Turing versus Gödel on Computability and the Mind

B. Jack Copeland and Oron Shagrir

Nowadays the work of Alan Turing and Kurt Gödel rightly takes center stage in discussions of the relationships between computability and the mind. After decades of comparative neglect, Turing's 1936 paper "On Computable Numbers" is now regarded as the foundation stone of computability theory, and it is the *fons et origo* of the concept of computability employed in modern theoretical computer science. Moreover, Turing's 1950 essay, "Computing Machinery and Intelligence," sparked a rich literature on the mind-machine issue. Gödel's 1931 incompleteness results triggered, on the one hand, precise definitions of effective computability and its allied notions, and on the other, some much-criticized arguments for the conclusion that the mathematical power of the mind exceeds the limitations of Turing machines. Gödel himself is widely believed to have held that minds are more powerful than machines, whereas Turing is usually said to have taken the opposite position. In fact, neither of these characterizations is much more than a caricature. The actual picture is subtle and complex. To complicate matters still further, Gödel repeatedly praised Turing's analysis of computability, and yet in later life he accused Turing of fallaciously assuming, in the course of this analysis, that mental procedures cannot go beyond effective procedures. How can Turing's analysis be "unquestionably adequate" (Gödel 1964, 71) and yet involve a fallacy?

We will present fresh interpretations of the positions of Turing and Gödel on computability and the mind. We argue that, contrary to first impressions, their views about computability are closer than might appear to be the case; and we will also argue that their views about the mind-machine issue are closer than Gödel—and others—have believed.[1] In section1.1, we show that Gödel's attribution of philosophical error to Turing is baseless; and we present a revisionary account of Turing's position regarding (what Gödel called) Hilbert's "rationalistic attitude." In section 1.2, we distinguish between two approaches to the analysis of computability, the cognitive and the noncognitive. We argue that Gödel pursued the noncognitive approach. As we will explain, we believe that Gödel mistook some cognitivist-style rhetoric in Turing's 1936 paper for an endorsement of the claim that the mind is computable. In section 1.3, we suggest that Turing held what we call the *Multi-Machine theory of mind*, according to which mental processes, when taken diachronically, form a finite procedure that need not be *mechanical*, in the technical sense of that term (in which it means the same as "effective").

1 Gödel on Turing's "Philosophical Error"

In about 1970, Gödel wrote a brief note entitled "A Philosophical Error in Turing's Work" (1972).[2] The note was, he said, to be regarded as a footnote to the postscript, which he had composed in 1964, to his 1934 undecidability paper. The main purpose of the 1964 postscript was to state generalized versions of incompleteness, applicable to algorithms and formal systems. It was in this postscript that Gödel officially adopted Turing's "analysis of the concept of 'mechanical procedure'… (alias 'algorithm' or 'computation procedure')" (1964, 72); and he there emphasized that it was "due to A. M. Turing's work, [that] a precise and unquestionably adequate definition of the general concept of formal system can now be given … [A] formal system can simply be defined to be any mechanical procedure for producing formulas, called provable formulas" (71–72). In the postscript, Gödel also raised the intriguing "question of whether there exist finite *non-mechanical* procedures" (72); and he observed that the generalized incompleteness results "do not establish any bounds for the powers of human reason, but rather for the potentialities of pure formalism in mathematics" (73).

Gödel's retrospective footnote to his 1964 postscript attributed the view that "mental procedures cannot go beyond mechanical procedures" to Turing's"On Computable Numbers." Gödel criticized an argument for this view that he claimed to find there:

A philosophical error in Turing's work. Turing in … [section 9 of "On Computable Numbers" (1936, 75–76)] gives an argument which is supposed to show that mental procedures cannot go beyond mechanical procedures. However … [w]hat Turing disregards completely is the fact that *mind, in its use, is not static, but constantly developing*… [A]lthough at each stage the number and precision of the abstract terms at our disposal may be *finite*, both (and, therefore, also Turing's number of *distinguishable states of mind*) may *converge toward infinity* in the course of the application of the procedure. (Gödel 1972, 306)

Gödel later gave Hao Wang a different version of the same note, in which Gödel says explicitly that Turing's argument involves "the supposition that a finite mind is capable of only a finite number of distinguishable states." The later version runs:

Turing … [in "On Computable Numbers"] gives an argument which is supposed to show that mental procedures cannot carry any farther than mechanical procedures. However, this argument is inconclusive, because it depends on the supposition that a finite mind is capable of only a finite number of distinguishable states. What Turing disregards completely is the fact that *mind, in its use, is not static, but constantly developing*. This is seen, e.g., from the infinite series of ever stronger axioms of infinity in set theory, each of which expresses a new idea or insight. … Therefore, although at each stage of the mind's development the number of its possible states is finite, there is no reason why this number should not converge to infinity in the course of its development. (Gödel in Wang 1974, 325)

However, Turing can readily be defended against Gödel's charge of philosophical error. In what follows we will show that it is far from the case that "Turing disregards completely … the fact that *mind, in its use, is not static, but constantly developing*." Gödel's blunt criticism of Turing is entirely misdirected.

In fact, we will argue in part 3 that the dynamic aspect of mind emphasized here by Gödel lies at the very center of Turing's account.

Gödel was too hasty in his claim that, in "On Computable Numbers," Turing put forward an argument supposed to show that mental procedures cannot go beyond mechanical procedures. There is no such argument to be found in Turing's paper; nor is there even any trace of a statement endorsing the conclusion of the supposed argument. Turing, on the page discussed by Gödel, was not talking about the general scope of mental procedures; he was addressing a different question, namely,"What are the possible processes which can be carried out in computing a number?"[3] Furthermore, there is a passage in "On Computable Numbers" that seemingly runs counter to the view attributed to Turing by Gödel. Having defined a certain infinite binary sequence δ, which he shows to be uncomputable, Turing says: "It is (so far as we know at present) possible that any assigned number of figures of δ can be calculated, but not by a uniform process. When sufficiently many figures of δ have been calculated, an essentially new method is necessary in order to obtain more figures" (1936, 79). This is an interesting passage. Turing is envisaging the possibility that the human mathematician can calculate any desired number of digits of an uncomputable sequence by virtue of creating new methods when necessary. Gödel, on the other hand, considered Turing to have offered an "alleged proof that every mental procedure for producing an infinite series of integers is equivalent to a mechanical procedure."[4]

Even without focusing on the detail of Turing's views on mind (which emerged in his post-1936 work), attention to what Turing actually said in his 1936 paper is sufficient to show that Gödel's criticism of Turing bears at best a tenuous relation to the text Gödel was supposedly discussing. Turing did not say that "a finite mind is capable of only a finite number of distinguishable states." He said: "We will … suppose that the number of states of mind which need be taken into account [for the purpose of analyzing computability] is finite" (75). He immediately acknowledges that beyond these there may be "more complicated states of mind," but points out that, again for the purpose of analyzing computability, reference to these more complicated states "can be avoided by writing more symbols on the tape" (76). Turing is in effect distinguishing between elementary states of mind and complex states of mind, and is noting that symbols on the tape can serve as a surrogate for complex states of mind. Turing nowhere suggests that the "more complicated states of mind" are finite in number (and nor does he suggest that they fail to be distinguishable unless finite in number).

1.1 Turing on Mathematical Intuition

In short, Turing's 1936 text does not support Gödel's interpretation. The situation becomes bleaker still for Gödel's interpretation when Turing's 1939 publication "Systems of Logic Based on Ordinals" is taken into account. There Turing emphasized the aspect of mathematical reasoning that he referred to as "intuition." He said:

> In pre-Gödel times it was thought by some that … all the intuitive judgments of mathematics could be replaced by a finite number of … [formal] rules. … In consequence of the impossibility of finding a formal logic which wholly eliminates the necessity of using intuition, we naturally turn to "non-constructive" systems of logic with which not all the steps in a proof are mechanical, some being intuitive. (Turing 1939, 192–193)

In Turing's view, the activity of what he called the *faculty* of intuition brings it about that mathematical *judgments*—again, his word—exceed what can be expressed by means of a single formal system (192). "The activity of the intuition," he said, "consists in making spontaneous judgments which are not the result of conscious trains of reasoning" (192). Turing's cheerful use of mentalistic vocabulary in this connection makes it very unlikely that Gödel was correct in finding an argument in the 1936 paper supposedly showing that mental procedures cannot go beyond mechanical procedures.

During the early part of the war, probably in 1940, Turing wrote a number of letters to Max Newman explaining his thinking about intuition. The following passage is illuminating:

> I think you take a much more radically Hilbertian attitude about mathematics than I do. You say "If all this whole formal outfit is not about finding proofs which can be checked on a machine it's difficult to know what it is about." When you say "on a machine" do you have in mind that there is (or should be or could be, but has not been actually described anywhere) some fixed machine on which proofs are to be checked, and that the formal outfit is, as it were about this machine. If you take this attitude (and it is this one that seems to me so extreme Hilbertian [sic]) there is little more to be said: we simply have to get used to the technique of this machine and resign ourselves to the fact that there are some problems to which we can never get the answer. On these lines my ordinal logics would make no sense. However I don't think you really hold quite this attitude because you admit that in the case of the Gödel example one can decide that the formula is true i.e. you admit that there is a fairly definite idea of a true formula which is quite different from the idea of a provable one. Throughout my paper on ordinal logics I have been assuming this too. … If you think of various machines I don't see your difficulty. One imagines different machines allowing different sets of proofs, and by choosing a suitable machine one can approximate "truth" by "provability" better than with a less suitable machine, and can in a sense approximate it as well as you please. The choice of a … machine involves intuition … (Turing to Newman, ca. 1940b, 215)

The picture described in this letter will be called Turing's *Multi-Machine* picture of mathematics. In this picture, the role of intuition is localized very precisely. Intuition is responsible for the selection of the appropriate theorem-proving machine (the appropriate Turing machine), and the rest is mechanical. The intuition involved in selecting the appropriate theorem-proving machine is, Turing said, “interchangeable” with the intuition involved in finding a proof of the theorem.

