N(A, s), [t, A], [A, t], [r, s, t], \lambda x\!:\!A.s(x), (s, t)$$

where, in the third case, s and t are of course lambda-terms. Every term t can be written uniquely as

$$t_0 t_1 \cdots t_n = (\cdots (t_0 t_1) \cdots t_n)$$

where $n \geq 0, t_0$ is simple and each of $t_1, \ldots t_n$ is either $1, 2$ or a term. Note that when t is normal and $n > 0$, then t_0 must be a variable. (It is to ensure this that the conversion rules for $N(A, s)t$ and $[r, s, t]u$ are needed, as Frank Pfenning pointed out to me.)

2 Proof of the Church-Rosser Theorem

In what follows, we just write $N(t)$ for $N(A, t)$, $[t]$ or, when necessary, $[t]_1$ for $[t, B]$, $[t]_2$ for $[B, t]$, and $\lambda xt(x)$ for $\lambda x\!:\!At(x)$. The Church-Rosser Theorem, in any case, has nothing to do with type structure. It applies to the set of 'terms' built up from type-less variables by means of the operations

$$N(t),\ [t],\ [r,s,t],\ (s,t),\ st,\ t1,\ t2,\ \lambda xt(x)$$

subject to the conversion rules stated above. (This is in contrast to the Wellfoundedness Theorem, which has everything to do with type structure—as witnessed by the example $[\lambda x(xx)]\lambda x(xx)$.)

Lemma 1 *If s' is obtained from s by simultaneously replacing terms t_i by terms t'_i, where $t_i\ 1-RED\ t'_i$, then $s\ 1-RED\ s'$.*

The proof is by induction on s. If s is one of the terms t_i, then there is nothing to prove. So let $s = S(p_1,\ldots,p_n)$, where $S(v_1,\ldots,v_n)$ is a matrix. Then $s' = S(p'_1,\ldots,p'_n)$ is the result of substituting the t'_i's for the t_i's in the p_j's. By the induction hypothesis, $p_j\ 1-RED\ p'_j$ for each j, from which the result follows by definition. □

Lemma 2 *Let $r\ CONV\ s$ and $r\ 1-RED\ t$. Then there is a term u such that $s\ 1-RE\ u$ and $t\ 1-RED\ u$.*

The proof is by induction on r.

If $s = t$ or $r = t$, set $u = s$ and if $r = s$, set $u = t$. So we assume that r, s and t are distinct. This implies that r does not convert to t. It follows that t is obtained by replacing subterms of r by terms to which they 1-reduce.

Case 1. $r = N(p)q]$, $s = N(p)$ and $t = N(p')q'$. Then set $u = N(p')$.

Case 2. $r = [[r_0]_e, \lambda xr_1(x), \lambda xr_2(x)]$. We can assume $e = 1$, the other case being exactly the same. So $s = r_1(r_0)$ and $t = [[r'_0]_e, \lambda xr'_1(x), \lambda xr'_2(x)]$, where the r'_j's are obtained by replacing subterms of the r_j's by terms to which they 1-reduce. Set $u = r'_1(r'_0)$.

Case 3. $r = [r_0, \lambda xr_1(x), \lambda xr_2(x)]p$, $s = [r_0, \lambda x(r_1(x)p), \lambda x(r_2(x)p)]$ and $t = [r'_0, \lambda xr'_1(x), \lambda xr'_2(x)]p'$. Set $u = [r'_0, \lambda x(r'_1(x)p'), \lambda x(r'_2(x)p')]$

Case 4. $r = (\lambda xp(x))q$, $s = p(q)$ and $t = \lambda xp'(x)q'$, where $\lambda p(x)\ 1-RED\ \lambda xp'(x)$ and $q\ 1-RED\ q'$. Set $u = p'(q')$.

Case 5 . $r = (s,p)1$ and $t = (s',p')1$. Set $u = s'$.

Case 6. $r = (p,s)2$ and $t = (p',s')2$. Set $u = s'$ □

Lemma 3 *Let $r\ 1-RED\ s$ and $r\ 1-RED\ t$. Then there is a term u such that $s\ 1-RED\ u$ and $t\ 1-RED\ u$.*

$$\begin{array}{ccc} r & \xrightarrow{1-RED} & s \\ {\scriptstyle 1-RED}\downarrow & & \downarrow{\scriptstyle 1-RED} \\ t & \xrightarrow[1-RED]{} & u \end{array}$$

Proof by induction on r. Let $r = r_0 \cdots r_n$, where r_0 is simple. The 1-reduction of r to s is called *internal* iff $s = s_0 \cdots s_n$, where s_0 is simple and $r_i\ 1-RED\ s_i$ for each i. If the 1-reduction is not internal, we call it *external.*

Case 1. The 1-reductions of r to s and t are both internal. Then $s = s_0 \cdots s_n$ and $t = t_0 \cdots t_n$, where r_i 1-reduces to both s_i and t_i for each i. Assume $n > 0$. Then the induction hypothesis applies to yield, for each i, a u_i such that both s_i and t_i 1-reduce to u_i. Then s and t 1-reduce to $u = u_0 \cdots u_n$.

