A FUNDAMENTAL FLAW IN INCOMPLETENESS PROOFS BY GREGORY CHAITIN

James R Meyer

http://www.jamesrmeyer.com

v1 26 Oct 2011

Abstract

There are several similar proofs published by Chaitin involving the concept of information-theoretic complexity, and Chaitin claims that these are proof of the incompleteness of formal systems. An elementary analysis of these proofs demonstrates that the proofs are not in fact proofs of incompleteness.

1 Introduction

Over a period of several years Gregory Chaitin has publishing several very similar proofs where the crucial claim by Chaitin is that these are proofs of the incompleteness of formal systems. There has been some criticism of these proofs, in particular by van Lambalgen[10], Raatikainen[8, 9], Franzén[7], and D'Abramo[6]. For example, Raatikainen[8] states of van Lambalgen's paper that, *'...his discussion certainly shows that there must be something wrong with the received interpretation ...'*. This is not surprising since the conventional interpretation is that Chaitin has proved the incompleteness of formal systems, whereas a logical analysis shows that this is not the case.

2 An Analysis of Chaitin's proof

An elementary analysis of Chaitin's proofs of incompleteness reveals the simple fact that the proofs are not in fact proofs of incompleteness. Chaitin has published several similar versions of his proof[1, 2, 3, 4, 5]. We analyse here one such proof; the fundamentally erroneous claim is common to all of them.

The essence of Chaitin's Theorems A and B of the paper in question[5] consists of the following:

Initial definitions:[a]
$C(p)$ represents a computer C that runs a program p

[a] Chaitin's proofs includes other material which is not necessary for this analysis

$H(x)$ is defined as the size in bits $|p|$ of the smallest program p that outputs the string x on a Universal computer U.

Chaitin asserts that, given a formal system, there is a computer C and a program which, when run on that computer, searches through all proofs of the formal system, and when it finds the smallest proof that proves the proposition $H(s*) > j$, where $s*$ is any specific bit string, and where j is a specific natural number that satisfies certain conditions (specified by Chaitin in his paper), then the computer prints out the string $s*$, and the program stops.

Chaitin's argument continues with the supposition that there is such a proof in the formal system, and he observes that a contradiction results from this argument. Chaitin concludes that this contradiction indicates that there must be some numbers $s*$ and j for which the proposition $H(s*) > j$ applies, but it must be the case that the formal system cannot prove this proposition.

That concludes the essence of Chaitin's proof.

But this proof is not a proof of incompleteness. A proof of the incompleteness of a formal system must be a proof that there is at least one proposition of the formal system that the formal system cannot prove to be either true or false. And Chaitin has not provided any proof that there are propositions in the formal system that state in some way the proposition $H(s*) > j$, where $s*$ is a specific bit string and j is a specific natural number.

If Chaitin had included a proof that the formal system can state in some way the proposition $H(s*) > j$, then his argument would demonstrate[b] that there is a proposition of the formal system that is correct, yet the formal system cannot prove it. And that would be a proof of the incompleteness of the formal system. But Chaitin has not provided so such proof. As such his claim that he has proved incompleteness of a formal system carries no logical validity whatsoever. All that Chaitin's argument demonstrates[b] is that: *'either there are propositions of the formal system that the formal system cannot prove true or false OR the formal system cannot prove propositions that cannot be stated within that system.'*

Since Chaitin's proof does not tell us which is the case, and since we know that the formal system cannot prove propositions that cannot be stated within that system, Chaitin's conclusion is not in the least surprising.

3 Discussion

Chaitin's claim that his proof is an incompleteness proof can be considered to be a case of attaching, a posteriori, an additional supposition into the proof. That additional supposition is that the formal system can state a certain type of proposition, namely a proposition that states in some way $H(s*) > j$. But now the proof contains two suppositions; and in a proof by contradiction, as Chaitin's proof is, the contradiction merely indicates that at least one of the suppositions used in the proof is incorrect, but it does not specify which one.

[b] If Chaitin's definition of H is a precise definition

And to accept, without proof, the supposition that the formal system can state propositions that state in some way $H(s*) > j$ is tantamount to assuming without proof the notion that the formal system is incomplete. For once the assumption is made that a formal system can state a certain type of self-referential statement, the proof of incompleteness of such a system is a trivially obtained consequence.

And while Raatikainen[8] states that Chaitin's proof avoids any direct self-reference, in fact Chaitin's definition of H is by way of a purported Universal computer; and that Universal computer is defined in terms that reference *every* program of *every* computer. So the purported program in Chaitin's proof that searches for the proposition $H(s*) > j$ is searching for a proposition whose definition includes a reference to that program itself.

Can propositions in a formal logical system reference their own length?

A logical consideration casts doubt on the notion that propositions of a logically coherent formal system can refer, in general, to the length of the propositions of that formal system. First, we note that formal systems do not directly refer to physical things; the formal system, of itself, contains no information about computers or computer programs. A formal system is simply a set of symbols and a set of rules regarding those symbols. Of the combinations of these symbols[c] some will be be logical operators, some will be variables, and some will be the specific values of that system. For example, a simple arithmetic formal system has specific values that are of the form 0, $s0$, $ss0$, $ss0$, ..., and which represent natural numbers.

In order for a proposition of a formal system to refer to the length of its own propositions, it would seem that it would have to reference all symbols of the system, and all combinations of these symbols. And that would appear to require that the formal system would have at least one variable with the domain of all symbol combinations of the system.[d] That would mean that all symbols of the system, and all symbol combinations of the system would be specific values in the system. But if that was the case, then there could be no symbol or symbol combination that could be a variable of the system - otherwise we would have the case where a symbol (or a symbol combination) would be at the same time a variable and a specific value - which is a logical absurdity.

The above demonstrates why it is not acceptable to simply assume that the formal system can state in some way the proposition $H(s*) > j$ rather than proving that it is the case.

[c]Here a single symbol is taken to be a combination of one symbol

[d]This domain includes all single symbols. The alternative, that there is at least one variable with the domain of all single symbols of the system, and at least one variable with the domain of all symbol combinations of the system of more than one symbol, makes no difference to the argument

References

[1] G. J. Chaitin. Computational Complexity And Gödel's Incompleteness Theorem. *ACM SIGACT News*, 9:11–12, 1971.

[2] G. J. Chaitin. Information-theoretic computational complexity. *IEEE Transactions on Information Theory*, IT-20:10–15, 1974.

[3] G. J. Chaitin. Information-Theoretic Limitations Of Formal Systems. *Journal of the ACM*, 21:403–424, 1974.

[4] G. J. Chaitin. Algorithmic Information Theory. *IBM Journal of Research and Development*, 21:350–359, 1977.

[5] G. J. Chaitin. Information Theoretic Incompleteness. *Applied Mathematics and Computation*, 52:83–101, 1992.

[6] Germano D'Abramo. On the information-theoretic approach to Gödel's incompleteness theorem. 2002, math-ho/0211078.

[7] Torkel Franzén. *Gödel's Theorem: An Incomplete Guide to its Use and Abuse.* A K Peters, 2005. ISBN: 1568812388.

[8] Panu Raatikainen. On interpreting Chaitin's incompleteness theorem. *Journal of Philosophical Logic*, 27:269–586, 1998.

[9] Panu Raatikainen. Algorithmic information theory and undecidability. *Synthese*, 123:217–225, 2000.

[10] M van Lambalgen. Algorithmic information theory. *Journal of Symbolic Logic*, 54:1389–1400, 1989.