1.2 Gödel and Turing on Rationalistic Optimism

“Rationalistic optimism” is the view that there are no mathematical questions that the human mind is incapable of settling, in principle at any rate, even if this is not so in practice (due, say, to the occurrence of the heat-death of the universe).[5] In a striking observation about the implications of his incompleteness result, Gödel said:

> My incompleteness theorem makes it likely that mind is not mechanical, or else mind cannot understand its own mechanism. If my result is taken together with the rationalistic attitude which Hilbert had and which was not refuted by my results, then [we can infer] the sharp result that mind is not mechanical. This is so, because, if the mind were a machine, there would, contrary to this rationalistic attitude, exist number-theoretic questions undecidable for the human mind. (Gödel in Wang 1996, 186–187)

What Gödel calls Hilbert’s “rationalistic attitude” was summed up in Hilbert’s celebrated remark that “in mathematics there is no *ignorabimus*”—no mathematical question that in principle the mind is incapable of settling (Hilbert 1902, 445). Gödel gave no clear indication whether, or to what extent, he himself agreed with what he called Hilbert’s “rationalistic attitude” (a point to which we shall return in section1.3). On the other hand, Turing’s criticism (in his letter to Newman) of the “extreme Hilbertian” view is accompanied by what seems to be a cautious endorsement of the rationalistic attitude. The "sharp result" stated by Gödel seems in effect to be that there is no *single* machine equivalent to the mind (at any rate, no more is justified by the reasoning that Gödel presented)—and with this Turing was in agreement, as his letter makes clear. Incompleteness, if taken together with a Hilbertian optimism, excludes the extreme Hilbertian position that the "whole formal outfit" corresponds to some one fixed machine.

Turing's view, as he expressed it to Newman and in "Systems of Logic Based on Ordinals," appears to have been that mathematicians achieve progressive approximations to truth via a non-mechanical process involving intuition. This picture, in which minds devise and adopt successive, increasingly powerful mechanical formalisms in their quest for truth, is consonant with Gödel's view that "mind, in its use, is not static, but constantly developing." Gödel's own illustration of his claim that mind is constantly developing is certainly related to Turing's concerns. Gödel said: "This [that mind is not static but constantly developing] is seen, e.g., from the infinite series of ever stronger axioms of infinity in set theory, each of which expresses a new idea or insight" (Gödel in Wang 1974, 325).

So the two great founders of the study of computability were perhaps not quite as philosophically distant on the mind-machine issue as Gödel supposed. We shall have more to say about their views on this issue in section 1.3. But first, let us look at what these founding fathers thought about the concept of computability itself. Gödel repeatedly praised Turing's analysis of computability, saying it produces a "correct and unique" definition of "the concept of mechanical" in terms of "the sharp concept of 'performable by a Turing machine'" (Gödel in Wang 1974, 84).[6] Yet Turing's analysis appears in the very same passages of his 1936 paper in which Gödel thought he found "an argument which is supposed to show that mental procedures cannot go beyond mechanical procedures" (Gödel ca. 1972, 306). How could Gödel praise Turing's analysis while at the same time rejecting what seems to be a key element in it, namely the constraint of a fixed bound on the number of internal states that a computer can be in? Our answer to this question will illuminate Gödel's reasons for thinking that in the course of his analysis of computability Turing proposed an argument about minds and machines.

2 Two Approaches to the Analysis of Computability

We will start with a brief summary of Turing's analysis of computability and will then describe Gödel's reaction to it in more detail. Against that background, we will distinguish between two approaches to the analysis of computability, which we call the *cognitive* and *noncognitive* approaches, respectively. We will then explain where Gödel, Turing, and Kleene stand vis-à-vis this distinction, especially with respect to the boundedness constraint on the number of states of mind. We argue that the distinction sheds light on the puzzle of how Gödel took the very same passages in Turing to provide both an erroneous philosophical argument about the limits of the mind *and* a unique and correct definition of computability.

2.1 Preamble: Turing's Analysis of Computability

Turing's 1936 analysis of computability has been explicated by Kleene (1952, 376–381)—who unfortunately misdated the analysis to 1937—and by Gandy (1988). Gandy's explication has been further developed by Sieg (1994, 2002). A key point, often misunderstood, is that Turing's "computability" concerns calculations by an ideal *human*, a human computer. Turing, as Gandy said, "makes no reference whatsoever to calculating machines" (1988, 77).[7]

In section 9 of his paper, Turing presents three arguments that his analysis catches everything that, as he put it, "would naturally be regarded as computable" (1936, 74). The first argument can be set out like this (Shagrir 2002):

Premise 1 ("the central thesis") A human computer operates under the restrictive conditions 1–5 (below).

Premise 2 ("Turing's theorem") Any function that can be computed by a computer operating under conditions 1–5 is Turing-machine computable.

Conclusion ("Turing's thesis") Any function that can be computed by a human computer is Turing-machine computable.

Turing calls this his "Type (a)" argument (1936, 74–77). He enumerates the five restrictive conditions somewhat informally. The first concerns the deterministic relationship between the computation steps:

1. "The behavior of the computer at any moment is determined by the symbols which he is observing, and his 'state of mind' at that moment" (75).

Turing then formulates boundedness conditions on each of the two determining factors, namely, the observed symbols and states of mind:

2. "There is a bound *B* to the number of symbols or squares which the computer can observe at one moment" (75).
3. "The number of states of mind which need be taken into account is finite" (75).

There are three "simple operations" (behaviors) that the computer may perform at each moment: a change in the symbols written on the tape, a change of the observed squares, and a change of state of mind. Turing gives additional boundedness conditions on the first and second type of operation (the third having already been dealt with):

4. "We may suppose that in a simple operation not more than one symbol is altered" (76).

5. “[E]ach of the new observed squares is within L squares of an immediately previously observed square” (76).

The second premise of the argument is a *reduction theorem* stating that *any* system operating under conditions 1–5 is bounded by Turing machine computability. Turing provides an outline of the proof (77); a more detailed demonstration is given by Kleene (1952). Gandy (1980) proves the theorem with respect to “Gandy machines,” which operate under more relaxed restrictions.[8]

2.2 Gödel on Computability

Gödel’s rather sparse statements on computability are now well documented.[9]We provide an overview of his thoughts on the subject. Gödel’s interest in a precise definition of computability stemmed from the incompleteness results. A precise definition is required for understanding not only the *philosophical implications* of the incompleteness results but also, first and foremost, for establishing the *generality* of the results. As the title of Gödel’s paper (1931) noted, the incompleteness results apply in their original forms to “Principia Mathematica and related systems.” More precisely, they apply to the formal system *P*, which is “essentially the system obtained when the logic of *PM* is superposed upon the Peano axioms” (1931, 151), and to the extensions of *P* that are the “ω-consistent systems that result from *P* when [primitive] recursively definable classes of axioms are added" (185, note 53). But it was still an open question whether there exist extensions of *P* whose class of theorems is effectively but not recursively enumerable. The precise definitions of computability that emerged later secured the generality of Gödel's incompleteness results. As Gödel put it in the 1964 *Postscriptum*, the precise definitions of computability imply the general definition of a formal system; hence "the existence of undecidable arithmetical propositions and the non-demonstrability of the consistency of a system in the same system can now be proved rigorously for *every* consistent formal system containing a certain amount of finitary number theory" (1964, 71).

As Gödel explained it, a formal system is governed by what we now call an effective procedure. Gödel did not use the term "effective" himself; he characterized the governing procedure as a *mechanical and finite* one. The property of *being mechanical* is spelled out in Gödel's 1933 address to the Mathematical Association of America (entitled "The Present Situation in the Foundations of Mathematics"). He opened with a rough characterization of formal systems, pointing out that the "outstanding feature of the rules of inference [is] that they are purely formal, i.e., refer only to the outward structure of the formulas, not to their meaning, so that they could be applied by someone who knew nothing about mathematics, or by a machine" (45). Gödel's reference to machines signals his fascination with calculating machines,[10] but also implies that

Gödel was primarily thinking of humans—even "someone who knew nothing about mathematics"—as the ones who proceed mechanically. He discussed the property of *finiteness* in his 1934 Princeton address, where he characterized a "formal mathematical system" (346) as follows:

> We require that the rules of inference, and the definitions of meaningful formulas and axioms, be constructive; that is, for each rule of inference there shall be a finite procedure for determining whether a given formula B is an immediate consequence (by that rule) of given formulas $A_1,\ldots,A_n$, and there shall be a finite procedure for determining whether a given formula A is a meaningful formula or an axiom. (346)[11]

At this point Gödel did not have a precise definition of what can be computed by a finite and mechanical procedure. A statement that seems to be much like the Church-Turing thesis appears in the printed version of Gödel's 1934 Princeton lectures, where he formulates what is generally taken to be the "easy" part of the Church-Turing thesis, namely, that "[primitive] [r]ecursive functions have the important property that, for each given set of values of the arguments, the value of the function can be computed by a finite procedure" (348). In a footnote to this statement, Gödel remarks that "[t]he converse seems to be true if, besides [primitive] recursions … recursions of other forms (e.g., with respect to two variables simultaneously) are admitted [i.e., general recursions]. This cannot be proved, since the notion of finite computation is not defined, but it serves as a heuristic principle" (348, note 3). However, in a letter to Martin Davis (on February 15, 1965) Gödel denied that his 1934 paper anticipated the Church–Turing thesis:

It is not true that footnote 3 is a statement of Church's Thesis. The conjecture stated there only refers to the equivalence of "finite (computation) procedure" and "recursive procedure." However, I was, at the time of these lectures, not at all convinced that my concept of recursion comprises all possible recursions; and in fact the equivalence between my definition and Kleene [1936] is not quite trivial. (Gödel in Davis 1982, 8)

Church, who first met Gödel early in 1934, gave some additional information in a letter to Kleene dated November 29, 1935:

In regard to Gödel and the notions of recursiveness and effective calculability, the history is the following. In discussion with him [*sic*] the notion of lambda-definability, it developed that there was no good definition of effective calculability. My proposal that lambda-definability be taken as a definition of it he regarded as thoroughly unsatisfactory. (Church in Davis 1982, 8)

Gödel's attitude changed not long after. In an unpublished paper dating from about 1938, he wrote:

When I first published my paper about undecidable propositions the result could not be pronounced in this generality, because for the notions of mechanical procedure and of formal system no mathematically satisfactory definition had been given at that time. This gap has since been filled by Herbrand, Church and Turing. (Gödel 193?, 166)[12]

So, just a few years after having rejected Church's proposal, Gödel embraced it, attributing the "mathematically satisfactory definition" of computability to Herbrand, Church, and Turing. Why did Gödel change his mind? Turing's work was clearly a significant factor. Initially, Gödel mentions Turing together with Herbrand and Church, but a few pages later he refers to Turing's work alone as having demonstrated the correctness of the various equivalent mathematical definitions: "[t]hat this really is the correct definition of mechanical computability was established beyond any doubt by Turing," he wrote (193?, 168). More specifically:

> [Turing] has shown that the computable functions defined in this way are exactly those for which you can construct a machine with a finite number of parts which will do the following thing. If you write down any number $n_1, \ldots, n_r$ on a slip of paper and put the slip into the machine and turn the crank, then after a finite number of turns the machine will stop and the value of the function for the argument $n_1, \ldots, n_r$ will be printed on the paper. (193?, 168).