If $n = 0$, then $r = r_0$ is a simple term $N(r'), [r'], [r', r'', r'''], (r', r'')$ or $\lambda x r'(x)$ and the result follows easily by the induction hypothesis.

Case 2. Both 1-reductions are external. First, suppose $r_0 \cdots r_k$ converts to some r', $s = r' s_{k+1} \cdots s_n$ and $t = r' t_{k+1} \cdots t_n$. By the induction hypothesis, there is a u_i for each $i > k$ such that s_i and t_i 1-reduce to u_i. So s and t 1-reduce to $u = r' u_{k+1} \cdots u_n$.

But there is another possibility, namely that

$$r_0 = [[p_0]_e, \lambda x p_1(x), \lambda x p_2(x)]$$

$s = p_e(p_0) s_1 \cdots s_n$, where the r_i's 1-reduce to s_i's and

$$t = [[p_0]_e, \lambda x (p_1(x) r_1), \lambda x (p_2(x) r_1)] t_2 \cdots t_n$$

where the r_i's 1-reduce to the t_i's fpr $i > 1$. By the induction hypothesis, s_i and t_i 1-reduce to some u_i for $i > 1$. Let $u' = p_e(p_0) s_1$. Then s and t 1-reduce to $u' u_2 \cdots u_n$.

Case 3. One of the 1-reductions, say the 1-reduction of r to s, is external and the other is internal. So $r_0 \cdots r_k\ CONV\ r'$, $s = r' s_{k+1} \cdots s_n$, where r_i1-reduces to s_i for $i > k$, and $t = t_0 \cdots t_n$, where r_i 1-reduces to t_i for each i. By Lemma 2, r' and $t_0 \cdots t_k$ 1-reduce to some u'. By the induction hypothesis, s_i and t_i 1-reduce to some u_i for $i > k$. So s and t 1-reduce to $u' u_{k+1} \cdots u_n$. □

Lemma 4 *If $r\ m-RED\ s$ and $r\ n-RED\ t$ then there is a u such that $s\ n-RED\ u$ and $t\ m-RED\ u$.*

Proof:

$$\begin{array}{ccccccc}
r = r_{11} & \longrightarrow & r_{12} & \longrightarrow & \cdots & \longrightarrow & r_{1m} = s \\
\downarrow & & \downarrow & & \downarrow & & \downarrow \\
r_{21} & \longrightarrow & r_{22} & \longrightarrow & \cdots & \longrightarrow & r_{2m} \\
\downarrow & & \downarrow & & \downarrow & & \downarrow \\
\vdots & \longrightarrow & \vdots & \longrightarrow & \vdots & \longrightarrow & \vdots \\
\downarrow & & \downarrow & & \downarrow & & \downarrow \\
t = r_{n1} & \longrightarrow & r_{n2} & \longrightarrow & \ldots & \longrightarrow & r_{nm} = u
\end{array}$$

where each arrow denotes a 1-reduction.

This completes the proof of the Church-Rosser Theorem.[3]

3 Proof of the Wellfoundedness Theorem

We want to prove that every term t is wellfounded, i.e. that every sequence $t = p_0 \;>\; p_1 \;>\; p_2 \;> \cdots$ is finite.

Definition of Computability. We define the notion of a *computable term* of type A by induction on A.

- A term of atomic type is computable iff it is well-founded.
- A term of type $A \vee B$ is computable iff it is well-founded and
  - if it reduces to a term $[s, B]$, then s is a computable term of type A.
  - If it reduces to a term of type $[A, s]$, then s is a computable term of type B.
- A term f of type $\forall x\!:\!A.F(x)$ is computable iff ft is a computable term of type $F(t)$ for every computable term t of type A.

[3]This method of proof was first presented for the type-free theory of combinators in a seminar at Stanford in the Spring of 1965. It was noted at the time that a suitable definition of 1-reduction could also be given for the type-free lambda calculus.

- A term p of type $\exists x\!:\!A.F(x)$ is computable iff $p1$ is a computable term of type A and $p2$ is a computable term of type $F(p1)$.

We define the notion of a *c-extension* of a term t of type A by induction on A. If A is atomic or a disjunction $B \vee C$, then t is the only c-extension of t. If $A = \forall x\!:\!BF(x)$, then the c-extensions of t are precisely the c-extensions of ts for all computable terms $s\!:\!B$. If $A = \exists x\!:\!BF(x)$, then the c-extensions of t are precisely the c-extensions of $t1$ and $t2$. So the c-extensions of t are all the terms of atomic or disjunctive type of the form $ts_1 \cdots s_n$, where the s_i are either 1, 2 or are computable. Clearly, t is computable iff all of its c-extensions are computable.

