What are the recursion theoretic properties of a set of axioms? Understanding a paper by William Craig

Armando B. Matos

armandobcm@yahoo.com

February 5, 2014

Abstract

This note is for personal use. It was written with the goal of understanding the paper "On Axiomatizability Within a System" by William Craig. Some of its consequences are also explored. Parts of two other papers on this subject are also transcribed and studied. These three papers are (or were) freely available at the internet. The recursion theoretic properties of a set of axioms are studied.

1 Introduction

This note is about the application of a recursion theoretical result discovered by Craig to logical formal systems. We transcribe and study parts of three papers related that result, known as Craig's Theorem or Craig's observation. We hope that some readers may enjoy this note, which is essentially a collection of views on the same subject.

General comments on formal deduction systems

Due to Gödel incompleteness theorem, a sufficiently strong consistent theory (view as the set of its theorems) cannot be decidable. Such a theory will be a recursively enumerable (or Σ_1^0) set.

Similarly, the set of axioms is also usually infinite, not decidable but recursively enumerable. We quote a Wikipedia entry on Gödel's incompleteness theorems.

> Many theories of interest include an infinite set of axioms [...] To verify a formal proof when the set of axioms is infinite, it must be possible to determine whether a statement that is claimed to be an axiom is actually an axiom. This issue arises in first order theories of arithmetic, such as Peano arithmetic, because the principle of mathematical induction is expressed as an infinite set of axioms (an axiom schema).

> A formal theory is said to be effectively generated if its set of axioms is a recursively enumerable set. This means that there is a computer program that, in principle, could enumerate all the axioms of the theory without listing any statements that are not axioms. This is equivalent to the existence of a program that enumerates all the theorems of the theory without enumerating any statements that are not theorems. Examples of effectively generated theories with infinite sets of axioms include Peano arithmetic and Zermelo–Fraenkel set theory.
> In choosing a set of axioms, one goal is to be able to prove as many correct results as possible, without proving any incorrect results. A set of axioms is complete if, for any statement in the axioms' language, either that statement or its negation is provable from the axioms. A set of axioms is (simply) consistent if there is no statement such that both the statement and its negation are provable from the axioms. In the standard system of first-order logic, an inconsistent set of axioms will prove every statement in its language (this is sometimes called the principle of explosion), and is thus automatically complete. A set of axioms that is both complete and consistent, however, proves a maximal set of non-contradictory theorems. Gödel's incompleteness theorems show that in certain cases it is not possible to obtain an effectively generated, complete, consistent theory.

A formal deduction system has several parts, namely: the set of theorems, also called "the theory", a recursively enumerable (or equivalently Σ^0_1) set of axioms, (possibly) a partial recursive function that extends the sets of axioms, a function that checks the correctness of a proof. This is a partial recursive function $\mathsf{check}(p)$ such that, if p is a proof, the computation $\mathsf{check}(p)$ halts with output TRUE; otherwise it may output FALSE or diverge. We also assume the existence of a total recursive function $\mathsf{theorem}$ that, given a proof p, outputs the theorem t that is proved, $t = \mathsf{theorem}(p)$. Notice that
$\vdash t \quad \text{iff} \quad \exists p : [\mathsf{check}(p) \wedge (\mathsf{theorem}(p) = t)]$.

It is easy to show that, if those two functions ($\mathsf{theorem}$ and check) exist, then the set of theorems as well as the set of formal proofs are recursively enumerable sets.

Given a formal system $\mathcal{F}$, it is possible to extend it with a partial recursive function A, resulting in a formal system $\mathcal{F}'$. If A is surjective (onto) with obtain exactly the same set of theorems.

$$\vdash_{\mathcal{F}'} t \;\;\Leftrightarrow\; \vdash_{\mathcal{F}} t \;\;\Leftrightarrow\;\; \exists x : (A(x) = p) \wedge \mathsf{check}(x) \wedge (\mathsf{theorem}(p) = t)$$

Although A will usually be used only once at the beginning of a proof, it could be in principle be used anywhere in a proof to deduce t, and we write ${}^{x}_{A}\!\vdash t$ if $A(x) = p$, $\mathsf{check}(p)$ holds, and $t = \mathsf{theorem}(p)$.

