

AN ERROR IN A COMPUTER VERIFIED PROOF OF INCOMPLETENESS BY JOHN HARRISON

James R Meyer

<http://www.jamesrmeyer.com>

v1 03 November 2011

Abstract

This paper examines a proof of incompleteness by John Harrison, who claims that his proof has been verified by computer. This paper demonstrates that the proof has a fundamental error of logic which renders the proof invalid.

1 Introduction

This paper demonstrates a fundamental error in a proof of incompleteness by John Harrison, the description of which is found in a book by Harrison [1] first published in 2005. This paper does not discuss the implications of the failure of the computer verification software to detect the error in the proof; that will be published elsewhere.

2 Harrison's proof

Harrison's proof was created using the HOL Light system [2], which uses what is called HOL logic within the OCaml computer language. The principle behind the HOL logic that it is based on a theory of types.

Before we deal with the nub of Harrison's proof, we shall summarize some of the notation and definitions that he uses:

\bar{n} represents what Harrison calls the *numeral* function^a, and which gives the *numeral* of n . It is a function whose variable n has the domain of natural numbers, and the function gives an expression of the formal system that is an expression for a natural number (that is, in the form $0, S0, SS0, \dots$).

$\lceil \phi \rceil$ represents the Gödel numbering function^b, and gives the Gödel number of ϕ . The domain of the variable ϕ is expressions of the formal system, and the function gives a unique numerical value for every expression of the formal system. This is a function of the meta language.

$\overline{\lceil \phi \rceil}$ is a function that is the result of the combination of the two functions above, where the domain of the variable ϕ is expressions of the formal system, and the function gives an expression of the formal system that is an expression for a natural number. This also is a function of the meta language.

$\lceil \bar{n} \rceil$ is also a function that is the result of the combination of the two functions \bar{n} and $\lceil \cdot \rceil$, but in a different order, and is the Gödel number of the *numeral* of n . Harrison also calls this function *gnumeral*(n). Again, this is a function of the meta language.

$QDIAG_x(n, y)$ is defined as a relation in \mathbb{N} (i.e., as a number-theoretic relation of natural numbers), where:^c

$$QDIAG_x(n, y) \Leftrightarrow$$

$$\exists k. GNUMERAL(n, k) \wedge y = \langle C_x, \langle 5, \langle \langle 1, \langle \langle 0, C_x \rangle, k \rangle \rangle, n \rangle \rangle \rangle$$

where $GNUMERAL(n, k)$ and $\langle C_x, \langle 5, \langle \langle 1, \langle \langle 0, C_x \rangle, k \rangle \rangle, n \rangle \rangle \rangle$ are defined as a number-theoretic relation and function in \mathbb{N} respectively.

$qdiag_x(p)$ is defined by Harrison^c as $qdiag_x(p) \equiv \exists x. x = \overline{\lceil p \rceil} \wedge p$. This is a function in the meta language.

$diag_x(p)$ is defined^c as $diag_x(p) \equiv \text{subst}(x \mapsto \overline{\lceil p \rceil})$, which represents the formula that results when the free variable x of the formula p is substituted by the value $\overline{\lceil p \rceil}$. This also is a function in the meta language.

$=_{\text{def}}$ according to Harrison, when a definition is made by $=_{\text{def}}$, that includes the claim that the corresponding equivalence applies in the system of natural numbers \mathbb{N} , where the term $=_{\text{def}}$ is replaced by \Leftrightarrow .

Harrison's computer code expressions will be printed here in monospace typewriter font, for example: `(arith_gnumeral s t)`.

In his description of his proof, Harrison points out that 'various sets of natural numbers, and relations over natural numbers, are definable in arithmetic'. His proof of incompleteness is dependent on his Lemma 7.3, which is:

Let $P[x]$ be a formula in the language of arithmetic with just the free variable x , and define $\phi \equiv_{\text{def}} qdiag_x(\exists y. QDIAG_x(x, y) \wedge P[y])$. Then $\phi \Leftrightarrow P[\overline{\lceil \phi \rceil}]$ holds in \mathbb{N} .

