A FUNDAMENTAL FLAW IN AN INCOMPLETENESS PROOF BY GEORGE BOOLOS

James R Meyer
http://www.jamesrmeyer.com

v1 8 Dec 2011

Abstract

This paper addresses a proof of incompleteness published by George Boolos. An analysis of this proofs demonstrates that there is an elementary error in the proof; the proof relies on the unproven assumption that the formal system can self-reference its own formulas.

1 Introduction

Boolos published his proof of incompleteness in 1989 [2]. It is also to be found in two books published subsequently [3, 4]. He claims that his proof is essentially different from previously published proofs, since it operates on the principle of Berry's paradox.He claims elsewhere [1] that his proof provides an essentially different reason for incompleteness.

Berry's paradox, of course, is a contradiction that is stated in natural language, which means that it is stated with a lack of precision of definition. One might expect that when a corresponding statement is used as part of a mathematical proof, it would be stated with sufficient precision so that it is evident that there is not a hidden contradiction in the statement.

However, Boolos's proof does not give a rigorously precise formulation of the Berry paradox. Instead, he relies on a assumption that the formal system for which he claims to prove incompleteness can self-reference its own formulas; Boolos provides no proof of this assumption, and since his proof is entirely reliant on that assumption, his proof cannot sensibly be called a proof at all. Rather than providing a different reason for incompleteness, as Boolos claims, the proof merely demonstrates that once an assumption is made that a formal system can have certain types of self-referential statements, the result of incompleteness is a trivially obtained consequence.

2 Boolos's proof

The outline of Boolos's proof is as follows:

Definition: The *language of arithmetic* consists of 16 symbols: $+$, $\times$, 0, s, $=$, $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$, $\forall$, $\exists$, $($, $)$, x, and $'$.

Assumption: There is an algorithm M that outputs all true statements of the *language of arithmetic* and no false ones.

Definition: *[n]* is the expression that consists of 0 preceded by n quantity of the symbol s.

Definition: The formula $F(x)$ is said to *name* the natural number n if the expression $\forall x(F(x) \leftrightarrow x = [n])$ is an output of the algorithm M.

Assumption: There is a formula in the *language of arithmetic* (i.e., using only the 16 symbols mentioned above) that states (by an appropriate encoding): '*x is a number that is named by some formula containing z symbols*'. The designation $C(x,z)$ is used to refer to this formula.

Definition: $B(x,y)$ is defined as $\exists z(z < y \wedge C(x,z))$.

Definition: $A(x,y)$ is defined as $(\neg B(x,y) \wedge \forall a(a < x \rightarrow B(a,y)))$.

Definition: k is defined as the 'number of symbols in' $A(x,y)$.

Definition: $F(x)$ is defined as $\exists y(y = ([10] \times [k]) \wedge A(x,y))$.

Boolos then defines a formula that is defined in terms of this $F(x)$ as $\forall x(F(x) \leftrightarrow x = [n])$. He asserts that this formula states that '*x is the least number not named by any formula containing fewer than* $10k$ *symbols*' He then states that this formula cannot be an output of the algorithm M, but that the formula is actually true. Boolos concludes that this contradiction indicates that his initial assumption that there is an algorithm M is incorrect, and that the contradiction proves that there is no such algorithm M. That completes the proof.

3 Analysis of Boolos's proof

In this proof, Boolos ignores a basic tenet of logic that, in any proof by contradiction, the contradiction indicates that at least one of the assumptions leading to that contradiction is incorrect, but it does not specify which one.

Boolos's proof, besides the assumption that there is an algorithm M, assumes that there is a formula $C(x,z)$ of the *language of arithmetic* which encodes the expression '*x is a number that is named by some formula containing z symbols*'. The formula that gives rise to the contradiction, $\forall x(F(x) \leftrightarrow x = [n])$, is defined in terms of this formula $C(x,z)$.

