인지적 계산가능성에 대한 메타수학적 연구

현 우 식
연세대학교 인지과학연구소 인공지능연구실

A Metamathematical Study of Cognitive Computability with Gödel's Incompleteness Theorems

Woo Sik Hyun
Artificial Intelligence Lab., Center for Cognitive Science, Yonsei University

Abstract

This study discusses cognition as a computable mapping in cognitive systems and relates Gödel's Incompleteness Theorems to the computability of cognition from a metamathematical perspective.

Understanding cognition as a form of computation requires not only Turing machine models but also neural network models. In previous studies of computation by cognitive systems, it is remarkable to note how little serious attention has been given to the issue of computation by neural networks with respect to Gödel's Incompleteness Theorems. To address this problem, first, we introduce a definition of cognition and cognitive science. Second, we deal with Gödel's view of computability, incompleteness and speed-up theorems, and then we interpret Gödel's disjunction on the mind and the machine. Third, we discuss cognition as a Turing computable function and its relation to Gödel's incompleteness. Finally, we investigate cognition as a neural computable function and its relation to Gödel's incompleteness.

The results show that a second-order representing system can be implemented by a finite recurrent neural network. Hence one cannot prove the consistency of such neural networks in terms of first-order theories. Neural computability, theoretically, is beyond the computational incompleteness of Turing machines. If cognition is a neural computable function, then Gödel's incompleteness result does not limit the computational capability of cognition in humans or in artifacts.

0. Introduction

The scientific analysis of an object requires that it be observed within the context of a specific model. Consequently, every model reflects the specific concerns and methods of its respective discipline. The development of computational models for studying cognition is extremely common these days and fundamental in cognitive science.

The purpose of our research is to explore the characteristics of neural computability and to analyze from a formal perspective the relationship between neural networks and Gödel's incompleteness theorem. The computability theory of neural networks is not just one more issue among many to be debated in cognitive science, but rather the most important underlying issue. An analysis of neural computability will provide a more proper understanding of cognition in humans and in artifacts. It is remarkable, however, to observe how little serious attention has been given to the theme of neural computability in cognitive science.

Nevertheless, we have to caution ourselves in that the world of formal science deals with the formal relationship between real objects, whereas empirical science is concerned with the real objects themselves. Therefore, it is expedient for us to perceive the object as being nothing more than a conceptual guide in our search for the theory that would be most appropriate for cognitive science.

1. Cognition in Cognitive Science

1.1 Cognition

By cognition, we mean a kind of computation. Thus, we may think of it as a mathematical mapping from a class of times to a class of spaces. Cognition is a plexus of properties rather than a single concept. In this research, the domain of cognition is defined by a class of discrete times, and its range a class of mental spaces, where mental spaces are restricted to the class of the human mind. For a rigorous discussion on cognition, we are restricting its scope and level to that of mathematical thinking.

1.2 Cognitive Science

The new characteristic of cognitive science lies in the study of the mind with respect to both the brain and the machine. Therefore, we need to consider the wide range of the mind, the brain, and the machine in terms of cognition. Thus only a valid theory for all these systems can lead to a viable theory in cognitive science.

More formally, we may define cognitive science as the study of the union of connectionist and symbolic approaches. Since what we refer to as the human mind is regarded as a product of the human brain and the computing machine, we can obtain two mathematical projections; one being from the mind class to the brain class, called the *connectionist approach*; and the other from the mind class to the computing machine class, called the *symbolic approach*. Thus, the former is concerned with mind-as-computing machine, whereas the latter is concerned with mind-as-computing brain. The metaphors of the brain and the computer will be used, respectively, to designate these two approaches. Cognitive science can be studied on the assumption that we can learn about the mind from studying the union of the brain and the machine.

