NOÛS 41:4 (2007) 686–713

The Turing Test as Interactive Proof

STUART M. SHIEBER
Harvard University

Abstract

In 1950, Alan Turing proposed his eponymous test based on indistinguishability of verbal behavior as a replacement for the question "Can machines think?" Since then, two mutually contradictory but well-founded attitudes towards the Turing Test have arisen in the philosophical literature. On the one hand is the attitude that has become philosophical conventional wisdom, viz., that the Turing Test is hopelessly flawed as a sufficient condition for intelligence, while on the other hand is the overwhelming sense that were a machine to pass a real live full-fledged Turing Test, it would be a sign of nothing but our orneriness to deny it the attribution of intelligence. The arguments against the sufficiency of the Turing Test for determining intelligence rely on showing that some extra conditions are logically necessary for intelligence beyond the behavioral properties exhibited by an agent under a Turing Test. Therefore, it cannot follow logically from passing a Turing Test that the agent is intelligent. I argue that these extra conditions *can* be revealed by the Turing Test, so long as we allow a very slight weakening of the criterion from one of logical proof to one of statistical proof under weak realizability assumptions. The argument depends on the notion of interactive proof developed in theoretical computer science, along with some simple physical facts that constrain the information capacity of agents. Crucially, the weakening is so slight as to make no conceivable difference from a practical standpoint. Thus, the Gordian knot between the two opposing views of the sufficiency of the Turing Test can be cut.

1. Introduction

In this paper, I attempt to reconcile two mutually contradictory but well-founded attitudes towards the Turing Test, Alan Turing's proposed replacement for the question "Can machines think?" On the one hand is the attitude

that has become philosophical conventional wisdom, viz., that the Turing Test is hopelessly flawed as a sufficient condition for intelligence, while on the other hand is the overwhelming sense that were a machine to pass a real live full-fledged Turing Test, it would be a sign of nothing but our orneriness to deny it the attribution of intelligence.

The arguments against the sufficiency of the Turing Test for determining intelligence rely on showing that some extra conditions are logically necessary for intelligence beyond the behavioral properties exhibited by an agent under a Turing Test. Therefore, it cannot follow logically from passing a Turing Test that the agent is intelligent. I will argue that these extra conditions *can* be revealed by the Turing Test, so long as we allow a very slight weakening of the criterion from one of logical proof to one of statistical proof under weak realizability assumptions. Crucially, this weakening is so slight as to make no conceivable difference from a practical standpoint. Thus, the Gordian knot between the two opposing views of the sufficiency of the Turing Test can be cut.

1.1. The Essence of the Turing Test

The Turing Test is, at its heart, a test of the adequacy of an agent's verbal behavior. Block (1981) characterizes it as a test of the ability to "produce a sensible sequence of verbal responses to a sequence of verbal stimuli."[1] Turing's original presentation of the Test is couched in terms of an imitation game between two entities, a person and a machine, with the goal of seeing if in repeated forced choices a judge can do no better than chance at determining which is which on the basis of verbal interactions with each. Much of this setup (and the preliminaries that he introduces regarding a gender-based version of the game) are incidental to the underlying goal, which is to determine if a machine has human-level verbal behavior.

The introduction of the human confederate and the forced choice merely serve to make more clear and operational what constitutes "sensibility" of the machine's responses, but there are other ways to achieve the same goal. For instance, the underlying idea could be implemented in a simpler form, in which a judge merely stipulates whether or not a machine has exhibited human-level behavior, except that without some sort of forced choice, a gaming of the test would be possible. Indeed, Turing presents this simpler more direct form in a little known 1952 BBC interview in which he describes the test as follows:

> The idea of the test is that the machine has to pretend to be a man, by answering questions put to it, and it will only pass if the pretence is reasonably convincing.... We had better suppose that each jury has to judge quite a number of times, and that sometimes they really are dealing with a man and not a machine. That will prevent them saying "It must be a machine" every time without proper consideration. (Newman, Turing, Jefferson, and Braithwaite 1952)

Here, he describes the point of the Test directly in the first sentence, and makes clear that the comparison issue (whether through repeated trials, as described in this selection, or one-on-one, as in the original paper) is an expedient to make the forced choice a real one.

