Computability and Humanity

Drew Morris

Brigham Young University

WRTG 316 Technical Communication

Professor Marsha Maxwell

10 July, 2023

Computability and Humanity

This literature review will examine how the limitations of computation can provide insight on the human condition and its limitations. Computation Theory is a branch of mathematics and computer science that studies the theoretical limitations of computers. These limitations provide benchmarks for which we can measure practical implementations and advancements against. Using some very abstract mathematics we can discuss a possible way to apply these computational limitations to the human condition and discuss the possible consequences of them. We will do this by first explaining what computability is and what limitations it has and then making a conjecture to show how the interactions of our universe can be considered computable. By the end of this essay, I hope to further our understanding of the question, “are there any differences between human intelligence and computer intelligence?”.

What Is Computability?

Most of us are familiar with computers. We use them to send emails, watch videos, write papers and so much more. Some of us have been fortunate enough to see the evolution of the computer over the past few generations: evolving from little more than a calculator to a near-inseparable aspect of our daily lives. Furthermore, in our modern age we have seen that, through machine learning, computers can create original works of art and can carry on in human conversation: performing in ways that were previously thought to be exclusively human. This sudden and rapid improvement in computer thinking can make us wonder what separates a human from a computer. At first, one might point out that computers require electricity to run or that they are made of inorganic materials. While this is true, it is a misunderstanding of the question. This practical view of a computer is very different from the vision of computers in computation theory. In computation theory, we study the limitations on what a computer can achieve outside of physical limitations such as time, memory, and energy. Sipser. (2019) This is called an idealized computer. For our purposes, idealized computers exist to solve problems and problems are any question that has a definitive and objective answer. Any such problem is computable if and only if it can be solved by a computer. Such a computer is extremely powerful and you may be wondering, “Wouldn’t such a machine be able to do anything?”. The answer is no. To understand why, we must understand how computers operate.

How Do Computers Operate?

All computers operate by sequentially executing atomic commands. Atomic commands are operations that are not made up of any smaller parts. In most modern computers (those using an Intel brand processor) these atomic commands are called the x86 architecture (or the equivalent y86 architecture). Bryant et al. (2016) To articulate the simplicity of the y86 architecture let's examine how the problem “Is x greater than 2?” is solved as shown by Bryant et al. (2017) To spare further explanation of the y86 architecture, the unfamiliar reader should think of y86 registers (Ex. %rax, %rdi) as mathematical variables. As such we begin by assuming the value of x is stored in the register, %rax.

main:

1. irmovl $2, %rbx (store the value, 2, in the register, %rbx)
2. subl %rbx, %rax (store the difference of %rax and %rbx in %rax)
3. rrmovl %rax, %eax (store the value of %rax in %eax)
4. jg true (if %eax is greater than 0, jump to true)
5. jle false (if %eax is less than or equal to 0, jump to false)
6. halt (end the program execution)

true:

1. irmovl $1, %ebp (store the value, 1, in the return register, %ebp)
2. ret (return the value in %ebp)

false:

1. irmovl $0, %ebp (store the value, 0, in the return register, %ebp)
2. ret (return the value in %ebp)

By this example, we can see that even a simple problem like "Is x greater than 2?" takes 7 steps to solve. Every computer, practical and idealized, has to use simple commands like those above to solve problems. This means that even our idealized computer has a theoretical limitation; it has to solve problems one simple step at a time. Sipser. (2019) Therefore, a problem is computable if and only if it can be solved in an orderable series of steps. One may say, "surely if the steps are orderable then there must only be a finite number of them. Otherwise, the computer could run forever." Once again, this is not necessarily the case.

What is Orderability?