From this assumption, we arrive at two formal models for an artificial mind: the Turing machine for the symbolic approach and the neural network for the connectionist one. Both models leads to the fundamental question in cognitive science: *How does one determine whether or not human cognition can be approached as artificial cognition?*

According to Roger Penrose [1994; 1997], there are at least four viewpoints concerning the relationship between mathematical thinking and computation: A. All thinking is computation; B. Awareness is a feature of the brain's physical action; C. Appropriate physical action of the brain evokes awareness; and D. Awareness cannot be explained by physical, computational, or any other scientific terms. Penrose then analyzes that Turing's viewpoint is contained within A, that is, the so-called "Strong AI" or "computational functionalism," and Gödel's view in D, that is, the mystical category [1994, pp.127-129; 1997, pp.112-113].

In spite of Penrose's contribution to showing the relevance of Gödel and cognitive science, I do not agree with him on the interpretation of Gödel's mystical position. First, his classification is not satisfactory in that viewpoint A is directed toward computation and thinking; B toward physical action, computation, and thinking; C toward physical action, computation, and thinking; and D toward physical action, computation, and thinking. He should, therefore, have indicated at least eight(2^3) viewpoints for a complete categorization. However, Penrose does not give any reason as to why the remaining viewpoints are omitted.

Secondly, less convincing is Penrose's assertion that Gödel's viewpoint is mystical(D). Although he uses Gödel's own statements [Gödel 1951] as evidence, the reference is very limited in that it does not show that Gödel refuted viewpoint A. Contrary to Penrose's claim, Gödel's conclusive disjunction [Gödel 1951] can be viewed as being logically consistent not only with viewpoint D, but also with viewpoint A.

To analyze Gödel's viewpoint and the implications of his celebrated incompleteness theorems for cognitive science, we need to address the disjunction because, according to Gödel [1951, p. 310] it is clearly inevitable with regard to both the human and artificial minds.

2. Gödel on Cognitive Science

2.1 Gödel's Incompleteness Theorems

We say that a formal system S is *sound* for a formula Π_1^0 in S whenever Π_1^0 is true in the structure of natural numbers if S proves Π_1^0. It follows that a formal system S is consistent if and only if S is sound for Π_1^0. Here, we may consider the Π_1^0 formula as $\forall y P(x,y)$ with a free variable

x and a decidable predicate P.[1] Let $Pf_s(x,y)$ be a decidable binary predicate expressing that y is a proof of the formula x in S.[2] Then $\forall y \neg Pf_S(x,y)$ is in the class of Π_1^0 formulas. Gödel constructed a Π_1^0 sentence, i.e., a formula without free variables, such as

$$G(S) \equiv \forall y \neg Pf(\gamma, y),$$

where γ is the Gödel number of $G(S)$.

THEOREM (Gödel's First Incompleteness) If a formal system S is consistent then $G(S)$ is not provable in S and $\neg G(S)$ is not provable in S.

THEOREM (Gödel's Second Incompleteness) If a formal system S is consistent then its own consistency, denoted by $Con(S)$, is not provable in the system.

The logical result of Gödel's theorems is clear. However, far from obvious remain the implications for the relationship between human cognition and artificial cognition. A great amount has been written on the implications of Gödel's Incompleteness Theorems for human cognition and artificial cognition, or for the mind-machine or brain-machine controversy: *Is the human mind, or the human brain, essentially superior to machines?*

The speed-up theorem[Gödel 1936] is relevant for increasing the range of the computing machine by adding new instructions. It is well known that Gödel had already shown that a logic of a higher order could prove formulas that a logic of lower order could not prove. This theorem implies that a certain function has no best algorithms. S. Feferman [1998, p.229] articulates this speed-up aspects of Gödel's Incompleteness Theorems from Gödel's own footnote 48a [Gödel 1931], in which Gödel clarified: "undecidable propositions constructed here become decidable whenever appropriate higher types are added." Feferman then calls it the *Gödel's doctrine*: the unlimited transfinite iteration of the power-set operation is necessary to account for finitary mathematics. According to him, the true reason for the incompleteness phenomena is that the formation of ever higher types can be continued into the transfinite in systems using types. It can lead to a new application of Gödel's theorems to cognitive science.