It is hard to tell, though, precisely why Gödel found Turing's definition correct "beyond any doubt." Possibly he regarded the concept of a mechanical and finite procedure as somehow captured by the notion of "a machine with a finite number of parts." Gödel is presumably referring to the reduction of human computability to Turing-machine computability. He does not mention that Turing characterized mechanical and finite procedures in terms of the finiteness conditions 1–5 on *human computation*.

In his 1946 Princeton lecture, Gödel returned to the issue of computability. Referring to Tarski's lecture at the same conference, he said:

> Tarski has stressed in his lecture (and I think justly) the great importance of the concept of general recursiveness (or Turing's computability). It seems to me that this importance is largely due to the fact that with this concept one has for the first time succeeded in giving an absolute definition of an interesting epistemological notion, i.e., one not depending on the formalism chosen. (150)

In referring to computability as an *epistemological* concept Gödel was quite likely thinking of the major epistemological role played by computability in Hilbert's finitistic program, and probably also of the normative role played by computability in logic and mathematics generally from the end of the nineteenth century.[13] The epistemological dimension highlights the tight relationship between formal systems and *human* calculators—it is an (ideal) human who decides, by means of a finite and mechanical procedure, whether a sequence of symbolic configurations is a formal proof or not.[14]

In his Gibbs lecture, Gödel was very explicit in his support for Turing's general approach to computability:

> The greatest improvement was made possible through the precise definition of the concept of finite procedure, which plays a decisive role in these [incompleteness] results. There are several different ways of arriving at such a definition, which, however, all lead to exactly the same concept. The most satisfactory way, in my opinion, is that of reducing the concept of finite procedure to that of a machine with a finite number of parts, as has been done by the British mathematician Turing. (1951, 304–305)

Again, Turing's way of arriving at a definition of the "concept of finite procedure" is "most satisfactory." Its satisfactoriness has something to do with the reduction of the concept of finite procedure to that of "a machine with a finite number of parts." Yet, as before, Gödel said nothing about the reduction itself, nor did he say why he thought it so successful.

In his 1964 postscript, Gödel emphasized the contribution of Turing's definition to the generality of the incompleteness results:

> In consequence of later advances, in particular of the fact that, due to A.M. Turing's work, a precise and unquestionably adequate definition of the general concept of a formal system can now be given, the existence of undecidable arithmetical propositions and the non-demonstrability of the consistency of a system in the same system can now be proved rigorously for *every* consistent formal system containing a certain amount of finitary number theory. Turing's work gives an analysis of the concept of "mechanical procedure" (alias "algorithm" or "computation procedure" or "finite combinatorial procedure"). This concept is shown to be equivalent with that of a "Turing machine." (71–72)

According to Gödel, then, Turing provided a precise and unquestionably adequate definition of the general concept of a formal system. Turing does so, Gödel says, by providing a *conceptual analysis*, an analysis of the concept of a finite and mechanical procedure.

During the last decade of Gödel's life, he continued to praise Turing's analysis in conversations with Wang, saying that it provides a "correct and unique" definition of "the concept of mechanical" in terms of "the sharp concept of 'performable by a Turing machine'." He said that computability is "an excellent example … of a concept which did not appear sharp to us but has become so as a result of careful reflection," and that it is "absolutely impossible that anybody who understands the question and knows Turing’s definition should decide for a different concept” (Gödel in Wang 1974, 84).

Gödel’s remarks on Turing’s analysis are not to be taken lightly. Gödel made them at a time when others were failing to ascribe any special merit to Turing’s analysis.[15] Logic and computer science textbooks from the decades following the pioneering work of the 1930s by and large ignored Turing’s analysis altogether, and that trend continues to this day.[16] The full significance of Turing’s analysis has been appreciated only relatively recently.[17]

It is thus somewhat surprising that Gödel complained to Wang that Turing’s analysis contains a philosophical error. The puzzle is twofold. First, how could Gödel embrace Turing’s analysis despite the error of its ways? Second, how could Gödel go so wrong in attributing to Turing an argument about minds and machines that Turing did not advance? What is it in Turing’s analysis of computability that prompted Gödel to think the analysis involves an argument that is supposed to show mental procedures cannot go beyond mechanical procedures? Answering these questions is no easy task. Given the sparse and sometimes obscure textual evidence, any interpretation is bound to include a grain of speculation. Our interpretation invokes a distinction between two ways of understanding computability. We believe that this distinction is important for reasons transcending its ability to make sense of Gödel’s remarks on Turing. The distinction accounts for differing perspectives that exist concerning the concept of an effective procedure (alias algorithm), and concerning the Church-Turing thesis and issues about computability in general.

2.3 Distinguishing the Two Approaches

The cognitive and the noncognitive approaches differ with respect to the status of the restrictive conditions 1–5. The cognitive approach offers conditions 1–5 as reflections of (or abstractions from) limitations on human cognitive capacities. These limitations *give warrant to* or *justify* the correctness of the restrictive conditions. According to the cognitive approach, computability is constrained by conditions 1–5 *because* these constraints reflect the limitations of human cognitive capacities as these capacities are involved in calculation—or, as we shall say for short, because these constraints reflect limitations of the *faculty of calculation*.[18] A cognitivist need not claim that these limitations apply to human mental processes *in general*. As we saw, Gödel accused Turing of being a cognitivist in this more general sense, which he was not. As we will see below, despite several rhetorical statements, it is likely that Turing was not even a cognitivist in the more confined sense, of relating the conditions 1-5 to the limitations of the faculty of calculation.

The noncognitivist, on the other hand, does not think that the restrictive conditions 1–5 necessarily reflect limitations on human cognitive capacities. The noncognitivist need not deny that the existence of a faculty of calculation; the claim is that its limitations do not warrant the restrictive conditions. According to the noncognitivist conditions 1–5 merely explicate the concept of effective computation as it is properly used and as it functions in the discourse of logic and mathematics. The noncognitivist offers no other justification for the five conditions. In fact, a call for further justification might not have a place at all in the analysis of computability, according to the noncognitivist.

The difference between the two approaches can be made crystal clear by considering what the consequences for the extension of the concept of computability would be should the human faculty of calculation be found to violate one or more of conditions 1–5. As we have seen, Gödel himself challenged the assumption that the number of states of mind is bounded. Let's imagine scientists discover that human memory can involve an unbounded number of states and, further, that this results in hypercomputational mental powers—i.e., results in humans being able to calculate the values of functions that are not Turing-machine computable.[19] Would these discoveries threaten Turing's analysis of computability?[20] The cognitivist and the noncognitivist give different answers.

The cognitivist answers “Yes”. If it turns out that humans could, as a matter of cognitive fact, encode an infinite procedure, perform supertasks, or even observe, at any given step, an unbounded number of symbols when calculating a value of a function, cognitivists would regard this as undermining the analysis. If some of the constraints among 1–5 do not reflect actual upper limits on the faculty of calculation, then on the cognitive approach these constraints have no place in the analysis. In the circumstances we are imagining, the cognitivist would discard, weaken, or otherwise modify some of the conditions in order to produce a set of restrictive conditions that do reflect our true cognitive capacities. The cognitivist who finds herself or himself in the situation we are describing will jettison Turing’s analysis of computability and will replace it with a nonequivalent analysis that deems some non–Turing-machine computable functions to be computable.

According to the noncognitivist, on the other hand, the answer is “No.” Discoveries about the human mind have no bearing on the analysis of computability. The noncognitivist does not exclude the empirical possibility of the discovery that human memory is unbounded; nor is noncognitivism inconsistent with other ways in which the human mind might violate conditions 1–5. Rather, the analysis of computability invokes a finite number of states of mind because the analyzed concept is that of computation *by means of a finite* *procedure*. The focus is on what can be achieved by “finite means”—not on whether, as a matter of fact, human beings are limited to calculation by finite means.

The differences between cognitivism and noncognitivism have far-reaching implications in discussion of foundational issues in logic and mathematics. What, for example, should one say about a mathematician who is able to calculate any assigned number of digits of Turing’s δ? The cognitivist would say that the mathematician is in the role of human computer and that the Church-Turing thesis is false, since the thesis identifies computability with Turing-machine computability. According to the noncognitivist, however, these spectacular claims are unwarranted. If Turing’s analysis of computability is correct, then the mathematician who calculates arbitrary numbers of digits of δ is doing something that a human computer cannot do, *qua* human computer.

Let us now consider some ways in which cognitivism and noncognitivism do *not* differ. First, the distinction is not between human computation and other-than-human computation. Both approaches tie computability in the first instance to the activity of human computers, idealized humans who calculate with (perhaps) pencil and paper; and both approaches assume (absent the discoveries imagined above) that the human computer operates under the restrictive conditions 1–5. The difference has to do with what is meant by a human *computer*. According to the cognitivist, whatever calculations can be carried out by means of the human faculty of calculation count as computations. The human computer operates under the restrictive conditions 1–5 simply because these restrictions reflect the cognitive limitations of the faculty of calculation. According to the noncognitivist, however, the human computer is characterized by the restrictive conditions 1–5 simply because this is part and parcel of what it is to be a human computer.