If we use a standard universal Turing machine U (which is surjective) as the partial recursive function A, we have the following results.

Theorem 1 *Suppose that in the formal system $\mathcal{F}$ a theorem t has a proof with length $|p|$. Then, in $\mathcal{F}+U$, t has a proof of length $K(p)$.*

In fact a stronger statement is possible.

Theorem 2 *In $\mathcal{F}+U$, every theorem t has a proof with Kolmogorov complexity $K(p) \leq K(t)+c$ where c is a constant.*

<u>Proof</u>. Use a Turing machine that enumerates the proofs of $\mathcal{F}$ until a proof of t is found. □

Organization of this note

The more general and abstract result, Theorem 4 (page 10) is only presented in Section 4 (page 10), after two motivating sections. Theorem 4 is a purely recursion theoretic result, not mentioning any logic concept; it's about recursive enumerability, binary relations and closures. When applying it to Logic, the sets are usually sets of axioms ("axiomata") or of theorems ("theories"), while the binaries relations are related to the logic deduction relation. In Sections 3 (page 7) and 2 (page 3) several particular results are proven; *these results can also be seen as corollaries of Theorem 4.* Although obvious from the titles, I should stress that Sections 2, 3, and 4 are directly based on the papers [5], [1], and [2], respectively.

When transcribing parts of a paper, my observations are interspersed with the transcribed text. I apologize for that. A part of text that is not transcribed is indicated by "[...]".

2 Mainly from Putman, "Craig's theorem" ([5])

In this section we present a partial transcription of [5] where the result mentioned in 4.3 (page 12) is proved and explained. The existence of recursive and of primitive recursive axiomatizations are proved in separate.

2.1 Craig's observation

In Craig's paper [2], the result (I) below is in the Applications Section (see below, page 11), being thus a corollary of general Theorem 4, page 10. While this theorem is a recursion theoretic result, not mentioning any logic concept (like axiom, deduction...), the result (I) is in fact an application Theorem 4 to formal system theory. As Craig writes in [2], "This observation [Theorem 4] can be applied to many formal systems S, by letting R correspond to the relation of deducibility in S...".

The "observation", as expressed by Putman in [5] is

> (I) Every theory that admits a recursively enumerable set of axioms can be recursively axiomatized.

2.2 General concepts

Some explanations are in order here:

1. A theory is an infinite set of wffs (well-formed formulas) which is closed under the usual rules of deduction. One way of giving a theory T is to specify a set of sentences S (called the axioms of T) and to define T to consist of the sentences S together with all sentences that can be derived from (one or more) sentences in S by means of logic.

2. If T is a theory with axioms S, and S' is a subset of T such that every member of S can be deduced from sentences in S', then S' is called an alternative set of axioms for T. Every theory admits of infinitely many alternative axiomatizations-including the trivial axiomatization, in which every member of T is taken as an axiom (i.e., $S = T$).

3. A set S is called recursive if and only if it is decidable-i.e., there exists an effective procedure for telling whether or not an arbitrary wff belongs to S. [...] For "effective procedure" one can also write "Turing machine".

   A theory is recursively axiomatizable (often simply "axiomatizable", in the literature) if it has at least one set of axioms that is recursive. Every finite set is recursive; thus all theories that can be finitely axiomatized are recursively axiomatizable. An example of a theory that can be recursively axiomatized but not finitely axiomatized is Peano arithmetic[1]

   Although Peano would have considered this a single "axiom", to write it down we have to write down an infinite set of wffs, one instance of 1 (page 5) for each wff S, that can be built up out of the symbols 0, E, S, T, F and logical symbols. Thus Peano arithmetic has an infinite set of axioms (and it has been proved that no finite alternative set of axioms exists). However, the usual set of axioms is recursive. To decide whether or not a wff is an axiom we see if it is one of the axioms that are not of the form 1 (one of seven axioms...), and, if it is not, we then see whether or not the wff in question has the form 1 (the "axiom" of mathematical induction; This can be effectively decided. Thus theories with an infinite set of axioms play an important role in actual mathematics; however, it is always required in practice that the set of axioms be recursive. For, if

[1]See the description at the end of this section, page 5.

there were no procedure for telling whether or not a wff was an axiom, then we could not tell whether or not an arbitrary sequence of wffs was a proof!