Nevertheless, the finite description issue and the consistency issue of cognitive systems in effect remain unresolved. This presents two problem to the cognitive scientist. One is to disregard the computational model of neural networks. The other is to erroneously apply Gödel's Incompleteness Theorems to the issue of cognition in natural or artificial systems. One of the best ways to avoid this problem is to analyze Gödel's own view on both the human and artificial minds. To understand Gödel's viewpoint is one thing, to use Gödel's theorems is another. It is thus necessary to precisely distinguish Gödel's own argument from so-called Gödelian arguments.

2.2 Gödel's Disjunctive Conclusion

According to Gödel, there seems to be two alternatives on the equivalence of the human mind and finite machines. Gödel asserted that the following disjunctive conclusion inevitable with respect to the undecidable:

> *Either mathematics is incompletable in this sense, that its evident axioms can never be comprised in a finite rule, that is to say, the human mind (even within the realm of pure mathematics) infinitely surpasses the powers of any finite machine, or else there exist absolutely unsolvable diophantine problems of the type specified* (where the case that both terms of the disjunction are true is not excluded, so that there are, strictly speaking, three alternatives). It is this mathematically established fact which seems to me of great philosophical interest. [italics in original, Gödel 1951, p.310]

For Gödel, it is the *intuitionists* in the foundation of mathematics who assert the first alternative of the disjunction and negate the second part. Gödel regarded *finitists* as opponents of the first disjunctive term. Epistemologically, these terms are more precise expressions rather than mechanism or anti-mechanism. Thus, for intuitionists and finitists, the theorem holds as an implication rather than a disjunction [see Gödel

[1] The class of Π_1^0 formulas is one of the form $\forall x P(x)$ representing for all variables x the predicate P holds, where the predicate P is a *decidable* property of natural numbers.

[2] $Pf_s(x,y)$ is a decidable relation between the two natural numbers x and y, that is, an algorithm exists to decide, for each choice of value of x and y, whether or not $Pf_s(x,y)$ holds.

1951, footnote 15]. Such a disjunctive conclusion illustrates that Gödel's position cannot be reduced to a mystical one.

The first alternative, however, does not necessarily mean that the incompleteness theorems preclude the existence of an idealized AI produced by a finite rule[Gödel 1951, p.312]. Although Gödel did not accept Turing's argument[Gödel 1972, p.306], he never claimed that the theorems refuted Turing's mechanistic view of mind. Intuitionists and finitists focused on only one alternative term, whereas Gödel refuted one logical implication. Those alternatives were not mutually exclusive. Rather, Gödel was firmly convinced of the truth of both(the third alternative). We, therefore, cannot simply say that Gödel's incompleteness theorems imply one specific alternative.

3. Cognition as a Turing Computable Function

Turing's analysis transformed the term *finite procedure* into the term *mechanical procedure.* Consequently, a function is computable or effectively calculable if it can be calculated by a finite mechanical procedure, that is, Turing machine. A function is Turing computable if it is definable by a Turing machine [Turing 1936].[3] Formally, a Turing machine is a function TM such that for some natural number n,

$$TM:\{1,2,\ldots,n\}\times\{0,1\}\rightarrow \{0,1\}\times\{L,R\}\times\{0,1,2,\ldots,n\},$$

where L stands for "move one left" and R "move one right." We should note that Gödel [1946; 1951; 1963] endorsed this concept as a generally accepted property of effective calculability, but not as a general recursion defined by himself.

Turing machine is a finite automata with unlimited tape as a memory device. As we remarked, it is mathematically equivalent to the class of the Herbrand-Gödel-Kleene equation system, i.e., the class of general recursive functions [Gödel 1934].

The Halting problem unsovable by a Turing machine is a question about Turing machines themselves, causing a metamathematical rather than mathematical question. The Halting function for the Turing machine is a mechanical implementation of Gödel's undecidable sentences.[4] Turing [1936] demonstrated the limitation of Turing computability, proving that there are unsolvable problems, e.g., the Halting Problem, in the Turing machine system. This is equivalent to Church's theorem that the decision problem for first-order calculus is not solvable. These along with Gödel's theorems show the limitations of the first-order calculus system or of recursive machines. Thus, if the mind is a Turing machine and cognition is a Turing computable function, then the mind would not be able to compute the Halting functions, because they would not be in the class of cognition.