^aSee Harrison's Section 7.2 *Tarski's theorem on the undefinability of truth*

^bSee Harrison's Section 7.2, para *Arithmetization of syntax*

^cSee Harrison's section 7.2, para *The fixpoint lemma*. Note that, for convenience, C_x is used here to represent the number corresponding to the formal system variable x ; in Harrison's text it is represented by `number(x)`.

This lemma is dependent on a prior assertion made by Harrison that $QDIAG_x(\overline{p}, y)$ and $y = \overline{qdiag_x(p)}$ are equivalent within \mathbb{N} . Harrison justifies that assertion by the following:^d

Since

$$QDIAG_x(\overline{p}, y) \Leftrightarrow \exists k, GNUMERAL(\overline{p}, k) \wedge y = \langle 10, \langle C_x, \langle 5, \langle \langle 1, \langle \langle 0, C_x \rangle, k \rangle \rangle, \overline{p} \rangle \rangle \rangle \rangle$$

and, by the definition of $GNUMERAL(n, k)$ and $gnumeral(n)$,

if $k = gnumeral(\overline{p})$, that is, if $k = \overline{\overline{p}}$, then

$GNUMERAL(\overline{p}, \overline{\overline{p}})$ holds, so that:

$$y = \langle 10, \langle C_x, \langle 5, \langle \langle 1, \langle \langle 0, C_x \rangle, k \rangle \rangle, \overline{p} \rangle \rangle \rangle \rangle \quad (2.1)$$

$$= \langle 10, \langle C_x, \langle 5, \langle \langle 1, \langle \langle 0, C_x \rangle, \overline{\overline{p}} \rangle \rangle, \overline{p} \rangle \rangle \rangle \rangle \quad (2.2)$$

$$= \langle 10, \langle C_x, \langle 5, \langle \langle 1, \langle \overline{x}, \overline{\overline{p}} \rangle \rangle, \overline{p} \rangle \rangle \rangle \rangle \quad (2.3)$$

$$= \langle 10, \langle C_x, \langle 5, \langle \overline{x} = \overline{\overline{p}}, \overline{p} \rangle \rangle \rangle \rangle \quad (2.4)$$

$$= \langle 10, \langle C_x, \overline{x} = \overline{\overline{p}} \wedge \overline{p} \rangle \rangle \quad (2.5)$$

$$= \overline{\exists x. x = \overline{\overline{p}} \wedge \overline{p}} \quad (2.6)$$

$$= \overline{qdiag_x(p)} \quad (2.7)$$

In the above Harrison assumes the equivalence:

$$GNUMERAL(n, k) \Leftrightarrow k = gnumeral(n).$$

that is, Harrison asserts that when $GNUMERAL(n, k)$ holds, then $k = gnumeral(n)$ holds, and vice-versa.

But that equivalence cannot apply within \mathbb{N} . The reason for this is elementary; a purely arithmetical system \mathbb{N} does not have variables with the domain of formulas of the formal system. But the function $gnumeral(n)$ is defined as $gnumeral(n) = \overline{\overline{n}}$, and since the function $\overline{\quad}$ is a function of the meta language (where the free variable of $\overline{\quad}$ has the domain of formulas of the formal system), then the function $gnumeral(n)$ (which is the function $\overline{\overline{\quad}}$) is a function of the meta language also, and cannot be a function of \mathbb{N} . Since it cannot be a function of \mathbb{N} , then equivalence regarding that function and the function $GNUMERAL(n, k)$ cannot be established in \mathbb{N} .^e

This means that Harrison's claim that 2.1 - 2.7 are all equivalent within \mathbb{N} is patently erroneous, since the series of equivalences depends on at least one equivalence that cannot be an equivalence within \mathbb{N} . The claim of a proof of incompleteness is invalid, since at least one step in the proof process involves untenable assertions.

^dSee Harrison's section *The fixpoint lemma* in his Chapter 7 *Limitations*.

^eThat is not to say that there cannot be a purely number-theoretic function that is *similar* to the function $gnumeral(n)$; there can be such a function, but that is quite beside the point here.