Thus, at base, the Turing Test is a test based on the idea that ability to produce sensible verbal behavior is an indication of intelligence. The syllogism that underlies the appropriateness of the Turing Test as a criterion for intelligence proceeds something like this:

> **Premise 1:** If an agent passes a Turing Test, then it produces a sensible sequence of verbal responses to a sequence of verbal stimuli.
>
> **Premise 2:** If an agent produces a sensible sequence of verbal responses to a sequence of verbal stimuli, then it is intelligent.
>
> **Conclusion:** Therefore, if an agent passes a Turing Test, then it is intelligent.

Block refers to a premise such as the second one as the "Turing Test conception of intelligence," and his (and others') repudiation of the Turing Test as a criterion for intelligence is based on a denial of this premise.

1.2. The Conceptual Basis for Turing-Test Denial

Philosophers of mind fall, roughly speaking, into two camps, the Turing-Test deniers, who think that passing a Turing Test cannot be used as a sufficient condition for intelligence, and the Turing-Test approvers, who think that it can. Turing-Test deniers think of intelligence like a bad cold. It has a hidden cause, a germ. Victoria can say of her friend Peter without sounding ridiculous things like "Oh, Peter's not really sick; he's just faking it to get out of school." Sickness can't be cashed out in terms of some disposition to exhibit sickness symptoms (coughing, complaining of stomach pain, staying in bed). There has to be a germ.

Turing-Test approvers, on the other hand, think of intelligence like being fluent in Italian. (In fact, they think it's *exactly* like being fluent in Italian.) Imagine you've been talking for an hour with Victoria's friend Pietro using perfect Italian. Now suppose Victoria were to say, "Oh, Pietro's not really fluent in Italian; he's just faking it to be eligible for an Italians-only scholarship." Such a statement is clearly silly. One can't exhibit the symptoms of being fluent in Italian and be faking, missing some essential "germ" of fluency; the symptoms *are* the fluency.

Now imagine a scenario in which Peter has been getting straight A's in school and just got two 800's on the SAT. Victoria says "Oh, Peter's not really intelligent; he's just faking it to get into a good school." Intelligence in this sense (which is not, of course, the sense that the Turing Test is meant to test for) is clearly like fluency in Italian, which is why the statement sounds ridiculous.

Finally, imagine Victoria takes you to a Searlian "Italian room" where you can insert slips of paper with Italian written on them through a slot in the door and get back other slips of paper with perfectly fluent Italian responses, sometimes clever, sometimes amusing, always insightful; the room is a brilliant conversationalist. After an hour or so of this, you're quite impressed, but Victoria, ever the spoilsport, says "Oh, that room isn't intelligent; it's just faking it." If you think that sounds silly prima facie, you can see why the Turing-Test deniers' view is so counter-intuitive. They seem to think that one could have the symptoms without the germ. Different philosophers diagnose this necessary causal agent differently. Searle (1980) thinks the germ is intentionality (though Dennett (1987) objects that Searle thinks it's consciousness); Davidson (1990) thinks it's semantics; Gunderson (1964) thinks it's flexibility of behavior; Block (1981) thinks it's "richness of information processing". But all (except Dennett) agree that intelligence is not testable in purely behavioral terms.

On the other hand, many find it hard to shake the intuition that a Turing-Test-passing entity must surely be intelligent. To such Turing-Test approvers, like Dennett (1985), no germ is necessary. "[T]he Turing test, conceived as he conceived it, is (as he thought) plenty strong enough as a test of thinking. I defy anyone to improve upon it." This intuition is quite strong. Nonetheless, intuitions may be wrong and a little philosophy might be just the thing to lead us to accept previously counterintuitive conclusions, for instance, that sentences like "that machine is just faking intelligence" aren't ridiculous at all.

1.3. The Argument Against Behaviorist Tests

In "Psychologism and Behaviorism", Block (1981) presents what I take to be the strongest argument to date of the inadequacy of the Turing Test as a criterion of intelligence. Through a series of thought experiments, Block argues that no conception of intelligence that relies solely on external behavior (as manifested in Premise 2) can be sufficient; some (at least minimal) internal conditions on the means by which the behavior is generated must be included. In particular, he faults the Turing Test for failing to demonstrate not only the fact of producing "a sensible sequence of verbal responses to a sequence of verbal stimuli" but of a general capacity for such behavior, and further, one derived from sufficient "richness of information processing"; the antecedent in Premise 2 is too weak. Because I think this is the strongest argument against the Turing Test as a sufficient condition of intelligence, it is the argument that I address in this paper. I argue that the Turing Test can in fact provide such a demonstration, thereby vitiating Block's argument against the sufficiency of the Turing Test as a test of intelligence.

