A FUNDAMENTAL FLAW IN AN INCOMPLETENESS PROOF BY STANISŁAW ŚWIERCZKOWSKI

James R Meyer

http://www.jamesrmeyer.com

v1 16 Oct 2014

Abstract

This paper examines a proof of incompleteness by Świerczkowski in a paper entitled "*Finite sets and Gödel's incompleteness theorems*".[10] Świerczkowski's proof is different to most other proofs of incompleteness because the formal system that is used for the proof is a system of hereditarily finite sets. Świerczkowski claims that this makes his proof simple and elegant, and that it has enabled his proof to be complete and without any gaps or omissions, nor relying on references to other publications. Świerczkowski claims that for this reason his proof his superior to most other proofs of incompleteness. However, the author makes an elementary error in his proof that renders the proof invalid.

1 Introduction

Since Gödel published his original proof of incompleteness[1] over eighty years ago, there have been numerous attempts (see, for example Smullyan,[9] or Smith[8]) to provide proofs of incompleteness that are superior to Gödel's original proof. This paper examines one such attempt which is given by Świerczkowski in a paper entitled "*Finite sets and Gödel's incompleteness theorems*".[10]

2 Background to Świerczkowski's Proof

Most incompleteness proofs use a coding system that creates a unique one-to-one correspondence between the symbol combinations of a formal system and natural numbers. Given such coding, a correspondence is then established between relations between formulas of the formal system and relations between numbers. In Świerczkowski's proof the coding system creates a one-to-one correspondence between expressions of the formal system and certain finite sets, where these sets correspond to the notion of natural numbers. Świerczkowski's formal system will only be described briefly here, since most of the details are peripheral to the demonstration of the flaw in the proof. In any case, full details of the formal system can be found in Świerczkowski's paper.

2.1 Hereditarily Finite Sets

For convenience the Hereditarily Finite Set system that Świerczkowski uses as the formal system is referred to by the abbreviation HF. The objects of the formal system are hereditarily finite sets; HF uses a language that has 7 basic symbols and infinitely many variable symbols. The 7 basic symbols are 0, ϵ, $\lhd$, =, $\vee$, $\neg$ and $\exists$. Since there are infinitely many variables, a symbol system such as $x_1, x_2, x_3, \ldots$ can be used for the variables of the system.[a] Further details of the system will not be described here, as the details can be found in Świerczkowski's paper.

2.2 The Coding Function

One of the key ideas behind most incompleteness proofs is that we can form a correspondence between formulas of the formal system and certain objects; in this case the objects are finite sets. The coding is such so that for every relationship between formulas of the formal system that relationship can be mapped precisely to relationships between finite sets; so that if a certain relationship between formal system formulas applies, then there is corresponding relationship between the corresponding finite sets which also applies.

Świerczkowski refers to the system used in his proof to map the formal system formulas to finite sets as the 'coding'. This coding is a function which gives a one-to-one correspondence from symbol combinations of the formal system HF to finite sets, and this function is designated by $\ulcorner \quad \urcorner$. The function has one free variable, although Świerczkowski does not specify any symbol for this variable. For convenience we will refer to the function with the free variable Ω, which gives the function as $\ulcorner \Omega \urcorner$, where Ω is a variable of a meta-language to the system HF, and which has the domain of all symbol combinations of the formal system HF, including all variables of HF.

The proof of incompleteness involves the claim that there is a formula of the formal system that expresses the coding function in some way. However, in a consistent logical system there is no variable of the formal system that has the domain that includes all variables of the formal system, including itself. Hence, a claim that there is an expression of the formal system that contains the information of the definition of the coding function is a quite remarkable claim, since there is no variable of the formal system that is comparable to the free variable Ω of the function $\ulcorner \Omega \urcorner$. Because of this extraordinary claim, it is worth expending some effort examining the method by which the proof asserts that the system HF can express the coding function.

2.3 Domains of Variables and Substitutability

As noted above, since there are infinitely many variables in HF, a symbol system such as $x_1, x_2, x_3, \ldots$ is required. Świerczkowski uses the term k to refer to the k^{th} suffix in a series of suffixes $1, 2, 3, \ldots$, so that x_k refers to the k^{th} variable in a series of variables $x_1, x_2, x_3, \ldots$.