4. A set is recursively enumerable if the members of the set are also the elements S_1, S_2, S_3,... of some sequence that can be effectively produced (i.e., produced by a Turing machine that is allowed to go on "spinning out" the sequence forever). [...]

   The set of theorems of T, where T is any recursively axiomatized theory, is also a recursively enumerable set. This can be shown by arranging all the proofs in T in an effectively produced sequence (say, in order of increasing number of symbols). If one replaces the ith proof in the resulting sequence Proof_1, Proof_2,... by the wff that is proved by Proof_i, one obtains a listing of all and only the theorems of T (with infinitely many repetitions, of course – however, these can be deleted if one wishes).

Note. An axiomatization of Peano arithmetic

The primitive predicates are $E(x,y)$ (also written $x = y$), $S(x,y,z)$ (also written $x+y = z$), $T(x,y,z)$ (also written $x \cdot y = z$ or $xy = z$), and $F(x,y)$ (also written $y = x'$ or $y = x + 1$).

The axioms are Peano axioms for number theory plus the four formulas that recursively define addition and multiplication which, in slighly abbreviated notation, are:

$$\forall x : x + 0 = x, \quad \forall x \, \forall y : x + y' = (x + y)',$$
$$\forall x : x \cdot 0 = 0, \quad \forall x \, \forall y : x \cdot y' = xy + x$$

The "axiom" of mathematical induction says

$$[S_0 \, \wedge \, \forall x \forall y : (S_x \wedge (y = x') \Rightarrow S_y)] \Rightarrow \forall x : S_x \tag{1}$$

where S_x is any wff not containing 'y', S_y contains 'y' where and only where S_x contains free 'x', and S_0 contains the individual constant '0' whenever S_x contains free 'x'.

2.3 Proof of Craig's observation

Is every recursively enumerable set recursive? According to a fundamental theorem of recursive-function theory, the answer is "no". There is a recursively enumerable set D of positive integers that is not recursive. In other words, there is a sequence a_l, a_2,... of numbers that can be effectively continued as long as we wish, but such that there is no method in principle that will always tell whether or not an arbitrary integer eventually occurs in the sequence.

The set of theorems of quantification theory (first-order logic) is another example of a recursively enumerable non-recursive set. The theorems can be effectively produced in a single infinite sequence; but there does not exist in principle an algorithm by means of which one can tell in a finite number of steps whether a wff will or will not eventually occur in the sequence – i.e., the "decision problem" for pure logic is not solvable.

We now see what the observation (I), see page 4, comes to. It says that all recursively enumerable theories can be recursively axiomatized. If the theory T is recursively enumerable (this is equivalent to having a recursively enumerable set of axioms), then a recursive set S can be found which is a set of axioms for T.

Craig's Proof of observation (I). Craig's proof of (I) is so remarkably simple that we shall give it in full.

Proof. Let T be a theory with a recursively enumerable set S of axioms, and let an effectively produced sequence consisting of these axioms be S_1, S_2,... We shall construct a new set S' which is an alternative set of axioms for T. Namely, for each positive integer i, S' contains the wff $S_i \wedge (S_i \wedge (\ldots))$, with i conjuncts S_i.

Clearly, each S_i can be deduced from the corresponding axiom in S' by the rule $A \wedge B$ implies A. Also, each axiom is S' can be deduced from the corresponding S_i by repeated use of the rules: A implies $A \wedge A$, and A, B imply $A \wedge B$. It remains to show that S' is recursive.