The *rank* $|A|$ of a formula A is defined by

$$|A| = Max\{|B| + 1 \mid B \text{ is a component of } A\}$$

Lemma 5 *For each type A*

a) Every variable v of type A is computable.

b) Every computable term of type A is well-founded.

c) If s is a computable term of type A and $sREDt$, then t is computable.

The proof is by induction on $|A|$.

a) We need only show that all c-extensions $s = vs_1 \cdots s_n$ of v are well-founded. (When s is of type $B \vee C$, it cannot reduce to a term of the form $[r, C]$ or $[B, r]$.) The s_i which are terms are of types of rank $< |A|$ and so, by the induction hypothesis, are well-founded. It immediately follows that s is well-founded.

b) Let s be computable and consider the c-extension $t = ss_1 \cdots s_n$ of s, where the terms among the s_i are variables. t is computable and hence well-founded. So s must be well-founded.

c) We need first to show that all c-extensions $tt_1 \cdots t_n$ of t are well-founded. But if such a term were not well-founded, then neither would the c-extension $st_1 \cdots t_n$ of s be, contradicting the computability of s. Secondly, if $tt_1 \cdots t_n$ reduces to $[r, C]$ or $[B, r]$, then so does $st_1 \cdots t_n$ and so r is computable. □

It follows from b) that in order to prove that every term is well-founded, it suffices to prove that every term is computable. By an *c-instance* of a term t we will mean any result of replacing each variable throughout t by a computable term of the same type. So by a) above, t is a c-instance of itself.

COMPUTABILITY THEOREM Every c-instance of a term t is computable.

The proof is by induction on t. If t is a variable, this is immediate. If $t = rs$, where r and s are terms, then by the induction hypothesis, the c-instances of r and s are computable. So any c-instance of t is computable by definition. Similarly, if $t = s1$ or $t = s2$, then s is computable by the induction hypothesis, and so t is computable by definition. So we need only consider terms t of the form $N(s, A), [q, r, s], [s, B], [B, s], (r, s)$ or $\lambda x : As(x)$.

Let t' be a c-instance of t. Consider any c-extension $p = t't_1 \cdots t_n$ of t'. We need first to show that p is well-founded. Consider a sequence

$$(1) \qquad p = p_0 > p_1 > \cdots$$

It will be convenient to assume that, for each k, p_{k+1} is obtained by converting exactly one occurrence of a subterm of p_k.

We have first to show that (1) is finite. Recall that if the step from p_k to p_{k+1} is obtained by converting an initial part of p_k, i.e.

$$p_k = rs_1 \cdots s_m, r\ CONV\ r', \quad p_{k+1} = r's_1 \cdots s_m,$$

we call it an *external reduction*; otherwise, an *internal reduction*.

Secondly, we must show that, if p reduces to a term of the form $[q, B]$ or $[B, q]$, then q is computable. But, if we have already shown that the sequence (1) is finite, we may assume that its last term p_m is normal. But then it follows from the Church-Rosser Theorem that p_m is of the form $[q, B]$ or $[B, q]$ (since the only reductions of terms of this form are internal). We need therefore only show that, for each sequence 1), if the last term is of one of these forms, then it is computable.

Given any sequence $t_0, t_1, \ldots$ of terms, by its $>$-*subsequence*, we mean the maximum subsequence $t_{i_0} > t_{i_1} > \ldots$.

Let $t = N(A, s)$ and $t' = N(A, s')$. Then p_m is of the form

$$N(A, q^m)q^m_{i+1} \cdots q^m_n,$$

where i is the number of external conversions in the sequence (1) up to p_m, $st_1 \cdots t_i$ RED q^m and t_j RED q_j^m for $j = i+1, \ldots, n$. There can be at most n such external conversions in this sequence. Let $p'_m = N(A, s't_1 \cdots t_n)$. Do for sufficiently large m $p'_m > p'_{m+1}$. So if the sequence of p_m's were infinite, so would be the $>$-subsequence of the p'_m's. But $s't_1 \cdots t_n$ is computable. Hence (1) is finite. Moreover, the last member of the sequence cannot be of the form $[q, B]$ or $[B, q]$.

Let $t = [q, r, s]$. Then $t' = [q', r', s']$. As in the previous case, we can assume that $n = 0$, so that $p_0 = t'$ and $p_m = [q_m, r_m, s_m]$. There are two cases: first, there are no external conversions in the sequence of p_m's. Then, since q', r', s' are all computable, the $>$-subsequences of the sequences of the q_m's, r_m's and s_m's must all be finite; and hence the sequence of p_m's is finite. Moreover, its last member cannot be of the form $[q, B]$ or $[B, q]$. The other case is that there are no external conversions up to $p_m = [q_m, r_m, s_m]$, but $q_m = [p, B]$ or $[B, p]$ and $p_{m+1} = r_m p$ or $s_m p$. But, r_m, s_m and p are computable. Since q' is of disjunctive type and reduces to $[p, B]$ or $[B, p]$, then p is by definition, computable. Hence, the term $p_{m+1} = r_m p$ or $s_m p$ is computable and the sequence of p_i's is finite. Let $t = [s, B]$ and $t' = [s', B]$. Then $n = 0$ and $p_m = [s_m, B]$ for all m. By the induction hypothesis, s' is computable. So the sequence of s_m's is finite; i.e. the sequence of p_m's is finite and its member are computable. Similarly for $t = [B, s]$.