Second, the distinction is not about empirical versus nonempirical analysis. Both approaches assume that Turing's analysis involves some form of *conceptual* analysis. The cognitive approach might be "empirical" only in the sense that the cognitivist's restrictive conditions reflect empirical facts about human cognition. But it is not the task of the analysis itself to discover these facts through empirical research; arguably, the fact that the human operates under these restrictive conditions is self-evident. Third, the distinction is not between the epistemic and the nonepistemic. According to both approaches, effective procedures play an important epistemic role, namely that of generating trustworthy results whose validity is beyond doubt. The approaches differ, rather, about the source of this epistemic role. According to the cognitive approach, the epistemic role is grounded in our calculative abilities. What counts as an effective procedure depends upon the upper limits of the faculty of calculation; thus discovering hypercomputational powers of the mind would immediately enlarge the class of trustworthy results. According to the noncognitive approach, the epistemic status of the effective procedures is rooted in their finite nature. An effective procedure generates trustworthy results *because* it is limited by finiteness constraints. (This is not to say, however, that effective procedures are the only way to generate trustworthy results. The discovery of hypercalculative abilities might indicate that there are other, noneffective, methods that generate trustworthy results. The noncognitivist's claim, rather, is that this discovery does not enlarge the class of effective, finite, computations.)[21]

There is not necessarily a sharp line between the cognitive and noncognitive approaches. One might hold, for example, that some of the restrictive conditions are arrived at by abstracting from cognitive capacities, while others arise from the nature of anything properly describable as "finite means." Emil Post has one foot, or possibly even both feet, in the cognitive camp, saying that the purpose of his analysis "is not only to present a system of a certain logical potency but also, in its restricted field, of psychological fidelity" (1936, 105). Post refers to Church's identification of effective calculability with recursiveness as being "not so much to a definition or to an axiom but to a *natural law*" (105), adding that "to mask this identification under a definition hides the fact that a fundamental discovery in the limitations of the mathematicizing power of Homo Sapiens has been made" (105, note 8).[22]

We will argue that Gödel's own allegiances lie with noncognitivism; but first we discuss Turing's and Kleene's positions.

2.4 Turing, a Pragmatic Noncognitivist

"The 'computable' numbers," Turing said in the opening sentence of his 1936 paper, are the numbers whose decimals "are calculable by finite means" (58). Although there is certainly some cognitivist rhetoric to be found in Turing's paper, this opening statement, and other textual evidence, makes it difficult to view him as a cognitivist. His remarks about the sequence δ are pertinent here. How could a cognitivist who accepts Turing's statement that "It is (so far as we know at present) possible that any assigned number of figures of δ can be calculated" (1936, 79) think that δ is known to be uncomputable? If it is in fact true that the faculty of calculation is such as to enable any assigned number of figures of δ to be calculated, then what reasons could a cognitivist have for thinking that δ is uncomputable? Turing, on the other hand, does say that δ is *not* computable; he says that it "is an immediate consequence of the theorem of [section] 8 that δ is not computable" (79). For Turing, δ is an example of a definable but uncomputable number that may nevertheless be calculable, although not by a uniform process.

Turing offers "the definitions" in his paper as a conceptual analysis of *calculable by finite means*.[23] Nevertheless, he is perfectly happy to appeal to cognitivist-style arguments from time to time, and his writing is a subtle blend of the two approaches. For example, he says that "the justification [of the definitions] lies in the fact that the human memory is necessarily limited" (59), a statement that will warm the cockles of any cognitivist's heart. He also appeals to the fact that "arbitrarily close" states of mind or symbols "will be confused" as a justification for disallowing the possibility of an infinity of (noncomplex) states of mind or of noncompound symbols (75–76). Turing is casting around for any viable "appeals to intuition" or "propaganda" that will help to win acceptance for his thesis "that the 'computable' numbers include all numbers which would naturally be regarded as computable" (74). "Propaganda is more appropriate to it than proof," he said elsewhere of a related thesis (1954, 588).[24]

A few pages after delivering a bouquet of cognitivist-style propaganda for his thesis, Turing eschews cognitive talk. He elides all reference to states of mind in favor of "a more physical and definite counterpart," the instruction-note (79).

2.5 Kleene and Fixed-in-Advance Public Processes

Gödel's assertion that the number of mental states could converge to infinity—which Kleene described as "pie in the sky"—has no "bearing on what number-theoretic functions are effectively calculable,"Kleene argued (1987, 493–494). He continued:

> For, in the idea of "effective calculability" or of an "algorithm" as I understand it, it is essential that all of the infinitely many calculations, depending on what values of the independent variable(s) are used, are performable—determined in their whole potentially infinite totality of steps—by following a set of instructions fixed in advance of all calculations. (Kleene 1987, 493)

However, this statement is in good accord with what Gödel thought. In his 1934 Princeton address (as we saw above), Gödel said that the computation procedure is finite, and he identifies it with primitive recursive operations. The interesting question for the cognitivist is whether or not this fixed-in-advanced-ness is a feature of our faculty of calculation; and if not, what is the justification of this restriction. Turing clearly thought that this is not a limitation on (what we are calling) the faculty of calculation, as Gödel did not.

Kleene suggested the requirement that a computation be *finite* is rooted in the necessity that *communication* be finite:

> The notion of an "effective calculation procedure" or "algorithm" (for which I believe Church's thesis) involves its being possible to convey a complete description of the effective procedure or algorithm by a finite communication, in advance of performing computations in accordance with it. My version of the Church-Turing thesis is thus the "*Public-Processes Version*" (Kleene 1987, 493–494).[25]

Yet why assume that communication must be carried out by a finite procedure?[26] For example, an accelerating Turing machine, which executes infinitely many steps in a finite period of time, is able to communicate an infinite amount of information in a temporally finite transmission.[27] One response, in accord with the cognitive approach, is that the necessity that communication be finite in all respects is rooted in the finiteness of our cognitive capacities. But this reply hardly addresses Gödel's arguments, since Gödel thought that the number of mental states could converge to infinity.

A noncognitivist, on the other hand, can cut across this issue of whether communication must be finite: whether or not knowledge can be conveyed by means of infinite procedures, we *begin* with the concept of a finite, fixed-in-advance mechanical procedure, and we analyze it in terms of a transition through a finite number of states (whether physical or mental).[28]

2.6 Gödel's Position, a Reconstruction

So how could Gödel embrace Turing's analysis of computability despite finding it to involve a fallacy, namely the imposition of a boundedness restriction on the number of states of mind? Unfortunately, Gödel says very little about his reasons for endorsing Turing's analysis, and any answer to this question is necessarily speculative. Gödel, it seems to us, was a thoroughgoing noncognitivist. As early as 1934 he was thinking of a computation procedure as a finite procedure, and at no point did he imply that this reflects, or is justified in terms of, limitations in human cognition. In fact, Gödel made no explicit mention of human computability at all. He suggested (in 1934) that sharpening the intuitive notion involves the formulation of a “set of axioms which would embody the generally accepted properties of this notion” (Gödel in Davis 1982, 9) and it is fair to assume that Gödel’s reading of Turing’s analysis was not along cognitivist lines.

In our view, Gödel probably regarded Turing’s statements about human cognition as entirely superfluous.[29]The fact that he disagreed with these statements was therefore no obstacle to his accepting the analysis. From a noncognitive perspective, the restrictive condition on the number of states of mind—that there is a fixed bound on the number of states that have to be “taken into account”—is correct, but not because “the human memory is necessarily limited,” nor because “if we admitted an infinity of states of mind, some of them will be ‘arbitrarily close’ and will be confused” (Turing 1936, 59, 79). It is exactly the other way around: the procedure can be implemented via a finite and fixed number of states (of mind, or more generally) *because* computability is analyzed in terms of a fixed finite procedure.

Thus Gödel embraced the analysis not because he thought that the finiteness of the procedure could be justified by other limitations (on the sensory apparatus, say). For Gödel the finiteness of the procedure is not grounded in the human condition at all—the restrictive condition on the number of states of mind (that “need be taken into account”) is adequate simply because this condition correctly explicates the finite, fixed-in-advance nature of the computation procedure. It might well be the case, as Gödel thought, that the number of distinguishable states of mind can converge to infinity, and that this convergence process is not effective, but all this is simply irrelevant to the analysis of the concept of computability.

Although Gödel was able to disregard Turing’s cognitivist rhetoric while the focus was the analysis of computability, he nevertheless took Turing to task for philosophical error once the focus shifted to the mathematical powers of the human mind more generally. Yet in fact Gödel misunderstood Turing, and their views about the mind were not as different as Gödel supposed.

3 Gödel and Turing on the Mind

3.1 The Mind-Machine Issue

Famously, the incompleteness and undecidability results of Gödel and Turing have been used to argue that there must be more to human mathematical thinking than can possibly be achieved by a computing machine.[30] In the Wang period (1967–1976[31]), Gödel’s discussions of the implications of these results were notably open-minded and cautious. For example, he said (in conversation with

Wang): “The incompleteness results do not rule out the possibility that there is a theorem-proving computer which is in fact equivalent to mathematical intuition” (Wang 1996, 186). On “the basis of what has been proved so far,” Gödel said, “it remains possible that there may exist (and even be empirically discoverable) a theorem-proving machine which in fact *is* equivalent to mathematical intuition, but cannot be *proved* to be so, nor even be proved to yield only *correct* theorems of finitary number theory” (184–185).

However, the textual evidence indicates clearly that Gödel’s position changed dramatically over time. In 1939 his answer to the question “Is the human mind a machine?” is a bold “No.” By 1951, his discussion of the relevant issues is nuanced and cautious. His position in his Gibbs lecture of that year seems to be that the answer to the question is not known. By 1956, however, he entertains—somewhat guardedly and with qualifications—the view that the “thinking of a mathematician in the case of yes-or-no questions could be completely replaced by machines” (1956, 375). In later life, his position appears to have moved once again in the direction of his earlier views (although the evidence from this period is less clear). We believe that a diachronic approach to the analysis of Gödel’s thinking about the mind-machine issue is illuminating. It has in the past been assumed that Gödel’s basic position on the issue remained essentially unchanged, although was refined over time, and thus that his earlier remarks (e.g., in 1939) could be used to guide exegesis of his later work.[32] However, such an approach risks conflating his earlier and later views.