Searle, in his "Minds, Brains, and Programs" (1980), presents a different argument against the Turing Test, his "Chinese room". This argument is based on an article of faith that is too woolly to argue against, namely, that

no formal system that merely manipulates symbols could bear intelligence. But Block doesn't go that far,[2] and indeed has argued against Searle on this point (Block 1980). Block is saying something simpler, that it is logically possible that some thing that not only is merely a symbol manipulator *but also is a trivial one* could pass the Turing Test. Furthermore, it not only can pass the Turing Test, but has a general capacity to do so. But if Block is right, why would we be inclined to attribute intelligence to a machine that passed a Turing Test?

It seems to me that Block *is* right in principle: Such a machine is conceptually possible; hence the Turing Test is not *logically* sufficient as a condition of intelligence. Let us suppose this view is correct and, as Block argues, some further criterion is needed regarding the manner in which the machine works. Some further criterion is needed, but how much of a criterion is that, and can the Turing Test test for it? Although Block calls this further internal property 'nonbehavioral', I will argue that *the mere behavior of passing a Turing Test can reveal the property*. Borrowing an idea from theoretical computer science, I argue that the Turing Test can be viewed as an interactive proof not only of the fact of sensible verbal behavior, but of a capacity to generate sensible verbal behavior, and to do so "in the right way". Assuming some extraordinarily weak conditions on physical realizability, any Turing-Test-passing agent must possess a sufficient property to vitiate Block's argument. In summary, Block's arguments are not sufficient to negate the Turing Test as a criterion of intelligence, at least under a very slight weakening of the notion of 'criterion'.

The argument I present does not demonstrate that the Turing Test is sufficient as a criterion for intelligence. It merely shows that Block's argument against its sufficiency fails. However, some other argument might hold; this possibility remains open.

2. Motivation

Before I argue for the resurrection of the Turing Test as a sufficient condition of intelligence, it merits mention of why such an argument is worth undertaking in the first place. Discussions such as the present one (and Block's) for or against the Turing Test as a definition or necessary or sufficient condition for intelligence might be denigrated (and have been) on the grounds that Turing didn't propose his Test as a criterion of intelligence. Rather, Turing wanted to *replace* the question "Can machines think?" with the question "Can machines pass the Turing Test?" But philosophers just won't listen. They insist on investigating the issue of whether the Turing Test is a good definition of intelligence, despite Turing's best efforts to avoid definitions entirely.

A few voices have kept up pressure to stop such useless bickering. "It is a sad irony that Turing's proposal has had exactly the opposite effect on the discussion of that which he intended," says Dennett (1985). "Alas,

philosophers—amateur and professional—have instead taken Turing's proposal as the pretext for just the sort of definitional haggling and interminable arguing about imaginary counterexamples he was hoping to squelch." Chomsky's view is that "Turing's sensible admonitions should also be borne in mind, more seriously than they sometimes have been, in my opinion." (Chomsky 2004)

But how can we know that Turing's test is an adequate replacement for the question "Can machines think?" if we can't compare the results of the Test with the corresponding answers to the question? I could request replacing the question "Can machines think?" with a test of their ability to perform arbitrary precision square roots, but one would be within rights to note that this is not a useful replacement. As Moor (1976, page 250) points out, "if Turing intends that the question of the success of the machine at the imitation game replace the question about machines thinking, then it is difficult to understand how we are to judge the propriety and adequacy of the replacement if the question being replaced is too meaningless to deserve discussion. Our potential interest in the imitation game is aroused not by the fact that a computer might learn to play yet another game, but that in some way this test reveals a connection between possible computer activities and our ordinary concept of human thinking." Thus, philosophers have been inexorably led to the question of the relationship between a machine's passing of the test and its thinking capacity.

Turing finds himself sliding down the slippery slope from replacement to definition for just this reason. "We cannot altogether abandon the original form of the problem, for opinions will differ as to the appropriateness of the substitution and we must at least listen to what has been said in this connection." (Turing 1950) He discusses, for instance, whether the test should be thought of as a necessary or sufficient condition for attributing intelligence, finding for the latter only.