Boolos justifies his assumption regarding the formula $C(x,z)$ by 'sketching' the construction of the formula, as follows:

> *"Let us now sketch the construction of a formula $C(x,z)$ that says that x is a number named by a formula containing z symbols. The main points are that algorithms like M can be regarded as operating on 'expressions', i.e., finite sequences of symbols; that, in a manner reminiscent of ASCII codes, symbols can be assigned code numbers (logicians often call these code numbers Gödel numbers); that certain tricks of number theory enable one to code expressions as numbers and operations on expressions as operations on the numbers that code them; and that these numerical operations can all be defined in terms of addition, multiplication, and the notions of logic."*

Here Boolos correctly states that one can assign code numbers that correspond to the 16 symbols of the *language of arithmetic*. He correctly states that one can use the coding of symbols to assign numbers that correspond to expressions of the *language of arithmetic*. And that one can, for any operation on such expressions, by such encoding, have a corresponding operation on the corresponding code numbers. And that any such numerical operation can be defined in terms of basic operations using the $+$, $\times$,$\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$, $\forall$, $\exists$ operators. Boolos continues:

> *"Discussion of symbols, expressions (and finite sequences of expressions, etc.) can therefore be coded in the language of arithmetic as discussion of the natural numbers that code them. ... tricks of number theory then allow all such talk of symbols, sequences, and the operations of M to be coded into formulas of arithmetic"*

From the above, since each specific expression of the *language of arithmetic* is encoded as a specific natural number, then it follows that in a statement of the meta-language that refers to expressions of the *language of arithmetic* in general, there will be a variable in that statement that has the domain of expressions of the *language of arithmetic*. It follows that upon encoding that statement, the encoding will have a corresponding variable whose domain is natural numbers.

For example, given the expression 's ***Proves*** t', where s and t are variables with the domain of expressions of the *language of arithmetic*, then the encoding gives some relation of the *language of arithmetic* $R(x,y)$, where x and y are variables with the the domain of natural numbers (in the format of the *language of arithmetic*), and which correspond to s and t respectively.

It also follows that on decoding an expression of the *language of arithmetic*, since every variable of the *language of arithmetic* has the domain of natural numbers, every free variable of the *language of arithmetic* will decode to a free variable whose domain is expressions of the *language of arithmetic*. For the example above, the relation $R(x,y)$ decodes to 's ***Proves*** t'.

So, Boolos's sketch outline regarding $C(x,z)$ tells us that decoding the formula $C(x,z)$ gives an expression of the meta-language with two free variables, both of which have the domain of expressions of the *language of arithmetic*. However, the expression which Boolos asserts is the decoding of the formula $C(x,z)$ is the expression *'x is a number that is named by some formula containing z symbols'*, which has two free variables x and z which have the domain, not of expressions of the *language of arithmetic*, but of natural numbers.

This demonstrates that Boolos's sketch of the construction of his formula $C(x,z)$ fails to substantiate his claim that that formula encodes *'x is a number that is named by some formula containing z symbols'*. Boolos has failed to show that there are valid *'tricks of number theory'* that can create an encoding of *'x is a number that is named by some formula containing z symbols'* to an expression of the *language of arithmetic*, and which ***also*** preserves the truth value of the expression.

Boolos's perfunctory justification only serves to introduce further assumptions, rather than provide any logical clarification. Boolos's claim that he has proved incompleteness carries no logical validity whatsoever.

References

[1] George Boolos. A Letter from George Boolos. *Notices of the American Mathematical Society*, 36:676, 1989.

[2] George Boolos. A New Proof of the Gödel's Incompleteness Theorem. *Notices of the American Mathematical Society*, 36:388–390, 1989.

[3] George Boolos. *Logic, Logic, and Logic.* Harvard University Press, 1999. ISBN: 9780674537675.

[4] Reuben Hersh. *What Is Mathematics, Really?* Oxford University Press, 1997. ISBN: 9780195113686.