3.2 The Younger Gödel

The early Gödel argued (on the basis of the limitative results) that the mind is not a machine:

> [I]t would actually be possible to construct a machine which would do the following thing: the supposed machine is to have a crank and whenever you turn the crank once around the machine would write down a tautology of the calculus of predicates. … So this machine would really replace thinking completely as far as deriving formulas of the calculus of predicates is concerned. It would be a thinking machine in the literal sense of the word. For the calculus of propositions, you could do even more. You could construct a machine in [the] form of a typewriter, such that if you type down a formula of the calculus of propositions then the machine would ring a bell [if it is a tautology] and if it is not it would not. You could do the same thing for the calculus of monadic predicates. But one can prove that it is impossible to construct a machine which would do the same thing for the whole calculus of predicates. So here already one can prove that [the] Leibnitzian program of the "calculemus" cannot be carried through i.e. one knows that the human mind will never be able to be replaced by a machine already for this comparatively simple question to decide whether a formula is a tautology or not. (Gödel in Cassou-Nogues 2009, 85)[33]

In his paper "Undecidable Diophantine Propositions" (193?), Gödel said—as he much later emphasized again in discussion with Wang—that Hilbert's rationalistic optimism "remains entirely untouched" by the incompleteness results (193?, 164). However, on neither occasion did Gödel himself endorse rational optimism, notwithstanding claims by Wang and Sieg to the contrary. Wang claimed that in the second version of the "philosophical error" note (quoted above) Gödel was "arguing for a 'rationalistic optimism' " (Wang 1996, 185), but there is no evidence whatsoever of this in the text. Sieg attributes to Gödel, in our view mistakenly, a "background assumption" of a "deeply rationalist and optimistic perspective" (Sieg 2007, 193).[34]

3.3 The Gibbs Lecture

By the time of the Gibbs lecture (1951), Gödel had drawn back from his firm conclusion of 1939 and offered a much more nuanced and guarded discussion. He introduced a sophisticated distinction between *subjective* and *objective* mathematics (1951, 309). Objective mathematics is "the system of all true mathematical propositions." Subjective mathematics is "the system of all [humanly] demonstrable propositions." Gödel pointed out that "no well-defined system of correct axioms can comprise all objective mathematics, since the proposition which states the consistency of the system is true, but not demonstrable in the system" (309). In the case of subjective mathematics, on the other hand, "it is not precluded that there should exist a finite rule" (309). Gödel appears to be leaving the question of whether there is such a rule open (in contradistinction to his 1939 remark).

Once the objective/subjective distinction has been framed, Gödel's 1939 remarks should probably be amended in something like the following way:

> But one can prove that it is impossible to construct a machine which would do the same thing [ring a bell if the typed formula is a tautology] for the whole calculus of predicates. So here already one can prove that *objective mathematics* will never be able to be replaced by a machine already for this comparatively simple question to decide whether a formula is a tautology or not. But it is not settled whether *subjective* mathematics will ever be able to be replaced by a machine; that is, one does not know whether the human mind will ever be able to be replaced by a machine.

The conclusion that Gödel draws in his Gibbs lecture is—unlike his bold statement of 1939—a cautious disjunction. He says:

> [I]f the human mind were equivalent to a finite machine, then objective mathematics not only would be incompletable in the sense of not being contained in any well-defined axiomatic system, but moreover there would exist *absolutely* unsolvable diophantine problems … where the epithet "absolutely" means that they would be undecidable, not just within some particular axiomatic system, but by *any* mathematical proof the human mind can conceive. So the following disjunctive conclusion is inevitable: *Either … the human mind … infinitely surpasses the powers of* any *finite machine* [*], *or else there exist absolutely unsolvable diophantine problems* [**]. (Gödel 1951, 310)

Concerning alternative [*], Gödel says only that "It is not known whether the first alternative holds" (312). He also says, "It is conceivable (although far outside the limits of present-day science) that brain physiology would advance so far that it would be known with empirical certainty … that the brain suffices for the explanation of all mental phenomena and is a machine in the sense of Turing" (309, note 13). Gödel takes alternative [**], asserted under the hypothesis that the human mind is equivalent to a finite machine, very seriously and uses it as the basis of an extended argument for mathematical Platonism.

3.4 The 1956 Letter to von Neumann

In a letter written to von Neumann five years after the Gibbs lecture, Gödel offers a new argument that it is "quite within the realm of possibility" that "the thinking of a mathematician in the case of yes-or-no questions could be completely replaced by machines, in spite of the unsolvability of the *Entscheidungsproblem.*"[35]

> Obviously, it is easy to construct a Turing machine that allows us to decide, for each formula F of the restricted functional calculus and every natural number n, whether F has a proof of length n [length=number of symbols]. Let $\psi(F, n)$ be the number of steps required for the machine to do that, and let $\varphi(n) = \max_F \psi(F, n)$.The question is, how rapidly does $\varphi(n)$ grow for an optimal machine? It is possible to show that $\varphi(n) \geq Kn$. If there really were a machine with $\varphi(n) \sim Kn$ (or even just $\sim Kn^2$) then this would have consequences of the greatest significance. Namely, this would clearly mean that the thinking of a mathematician in the case of yes-or-no questions could be completely replaced by machines, in spite of the unsolvability of the Entscheidungsproblem. n would merely have to be chosen so large that, when the machine does not provide a result, it also does not make any sense to think about the problem. Now it seems to me to be quite within the realm of possibility that $\varphi(n)$ grows that slowly. (Gödel to von Neumann 1956, 375)

3.5 The Wang Period

Gödel's note on Turing's "philosophical error" perhaps indicates that, by this stage, Gödel was once again setting aside the mechanical view of subjective mathematics. However, it is also perfectly possible that Gödel was merely objecting to what he took to be Turing's *argument* for a positive answer to the question, "Is the human mind a machine?" Everything that Gödel says in both versions of the note is consistent with the view that this question is open.

However, Wang reported that in 1972, in comments at a meeting to honor von Neumann, Gödel said: "The brain is a computing machine connected with a spirit" (Wang 1996, 189). In discussion with Wang at about that time, Gödel amplified this remark:

> Even if the finite brain cannot store an infinite amount of information, the spirit may be able to. The brain is a computing machine connected with a spirit. If the brain is taken to be physical and as a digital computer, from quantum mechanics there are then only a finite number of states. Only by connecting it to a spirit might it work in some other way. (Gödel in Wang 1996, 193)

Some caution is required in interpreting the remarks recorded by Wang, since the context is not always clear. Nevertheless Wang's reports create the impression that, by the time of his note about Turing, Gödel was again tending toward a negative answer to the question, "Is the human mind replaceable by a machine?"

3.6 Turing and the "Mathematical Objection"

Gödel's way—immaterialism—was certainly not Turing's way. Turing's subtle and interesting treatment of what he called the *Mathematical Objection* has a lot to teach us about his view of mind—which, like Gödel's view in the Wang period, had at its center the fact that mind is not static but constantly developin*g*.

The mathematician Jack Good, formerly Turing's colleague at Bletchley Park, Britain's wartime code-breaking headquarters, gave a succinct statement of the Mathematical Objection in a 1948 letter to Turing:

> Can you pin-point the fallacy in the following argument? "No machine can exist for which there are no problems that we can solve and it can't. But we are machines: a contradiction." (Good 1948)

At the time of Good's letter Turing was already deeply interested in the Mathematical Objection. More than eighteen months previously he had given a lecture, in London, in which he expounded and criticized an argument that flowing from his negative result concerning the *Entscheidungsproblem*, and concluding that "there is a fundamental contradiction in the idea of a machine with intelligence" (1947, 393). Refined forms of essentially the same argument have been advocated by Lucas, Penrose, and others (despite Turing's definitive critique, of which few writers seem aware).[36]The earliest formulation that we have encountered of the argument was by Emil Post: "We see that a *machine* would never give a complete logic; for once the machine is made *we* could prove a theorem it does not prove," Post wrote in 1941 (417). He concluded that mathematicians are "much more than a kind of clever being who can do quickly what a *machine* could do ultimately."

Turing believed that the Mathematical Objection has no force at all as an objection to machine intelligence—but not because the objection is necessarily mistaken in its claim that what the mind does is not always computable. He gave this pithy statement of the Mathematical Objection in his 1947 lecture:

> [W]ith certain logical systems there can be no machine which will distinguish provable formulae of the system from unprovable … On the other hand if a mathematician is confronted with such a problem he would search around and find new methods of proof, so that he ought eventually to be able to reach a decision about any given formula. (393–394)

As we showed in section 1.1, this idea—that the devising of new methods is a nonmechanical aspect of mathematics—is found in Turing's logical work from an early stage. He also mentions the idea in another of his wartime letters to Newman:

> The straightforward unsolvability or incompleteness results about systems of logic amount to this
>
> α) One cannot expect to be able to solve the Entscheidungsproblem for a system
> β) One cannot expect that a system will cover all possible methods of proof. (Turing to Newman, ca. 1940a, 212)

Here Turing is putting an interesting spin on the incompleteness results, which are usually stated in terms of there being true mathematical statements that are not provable. On Turing's way of looking at matters, the incompleteness results show that no single system of logic can include all methods of proof; and he advocates a progression of logical systems—his ordinal logics—each more inclusive than its predecessors. He continued in the letter:

> [W]e… make proofs … by hitting on one and then checking up to see that it is right. … When one takes β) into account one has to admit that not one but many methods of checking up are needed. In writing about ordinal logics I had this kind of idea in mind. (212–213)

3.7 Turing and Rationalistic Optimism

We suggested in part 1 that Turing gave cautious expression to a form of rationalistic optimism when he said in 1936, "It is (so far as we know at present) possible that any assigned number of figures of δ can be calculated, but not by a uniform process" (79). Scattered throughout his later writings are passages indicating that Turing was in some sense a rationalistic optimist. He said in a lecture given circa 1951: "By Gödel's famous theorem, or some similar argument, one can show that however the [theorem-proving] machine is constructed there are bound to be cases where the machine fails to give an answer, *but a mathematician would be able to*" (italics added; ca. 1951,472). We have already noted that in his pithy 1947 statement of the Mathematical Objection he said: "On the other hand if a mathematician is confronted with such a problem he would search around and find new methods of proof, so that he ought eventually to be able to reach a decision about any given formula.” And in 1948, again discussing the Mathematical Objection, he said: “On the other hand the human intelligence seems to be able to find methods of ever-increasing power for dealing with such problems, ‘transcending’ the methods available to machines” (411).

In 1950 Turing considered the obvious countermove against the Mathematical Objection, namely the move of denying rationalistic optimism—but he did not endorse it. He said:

> The short answer to [the Mathematical Objection] … is that although it is established that there are limitations to the powers of any particular machine, it has only been stated, without any sort of proof, that no such limitations apply to the human intellect. (Turing 1950, 451)

Rather than letting matters rest there, however, he continued: “But I do not think [the Mathematical Objection] … can be dismissed quite so lightly.”