I therefore take seriously the issue as to whether passing a Turing Test is a sufficient criterion for intelligence. (The arguments against the Turing Test as a necessary condition, and therefore as a definition, of intelligence are simple, clear, uncontroversial, and need not be restated. The views of French (1990) are particularly trenchant on the matter.) In the next section, I rehearse conventional philosophical wisdom on the matter (Dennett notwithstanding).

3. Turing Test Conceptions of Intelligence

Whether one thinks that the Turing Test is a sufficient condition for intelligence or not depends in large part on one's interpretation of particular aspects of the role of the Turing Test in the stating of the condition. In Block's phraseology, it depends on the "Turing Test conception of intelligence" that one has in mind. Block takes the upshot of passing a Turing Test as demonstrating that the subject-under-test can "produce a sensible sequence of verbal

responses to a sequence of verbal stimuli." The Turing Test conception of intelligence thus provides the connection between such production and the possession of intelligence. In its most direct form, the relation is expressed as in Premise 2 above, repeated here under the name "the occasional conception of intelligence".[3]

> **The occasional conception:** If an agent produces a sensible sequence of verbal responses to a sequence of verbal stimuli, then it is intelligent.

This conception, together with Premise 1—which asserts that passing a Turing Test demonstrates the antecedent—allows the conclusion that the agent is intelligent.

It is simple to argue that this conception (admittedly a straw man, as no one to my knowledge, including Turing, has ever claimed it) is flawed. Imagine a machine that responds to the interrogator's queries by emitting a random sequence of keystrokes. (The idea is conventionally implemented using monkeys and typewriters.) There is some (admittedly astronomically small) probability that these keystrokes will fortuitously spell out perfectly plausible responses to the queries, and the interrogator would therefore be fooled into confusing the random keystroke generator with a human. If one holds the stance that the random typing responses were not true intelligent behavior (and why would they be?), then the *mere possibility* of such an occurrence, by itself, demonstrates that passing the Turing test is not a sufficient condition for intelligent behavior, at least under the occasional conception.

Of course, even Turing admitted as much. He thought of the test as having a statistical component, requiring more than a single occasion of passing. This is clear from his 1952 interview, quoted above. His statements about passing the Test were statistical too, as in his famous prediction that "an average interrogator will not have more than 70 per cent. chance of making the right identification after five minutes of questioning." (Turing 1950)

But subjecting the monkeys to multiple Tests, or longer ones, doesn't solve this problem; it merely adjusts the odds of a false positive. Instead, what is needed is a change in the conception of intelligence, along the lines that Block argues for in "Psychologism and Behaviorism". I skip ahead to his "neo-Turing-Test conception of intelligence", which I will call

> **The capacity conception:** If an agent has the capacity to produce a sensible sequence of verbal responses to a sequence of verbal stimuli, whatever they may be, then it is intelligent.

Arguably, this revised conception already gives up on a purely behaviorist view. How could one know that an agent has a (counterfactual) capacity without resort to analysis of its internal workings, that is, to a theory of its behavior, rather than a mere sample of it? In particular, how could the fact

of passing one or more Turing Tests allow one to conclude the antecedent of *this* conception?

I will pay this promissory note later. But Block is willing to grant capacities to the Turing-Test approvers anyway, ex hypothesi, because he has an argument even against this conception.

Imagine (with Block) a hypothetical machine that stores a tree of interactions providing a sensible response for each possible interrogator's input in each possible conversational context of up to, say, one hour long. (These responses might be modeled on those that Block's fictional Aunt Bertha would have given.) Such a tree would undeniably be large, but processing in it would be conceptually straightforward. By hypothesis, such an "Aunt Bertha machine" would pass a Turing Test of up to one hour, because its responses would be indistinguishable from that of Aunt Bertha, whose responses it recorded. Such a machine is clearly not intelligent, by the same token that the teletype that the interrogator interacts with in conversation with the human confederate in a Turing Test is not intelligent; it is merely the conduit for some other person's intelligence, the human confederate. Similarly, the Aunt Bertha machine is merely the conduit for the intelligence of Aunt Bertha. Yet just as surely, it can pass a Turing Test, and more, has the *capacity* to pass arbitrary Turing Tests of up to an hour. The mere logical possibility of an Aunt Bertha machine is sufficient to undermine the capacity conception.[4]

Block pursues a number of potential objections to his argument that the capacity conception is flawed, the most significant of which (his Objection 8) is based on the fact that the Aunt Bertha machine is exponentially large, that is, its size is exponential in the length of the conversation.