Naturally, a collection is orderable if it can be ordered. In other words, a collection is orderable if you can arrange every item in said collection in a single-file line without missing any items. Nielsen. (2022) For example, we can order the numbers 1 through 10 in its standard order (1,2,3,4,5,6,7,8,9,10) or alphabetically (8,5,4,9,1,7,6,10,3,2) or in my favorite order, by the minimal number of steps to converge to 1 in the Collatz sequence (1,2,4,8,5,10,3,6,7,9). For now, we don't care what order a collection is in but rather just that it can be ordered. Naturally, we start to see that any finite collection of items is orderable, but are there any infinite collections that can be ordered? Yes. For example, there are an infinite amount of natural numbers and yet we can always determine which of two natural numbers is larger by its standard ordering (0,1,2,3,...). This kind of infinite size is known as countably infinite because of its original basis in "the counting numbers." By the definition of set cardinality (the size of sets), two sets are of equal cardinality if and only if there exists a bijection (a one to one function) between them. Doud et al. (2019) In other words, a set is countably infinite if it can be ordered like the natural numbers. From this we see that the integers are countably infinite via an alternating ordering (0,1,-1,2,-2,3,-3,...). Even stranger, we see that the rationals (which are infinitely dense under their standard ordering) are also countably infinite via Cantor's Diagonalization ordering. Doud et al. (2019) This ordering alternates like the integers but, to avoid confusion, the following is the ordering of only the positive rationals $(1,2\frac{1}{2}, 3,\frac{1}{3},4,\frac{3}{2}, \frac{2}{3}, \frac{1}{4}, 5, \frac{1}{5}, 6, \frac{5}{2}, \frac{4}{3}, \frac{3}{4}, \frac{2}{5}, \frac{1}{6},\ldots)$. In short, countable infinity is confusingly large. To further explore that, let us consider the following thought experiment famously known as Hilbert's Hotel.

Hilbert's Hotel

Imagine you are the owner of an infinite hotel. Your hotel has an infinite number of rooms, each with their own room number. To your surprise, you have no vacancies. Suppose someone walks into your infinite hotel with no vacancies and asks for a room. How could you fit this additional guest into your hotel? The answer is simple. Have every guest move down to the next room and place the new guest into the first room. Similarly, you can fit two new guests into your infinite hotel by having all your current guests move down two rooms and putting the two new guests in the newly opened, first two rooms. This shows us that adding a finite amount to countable infinity will still yield countable infinity. Andersen. (2021) Now suppose that an infinite bus with an infinite number of new guests pulls up to your hotel (with no vacancies) and they all request a room. Is it possible to fit every additional guest into your hotel? Yes. Just ask every current guest to move down to the room number double their current room number and put the new guests in all the odd numbered rooms. This shows us that multiplying countable infinity by a finite amount will yield countable infinity. Andersen. (2021) Now suppose that an infinite line of infinite buses pulls up to your hotel, which to reiterate has no vacancies, and every guest would like a room at your hotel. Surely you can't fit this many people in your hotel; however, you can. One way is to ask every current guest to move to the square of their current room number and add one guest from each of the buses (starting from the front) to each newly opened section of rooms. This shows us that raising countable infinity to a finite power still yields countable infinity. Andersen. (2021) Now the time has come to break the infinite hotel. Suppose your infinite hotel is entirely vacant and further suppose that you have been given a list of guests that you are to cater to. This list is composed of id strings of every possible combination of 0s and 1s of every possible length, including countably infinite length. No matter which order you decide to place these guests, there will always be a guest left without a room. To prove this, think of any ordering you like. Take the first digit of the first guest's id, the second digit of the second guest's id, and so on to create a new id. Now change every 1 in your new id to a 0 and every 0 to a 1. We now have the id of a guest who has no room number because, for every room number, their id is different from the assigned guest's id by at least one digit. This shows us when we begin to raise countable infinity to a countably infinite power we obtain a degree of infinity that is larger than the one we started with. Nielsen. (2022) Interestingly enough, this technique of describing things using arbitrarily long combination strings of finitely many symbols is called First Order Language and it is the foundation of computation: including the construction of processor architectures like the x86 and y86 architectures. Nielsen. (2022)

First Order Language

A First Order Language is any language that consists of a finite set of symbols representing constants, variables, and operations, whose rules are consistent with First Order

Logic. Nielsen. (2022) First Order Logic is the system of binary logic that most of us are familiar with using even if we are not aware of it. For example, if we know that if it is raining then the road will be wet and we know that the road is not wet then we can determine that it is not raining. This process of deduction is the foundation of First Order Logic. Additionally, First Order Languages are often used under a theory written in that language. A theory, in the context of a First Order Language, is a collection of statements written in said language that are assumed to be true without a logical proof: also known as axioms. Nielsen. (2022) Given any First Order Language, the set of strings (of finite length) that can be constructed under that language is countably infinite. Nielsen. (2022) This means that any such statement that can be written in such a language is computable. In other words, if we can describe a problem using a language of finitely many atomic symbols with countably many n-ary operators that are bound by the confines of First Order Logic then that problem is computable. Let's consider how this can apply to the material universe that we exist in.