To overcome this computational limitation, Turing [1939] proposed an extension of his machine model. Referred to as the *oracle Turing machine,* this Turing machine features a special extra "read only" tape, called the oracle tape, on which is written the characteristic function of a set called *oracle* and whose symbols cannot be printed over. This idea gave rise to the important issues such as arithmetical hierarchy and relative recursiveness.

The oracle model is clearly more powerful than the old one, but it is also clear that the power comes from the addition of a function that was previously not computable. Hence, this led to a recursive function that accepts members of the uncountable N^N as inputs, which raises the problem of relative computations on recursive infinite functions. The extension model, however, still cannot give us any real idea of how to compute the Halting function [Parberry 1996]. Moreover, such an infinite machine is beyond the scope of our

[3] A Turing machine is specified by: (1) a list of states called by Turing machine configurations; (2) a finite alphabet of symbols including the blank; and (3) a finite list of instructions.

[4] There is no Turing machine M such that, for all e and n, if the Turing machine Gödel-numbered e produces something on input n then M produce 0 on input (e,n); if the Turing machine Gödel-numbered e produces nothing on input n then M produce 1 on input (e,n). This result is known as the effective unsolvability of the Halting Problem for Turing machine.

debate, for it does not satisfy the assumptions underlying the finite machine, the type specified, or the consistency condition.

4. Cognition as a Neural Computable Function

The term "neural networks" may refer to the circuitry of real brains. By a neural network, we mean a formal model of the brain rather than an actual brain. A classical neural network is said to be a collection of MP(McCulloch-Pitts) formal neurons [Arbib 1987]. Unifying the studies of neurophysiology and mathematical logic, W. McCulloch and W. Pitts [1943] formulated a formal neuron model as a threshold unit that could act as a control device for any Turing machine. McCulloch and Pitts offered a brain model of the computable, whereas Turing offered a mind model of the computable. It is well known that the MP neural network is equivalent to finite automata [Kleene 1956] and that any finite automaton can be simulated by a MP neural network [Arbib 1964; Minsky 1967]. Thus any computation by a Turing machine can be performed by a neural network [Franklin & Garzon 1991].[5]

Formally, a neural network is regarded as an arbitrarily graph of a mapping. If the nodes, the formal neurons, are finitely numbered, then the neural network is finite; otherwise, it is infinite. Like the Turing machine, the language of finite neural networks consists of finite alphabets, has some activation rules as the inference rule, such that $y(t+1)=1$ if and only if $\sum_i x_i w_i(t) \geq \theta$, where x is inputs, y one output, $t=1,2,\ldots$ time scale, θ threshold, and w weights. We may define a finite neural network as a 5-tuple $NN = < V, X, Y, E, g >$, where V is a finite ordered set of nodes, $X \cap V = \varnothing$ is a set of inputs, $Y \subseteq V$ is a set of outputs, $< V \cup X, E >$ is a weighted graph, and $g: V \to F$ is a node assignment function (F is the node function set). Thus, this can be characterized by a function to a set of outputs from a product of input set, weight set, and threshold set.

Theoretically, neural ntworks with even Boolean weights are more powerful than the Turing machines. The computability of these neural networks has been explored by scholars such as S. Franklin and M. Garzon [1996] and H. Siegelmann and E. Sontag [1992]. Franklin and Garzon prove that the Halting problem for the Turing machine is solvable in an infinite recurrent Boolean neural network. The Halting function is thus computable in this neural network, and hence the computability of neural networks is greater than Turing computability [Garzon 1995]. Siegelmann and Sontag prove that any function computable by a Turing machine can be computed by a finite recurrent neural network with rational weights. They show that their model can simulate a multitape Turing machine in linear time.

These results are convincing. However, they are possible because of the infinite property in their neural network model. Similarly, this property of infinitude may give rise to a certain Turing computability for the Halting function if the uncountable infinity for the oracle Turing machine is allowed.