Let A be a wff, and consider the problem of deciding whether or not A belongs to S'. Clearly, if $A \neq S_1$ and A is not of the form $(B \wedge (\ldots))$, A is not in S'. If A is of the form $(B \wedge (\ldots))$ with k B's, then A belongs to S' if and only if $B = S_k$. So we just continue the sequence S_1, S_2,... until we get to S_k and compare B with S_k. If $B = S_k$, A is in S'; otherwise A is not in S'. The proof is complete! □

Notice that, although we have given a method for deciding whether or not a wff A is in S', we still have no method for deciding whether or not an arbitrary wff C is in S, even though S and S' are trivially equivalent sets of sentences, logically speaking. For we don't know how long an initial segment $S_1, \ldots S_k$ we must produce before we can say that if C is not in the segment it is not in S at all. The fact that S' is decidable, even if S is not, constitutes an extremely instructive example.

Note. In the above proof the axioms of S' contain information about a specific Turing machine (TM) that enumerates the axioms of S. There are infinitely ways of enumerating them, depending on the order in which they are printed by M. So, to define an axiom of S' we need a particular enumerating TM... and perhaps a proof that this machine enumerates in fact the axioms of S'. $\square$

2.4 Craig's Theorem about predicate letters

We now state and prove another theorem from Craig. Essentially it states that in a theory T we can select any set V_B of predicate letters used in T and recursively axiomatize those formulas of T that only mention the letters in V_B.

Theorem 3 (Craig's Theorem about predicate letters) *Let T be a recursively enumerable theory, and consider any division of the predicate letters of T into two disjoint sets, say $V_A = \{T_1, T_2, \ldots\}$ and $V_B = \{O_1, O_2, \ldots\}$. Let T_B consist of those theorems of T which contain only predicate letters from V_B. Then T_B is a recursively axiomatizable theory.*

Proof. Let S_1, S_2,... be an effectively produced sequence consisting of the theorems of T. By leaving out all wffs which are not in the sub-vocabulary V_B, we obtain the members of T_B, say as V_1, V_2,... Thus T_B is a recursively enumerable theory, and possesses a recursively enumerable axiomatization (take T_B itself as the set S of axioms). Then, by observation (I), T_B is recursively axiomatizable. $\square$

The reader will observe that the proof assumes that the sets of predicate letters V_A and V_B are themselves recursive; strictly speaking we should have stated this. In practice these sets are usually finite sets, and thus trivially recursive.

Some philosophical implications of this result are discussed in [5].

3 Mainly from the Oxford Dictionary of Philosophy ([1])

We now transcribe the entry "Craig's Theorem" in the Oxford Dictionary of Philosophy, "Craig's theorem: a theorem in mathematical logic, held to have implications in the philosophy of science".

The logician William Craig at Berkeley showed how, if we partition the vocabulary of a formal system (say, into the T or theoretical terms, and the O or observational terms), then if there is a fully formalized system T with some set S of consequences containing only O terms, there is also a system O containing only the O vocabulary but strong enough to give the same set S of consequences.

The theorem is a purely formal one, in that T and O simply separate formulae into the preferred ones, containing as non-logical terms only one kind of vocabulary, and the others. The theorem might encourage the thought that the theoretical terms of a scientific theory are in principle dispensable, since the same consequences can be derived without them.

However, Craig's actual procedure gives no effective way of dispensing with theoretical terms in advance, i.e. in the actual process of thinking about and designing the premises from which the set S follows. In this sense O remains parasitic upon its parent T.

3.1 Recursive axiomatization

Let A_1, A_2,... be an enumeration of the axioms of a recursively enumerable set T of first-order formulas. Construct another set $T^\star$ consisting of

$$\underbrace{A_i \wedge \ldots \wedge A_i}_{i} \tag{2}$$

for each positive integer i. The deductive closures of $T^\star$ and T are thus equivalent; the proof will show that $T^\star$ is a decidable set. A decision procedure for $T^\star$ lends itself according to the following informal reasoning. Each member of $T^\star$ is either A_1 or of the form

$$\underbrace{B_j \wedge \ldots \wedge B_j}_{j}.$$

Since each formula has finite length, it is checkable whether or not it is A_1 or of the said form. If it is of the said form and consists of j conjuncts, it is in $T^\star$ if it is the expression A_j; otherwise it is not in $T^\star$. Again, it is checkable whether it is in fact A_n by going through the enumeration of the axioms of T and then checking symbol-for-symbol whether the expressions are identical.