Let $t = (r, s)$ and $t' = (r', s')$. If there are no external reductions in the sequence of p_m's, then $p_m = (r_m, s_m)t_1^m \cdots t_n^m$ and, since r', s' and the terms among the t_i's are computable, the $>$-subsequences of the r_m's and s_m's are finite; hence so is the sequence of p_m's. Otherwise some p_m is of the form $r_m t_2 \cdots t_n$ or $s_m t_2 \cdots t_n$. But, by the induction hypothesis, r' and s' are computable. So, for large enough m, p_m is computable. If there is an external reduction in the sequence, let the first one be at p_{k+1}, so that $p_k = (r_k, s_k)$ and $p_{k+1} = r_k$ or s_k. But r' RED r_k and s' RED s_k and, by the induction hypothesis, r' and s' are computable.

Finally, let $t = \lambda x s(x)$ (dropping the type for the sake of brevity). Then $t' = \lambda x s'(x)$. Again, if there are no external conversions in the sequence of p_m's, then $p_m = (\lambda x s_m(x))t_1^m \cdots t_n^m$ and the finiteness of the sequence (1) follows from the computability of $s'(v), t_1, \ldots, t_n$. So let there be an external conversion, first occurring at $p_{m+1} = s_m(t_1^m)t_2^m \cdots t_n^m$. But, by the induction hypothesis, since t_1 is a computable term, $s'(t_1)$ is computable, being a c-instance of $s(v)$. Hence, the subsequence of p_j's for $j > m$ is a sequence of computable terms. □

So, again, by part (b) of Lemma 5, the Wellfoundedness Theorem is proved.[4]

4 Embedding HL in CH

Let D be a fixed type symbol. (Think of D as the domain of individual over which the first-order variables range.) We write $D \wedge D \wedge D$ for $D \wedge (D \wedge D)$, $D \wedge D \wedge D \wedge D$ for $D \wedge (D \wedge (D \wedge D))$, etc., and $(r, s, t))$ for $(r, (s, t))$, (r, s, t, u) for $(r, (s, (t, u)))$, etc. First-order terms are terms of type D built up in the usual way from variables of type D and n-ary *first-order function* variables, i.e. variables of type $D_n = [D \rightarrow (D \rightarrow (\cdots (D \rightarrow D) \cdots))]$ ($n+1$ occurrences of D). The atomic first-order formulas are either $\mathbf{0}$ or are of the form $R(t_1, \ldots, t_n)$, where R is a symbol for a propositional function defined on $D \wedge \cdots \wedge D$ (n D's) and the t_k are first-order terms. The first-order quantifiers are introduced by

$$\forall x F(x) := \forall x\!:\!D.F(x)$$

$$\exists x F(x) := \exists x\!:\!D.F(x)$$

So, the first-order formulas are built up from atomic first-order formulas using $\rightarrow$, $\wedge$, $\vee$ and the first-order quantifiers. These are the formulas of HL. The rules of inference of HL are the usual ones of intuitionistic predicate logic, including the rule of inference from $\mathbf{0}$ to any formula A.

Let

$$\Delta := \{D_n \mid n < \omega\}$$

First-order terms t are built up from variables with signs in Δ; i.e. they are deductions of D from Δ in CH. Namely, if t is a variable, it is of type D.

[4] This method of proof, using 'computability' predicates, goes back to [Tait, 1963; Tait, 1967], where it is applied to prove the existence of normal forms for closed terms in the case of a single atomic type $\mathbf{N}$ with no existential quantification and no disjunction and with the following introduction and elimination rules for $\mathbf{N}$. The introduction rules are $0\!:\!\mathbf{N}$ and $t\!:\!\mathbf{N} \Rightarrow S(t)\!:\!\mathbf{N}$, and the elimination rule is

$$r\!:\!A, s\!:\!\mathbf{N} \rightarrow (A \rightarrow A),\ t\!:\!\mathbf{N} \quad \Rightarrow \quad R(r, s, t)\!:\!A$$

where R is defined by the conversion rules

$$R(r, s, 0)\ CONV\ r \qquad\qquad R(r, s, S(t))\ CONV\ stR(r, s, t).$$

Let $t = vt_1 \cdots t_n$, where v is of type D_n and the t_i are first-order terms. Then each t_i is a deduction of D from Δ and so, by n applications of $\forall$-elimination, so is t. So all valid formulas of HL are valid in CH under the assumption of Δ. The proof-theoretic version of this is

EMBEDDING THEOREM

$$\Gamma \vdash_{HL} A \;\Rightarrow\; \Gamma \cup \Delta \vdash_{CH} A$$

The proof is routine,[5] by induction on the length of the given deduction of A in HL, except for when the last inference of the deduction is an instance of $\forall$-elimination or $\exists$-elimination.