Elementary Particles

According to the Standard Model of Physics, there are believed to be a finite number of fundamental or elementary particles (atomic symbols). Furthermore, these particles are believed to form compounds of only finitely many particles (set of strings of finite length). Finally, these particles are believed to have only finitely many possible interactions (n-ary operators). This model contains all the necessary components for making a First Order Language other than being determined by the confines of First Order Logic. Assuming this Standard Model of Physics, as described by Martin et al. (2020), is an accurate model of this universe and that all interactions under this model are determined by First Order Logic, we can describe our universe and everything in it using a First Order Language. This means that the set of every material interaction that has taken place, is taking place, and will ever take place, can be determined by an idealized computer that can read said First Order Language. This conclusion allows us to apply the limitations of computability to every process that exists in our material universe. Now, before we discuss these applications there are a few discrepancies that must first be addressed.

First, this application requires us to assume that our universe behaves in a logical manner and will continue to behave that way ad infinitum. Second, this application requires us to assume our universe does fit the confines outlined in this Standard Model of Physics. If, for example, there are actually an infinite number of elementary particles, this application will no longer apply to our material universe. Third, we must accept these prior two discrepancies as axioms as there is no way to deductively prove them due to the inductive nature of scientific proof. We will now take a moment to discuss this third discrepancy in more detail.

Inductive Logic vs Deductive Logic

There are two main families of logic. The first family is called Inductive Logic. Inductive Logic is the foundation of A Posteriori studies like biology, chemistry, and physics. A Posteriori directly translates to "of the posterior" meaning, in this case, that we apply our logical reasoning after we perform observations. Pojman et al. (2011) The benefit of using Inductive Logic is that proving a statement using it does not require us to assume any prior statement without proof: meaning that it does not require an axiom schema to function. For example, consider the following scenario given by Hogan. (2022) Suppose you are observing the organism populations of a local lake; specifically, you are studying the characteristics and their frequency of the geese that inhabit said lake. Suppose that you have been observing this lake for 3 months and over that time you have recorded over 5,000 geese sightings. Notably, every goose that you have observed has had all black feathers. Therefore, you conclude that every goose in this lake is black. Sadly, this conclusion is not a certain one and this shows the main issue of Inductive Logic. Inductive Logic can only prove with statistical confidence and never with certainty.

The second family is called Deductive Logic. Deductive Logic is the foundation of A Priori studies like mathematics and computation theory. A Priori directly translates to "of the prior" meaning, in this case, that we apply our logical reasoning before we perform observations. Pojman et al. (2011) The benefit of using Deductive Logic is that, when a statement is proven using it, that statement is true with absolute certainty. For example, consider the following proof of "2 + 2 = 4" given by Nielsen. (2022) For context, "++" represents the successor function or "+1" function.

1. Assume Robinson Arithmetic
2. 2 = 1++
3. 1 = 0++ (Axiom 2)
4. 0 exists (Axiom 2)
5. 2 + 2 = 2 + 1++ (Axiom 1)
6. 2 + 1++ = (2 + 1)++ (Modus Ponens)
7. 2 + 1 = 2 + 0++ (Axiom 4)
8. 2 + 0++ = (2 + 0)++ (Modus Ponens)
9. 2 + 0 = 2 (Axiom 4)
10. (2 + 0)++ = 2++ (Axiom 3)
11. 2++ = 3 (Modus Ponens)
12. (2 + 1)++ = 3++ (Axiom 2)
13. 3++ = 4 (Modus Ponens)

∴ 2 + 2 = 4 (Axiom 2)

(Modus Ponens)

Notice how this proof depends on assuming the axioms of Robinson Arithmetic to work. This is the main issue of deductive logic. Every proof using Deductive Logic requires that we assume some set of statements (axioms) without proof and even if we were to prove said axioms, that would require us still assume some axioms (new or otherwise) to prove them. Otherwise, our axioms would be self-proving: otherwise known as circular reasoning whose proofs are of no substance.

Computation Theory is a deductive science which means our prior discussion and proofs in Computation Theory are true with certainty given the axioms of Computation Theory. Physics is an inductive science which means that even if we were to discover a Standard Model of Physics that perfectly describes the material universe that we exist in we would never know that that Standard Model of Physics is perfect with certainty. Furthermore, the application of any deductive science to the material world is in itself an inductive claim which falls victim to the same irresolvable uncertainty. Thus, as we move forward into discussing how these computational limitations affect the human condition, keep in mind the assumptions that have been made to allow us to make these applications.