3 Response from Harrison to the demonstration of the error in his proof

Harrison's response to the demonstration of an error in his proof is as follows:

'In Section 3 the author considers Harrison's discussion of the proof by HOL Light. However, the author's criticisms appear not to be based on the actual formal proof itself, but rather on an informal proof in the book. This rather oblique line of criticism is doubly flawed:

- 1. The author is not in fact discussing the formal proof itself, but what the author claims, without obvious justification, to be an informal description of it. (Where does the book claim that, by the way?)'*

The section in Harrison's book with the heading 'Gödel's incompleteness theorem' is quite obviously a description of a proof of incompleteness which is written in computer code. Why Harrison should imply that the book might not be an accurate description of his proof when his subsequent text (see below) indicates that he accepts that it *is* such a description, is perplexing; Harrison addresses the issues raised by this author by referring to the correspondence between the book and the proof, as can be seen below.

Harrison continues:

- '2. The author's criticisms are connected with a typical abbreviatory abuse of language that occurs only in the informal presentation and NOT the formal proof. If anything, the author's criticisms emphasize the imprecision of informal language and proofs.'*

This author has proceeded in precisely the same way as would any reviewer of an article that consisted of a proof in a non-standard notation and a description of that proof. When a reviewer discovers that the description indicates that the proof contains an error, it would not be expected that the reviewer then performs a complete in depth examination of the non-standard notation of the proof itself. If we cannot place any reliance whatsoever on Harrison's description, one is given to wonder what Harrison's objective was in providing that description in the first place. And, in any case, as will be seen below, Harrison's attempt to clarify what is occurring in his 'formal proof' fails to resolve the error - his code, as well as his description of it, is fundamentally flawed.

Harrison continues:

'After a sound summary of the notation and general setup from the book [9], the author presents the nub of his objection which is the following equivalence in N :

$$GNUMERAL(n,k) \Leftrightarrow gnumerical(n)$$

The author objects to this equivalence because $GNUMERAL(n,k)$ is a formula of the object language while $k = gnumerical(n)$ is a formula of the meta-language. It is true that the book is implicitly treating $GNUMERAL(n,k)$ and

many other formulas as meta-level expressions, for notational convenience. This is discussed near the beginning of the section “Definable relations, sets and functions” where Harrison talks about “fussy distinctions between variables and their interpretation”.

‘If the author cares to examine the formal proof, however, he will see that his critique does not apply to it. On the contrary, one of the merits of completely formal proofs is that they don’t let one get away with any sloppiness at all, and are far more precise and pedantic than this author or any other human being.’

‘Examining the file where the relevant definitions and formal theorems are found: <http://code.google.com/p/hol-light/source/browse/trunk/Arithmetic/tarski.ml> one will see that the distinctions that the author cares about are maintained in a very precise way. The naming and other minor aspects of the formal proof are slightly different from those in the book; the **gnumeral** function is actually defined as a binary relation rather than a function:

```
gnumeral m n = (gterm(numeral m) = n)
while GNUMERAL(n,k) is actually called arith_gnumeral n p:
arith_gnumeral n p
  formsubst ((0 |-> n) ((1 |-> p) V))
            (arith_gnumeral1' (arith_pair Z (numeral 3))
              (arith_pair (V 0) (V 1))).’
```

‘But the essential distinction the author notes is still there: **gnumeral** is a meta-level concept while **arith_gnumeral n p** (for given parameters **n** and **p**) is a formula of the object logic. The equivalence at issue is stated formally as follows (the formal theorem called **ARITH_GNUMERAL**): holds **v** (arith_gnumeral **s t**) <=> gnumeral (termval **v s**) (termval **v t**).’

‘This is precisely reflecting the distinction between just the formula itself (arith_gnumeral **s t**) and the fact that it holds in N with respect to a particular valuation **v** of the free variables (holds **v** (arith_gnumeral **s t**)). The equivalence is stated with **gnumeral** applied to the corresponding valuations of the subterms **s** and **t**. In summary, the author’s criticisms, such as they are, emphatically do not apply to the formal proof itself.’