For the first of these, suppose that $A = F(t)$ is obtained in HL from $\forall x F(x) = \forall x\!:\!D.F(x)$, where t is a first-order term. As we have just noted, t is a deduction of D from Δ. By the induction hypothesis, there is a deduction s in CH of $\forall x F(x)$ from $\Gamma \cup \Delta$. Hence st is the required deduction of A.

For the second, let A be obtained in HL from $\exists x F(x)$ and $\forall x(F(x) \to A)$, where v is not in A. By the induction hypothesis, there are deductions p and q in CH of these premises, respectively, from $\Gamma \cup \Delta$. So $p2$ is a deduction of $F(p1)$ and $q(p1)$ is a deduction of $F(p1) \to A$. Thus, $[[p,q]] = q(p1)(p2)$ is the required deduction of A. $\Box$

So, in this sense, CH is an extension of HL. We now need to show that it is conservative (in this sense) over HL, that is that $\Gamma \cup \Delta \vdash_{CH} A$ implies $\Gamma \vdash_{HL} A$.

5 Quasi First-Order Terms and Formulas

Call a term p of CH *existential* if its type is of the form $\exists y G(y)$. Call a term or formula X of CH *quasi-first-order (qfo)* iff there is a first-order term or first-order formula X', respectively, called an *original* of X, and there is a list $p_1, \ldots, p_n$ $(n \geq 0)$ of distinct normal existential terms, called an *e-list* for X, such that

- the type of p_i is obtained by substituting $p_1 1, \ldots, p_{i-1} 1$ for distinct free variables of type D in a first-order formula.

[5] It is essentially proved in [Martin-Löf, 1998].

- X is obtained by substituting $p_1 1, \ldots, p_n 1$ for distinct free variables in X'.

(Note that a qfo term or formula which is not first-order will have infinitely many originals, obtained from one another by renaming free variables.) If X has the e-list $p_1, \ldots, p_n$, then the type of p_k for $k = 1, \ldots, n$ is qfo and has the e-list $p_1, \ldots, p_{k-1}$. If $\forall x F(x)$ and $t\!:\!D$ are qfo, then so is $F(t)$. Just concatenate the e-list of p's for $\forall x F(x)$ with the e-list for t. Obviously, if $A \to B$, $A \wedge B$ or $A \vee B$ is qfo, then so are A and B.

Lemma 6 *Let $t = vt_1 \cdots t_n$ be a term of type $A \neq D$, where v is a variable of qfo sign and the terms among the t_i are all qfo. Then A is qfo.*

The proof is by induction on n. If $n = 0$, the result is trivial. Assume $n > 0$ and that $vt_0 \cdots t_{n-1}$ is of qfo type A'.

If A' is $B \to C$, then $A = C$ and the result is immediate.

If $A' = B \wedge C$, then A is B or C and again the result is immediate.

If $A' = \forall x.F(x)$, then $A = F(t_n)$, where t_n is of type D and so, by assumption, qfo. Hence, as we noted above, A is qfo.

If $A' = \exists x.F(x)$, then $t_n = 1$ or 2. If it is were 1, then A would be D. So it must be 2. $A = F(t_0 1 \cdots t_{n-1} 1)$. If $p_1, \ldots, p_k$ is a concatenation of an e-list for A' and one for t, then $p_1, \ldots, p_k, t_0 \cdots t_{n-1}$ is an e-list for A. □

Lemma 7 *Let t be a normal deduction in CH of D from $\Gamma \cup \Delta$, where the formulas in Γ are qfo, and assume that t has no subterm parts of the form $N(D, s)$ or $[r, s, u]$ of type D. Then t is qfo.*

The proof is by induction on t. Let $t = t_0 t_1 \cdots t_n$, where t_0 is simple. If $n = 0$, then, since $t = N(D, u)$ and $[r, s, u]$ of type D are excluded, t must be a variable of type D and so is a first-order term. So let $n > 0$. t_0 must be either a first-order function variable or a variable of first-order sign. In the first case, $t_1, \ldots, t_n$ are all normal terms of type D and so, by the induction hypothesis, satisfy the condition. Let $p_{i1}, \ldots, p_{ik_i}$ be an e-list for t_i. Then $p_{11}, \ldots, p_{nk_n}$ is an e-list for t. Since the originals of all the t_i are first-order terms, so are the originals of t.