Slight reformulation

Let B_1, B_2,... be an enumeration of the axioms of a recursively enumerable set $B = \{B_1, B_2 \ldots\}$ of axioms. The closure of B under the binary deduction relation R is denoted by $C = B^R$, the theory (set of theorems) generated by B.

Construct another set of axioms

$$A = \{\underbrace{B_i \wedge \ldots \wedge B_i}_{i} \mid i = 1, 2 \ldots\} = \{B_1,\ B_2 \wedge B_2,\ B_3 \wedge B_3 \wedge B_3, \ldots\}$$

The deductive closures of B and A (under R) are thus equivalent.

We now show that A is a decidable (recursive) set. A decision procedure for A is

> Input: first-order formula F.
> Output: 1 if $F \in A$, 0 otherwise
>   enumerate the first element B_1 of B.
>   if $F = B_1$ output 1 and stop.
>   check if F is the conjunction of $i \geq 2$ equal formulas H.
>   if it is not, output 0 and stop.
>   if it is:
>   enumerate the ith element B_i of B.
>   if $F = B_i$ output 1 else output 0

3.2 Primitive recursive axiomatizations

The proof above shows that for each recursively enumerable set of axioms there is a recursive set of axioms with the same deductive closure. A set of axioms is primitive recursive if there is a primitive recursive function that decides membership in the set. To obtain a primitive recursive axiomatization, instead of replacing a formula A_i with

$$\underbrace{A_i \wedge \ldots \wedge A_i}_{i} \tag{3}$$

one instead replaces it with

$$\underbrace{A_i \wedge \ldots \wedge A_i}_{f(i)} \tag{4}$$

where $f(x)$ is a function that, given i, returns [an integer that encodes] a computation history showing that A_i is in the original recursively enumerable set of axioms. It is possible for a primitive recursive function to parse an expression of form (4) to obtain A_i and j. Then, because Kleene's T predicate is primitive recursive, it is possible for a primitive recursive function to verify that j is indeed a computation history as required.

4 Mainly from Craig, "On axiomatizability within a system" [2]

This is the more general and abstract section. Many of the previous results are consequences of Theorem 4 below. This recursion theoretic result is not about Logic; it's about sets, binary relations and closures.

4.1 An "observation"...

A set X is closed under a (binary) relation R if $x \in X$ and $R(x, y)$ implies $y \in X$. This property could be called "forward closure" and we could similarly define "backward closure" by the property $x \in X$ and $R(y, x)$ implies $y \in X$; however, we will not use other forms of closure. The closure of the set X under the relation R is the smallest set Y that contains X and is closed under R. Such set always exist. We denote it by X^R. It is

$$X^R = \bigcap_Y [Y \supseteq X] \wedge [Y \text{ is closed under } R] \tag{5}$$

Theorem 4 Suppose that

1. R is a (binary) relation.
2. B is a recursively enumerable set. Let $C = B^{\vdash s}$
3. There is a primitive recursive relation Q such that (i) Q is a symmetric sub-relation[a] of R and (ii): $\forall m \in B,\ \exists^\infty n : Q(m, n)$ where "$\exists^\infty$" means "there exist infinitely many".

Then there is a primitive recursive set A, such that $A^R = B^R = C$. □

[a] That is, $Q \subseteq R$ and $\forall m, n : Q(m, n) \Rightarrow Q(n, m)$.

That is, under the conditions stated, there is a primitive recursive set A that generates the closure of B under R. In the corollaries below (pages 11–13) we will typically have that $R(F, G)$ is a deduction relation (that is, $F \vdash G$) and B will be a set of axioms; in this case the theorem states that there is a primitive recursive set of axioms from which the same consequences can be deduced. *These corollaries of Theorem 4 contain in a condensed form the results mentioned in Sections 2 (page 3, based on [5]) and 3 (page 7, based on [1]).*