It is important to maintain the distinction between the domain of a variable and the expressions that may be substituted for a variable. Substitutability of a variable by an expression does not imply that the expression is a member of the domain of the variable. For example, in conventional arithmetic,

[a] For convenience, Świerczkowski often uses other symbols such as x, y, z, etc, to indicate some such variable of the system HF.

given the formula $x+3>14$, it is valid to substitute the x by another variable, for example y, to give an equivalent formula $y+3>14$. One can also substitute an expression such as $y-2$ for the variable x to give another valid formula $y-2+3>14$ (where the set of values of y that satisfy the equation are different to the set of values of x that satisfy the original formula). In these cases, the variable is simply replaced by another variable expression which, like the original variable, has no specific value. In contrast, the domain of a variable necessarily consists only of expressions that are not variables; only specific values can be members of the domain of a variable.[b]

While Świerczkowski does not actually define the domain of the variables of HF, it is evident that the domain of these variables is all specific sets of HF.

2.4 Σ-formulas

Here we give a very brief mention of Σ-formulas as the details are peripheral for our purposes. Essentially Σ-formulas are formulas that are necessarily decidable, given any specific values of their free variables. The purpose of the definition is similar to the purpose of the use of primitive recursion in most incompleteness proofs.

2.5 Definitions of relations of HF

In Section 3, Świerczkowski introduces a series of relations, which are so defined that if the free variables of the relations correspond by the coding system to symbol combinations of the system HF, then there is a corresponding relation between those symbol combinations in a meta-language that can express such relations. For example, Świerczkowski's first relation is $\mathrm{Var}(x)$, which corresponds to the assertion that X is a variable of the system HF, so that if $x = \ulcorner X \urcorner$, and if X is a variable of the system HF, then the relation $\mathrm{Var}(x)$ is provable in the system HF. Note that here X is a variable of the meta-language.

It should be noted that Świerczkowski often uses the coding term within expressions which are meant to represent expressions of the system HF; for example, the definition of the relation $\mathrm{SeqTerm}(s,k,t)$ is:

$$\mathrm{LstSeq}(s,k,t) \wedge \forall (l \in k)\{s_l = 0 \vee \mathrm{Var}(s_l) \vee \exists (m,n \in l)[s_l = \langle \ulcorner \triangleleft \urcorner, s_m, s_n \rangle]\}$$

where $\ulcorner \triangleleft \urcorner$ is not an expression of the system HF, but is a term in the meta-language that ***represents*** a unique specific expression in the system HF. As long as this distinction is always maintained and recognized, then there should be no confusion of meta-language and object language.

Since $\ulcorner \Omega \urcorner$ is not an expression of HF, and since it does it represent any expression of HF, it should not appear in any expression that is claimed to represent an expression of HF; there is no expression of HF for which $\ulcorner \Omega \urcorner$ is a valid substitution. However, as shown above, certain terms that include the coding function $\ulcorner \ \urcorner$, such as the expression $\ulcorner \triangleleft \urcorner$ may be substituted, *in the meta-language*, for expressions of HF, provided certain conditions are met, such as recognizing that the expression only represents an expression of HF, and is not itself an expression of HF.

[b] Note that in a meta-language to an object language, the variables of the *object* language can be members of the domain of a variable of the meta-language.

2.6 *p-functions*

In Section 5, Świerczkowski introduces the concept of a *p-function*.[c]

If φ is a formula of HF with a free variable y and n other free variables, then if $\exists! y\varphi$ is a theorem of HF ($\exists! y$ means that there exists one and only one such y), then there is an associated function that is represented by the term F^y_φ. Such functions can have any finite number of free variables, and the function and its variables are represented by $F^y_\varphi(z_1,\ldots,z_n)$, and the number of free variables of this function F^y_φ is one less than for the formula φ.

By adding the symbols of such *p-functions* to the language of HF, an expanded language is created. Although they are not formulas of HF, for any such *p-function* in this language, there is a corresponding formula of HF.

Recalling what was said in Section 2.3 above, we note that no additional specific objects are defined that can be members of the domain of a variable of this expanded language. Hence the domain of the variables of this expanded language is precisely the same as the domain of the variables of HF.

And as was noted in Section 2.3 above, under certain conditions, an expanded set of expressions can be validly substituted for a variable within this expanded language. A brief description of such substitutions as described by Świerczkowski follows below.