If all the results of neural computability discussed here are accepted for the computability of the mind, one could clearly say that the mental computability of a neural network goes beyond the mental computability of a Turing machine. However, for a rigorous analysis of mental computability as neural computability, we must investigate the consistency of the finite neural networks that have been specified.

In order to analyze the consistency of a finite neural network, we need to introduce some preliminary metamathematical theorems to classify the order of the systems. These theorems include the Compactness theorem, the Löwenheim-Skolem theorem and the Lindström theorem. The Compactness theorem states that a set of sentences has a model if and only if each finite subset of the set of sentences has a model. The Löwenheim-Skolem theorem shows that if a set of sentences

[5] The input-output relation in the McCulloch-Pitts neural network MP acts as a kind of Boolean function. Thus there are non-computable functions such as XOR function, i.e., the exclusive or predicate [Minsky & Papert 1988]. However, MP can compute any Boolean function, (1) if the interaction of inputs to neurons is allowed for the given MP neural network [Arbib 1964]; and (2) if the hidden units are employed [Rumelhart, Hinton, & Williams 1986].

has a model then it has an at most countable model. The Lindström theorem proves that the first-order formal system is the maximal system that satisfies both the compactness theorem and the Löwenheim-Skolem theorem.

By the Lindström theorem, we can claim that the class NN of finite neural networks is not first-order definable. To do this, it is enough to show that NN is not a compact system if it is consistent. If NN is consistent, then it has a model, and hence all connection relations between nodes must be defined in NN. It is, however, observed that the collection of directed graphs of NN is not first-order definable. Let NN be the class of finite recurrent neural networks. There exist a connection that is not definable in NN. Let $connection_n(x,y)$ define the predicate expressing "there is a connection from node x to node y of length n." Then we have $\{\neg connection_n(x,y): n \geq 1\}$ which logically satisfies $\neg connection_n(x,y)$, for constants a, b in NN. Suppose NN is a first-order system. Then, by the Compactness theorem, there must be a natural number k such that $\{\neg connection_n(a,b): 1 \leq n \leq k\}$ logically satisfies $\neg connection_n(a,b)$. However, this is not valid. Hence, the predicate is not definable in the graphs of NN. Consequently, the connection relation of NN is not definable in NN, and thus we have a contradiction. Hence, NN is not a first-order system and is not compact.

In the representation power, we can claim that the finite neural network NN is more powerful than the first-order representing system such as Peano Arithmetic and the Turing machine. It is enough to prove that NN represents a Π_1^0 formula. Take the formula $\forall y \neg Pf(x,y)$ in section 2.1. The expression,

$$\begin{array}{l} \forall y_1 \neg Pf(x_1,y_1) \\ \qquad\qquad\qquad Computation[x_1,y_1,x_2,y_2] \\ \forall y_2 \neg Pf(x_2,y_2) \end{array}$$

is not representable in any first-order (sequential) computing system. This expression with a branching quantifier is equivalent to a second-order expression,

$$\exists P_1, \exists P_2 Computation[x_1, P_1(x_1), x_2, P_2(x_2)].$$

This formula is not representable in the first-order systems since it is being quantified over a set of variables. However, the same formula is representable in NN because connectionist formal systems can compute many things simultaneously. Hence, NN is not a first-order but second-order formal system. Moreover, even in the case of the Gödel sentence $Con(NN) \equiv \forall y \neg Pf(\gamma, y)$, where γ is the Gödel number of $Con(NN)$, there is no reason why NN cannot compute the Gödel sentence of NN. Although $x_1 = x_2$, no contradiction appears.

NN is not a first-order system. But suppose that NN were a first-order system. By Gödel's Second Incompleteness Theorem, NN is not a consistent system because it can compute such a Gödel sentence.