Proof. It is well known that a non-empty set is recursively enumerable if and only it is the range of a primitive recursive function. Let f be a primitive recursive function that enumerates B. Define

$$A = \{n \mid \exists p : (p \le n) \wedge (Q(f(p), n))\}$$

We show that A has the desired properties. Using the definition of A and property 3.(ii) above, we see that for every $m \in B$ there is a $n \in A$ (in fact there are infinitely many such n's) such that $Q(m,n)$ and thus $Q(n,m)$. Therefore $m \in A^Q$ and thus $B \subseteq A^Q \subseteq A^R$, which, given the property (5), implies $B^R \subseteq A^R$.
Conversely, for every $n \in A$ there is a $m \in B$ such that $Q(m,n)$. Thus $A \subseteq B^Q$ and, since Q is a sub-relation of R, $A \subseteq B^R$. From (5) it follows that $A^R \subseteq B^R$. Thus $A^R = B^R$.
Finally, that A is primitive recursive follows from [3], page 80; for an alternative justification consider the following LOOP schematic program[2] $P(n)$ that outputs 1 if $n \in A$ and 0 otherwise.

```
for p=0 to n:
  m ← f(p) // f(p) is a primitive recursive function
  if Q(m,n): // Q(m,n) is a primitive recursive relation
    output 1 else output 0
```

□

4.2 Corollary: application to formal systems

Theorem 4 (page 10) can be applied to many formal systems by letting R correspond to the relation of deducibility in S, that is, $R(m,n)$ iff m is the Gödel number of a formula or sequence of formulas of S from which, together with the axioms of S, a formula with the Gödel number n can be obtained by application of the rules of inference of S. In symbols,

$$R(m,n) \;\;\Leftrightarrow\;\; \ulcorner F \urcorner = m,\;\; \ulcorner G \urcorner = n,\; F \vdash_S G \tag{6}$$

where F and G are formulas or sequences of formulas of S.
Consider a system S in which, if F is a formula and $F \wedge \ldots \wedge F$ is a conjunction from an arbitrary number of occurrences of F, then F and $F \wedge \ldots \wedge F$ are deducible from each other. In symbols, $F \vdash_S F \wedge \ldots \wedge F$ and $F \wedge \ldots \wedge F \vdash_S F$.

Note. Other properties of S could have been chosen, for instance, that any formula F of S has infinitely many variants G such that F and G are deducible in S from one another, two formulas being variants if they are obtainable from one another by one or more substitutions thorough one variable by another. □

Let $Q(m,n)$ be the primitive recursive relation which holds between m and n

[2]LOOP programs compute exactly the primitive recursive functions, [4].

iff, for some formula of S, $\ulcorner F \urcorner = m$ and $\ulcorner F \wedge \ldots \wedge F \urcorner = n$ or vice versa. In symbols,

$$Q(m,n) \;\Leftrightarrow\; \exists F \;:\; \ulcorner F \urcorner = m \text{ and } \ulcorner F \wedge \ldots \wedge F \urcorner = n \text{ or vice versa} \qquad (7)$$

Consider any recursively enumerable set B and let $C = B^{\vdash_S}$ be the set of formulas deducible in S from B. We apply Theorem 4 (page 10) with the correspondences:

$$\begin{array}{lcl} \mathbb{N} & \leftrightarrow & \text{Gödel numbers of } S\text{-formulas} \\ R & \leftrightarrow & \text{see (6) above} \\ Q & \leftrightarrow & \text{see (7) above} \end{array}$$

Then from Theorem 4 we conclude that there is a primitive recursive set A of formulas of S, such that, if A is considered as the axiom set of S, then the theorems of the resulting system constitute $B^{\vdash_S}$, or, in symbols, $A^{\vdash_S} = B^{\vdash_S}$. The set of theorems $C = B^{\vdash_S}$, that is, those that are originally deducible in S, is then called a *primitive recursive axiomatizable in* S, meaning that, under the assumptions mentioned above, we can replace the recursively enumerable set of axioms B by a primitive recursive set of axioms A.