For the purposes of the expanded language, F^y_φ is called a *p-symbol* and any formula that includes the term F^y_φ is called a *p-formula*. A *p-term* is any term of the expanded language; *p-formulas* contain *p-terms*, and *p-formulas*, while not expressions of HF, represent expressions of HF.

Given an expression $F^y_\varphi(\tau_1,\ldots,\tau_n)$ of the expanded language, if $\tau_1,\ldots,\tau_n$ are substitutable only by members of the domain of variables of HF, then $\tau_1,\ldots,\tau_n$ are called simple *p-terms*; otherwise they are non-simple *p-terms*.

Świerczkowski describes a process that he calls '*reduction*', by which the number of non-simple *p-terms* in a *p-formula* can be reduced until there are only simple *p-terms* remaining in the expression.

For example, following Świerczkowski's description, suppose that α is a *p-formula* expression which has only one non-simple *p-term*. There is a corresponding formula φ of HF with the free variables x and y, which we write as $\varphi(x,y)$.

The simple *p-term* of the *p-formula* α occurs in some atomic sub-formula of α, and we call this atomic sub-formula $\beta(z)$. Then there is a simple *p-term* τ such that $\beta(F^y_\varphi(\tau))$ is a sub-formula of α. The reduction process is the replacement of this atomic sub-formula $\beta(F^y_\varphi(\tau))$ by:

$$\exists w\beta(w)\wedge\widetilde{\varphi}(\tau,w),$$

where w is a variable that does not occur in α nor τ, and where $\widetilde{\varphi}$ is equivalent to φ but such that τ and w are *substitutable* for x and y in $\widetilde{\varphi}$.[d] The result of this reduction is a formula with one free variable which is *substitutable* only by members of the domain of variables of HF.

As noted above, this process of reduction does not affect the domain of the variables in the formulas, which remains as all specific sets of HF.

[c] Świerczkowski also gives more details of *p-functions* in his Appendix 3: *p-functions*.

[d] This means that τ and w can be substituted in $\widetilde{\varphi}$ because there is no instances already of τ and w in $\widetilde{\varphi}$ such that the substitution would result in additional instances of τ or w that would alter the logical content of the formula φ.

3 Świerczkowski's Proof of Incompleteness

Throughout his paper, Świerczkowski exhibits a careless disregard for the distinction between meta-language and object-language. The point at which this culminates in a logical absurdity occurs in Świerczkowski's Section 6. In this section a *p-function* W is defined as a recursive function by:

$$W(x) = \begin{cases} 0 & \text{if } x = 0 \text{ or if } x \text{ is not an ordinal} \\ \langle \ulcorner \lhd \urcorner, W(x^-), W(x^-) \rangle & \text{if } x \text{ is a non-zero ordinal} \end{cases}$$

Świerczkowski's Lemma 6.1 asserts:

There is a p-function W such that (6.1.A)

$\vdash W(\ulcorner x_k \urcorner) \;=\; \ulcorner(\ulcorner x_k \urcorner)\urcorner$ (6.1.B)

for every variable x_k. (6.1.C)

Now, while W may be a *p-function*, the function $W(\ulcorner x_k \urcorner)$ in 6.1.B is a *composite* of the W function and the $\ulcorner\;\urcorner$ function. Świerczkowski's Lemma 6.1 asserts that the system HF can prove the proposition:

$$W(\ulcorner x_k \urcorner) \;=\; \ulcorner(\ulcorner x_k \urcorner)\urcorner$$

But $\ulcorner x_k \urcorner$ cannot be a *p-function*, since its free variable x_k has the domain of all variables of HF, that is, $x_1, x_2, x_3, \ldots$, and as explained above in Section 2.6, no variable of HF or of the expanded language of *p-functions* has this domain, otherwise it would include itself in its own domain, and it would be a variable and an object of the language at the same time, which is a logical absurdity. Hence the free variable x_k is not a variable of HF, nor can it be a variable of the expanded language of *p-functions*. It follows that $\ulcorner x_k \urcorner$ cannot be a *p-function*, and the composite function $W(\ulcorner x_k \urcorner)$ also cannot be a *p-function*.