What is Humanity

We have finally come to the conclusion of this literature review where we will now discuss the human condition in this new light of computable limitation. Think of everything you can that makes humans special: empathy, agency, sentience, imagination, creativity, etc. Where do these human characteristics come from? Why do these characteristics exist in humans and furthermore, why do humans believe that they are unique and exclusive traits to their species, or rather, existence?

To the reader who believes that these characteristics are the result of some immaterial or metaphysical phenomenon (such as containing a spirit, like in most Abrahamic religions, or being separated from a metaphysical singularity, like Nirvana) I present a bittersweet conclusion. On the one hand, your belief yields that your humanity is indeed special and unique to you and all other humans. This is due to the fact that all of the discussion thus far has built a connection between computability and material existence specifically. We did this because such a connection between computability and immaterial existence could only be proven trivially: meaning that it could only be true by direct assumption or by circular reasoning. All immaterial and metaphysical proofs are this way and, while they are logically fallacious, the beliefs they defend are not inherently wrong or bad. For example, on multiple occasions I have encountered the argument that the Abrahamic God is real because the Bible says so. Some slight examination however reveals that the Bible is only true if that God is real. Thus, this argument is circular and this proof is trivial. Nonetheless, there are many people who have done amazingly charitable acts only because they believe that the Abrahamic God is real. Similarly, many people have committed heinous atrocities for the exact same reason. It is important to see that any statement of a nature that cannot be observed falls victim to such an end; it can never be proven nor disproven. Instead, the adoption of such a view can only be done by a "leap of faith" as stated by Kierkegaard. Pojman et al. (2011) In other words, when presented with a statement that can neither be proven nor disproven it is the burden of the observer and them alone to adopt that statement as truth.

Now, to the reader who instead believes that these characteristics are a result of our material composition (regardless of its origin) I present what I believe to be a far more interesting conclusion. Because our material composition is computable by our previous discussion, all resulting behaviors and phenomena of said composition are also computable. This means that there is no fundamental difference between that which can be achieved by a computer and that which can be achieved by a human. Every musical composition, mathematical proof, scientific theory, theatrical performance and all other conceivable creations are not unique to humans. This is nothing to be afraid of but it is important. All this really means is that anything we believe that humans could do, computers could do also. Similarly, anything we believe that computers cannot do, humans cannot do also. Can humans make decisions that are truly their own? If so, then computers can also make decisions that are truly their own. Can computers feel genuine emotions? If not, then neither can humans. Perhaps such a thought could further our understanding of both computers and humanity itself. Regardless, I invite every reader to be mindful of the decisions you make and why.

References

Doud, D., & Nielsen, P. P. (2019). Understanding set cardinality and infinity. *A Transition to Advanced Mathematics* (1st ed., pp. 215-242). Brigham Young University.

Nielsen, P. P. (2022). Understanding first order logic, first order languages, and mathematical decidability. *A Journey Through Formal Mathematics* (1st ed., pp. 23-166). Brigham Young University.

Sipser, M. (2019). Understanding theoretical computers, Turing machines, computability, and incomputibility. *Introduction to the Theory of Computation* (M. Lee, Ed.) (3rd ed., pp. 193-272). Cengage Learning.

Bryant, R. E., & O'Hallaron, D. R. (2017). Understanding machine-level computing, processor architecture, and atomic-level programming. *Computer Systems: A Programmer's Perspective* (M. Goldstein & K. Loanes, Eds.) (3rd ed., pp. 163-494). Pearson Education.

Andersen, N. (2021, December 3). Understanding countable infinity and axiomatic construction. *Constructing the Naturals, Integers, and Rationals.* [Lecture].

Martin, S. P., & Wells, J. D. (2020). *Elementary Particles and Their Interactions* (1st ed.). Springer Charm.

Hogan, J. (2022, July 14). Understanding the differences between Deductive and Inductive Logic. *Kant, the Categorical Imperative, and the Unification of Rationalism and Empiricism.* [Lecture].

Pojman, L., & Vaugh, L. (2011). Understanding Immanuel Kant and Soren Kierkegaard. *Classics of Philosophy* (3rd ed., pp 815-920, 940-952). Oxford University Press.