5. Conclusion

Main results may be characterized as follows:

1. Gödel's Incompleteness Theorems are consistent with his Speed-up theorem, Doctrine, and Disjunctive conclusion.
2. Gödel's view on the human mind and the finite machine asserts "Disjunctive Conclusion" not a specific alternative.
3. A recurrent finite neural network is a second-order, not first-order formal system.
4. Neural computability goes beyond Turing computability with respect to Gödel's incompleteness results.
5. In terms of neural computation, Gödel's Incompleteness implies neither the superiority of human cognition to artificial cognition nor the equivalence of them.
6. Gödel's Incompleteness shows that the cognitive computability requires an ever higher computability.

References

Arbib, M. A. (1964) *Brains, Machines, and Mathematics*, McGraw-Hill Book Company.

Arbib, M. A. (1987) *Brains, Machines, and Mathematics(2nd ed.)*, Springer-Verlag.

Feferman, S. (1998) *In the Light of Logic,* Oxford University Press.

Franklin, S. & Garzon, M. (1996) Computation by Discrete Neural Nets, In: *Mathematical perspectives on neural networks,* eds. P. Smolensky, M.C.Mozer, and D.E. Rumelhart, Lawrence Erlbaum Association.

Garzon, M. (1995) *Models of Massive Parallelism: Analysis of Cellular Automata and Neural Networks*, Springer.

Gödel, K. (1931) On formally undecidable propositions of Principia Mathematica and related systems I, In: *Kurt Gödel Collected Works I,* eds. S. Feferman, et.al. (1986), Oxford University Press.

Gödel, K. (1934) On undecidable propositions of formal mathematical systems, notes by S. C. Kleene and J. B. Rosser on lectures at the Institute for Advanced Study, Princeton, New Jersey, In: *Kurt Gödel Collected Works I.*

Gödel, K. (1936) On the length of proofs, In: *Kurt Gödel Collected Works I.*

Gödel, K. (1951) Some basic theorems on the foundations of mathematics and their implications, In: *Kurt Gödel Collected Works III: Unpublished Essays and Lectures*, eds. S. Feferman, et. al. (1995), Oxford University Press.

Gödel, K. (1963) Postscriptum to Gödel 1931, In: *Kurt Gödel Collected Works I.*

Gödel, K. (1972) Some remarks on the undecidability results, In: *Kurt Gödel Collected Works II,*eds. S. Feferman, et.al. (1990), Oxford University Press.

Kleene, S. C. (1956) Representation of events in nerve nets and finite automata, In: *Automata Studies,* eds.C. E. Shanon & J. McCarthy, Princeton University Press.

McCulloch, W. S. & Pitts, W. (1943) A logical calculus of the ideas immanent in nervous activity, *Bulletin of Mathematical Biophysics* 5:115-33.

Minsky, M & Papert, S. (1988) *Perceptrons (Expanded Edition)*, The MIT Press.

Minsky, M. (1967) *Computation: Finite and infinite machines*, Prentice-Hall, Englewood Cliffs.

Parberry, I. (1996) Circuit Complexity and Feedforward Neural Networks, In: *Mathematical perspectives on neuralnetworks*, eds. P. Smolensky, M. C. Mozer, & D. E. Rumelhart, Lawrence Erlbaum Associates.

Penrose, R. (1994) *Shadows of the Mind,* Oxford University Press.

Penrose, R. (1997) *The Large, the Small and the Human Mind*, Cambridge University Press.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986) Learning Internal Representations by Error Propagation, In: *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, vol.1: Foundations*, Rumelhart, D.E., McClelland, L., & the PDP Research Group, The MIT Press, pp.318-362.

Siegelmann, H. T. & Sontag, E. D. (1992) On the computational power of neural nets, *Proceedings of the 5th ACM Workshop on Computational Learning Theory*:440-449.

Siegelmann, H. T. (1998) *Neural Networks and Analog Computation: Beyond the Turing Limit*, Birkhäuser.

Turing, A. (1936) On Computable Numbers, with an application to the Entscheidungsproblem, *Proceedings of the London Mathematical Society* 42:230-265; A correction, ibid., 43(1937):544-546.

Turing, A. (1939) Systems of logic based on ordinals, *Proceedings of the London Mathematical Society* 2(45):161-228.