Świerczkowski's objective in his definition of *p-functions* is that, given any *p-function*, there is a corresponding formula of the system HF. But the expression $W(\ulcorner x_k \urcorner) \;=\; \ulcorner(\ulcorner x_k \urcorner)\urcorner$ includes expressions that are not *p-functions*. Świerczkowski's 'proof' of his Lemma 6.1 completely ignores this crucial fact , so that nothing in Świerczkowski's 'proof' establishes that there can be any expression in HF corresponding to $\ulcorner x_k \urcorner$ or $\ulcorner(\ulcorner x_k \urcorner)\urcorner$, let alone a proof in HF regarding such expressions.

This is no trivial error. It is a fundamental error for which there is no simple resolution. Since the remainder of Świerczkowski's paper relies on the Lemma 6.1, Świerczkowski's 'proof' of incompleteness fails and his paper is fatally flawed.

4 Conclusions

Świerczkowski claims that his paper is unique in that it can "*present all arguments completely, without omissions . . . such a degree of completeness in proving Gödel's results has never been attained before. . . . all [other] published proofs of Gödel's incompleteness theorems contain gaps, omissions or references to key results in other publications.*" But as in so many other incompleteness proofs[8] [9] [2] [5] the facade of detail hides a very simple unfounded crucial assumption, which invariably results in the formal system appearing to self-reference when a fully detailed logical analysis shows nothing of the sort.

It can be observed that, in common with Gödel's original proof, and most other incompleteness proofs, Świerczkowski's coding function maps expressions of the formal system to sets which ostensibly are objects of the meta-language, but which are, rather conveniently, expressions that are precisely in the same format in this meta-language as the sets of the formal system HF. And in common with the authors of the afore mentioned proofs, Świerczkowski offers no explanation as to why this convenient identical format is used, rather than some other format being used for sets in the meta-language.

It might be noted that Lawrence Paulson has published some papers claiming a machine assisted proof of incompleteness that is based on Świerczkowski's paper. In one paper Paulson gives an overview[6] and in another the computer code.[7] Paulson's computer code does not appear to include any checking regarding the domain of the variable in the code that corresponds to the free variable referred to above, and hence it is flawed in the same way as Świerczkowski's proof. This type of omission has also been observed in other claims of computer checked proofs of incompleteness.[4][3]

References

[1] GÖDEL, K. *Über formal unentsceidebare Sätze der Principia Mathematica und verwandter Systeme I.* Monatshefte für Mathematik und Physik *38* (1931), 173–198.

[2] LINDSTROM, P. *Lecture Notes: Aspects of Incompleteness.* Springer-Verlag, 1997. ISBN: 3540632131.

[3] MEYER, J. R. *An Error in a Computer Verified Proof of Incompleteness by John Harrison*, 2011. http://www.jamesrmeyer.com/pdfs/ff_harrison.pdf.

[4] MEYER, J. R. *An Error in a Computer Verified Proof of Incompleteness by Russell O'Connor*, 2011. http://www.jamesrmeyer.com/pdfs/ff_oconnor.pdf.

[5] MOSTOWSKI, A. *Sentences Undecidable in Formalized Arithmetic.* Greenwood Press, 1982. ISBN: 9780313231513.

[6] PAULSON, L. C. *A Machine Assisted Proof of Gödel's Incompleteness Theorems for the Theory of Hereditarily Finite Sets.* The Review of Symbolic Logic (2013), 1–15. http://www.cl.cam.ac.uk/~lp15/papers/Isabelle/Goedel-logic.pdf.

[7] PAULSON, L. C. *Gödel's Incompleteness Theorems (Machine Code)*, Nov, 2013. http://afp.sourceforge.net/browser_info/current/AFP/Incompleteness/document.pdf.

[8] SMITH, P. *An introduction to Gödel's theorems*, Second ed. Cambridge University Press, 2013. ISBN: 9781107022843.

[9] SMULLYAN, R. M. *Gödel's Incompleteness Theorems.* Oxford University Press, 1992. ISBN: 0195046722.

[10] ŚWIERCZKOWSKI, S. *Finite sets and Gödel's incompleteness theorems*, vol. 422. Polska Akademia Nauk, Instytut Matematyczny, 2003. http://journals.impan.gov.pl/cgi-bin/dm/pdf?dm422-0-01.