So assume that t_0 is of first-order type. Then $p = t_0 \cdots t_{n-1}$ must be of type of the form $\exists y G(y)$ and $t_n = 1$. If $\exists y.G(y)$ is qfo with e-list $p_1, \ldots, p_k$, then we are done: $p_1, \ldots, p_k, p$ is an e-list for t and the originals of t are variables v_D. But, since the terms among $t_0, \ldots, t_{n-1}$ are qfo by the induction hypothesis, $\exists y G(y)$ is qfo by Lemma 6. □

Lemma 8 *If t is a normal deduction of A from $\Gamma \cup \Delta$, where Γ consists of first-order formulas, then every existential term occurring in A is a subterm of t.*

(The restriction of Γ to first-order formulas is necessary in virtue of our decision to count a variable v_B as an atomic symbol.) For, if $B \in \Gamma$ were to contain an existential term, then v_B would be a counterexample to the lemma. The proof is routine by induction on t. There are two cases in which an existential term p may be introduced into a formula: One is $A = F(s)$, where s contains p, $t = fs$, and f is of type $\forall x : B.F(x)$. The other is $A = F(p1)$, where $t = p2$ and $p : \exists x F(x)$. □

Not all formulas derivable from first-order formulas and Δ are qfo. For example, let v be of sign $\forall x \exists y F(x, y)$. Then $\lambda x : Dvx2$ is a deduction of $\forall x F(x, vx1)$ from $\forall x \exists y F(x, y)$. Call a term of CH *pure* iff the type of each of its subterms is either in Δ or is qfo. We need to prove that, if there is a deduction t of a qfo formula A from $\Gamma \cup \Delta$, where Γ consists of qfo formulas, then there is a pure such deduction. The difficulty in proving this straightforwardly by induction on t arises from the possible presence in t of disjunction-elimination: for example, t might have a subterm of the form $[q, r, s]$, where the type of q is a non-qfo formula $F \vee G$. We shall call such a term $[q, r, s]$ a *critical term.* When $[q, r, s]$ is a critical term and neither q, r, nor s have critical subterms, we call $[q, r, s]$ a *minimal* critical subterm.

Lemma 9 *Let t be a normal deduction in CH from $\Gamma \cup \Delta$ of A, where Γ consists of qfo formulas, and let $t = [q, r, s]$ be a minimal critical term. Then there is a normal deduction $t' = [q', r', s']$ of A from $\Gamma \cup \Delta$ which contains no critical subterms (and so is not itself critical).*

The proof is by induction of the length $|t|$ of t as a string of symbols. Let q be of type $F \vee G$. q cannot be of the form $vt_1 \cdots t_n$, since otherwise $F \vee G$ would be qfo by Lemma 6. It cannot be of the form $[p, G]$ or $[F, p]$, since otherwise t would not be normal. If it is of the form $N(F \vee G, p)$, then $t' = N(A, p)$ suffices. The only other possibility is that $q = [q^*, r^*, s^*]$, where the type $B \vee C$ of q^* must then be qfo. Let

$$r^* = \lambda x : B r_0(x) \qquad\qquad s^* = \lambda x : C s_0(x).$$

$r_0(v_B)$ is normal. If it is of the form $[p(v_B), G]$, then let $t'_1(v_B)$ be the normal form of $rp(v_B)$. If $r_0(v_B) = [F, p(v_B)]$, then let $t'_1(v_B)$ be the normal form of $sp(v_B)$ In either case, $t'_1(v_B)$ is a deduction from $\Gamma \cup \Delta \cup \{B\}$ of A without a critical subterm. If r_0 is of neither of these forms, then $t_1(v_b) = [r_0(v_B), r, s]$ is a normal critical term of type A with $|t_1(v_B)| < |t|$. So, by the induction hypothesis, there is again a deduction $t'_1(v_B)$ of A from $\Gamma \cup \Delta \cup \{B\}$ without a critical term. Similarly, replacing B and $r_0(v_B)$ by C and $s_0(v_C)$, respectively, we obtain a deduction $t'_2(v_C)$ of A from $\Gamma \cup \Delta \cup \{C\}$. Hence $[q', \lambda x\!:\!Bt'_1(x), \lambda x\!:\!Ct'_2(x)]$ is a deduction of A from $\Gamma \cup \Delta$ without critical subterms. □

If a term t' is obtained by replacing all occurrences of $[q, r, s]$ in the normal term t by occurrences of a normal term $[q', r', s']$ of the same type, then t' is normal. For the replacement creates no convertible subterms.

Lemma 10 *If $\Gamma \cup \Delta \vdash_{CH} A$, where Γ consists of qfo formulas and either A is qfo or else $A = D$, then there is a pure deduction t' in CH of A from $\Gamma \cup \Delta$.*

Let t be a normal deduction in CH of A from $\Gamma \cup \Delta$. The proof is by induction on the number $k(t)$ of critical subterms of t.

Let $k(t) = 0$. We proceed by induction on t.