4.3 Corollary: primitive recursive axiomatization

As a particular case of the general application of Theorem 4 to formal systems (Section 4.2, page 11), suppose that C is the set of formulas of S that are theorems of a system T, where T contains all the axioms and rules of inference of S and also other formulas or rules of inference. Then, provided the set of formulas of S and the set of theorems of T are recursively enumerable, C can be primitive recursively axiomatized in S, and hence formalized without the aid of additional formulas or rules of inference. Indeed C can be formalized with the only rule of inference $F \wedge \ldots \wedge F \to F$.

Example. If T is a system employing higher type quantifiers which expresses an analytic theory of numbers, then there exists a system which expresses the corresponding elementary theory of numbers, its theorems being those theorems of T which contain no higher type quantifiers.

4.4 A corollary (apparently) with philosophical interest

The following corollary of Theorem 4, page 10, has already been independently mentioned and proved above, see Theorem 3, page 7. Philosophers seem to have found this interesting, see for example [5].

If K is any recursive set of non-logical (individual, function, predicate) constants, containing at least one predicate constant, then there exists a system whose theorems are exactly those theorems of T in which no constant, other than those of K occur.

In particular, suppose that T expresses a portion of a natural science, that the constants of K refer to things or events regarded as "observable", and that the other constants do not refer to "observables" and hence may be regarded as "theoretical" or "auxiliary". Then there exists a system which does not employ "theoretical" or "auxiliary" constants and whose theorems are the theorems of T concerning observables.

4.5 Corollary: no additional formulas or rules of inference needed

Suppose that S is completable and hence that there exists a complete and consistent system T whose set of theorems is recursively enumerable and whose axioms and rules of inference include those of S.

Then, provided the set of formulas of S is recursively enumerable, S can be completed without the use of additional formulas or rules of inference.

4.6 Kleene-Mostowski hierarchy: generalization of Theorem 4

Consider the Kleene-Mostowski hierarchy and let "$\{\eta_1\}$", "$\{\eta_2\}$", "$\{\eta_3\}$"... stand for $\forall k_1$", $\forall k_1 \, \exists k_2$", $\forall k_1 \, \exists k_2 \, \forall k_3$",... and "$\eta_1$", "$\eta_2$", "$\eta_3$"... for "$k_1$", "$k_2$", "$k_3$"...

Suppose that C, B, R and Q are as described earlier, except that B and Q, instead of being recursively enumerable and primitive recursive respectively, are $\{m | \exists p \, \{\eta_r\} P(m, p, \eta_r)\}$ and $\{m, n | \{\eta_r\} N(m, n, \eta_r)\}$, respectively for some $r \geq 1$ and some recursive P and N. Now, let

$$A = \{n \mid \exists m \, \exists p : [m \leq n \wedge p \leq n \wedge \{\eta_r\} P(m, p, \eta_r) \{\eta_r\} N(m, n, \eta_r)]\}$$

Then C is the closure under R of A.

5 General comments

[In construction] The new axioms are dependent on the enumerating Turing machine that is chosen to enumerate the old axioms. See for instance 3.1 (page 8), where the number of times that A_i, the ith formula printed by M, occurs in the new axiom is i, and in 4 (page 9), where the number of times that B_i occurs specifies (is an encoding of) the computation history of M until B_i is printed. Thus, in these constructions of one new axiomata from another,

– The new axioms may contain a lot of information about the enumeration of the old axioms.

– That information (and the new axioms) are highly dependent on the Turing machine that is selected (or constructed) for the enumeration of the initial set of axioms.

In my view these properties are undesirable.

References

[1] Simon Blackburn. *The Oxford Dictionary of Philosophy.* Oxford University Press, 2008. "Craig's Theorem" entry.

[2] William Craig. On axiomatizability within a system. *Journal of Symbolic Logic*, 18(1):30–32, 1953.

[3] Kurt Gödel. *On Formally Undecidable Propositions of Principia Mathematica and Related Systems.* Dover Books on Mathematics Series. Dover Publications, 1992.

[4] A. R. Meyer and D. M. Ritchie. The complexity of loop programs. *Proceedings of 22nd National Conference of the ACM*, pages 465–469, 1967.

[5] Hilary Putnam. Craig's theorem. *The Journal of Philosophy*, 62(10):251–260, 1965. Reprinted in *Mathematics, Matter and Method*, 1975.