3.8 What Would Turing Have Said about Gödel’s “Sharp Result”?

Gödel might appear to have hit the nail right on the head with what he called his “sharp result” (“If my result is taken together with the rationalistic attitude … then [we can infer] the sharp result that mind is not mechanical. This is so, because, if the mind were a machine, there would, contrary to this rationalistic attitude, exist number-theoretic questions undecidable for the human mind."[Gödel in Wang 1996, 186–187]). The incompleteness results by themselves certainly do not show that the mind is not a computer. An essential extra ingredient that must be added to the incompleteness results is, as Gödel said, the premise of rationalistic optimism. But even if this premise is added, is the inference that the mind is not mechanical validly drawn?

If our interpretation of Turing is correct, Turing would have answered this question negatively. He would have emphasized the point made in his letter, quoted earlier:

> If you think of various machines I don't see your difficulty. One imagines different machines allowing different sets of proofs …

What is the significance of Turing's statement? The Gödel argument is usually thought of as being a *reductio*. Assume that the mind is equivalent to some Turing machine, say T. Then the usual moves show that the mind cannot be equivalent to T; from which it is concluded that the mind is not mechanical. However, as we pointed out earlier, Gödel's sharp result tells us only that there is no *single* machine that is equivalent to mathematical intuition. If one allows that, from time to time, the mind is identical to *different* Turing machines, allowing *different* sets of proofs, then there is no inconsistency between the sharp result and the claim that every stage of a dynamically changing mind (or of a dynamically changing collection of a number of minds) is a Turing machine. Turing employed the image of a sequence of increasingly powerful proof-producing machines in his 1950 discussion of the Mathematical Objection:

> In short, then, there might be men cleverer than any given machine, but then again there might be other machines cleverer again, and so on. (Turing 1950, 451)

If we are correct, Turing's answer to the Mathematical Objection is as follows: Pick any Turing machine T, then there may be a (developmental stage of) some mind M that is cleverer than T, but this has no tendency to show that (this stage of) M is not itself a Turing machine.—*It is perfectly consistent with the sharp result that this stage of M is a proof-producing Turing machine.*

Underlying this answer to the Mathematical Objection is what we call the *Multi-Machine theory of mind*: human minds are Turing machines, in the sense that each developmental stage of a mind M is equivalent to some Turing machine, while different stages of M are equivalent to *different* Turing machines.The interpretationappears in Copeland (2004b); and sections 1.3.6 – 1.3.10 of the present paper are based on Copeland 2006.[37] (Wilfried Sieg puts forward a similar interpretation of Turing in his paper in this volume.)

3.9 The Post--War Turing on Learning

Turing did not attempt to explain what he called the "activity of the intuition," either in his 1939 paper nor the wartime letters to Newman. A human mathematician working according to the rules of a fixed logical system is in effect a proof-producing machine; and when intuition supplies the mathematician with some new means of proof, he or she becomes a different proof-producing machine, capable of a larger set of proofs. How does the mathematician achieve this transformation from one proof-finding machine to another? The pre-war Turing was content to leave this question to one side, but the post-war Turing had a lot to say that is relevant to this question. In his post-war writing on mind, the term "intuition" drops from view and what comes to the fore is the idea of *learning*—in the sense of devising or discovering—new methods of proof.[38]

Turing's discussions of learning repeatedly emphasized:

- The importance of search: hehypothesized boldly that "intellectual activity consists mainly of various kinds of search." (1948, 431)
- The importance of the learner making and correcting mistakes: "[T]his danger of the mathematician making mistakes is an unavoidable corollary of his power of sometimes hitting upon an entirely new method." (ca. 1951, 472)
- The importance of involving a random element: "[O]ne feature that … should be incorporated … is a "random element." … This would result in the

behavior of the machine not being by any means completely determined by the experiences to which it was subjected." (ca. 1951, 475)

- The importance of instruction modification: "What we want is a machine that can learn from experience. The possibility of letting the machine alter its own instructions provides the mechanism for this. … One can imagine that after the machine had been operating for some time, the instructions would have altered out of all recognition." (1947, 393)

Instruction-modification leads from one Turing machine to another; and underpins the central feature of the Multi-Machine theory of mind, the transformation of one Turing machine into another. Different instruction table, different Turing machine. In a lecture given circa 1951, Turing made it clear that his—then radical—idea that machines can learn is the crux of his reply to the Mathematical Objection; and he stressed the importance of the idea of learning new methods of proof in his 1947 discussion of the objection, describing the mathematician as searching around and finding new methods of proof.

Modifying the table of instructions in effect transforms the learning machine into a different Turing machine. So a machine with the ability to learn is able to traverse the space of proof-finding Turing machines. The learning machine successively mutates from one proof-finding Turing machine into another, becoming capable of wider sets of proofs as it searches for and acquires new, more powerful methods of proof.

How is this learning of new methods of proof actually to be accomplished? Of course, Turing did not say in any detail. He posed a question about the human mind whose answer is still fundamentally unknown.

3.10 Computability and the Multi-Machine Theory of Mind

Figure 1.1 shows a diagrammatic representation of the multi-machine theory of mind. The Turing machines are enumerated as points on the *y*-axis, beginning with the first (relative to some unspecified underlying ordering of the Turing machines). Along the *x*-axis are the successive stages of development of learning mind *M*. *M* successively mutates (in discrete jumps) from one theorem-proving Turing machine into another. Idealizing away death and other contingent resource-constraints, we can imagine the trajectory continuing indefinitely to the right.

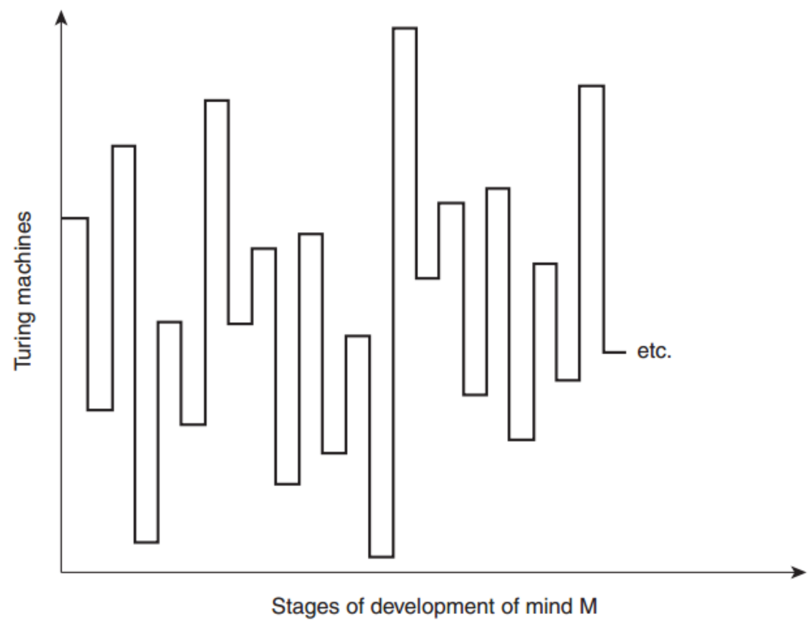

Figure 1.1

The Multi-Machine theory of mind.

Is the function from mind-stages to Turing machines computable? Not if rationalistic optimism is true. If it is an open question whether some appropriate version of rationalistic optimism is true, then it is an open question whether this function is computable.

How could this function *fail* to be computable? Where could the uncomputability come from? Had Gödel commented specifically on the Multi-Machine theory of the human mind, he might have said the following (he was actually commenting on the idea of a “race” of theorem-proving machines, analogous to the mind stages of the Multi-Machine theory):

Such a state of affairs would show that there is something nonmechanical in the sense that the overall plan for the historical development of machines is not mechanical. If the general plan is mechanical, then the whole race can be summarized in one machine. (Gödel in Wang 1996, 207)[39]

However, Gödel did not cash out this notion of an "overall plan," and although his remark gets the issues into sharp focus, it does not really help very much with the specific question of *where* the uncomputability might come from. We can only guess how Turing would have answered if questioned on this point. But given the emphasis that he placed on the inclusion of a random element in the learning process, he *might* have said:The answer to your question is simple—the source of the uncomputability is randomness. For, as Church had pointed out in 1940, "If a sequence of integers a_1, a_2, ... a_n, … is random, then there is no function $f(n) = a_n$ that is calculable by Turing machine" (134–135).

The core idea is that the partially random learning machine emulates the "activity of the intuition" in its walk through the space of proof-finding Turing machines. Heuristic and other forms of search, together with learning algorithms and instruction modification, coupled with the activity of a random element from time to time, produce the evolving sequence of proof-finding Turing machines described by the Multi-Machine theory. Turing spoke in several places about the idea of including "a random element in a learning machine", saying, "A random element is rather useful when we are searching". He emphasized that the same process of search involving a random element "is used in the analogous process of evolution" (1950. 463).

In the situation envisaged by the Multi-Machine theory, the trajectory through the space of proof-finding Turing machines could indeed be uncomputable, in the precise sense that the function on the non-negative integers whose value at i is the i^{th} Turing machine on the trajectory might be uncomputable. If this trajectory is uncomputable, then although the mind is at every stage identical to some Turing machine, it is not the case that the mind as a whole "can be summarized in one machine."

4 Conclusion

Gödel's critical note on Turing is of interest not because Turing committed a philosophical error about the mind-machine issue, for he did not, but because of the light that the critical note helps to shed on the similarities, and the differences, in the views of these two great founders of the study of computability. Gödel praised Turing's analysis of computability but objected to the thought that Turing's restrictive conditions are grounded in facts about human cognition, and he misinterpreted some of Turing's statements about the analysis of computability as claims about the mind in general. In fact, Turing agreed with Gödel that the mind is more powerful than any given Turing machine. Unlike Gödel, however, Turing did not think that the mind is something different from machinery.

Acknowledgments

Copeland's research was supported in part by the Royal Society of New Zealand Marsden Fund, grant UOC905, and Shagrir's research was supported by the Israel Science Foundation, grant 1509/11.