We may suppose that t does not contain subterm parts of the form $N(D, s)$, since otherwise the induction hypothesis yields a pure deduction s' of $\mathbf{0}$ and we may take $t' = N(A, s')$. So we have only the following cases:

If $t = vt_1 \cdots t_n)$, where the sign of v is in $\Gamma \cup \Delta$, the terms among the t_i are of type D. So, by Lemma 7, each term t_i is qfo. Hence, by Lemma 6, the type A_j of $vt_1 \cdots t_j$ is qfo for $j = 1 \ldots n$. Hence, $t' = t$ is pure.

If $t = [s, B]$, where $A = B \vee C$ or $C \vee B$, then $t' = [s', B]$

Let $t = (r, s)$. If $A = B \wedge C$, set $t' = (r', s')$. If $A = \exists x, F(x)$, then r is of type D and so, as we have already seen, $r' = r$. Hence s' is a pure deduction of $F(r')$ and so, again, we may set $t' = (r', s')$.

If $t = \lambda x\!:\!Bs(x)$, apply the induction hypothesis to the deduction $s(v_B)$ from $\Gamma \cup \Delta \cup \{B\}$ to obtain $s'(v_B)$. Then $t' = \lambda x\!:\!Bs'(x)$.

Now assume $k(t) > 0$. There is a minimal critical subterm $[q, r, s]$ of t. By Lemma 9, we may replace each occurrence of $[q, r, s]$ in t by the corresponding occurrence of a normal term $p = [q', r', s']$ of the same type with $k(p) = 0$, obtaining a normal deduction t^* of A from $\Gamma \cup \Delta$ with $k(t^*) < k(t)$. By the remark above, t^* is normal, and so by the induction hypothesis we obtain $t' = (t^*)'$. □

6 Elimination of Left Projections

Let t be a normal pure deduction from $\Gamma \cup \Delta$. A list $p_1, \ldots, p_n$ of distinct existential terms is called a *base* for t iff it is a shortest e-list for the type of t.

In the following lemma, let $\Gamma(v)$ and $A(v)$ be obtained by everywhere replacing $p1$ in (the formulas in) $\Gamma(p1)$ and in $A(p1)$, respectively, by $v = v_D$.

Lemma 11 *Let t be a pure normal deduction of $A(p1)$ from $\Gamma(p1) \cup \Delta$, where the formulas in $\Gamma(p1)$ are qfo and $A(p1)$ is either qfo or $= D$. Let $p, p_1, \ldots, p_n$ is a base for t, where $p_1, \ldots, p_n$ are pure normal terms. Let $\exists x G(x)$ be the type of p. Then there is a pure deduction t' of $A(v)$ from $\Gamma(v) \cup \Delta \cup \{G(v)\}$ with base $p'_1, \ldots, p'_n$.*

If t is a variable, then $A(p1)$ is in $\Gamma(p1)$. So let t' be a new variable u of sign $A(v)$.

If $t = N(A(p1), s)$, then $t' = N(A(v), s')$.

If $t = [s, B(p1)]$, then $t' = [s', B(v)]$.

If $t = [B(p1), s]$, then $t' = [B(v), s']$.

If $t = [r, s, u]$, then $t' = [r', s', u']$.

If $t = \lambda x{:}C(p1).s(x)$, then $t' = \lambda x{:}C(v).s(x)'$.

If $t = (r, s)$, then $t' = (r', s')$.

If $t = fs$, where s is a term, then $t' = f's'$.

If $t = qe$, where e is 1 or 2 and $q \neq p$, then $t' = q'e$.

If $t = p1$, then $t' = v$.

If t is $p2$, then $A(p1) \equiv G(p1)$ and t' is a new variable of sign $G(v)$. □

Lemma 12 *Let $\Gamma \cup \{A\}$ be a set of qfo formulas. If t is a pure normal deduction in CH of A from $\Gamma \cup \Delta$ with a null base and if Γ' is the set of all originals of each formula in Γ and A' is an original of A, then $\Gamma' \vdash_{HL} A'$.*

The proof is by induction on t.

If $t = v_A$, then A' is in Γ'.

If $t = N(B, s)$, Then by the induction hypothesis, $\Gamma' \vdash_{HL} \mathbf{0}$ and so $\Gamma' \vdash_{HL} A'$.

If $t = [s, C]$, then $s : B$, where $A = B \vee C$. By the induction hypothesis, $\Gamma' \vdash_{HL} C'$ and so $\Gamma' \vdash_{HL} A'$. Similarly for $t = [B, s]$.

If $t = [r, s, u]$, where $r : B \vee C$, then by the induction hypothesis, $\Gamma' \vdash_{HL} B' \vee C'$, $\Gamma' \vdash_{HL} B' \to A'$ and $\Gamma' \vdash_{HL} C' \to A'$. So $\Gamma' \vdash_{HL} A'$.

If $t = \lambda x : B.s(x)$, then either $A = B \to C$ or else $B = D$ and $A = \forall x F(x)$. In the first case, B is qfo and so, by the induction hypothesis, we have $\Gamma' \cup \{B'\} \vdash_{HL} C'$ from which follows $\Gamma' \vdash_{HL} A'$. In the second case, $\Gamma' \vdash_{HL} F(v)'$ and hence $\Gamma' \vdash_{HL} A'$.