Harrison claims that the error in his description in his book is not replicated in his ‘formal proof’. But an analysis of his actual code shows that the same careless disregard for the actual definition of the domains of variables applies both to his informal description and to the actual code of his proof. In the code that Harrison refers to:

```
holds v (arith_gnumeral s t) <=> gnumeral (termval v s) (termval v t)
```

the terms **v**, **s** and **t** are all variables; **termval** is defined by Harrison as a function where its second variable is defined as having a domain that is **not** the domain of natural numbers. This can be ascertained by examining the code, which is viewable online at

<http://code.google.com/p/hol-light/source/browse/trunk/Arithmetic/fol.ml>, and the HOL-light and OCaml reference manuals [3, 4].

So, considering the lines of Harrison’s proof which are:

```
let ARITH_GNUMERAL = prove
  (`!v s t. holds v (arith_gnumeral s t) <=>
   gnumeral (termval v s) (termval v t))`...
```

if, as Harrison asserts, the relation `arith_gnumeral` is a formula of the formal system that is the object of the proof (which Harrison calls the ‘object logic’), we can see immediately that in this code there is a confusion of variable domains, since the variable `s` where it occurs in `arith_gnumeral s t` must have the domain of natural numbers, but where the *same* variable `s` occurs in `termval`, it must have a domain that is *not* the domain of natural numbers. The only alternative is that Harrison’s code has not correctly defined the domain of the variables of `arith_gnumeral` as restricted to the natural numbers; if `arith_gnumeral` is to represent a formula of the formal system that is the object of Harrison’s proof, then the variables of that formula must be defined as the domain of natural numbers.^f

In either case, it is evident that Harrison’s code fails to observe elementary mathematical principles regarding the domains of variables; Harrison’s response only serves to confirm that his ‘formal proof’ relies on an erroneous confusion of the domains of the variables in the proof. Despite Harrison’s conviction that his computer software system cannot allow any ‘sloppiness’ in his code, the evidence is that it does indeed allow such ‘sloppiness’, and that it fails to detect a violation of elementary mathematical principles.

References

- [1] J. Harrison, *Handbook of Practical Logic and Automated Reasoning*. Cambridge University Press, 2009. ISBN: 9780521899574 (eBook format: ISBN: 9780511508653).
- [2] HOL, “The HOL Light theorem prover.” University of Cambridge Computer Laboratory website: <http://www.cl.cam.ac.uk/~jrh13/hol-light/>.
- [3] J. Harrison, “The HOL Light System - REFERENCE.” http://www.cl.cam.ac.uk/~jrh13/hol-light/reference_220.pdf, Oct, 2011.
- [4] “The OCaml system.” <http://caml.inria.fr/distrib/ocaml-3.12/ocaml-3.12-refman.pdf>, July, 2011.

^fIf the reader is interested as to which of the two possibilities is the primary source of Harrison’s error, it is suggested that the reader contact Harrison with that question. This author sees no point in expending further analysis on a proof which is fundamentally flawed, when Harrison, who is presumably quite familiar with the details of his proof, should be able to answer the question without difficulty.

Bibliography - Errors in other incompleteness proofs

- [a] J. R. Meyer. “*An Error in a Computer Verified Proof of Incompleteness by Russell O’Connor.*” http://www.jamesrmeier.com/pdfs/ff_oconnor.pdf, 2011
- [b] J. R. Meyer. “*An Error in a Computer Verified Proof of Incompleteness by Natarajan Shankar.*” http://www.jamesrmeier.com/pdfs/ff_shankar.pdf, 2011
- [c] J. R. Meyer. “*A Fundamental Flaw in an Incompleteness Proof in the book ‘An Introduction to Gödel’s Theorems’ by Peter Smith.*” http://www.jamesrmeier.com/pdfs/ff_smith.pdf, 2011
- [d] J. R. Meyer. “*A Fundamental Flaw In Incompleteness Proofs by Gregory Chaitin.*” http://www.jamesrmeier.com/pdfs/ff_chaitin.pdf, 2011
- [e] J. R. Meyer. “*A Fundamental Flaw In Incompleteness Proofs by S. C. Kleene.*” http://www.jamesrmeier.com/pdfs/ff_kleene.pdf, 2011