Notes

1. See also Copeland (2011), "Epilogue: The Computer and the Mind."
2. Wang reports that Gödel wrote the note "around 1970" (1996, 195). The note is included in the *Collected Works* (Gödel 1990, 305–306), with an introduction by Judson Webb. In 1972, Gödel gave Wang a revised version of the note, which Wang published in his *From Mathematics to Philosophy* (1974, 325–326).
3. The quotation is from the preceding page of "On Computable Numbers" (Turing 1936, 74).
4. Quoted by Wang in *A Logical Journey* (1996, 197).
5. The term "rationalistic optimism" seems to be Wang's (see Wang 1974, 325).
6. Gödel quoted by Wang.
7. See also Copeland (1997a, 1998a, 2000).
8. See also Sieg and Byrnes (1999), Sieg (2002), Copeland and Shagrir (2007).
9. See, for example, Copeland's "Computable Numbers: A Guide," in *The Essential Turing* (Copeland, 2004a); Shagrir (2006); Sieg (2006); and Soare(this volume).
10. See the discussion in Sieg (2007).

11. Gödel often vacillates between the terms "mechanical procedure" and "finite procedure" when referring to formal systems (see the discussion in Shagrir 2006). But from his 1972 note it is clear that he does not take the two to be synonymous; he talks there about *finite but non-mechanical* procedures.

12. Davis dates the article to 1938 (see his introduction to Gödel 193? in the *Collected Works*, [Gödel 1995, 156–163]).

13. For a pertinent discussion see Sieg (2009).

14. In a footnote to this lecture, added in 1964, Gödel defines computability in terms of representability; a function is computable just in case (to simplify a bit) the corresponding representing formal expression is a theorem in the relevant formal system. Similar characterizations are advanced by Church (1936), Turing (1936, section 9, part 2); see the discussion by Kripke (this volume), and Hilbert and Bernays (1939).

15. Church describes Turing's identification of effectiveness with Turing-machine computability as "evident immediately" (1937a, 43), "an adequate representation of the ordinary notion" (1937b, 43), and as having "more immediate intuitive appeal" (1941, 41), but he does not say it is more convincing than other arguments; see also Kleene (1952).

16. Turing's argument is mentioned in the early days of automata theory, e.g., by McCulloch and Pitts (1943); Shannon and McCarthy (1956), in their introduction to *Automata Studies*; and Minsky (1967, 108–111), who cites it almost in full in his *Finite and Infinite Machines*. Yet even Minsky asserts that the "strongest argument in favor of Turing's thesis is the fact that … satisfactory definitions of 'effective procedure' have turned out to be equivalent" (111). As far as we know there is no mention of Turing's argument in current logic and computer science textbooks. The two arguments always given for the Church-Turing thesis are the confluence (equivalence) of definitions, and the lack of counterexamples; see, e.g., Boolos and Jeffery (1989, 20) and Lewis and Papadimitriou (1981, 223–224).

17. See Gandy (1988), Sieg (1994), Copeland (1997a, 2000). The relatively recent rediscovery of Turing's analysis is underscored by Martin Davis's comment (1982, 14, note 15) that "this [Turing's] analysis is still very much worth reading. I regard my

having failed to mention this analysis in my introduction to Turing's paper in Davis (1965) as an embarrassing omission."

18. The term "faculty of calculation" does not necessarily refer to a designated functional or neural module but to the cognitive capacities as they are involved in calculating numbers or the values of functions.

19. For discussion of hypercomputation, see Copeland (1997b, 1998b, 2002b), Copeland and Proudfoot (1999), Copeland and Sylvan (1999). On the relationships between acceleration and hypercomputation, see Copeland and Shagrir (2011).

20. One could question the possibility of such discoveries. Following Kripke and Putnam, one could argue that *if* it is true that the restrictions 1–5 reflect the upper limits on the faculty of calculation, then it is necessarily true (analogously to the claim that if it is true that water is H_2O, then it is necessarily true). This is also in accord with the wisdom that if the Church-Turing thesis is true then it is necessarily true. Thus if it is true that the restrictions 1–5 reflect the upper limits on the faculty of calculation, it is not possible to discover that the faculty of calculation violates one of these restrictions (for discussion see Yaari 2005, who concludes that the thesis is of the *necessary a posteriori* kind). Our suggestion is that these scenarios are conceivable even if metaphysically impossible (for the distinction between conceivability and possibility see, e.g., Chalmers 2004). Our question is whether, in this scenario, the first premise in Turing's argument is true or false.

21. It should be mentioned in this context that Gödel himself proposed in the 1930s finite and nonmechanistic ("constructive") methods, partly as an attempt to rescue Hilbert's program. The hope of Gödel and others was to devise trustworthy "initary" methods that transcend the limitation of the incompleteness results. Yet he never claimed (as emphasized below) that these methods enlarge the class of effective computations.

22. Contemporary researchers who take this approach are Yaari (2005) and Bringsjord (see Bringsjord and Arkoudas [2006] and Bringsjord and Sundar [2011]).

23. See, for example, Turing (1936, 59).

24. See also Copeland (2004a, 577–578).

25. See also Kleene (1988, 50).

26. See also Kripke (this volume), who challenges this requirement altogether.

27. On accelerating Turing machines see Copeland (1998b, 1998c, 2002a), Shagrir (2004), and Copeland and Shagrir (2011).

28. Following Kleene, Sieg also grounds the finiteness of the procedure in the "normative requirement on the fully explicit presentation of mathematical proofs in order to insure inter-subjectivity" (2009, 532: see also 2008); see also Seligman (2002, section 4) [[Ref: SELIGMAN, J. 'The Scope of Turing's Analysis of Effective Procedures' Minds and Machines, Vol 12, 203-220, 2002.]]. On the other hand, Sieg (2006) contends that two "central restrictive conditions" on our sensory apparatus suffice to secure the conclusion of the analysis. One restrictive condition is that "[t]here is a fixed finite bound on the number of configurations a computer can immediately recognize" (boundedness). The other is that "[a] computer can change only immediately recognizable (sub-) configurations" (locality; 203). It thus seems to us that Sieg holds a dual position that is a mixture of cognitive and noncognitive elements.

29. See also Shagrir (2006).

30. For example, Lucas (1961, 1996), Penrose (1994; see e.g., 65).

31. See Wang (1996, xi, 7).

32. See, for example, Sieg (2006).

33. Extract from Gödel's notes for his 1939 introductory logic course at the University of Notre Dame.

34. See also the first paragraph of Sieg'ssection 2 (2007,191).

35. This letter is discussed in Wigderson (2011).

See also: Copeland's "Turing's Answer to the Mathematical Objection" in *The Essential Turing* (Copeland 2004a, 469–70);and Copeland and Proudfoot (2007).

36. A full history of the argument will be presented in Copeland and Piccinini (in preparation). Modern scholarship on Turing's mathematical objection began with Piccinini (2003).

37. See also: Copeland (2008); Copeland's "Turing's Answer to the Mathematical Objection" in *The Essential Turing* (Copeland 2004a, 469–470); and Copeland and Proudfoot (2007).

38. See also Copeland (2006, 168).

39. Gödel quoted by Wang.

References

Baaz, M., C. H. Papadimitriou, H. W. Putnam, D. S. Scott, and C. L. Harper, Jr., eds. 2011. *Kurt Gödel and the Foundations of Mathematics*. New York: Cambridge University Press.

Boolos, G. S., and R. C.Jeffrey. 1989. *Computability and Logic*. 3rd ed. Cambridge: Cambridge University Press.

Bringsjord, S., and K. Arkoudas. 2006. On the provability, veracity, and AI-relevance of the Church-Turing thesis. In *Church's Thesis after 70 Years,*eds. A. Olszewski, J. Wolenski, and R. Janusz, 66–118.Frankfurt: Ontos-Verlag.

Bringsjord, S., and G. N. Sundar. 2011. In defense of the unprovabilityof the Church-Turing thesis. *Journal of Unconventional Computing* 6:353–373.

Cassou-Nogues, P. 2009. Gödel's introduction to logic in 1939. *History and Philosophy of Logic* 30:69–90.

Chalmers, D. 2004. Epistemic two-dimensional semantics. *Philosophical Studies* 18:153–226.

Church, A. 1936. An unsolvable problem of elementary number theory. *American Journal of Mathematics* 58:345–63.Reprinted in Davis, *The Undecidable*, 88–107.

Church, A. 1937a. Review of Turing (1936). *Journal of Symbolic Logic* 2:42–43.

Church, A.1937b. Review of Post (1936). *Journal of Symbolic Logic* 2:43.

Church, A. 1940. On the concept of a random sequence. *Bulletin of the American Mathematical Society* 46:130–135.

Church, A. 1941. *The Calculi of Lambda-Conversion* .Princeton: Princeton University Press.

Copeland, B. J. 1997a. The Church-Turing thesis. *The Stanford Encyclopedia of Philosophy*, ed. E. Zalta. http://plato.stanford.edu.

Copeland, B. J. 1997b. The broad conception of computation. *American Behavioral Scientist* 40:690–716.

Copeland, B. J. 1998a. Turing's O-machines, Penrose, Searle, and the brain. *Analysis* 58:128–138.

Copeland, B. J. 1998b. Even Turing machines can compute uncomputable functions. In *Unconventional Models of Computation*, eds. C. Calude, J. Casti, and M. Dinneen, 150–164. London: Springer-Verlag.

Copeland, B. J. 1998c. Super Turing-machines.*Complexity* 4:30–32.

Copeland, B. J. 2000. Narrow versus wide mechanism.*Journal of Philosophy* 97:5–32. Reprinted in *Computationalism: New Directions*, ed. M. Scheutz, 59–86.Cambridge: MIT Press, 2002.

Copeland, B. J. 2002a. Accelerating Turing machines. *Minds and Machines* 12:281–301.

Copeland, B. J. 2002b. Hypercomputation, in Hypercomputation, special issue, *Minds and Machines* 12(4):461–502.

Copeland, B. J., ed. 2002–2003. Hypercomputation. Special issue, *Minds and Machines* 12, 13.

Copeland, B. J. 2004a. *The Essential Turing*. Oxford: Oxford University Press.

Copeland, B. J. 2004b. Hypercomputation: philosophical issues. *Theoretical Computer Science* 317:251-267.

Copeland, B. J. 2006. The mathematical objection: Turing, Penrose, and creativity. A lecture at the MIT Computer Science and Artificial Intelligence Laboratory, December 2, 2006.