If $t = (r, s)$, then either $A = B \wedge C$, $r : B$ and $s : C$ or else $A = \exists x F(x)$, x occurs in $F(x)$, $r : D$ and $s : F(r)$. In the first case, $\Gamma' \vdash_{HL} B'$ and $\Gamma' \vdash_{HL} C'$ and so $\Gamma' \vdash_{HL} A'$. In the second case, $\Gamma' \vdash_{HL} F(r)'$. Let $\exists x F'(x)$ be an original of $\exists x F(x)$. r is qfo by Lemma 7. Hence, $F(r)' = F'(r')$, where r' is an original for r. So $A' = \exists x(F(x)')$ follows by $\exists$-elimination in HL from $F(r)'$. So $\Gamma' \vdash_{HL} A'$.

Let $t = fs$, where s is a term. Then either $f : B \to A$ and $s : B$, in which case the induction hypothesis yields $\Gamma' \vdash_{HL} B \to A$ and $\Gamma' \vdash_{HL} B$, and so $\Gamma' \vdash_{HL} A$, or else $f : \forall x F(x)$, $s : D$ and $A = F(r)$. In the latter case, r is a normal term of type D and index 0. Hence, it is a term of HL and we have $\Gamma' \vdash_{HL} \forall x F(x)$ and so $\Gamma' \vdash_{HL} A$.

If $t = pe$, where e is 1 or 2. Let p be of existential type $\exists x F(x)$. $e = 2$, since otherwise $A = D$. So x is not in $F(x)$, since otherwise $A = F(p1)$ would contain an occurrence of $p1$ and so t would not have a null base. By the induction hypothesis, $\Gamma' \vdash_{HL} \exists x A'$ from which $\Gamma' \vdash_{HL} A'$ follows. Now suppose that p is of type $A \wedge B$ (if $e = 1$) or $B \wedge A$ (if $e = 2$). In any case, then, $\Gamma' \vdash_{HL} A' \wedge B'$ and so $\Gamma' \vdash_{HL} A'$. $\square$

7 Completeness of HL

Now we are in position to prove that CH is conservative over HL.

CONSERVATION THEOREM *Let* Γ *consist of first-order formulas, let* A *be* qfo *with original* A' *and let*

$$\Gamma \cup \Delta \vdash_{CH} A.$$

Then

$$\Gamma \vdash_{HL} A'.$$

Let t be a normal pure deduction of A from $\Gamma \cup \Delta$. The proof is by induction on the length of the bases of t. Since Γ consists of first-order formulas, it follows from Lemma 8 that every element of a base of t is a subterm of t and hence is pure and normal.

If t has a null base, then A is first-order, i.e. $A = A'$. So the result follows by Lemma 12.

So let t have base $p, p_1, \ldots, p_n$ and let $A = A(p1)$ and let p be of type $\exists x G(x)$. Since $p, p_1, \ldots, p_n$ is a base, $\exists x G(x)$ contains no existential terms, and so is first-order. Since p is pure and normal and has a null base,

$$\Gamma \vdash_{HL} \exists x G(x)$$

by Lemma 12. $p_1, \ldots, p_n$ are pure and normal, and so by Lemma 11, there is a pure normal deduction t' of $A(v_D)$ from $\Gamma \cup \Delta \cup \{G(v_D)\}$ with base $p'_1, \ldots, p'_n$. By the induction hypothesis, we therefore have

$$\Gamma \cup \{G(v)\} \vdash_{HL} A.$$

So, by $\exists$-elimination in HL, $\Gamma \vdash_{HL} A$. $\square$

References

Hindley, J. and Sheldon, J. (eds) [1980]. *To H.B. Curry: Essays on Combinatorial Logic, Lambda Calculus and Formalism*, London: Academic Press.

Howard, W. [1980]. The formula-as-types notion of construction, *in* J. Hindley and J. Seldin (eds), *[Hindley and Sheldon, 1980]*, pp. 479–490.

Martin-Löf, P. [1998]. An intuitionistic theory of types, *In [Sambin and Smith, 1998, 221-244]* .

Sambin, G. and Smith, J. (eds) [1998]. *Twenty-Five Years of Constructive Type Theory*, Oxford: Oxford University Press.

Tait, W. [1963]. A second order theory of functionals of higher type, with two appendices. Appendix A: Intensional functionals. Appendix B. An interpretation of functionals by convertible terms., *Stanford Seminar Report* pp. 171–206. A published version is [Tait, 1967].

Tait, W. [1967]. Intensional interpretations of functionals of finite type i, *Journal of Symbolic Logic* **32**: 198–212.

Tait, W. [1994]. The law of excluded middle and the axiom of choice, *in* A. George (ed.), *Mathematics and Mind*, Oxford: Oxford University Press, pp. 45–70.