Copeland, B. J. 2011. From the *Entscheidungsproblem* to the personal computer—and beyond (Epilogue: The Computer and the Mind). In *Kurt Gödel and the Foundations of Mathematics*, eds. Baaz, C. H.Papadimitriou, D. S.Scott, H.Putnam, and C. L.Harper. New York: Cambridge University Press, 151–184.

Copeland, B. J., and G. Piccinini. On Turing on uncomputability and the Mind Manuscript in preparation.

Copeland, B. J., and D. Proudfoot. 1999. Alan Turing's forgotten ideas in computer science. *Scientific American* 280:76–81.

Copeland, B. J., and D. Proudfoot. 2007. Artificial intelligence: History, foundations, and philosophical issues. In *Handbook of the Philosophy of Psychology and Cognitive Science*, ed. P.Thagard, 429–482.Amsterdam: Elsevier Science.

Copeland, B. J., and O.Shagrir. 2007. Physical computation: How general are Gandy's principles for mechanisms? *Minds and Machines*17:217–31.

Copeland, B. J., and O. Shagrir. 2011. Do accelerating Turing machines compute the uncomputable? *Minds and Machines* 21:221–239.

Copeland, B. J., and R. Sylvan. 1999. Beyond the universal Turing machine. *Australasian Journal of Philosophy* 77:46–66.

Davis, M., ed. 1965. *The Undecidable: Basic Papers on Undecidable Propositions, Unsolvable Problems and Computable Functions*. New York: Raven.

Davis, M. 1982. Why Gödel didn't have Church's thesis. Information and Control 54:3–24.

Gandy, R. 1980. Church's thesis and principles of mechanisms. In The Kleene Symposium, eds. J. Barwise, H. J. Keisler, and K. Kunen, 123–148. Amsterdam: North-Holland.

Gandy, R. 1988. The confluence of ideas in 1936. In Herken, *The Universal Turing Machine*, 51–102.

Gödel, K. 1931. On formally undecidable propositions of Principia Mathematica and related systems. MonatsheftefürMathematik und Physik 38:173–198.In Gödel, *Collected Works I*, 144–195.

Gödel, K. 1933. The present situation in the foundations of mathematics. In Gödel, *Collected Works III*, 45–53.

Gödel, K. 1934. On undecidable propositions of formal mathematical systems. In Gödel, *Collected Works I,* 346–369.

Gödel, K. 193?. Undecidable Diophantine propositions. In Gödel, *Collected Works III*, 164–175.

Gödel, K. 1939. Notre Dame lectures on logic. In Gödel, *Collected Works V,* 527–528.

Gödel, K. 1946. Remarks before the Princeton bicentennial conference on problems in mathematics. In M. Davis, *The Undecidable*, 84–88, and in Gödel, *Collected Works II,* 150–153. Page references are to *Collected Works*.

Gödel, K. 1951.Some basic theorems on the foundations of mathematics and their implications. In Gödel, *Collected Works III,* 304–323.

Gödel, K. 1956. Letter to John von Neumann, March 20, 1956. In Gödel, *Collected Works V,* 372–377.

Gödel, K. 1964.Postscriptum to Gödel 1934.In Gödel, *Collected Works I,* 369–71, and in M. Davis, *The Undecidable*, 71–73. Page references are to Davis 1965.

Gödel, K. 1972. Some remarks on the undecidability results. In Gödel, *Collected Works II,* 305–306.

Gödel, K. 1986. *Collected Works, Vol. I: Publications 1929–1936*. eds. S. Feferman, J. W. Dawson Jr., S. C. Kleene, G. H. Moore, R. M. Solovay, and J. van Heijenoort.Oxford: Oxford University Press.

Gödel, K. 1990. *Collected Works, Vol. II: Publications 1938–1974*. Eds. S. Feferman, J. W.Dawson Jr., S. C. Kleene, G. H. Moore, R. M. Solovay, and J. van Heijenoort. Oxford: Oxford University Press.

Gödel, K. 1995. *Collected Works, Vol. III: Unpublished Essays and Lectures*. eds. S. Feferman, J. W.Dawson Jr., W. Goldfarb, C. Parsons, and R. M. Solovay. Oxford: Oxford University Press.

Gödel, K.2003. *Collected Works, Vol. V: Correspondence, H–Z*. Eds.S. Feferman, J. W. Dawson Jr., W. Goldfarb, C. Parsons, and W. Sieg. Oxford: Clarendon Press.

Good, I. J. 1948 .Early notes on electronic computers. Unpublished. University of Manchester National Archive for the History of Computing, MUC/Series 2/a4.

Herken, R., ed. 1988. *The Universal Turing Machine: A Half-Century Survey*. Oxford: Oxford University Press.

Hilbert, D. 1902. Mathematical problems: Lecture delivered before the International Congress of Mathematicians at Paris in 1900. *Bulletin of the American Mathematical Society* 8:437–479.

Hilbert, D., and P. Bernays. 1939. *Grundlagen der Mathematik II*. Berlin: Springer-Verlag.

Kleene, S. C. 1936. General recursive functions of natural numbers. *Mathematische Annalen* 112:727–742.Reprinted in M. Davis, *The Undecidable* 236–253.

Kleene, S. C. 1952. *Introduction to Metamathematics*. Amsterdam: North-Holland.

Kleene, S. C. 1987. Reflections on Church's thesis. *Notre Dame Journal of Formal Logic* 28:490–498.

Kleene, S. C. 1988 .Turing's analysis of computability, and major applications of it. In Herken, *The Universal Turing Machine*,17–54.

Lewis, H. R., and C. H. Papadimitriou. 1981. *Elements of the Theory of Computation*. Englewood Cliffs, NJ: Prentice-Hall.

Lucas, J. R.1961. Minds, machines and Gödel. *Philosophy* 36:112–127.

Lucas, J. R. 1996. Minds, machines and Gödel: A retrospect. In *Machines and Thought: The Legacy of Alan Turing, Vol. 1*, eds. P. Millican and A. Clark, 103-124. Oxford: Clarendon Press.

McCulloch, W. S., and W. Pitts. 1943. A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics* 5:115–133.

Minsky, M. L. 1967. *Computation: Finite and Infinite Machines*. Englewood Cliffs, NJ: Prentice-Hall.

Penrose, R. 1994. *Shadows of the Mind: A Search for the Missing Science of Consciousness*. Oxford: Oxford University Press.

Piccinini, G. 2003. Alan Turing and the mathematical objection. *Minds and Machines* 13(1):23–48.

Post, E. L. 1936. Finite combinatory processes—Formulation I. J*ournal of Symbolic Logic* 1:103–05. Reprinted in Davis, *The Undecidable*, 288–291.

Post, E. L. 1941. Absolutely unsolvable problems and relatively undecidable propositions: Account of an anticipation. In Davis, *The Undecidable*, 338–443.

Seligman, J. 2002. The scope of Turing's analysis of effective procedures. *Minds and Machines* 12:203–220.

Shagrir, O. 2002. Effective computation by humans and machines. *Minds and Machines* 12:221–240.

Shagrir, O. 2004. Super-tasks, accelerating Turing machines and uncomputability. *Theoretical Computer Science* 317:105–114.

Shagrir, O.2006.Gödel on Turing on computability. In In*Church's Thesis after 70 years*, eds. A. Olszewski, J. Wolenski and R. Janusz, 393–419. Frankfurt: Ontos-Verlag.

Shannon, C. E., and J. McCarthy, eds. 1956. *Automata Studies*34.Princeton: Princeton University Press.

Sieg, W. 1994. Mechanical procedures and mathematical experience. In *Mathematics and Mind*, ed. A. George, 71–117. Oxford: Oxford University Press.

Sieg, W. 2002. Calculations by man and machine: Conceptual analysis. In *Reflections on the Foundations of Mathematics*, eds. W. Sieg, R. Sommer, and C. Talcott. Lecture Notes in Logic,Vol. 15, 396–415. Natick, MA: Association for Symbolic Logic.

Sieg, W. 2006. Gödel on computability. *PhilosophiaMathematica* 14:189–207.

Sieg, W. 2007. On mind and Turing's machines. *Natural Computing* 6:187–205.

Sieg, W. 2008. Church without dogma: Axioms for computability. In *New Computational Paradigms*, eds. B. Lowe, A. Sorbi, and B. Cooper, 139–152. Springer Verlag.

Sieg, W. 2009. On computability.In Handbook of the Philosophy of Mathematics, ed. A. Irvine, 535–630.Amsterdam: Elsevier.

Sieg, W., and J. Byrnes. 1999. An abstract model for parallel computations: Gandy's thesis. *Monist* 82:150–164.

Turing, A. M. 1936. On computable numbers, with an application to the *Entscheidungsproblem*. In Copeland, *The Essential Turing*, 58–90.

Turing, A. M. 1939. Systems of logic based on ordinals. In Copeland, *The Essential Turing*, 146–204.

Turing, A. M.ca. 1940a. Letter to Max Newman.In Copeland,*The Essential Turing*, 211–213.

Turing, A. M. ca.1940b. Letter to Max Newman.In Copeland,*The Essential Turing*, 214–216.

Turing, A. M. 1947. Lecture to the London Mathematical Society on February 20, 1947.In Copeland, *The Essential Turing*, 378–394.

Turing, A. M. 1948. Intelligent machinery.In Copeland, *The Essential Turing*, 410–432.

Turing, A. M. 1950. Computing machinery and intelligence.In Copeland, *The Essential Turing*, 441–464.

Turing, A. M.ca.1951. Intelligent machinery, a heretical theory.In Copeland, *The Essential Turing*, 472–475.

Turing, A. M. 1954. Solvable and unsolvable problems.In Copeland, *The Essential Turing*, 582–595.

Wang, H. 1974. *From Mathematics to Philosophy*. London: Routledge & Kegan Paul.

Wang, H. 1996. *A Logical Journey: From Gödel to Philosophy*. Cambridge: MIT Press.

Webb, J. C.1990. Introductory note to remark 3 of Gödel 1972. In Gödel, *Collected Works II*, 292–304.

Wigderson, A. 2011. The Gödel phenomenon in mathematics: A modern view. In M. Baaz, C. P. Papadimitriou, H. Putnam, S. Scott, and C. L. Harper, eds., *Kurt Gödel and the Foundations of Mathematics*, 475–508.

Yaari, J. 2005. *On the status and justification of the Church–Turing thesis*. M. Phil Thesis, University of London.