Objection 8 leads to his "amended neo-Turing-Test conception": "Intelligence is the capacity to emit sensible sequences of responses to stimuli, *so long as this is accomplished in a way that averts exponential explosion of search*" (emphasis in original). It is not exactly clear what "exponential explosion of search" is intended to indicate in general. In the case of the Aunt Bertha Machine, exponentiality surfaces in the size of the machine, not the time complexity of the search. Further, the aspect of the Aunt Bertha machine that conflicts with our intuitions about intelligence is its reliance upon *memorization*. Removing the possibility of exponential storage amounts to a prohibition against memorization.[5] Consequently, an appropriate rephrasing of this conception is

> **The compact conception:** If an agent has the capacity to produce a sensible sequence of verbal responses to a sequence of verbal stimuli, whatever they may be, and without requiring storage exponential in the length of the sequence, then the agent is intelligent.

Again, Block notes that this additional condition is psychologistic in mentioning a nonbehavioral condition, viz., that the *manner* of the processing

must avert combinatorial explosion of storage. He claims that insofar as the condition is psychologistic, a Turing Test cannot test for it.

To summarize, Block's Aunt Bertha argument forces us to pay up on two promissory notes. For the purely behavioral Turing Test to demonstrate intelligence, it must suffice as a demonstration of the antecedent of the compact conception of intelligence, that is, it must indicate a *general capacity* to produce a sensible sequence of verbal responses and it must demonstrate *compactness* of storage of the agent. It requires us to demonstrate a basis for an alternative to Premise 1:

> **Premise 1′:** If an agent passes a Turing Test, then it has the capacity to produce a sensible sequence of verbal responses to a sequence of verbal stimuli, whatever they may be, and without requiring storage exponential in the length of the sequence.

To invalidate Block's argument, then, it is sufficient to provide a basis for the new, stronger, view of the Turing Test codified in Premise 1′.

4. The Deductive, Inductive, and Abductive Basis for the Turing Test

One potential way of salvaging the Turing Test is to change the notion of "demonstrate" in the claim that passing a Turing Test demonstrates intelligence. For instance, James Moor's view (1976) is that Turing Tests should be viewed not as deductive proofs of intelligence (as Block would have it), but as "a source of good inductive evidence."

He calls the evidence inductive evidence, but what kind of induction could a Turing Test be evidence for? Induction, in one guise at least, is the form of reasoning from instances of a universal to the universal. The instances we see in a Turing Test are the agent "producing a sensible sequence of verbal responses to a sequence of verbal stimuli" as Block would say. The natural inductive conclusion to draw from such data is that the agent has the "capacity to produce a sensible sequence of verbal responses to arbitrary sequences of verbal stimuli." Moor's inductive evidence is evidence for the antecedent in the capacity conception of intelligence. Already, we see that by relaxing our notion of demonstration, we can make some headway on the path from Premise 1 to 1′.

Nonetheless, the Test is only inductive evidence for the consequent if the capacity conception is sound. Thus, if Block is right, and the capacity conception fails, so does the inductive evidence reconstruction.

But what Moor is getting at goes beyond the inductive view of the Turing Test, and is made clearer by Stalker's reply (1978) and Moor's response (1978). Stalker refers to the evidence not as inductive evidence, but as explanatory evidence. More properly, though Stalker doesn't use the terminology, it appeals to reasoning by *abduction*, that is, reasoning to the best explanation.

We can caricature the types of reasoning as follows: *Deduction* is reasoning from P and $P \rightarrow Q$ to Q; *induction* is reasoning from (repeated instances of) P and Q to $P \rightarrow Q$; *abduction* is reasoning from P and $Q \rightarrow P$ to Q.[6] Of course, such abductive reasoning is deductively unsound, and is appropriately limited to special cases where $Q \rightarrow P$ holds because Q is a cause of P, and if there are multiple Q_i such that $Q_i \rightarrow P$, we select the Q_i that serves as the "best" explanation as cause of P. (What "best" means is a tricky issue, of course; it is where all the action is in formalizing abductive reasoning.)

In the case at hand, we take P to be the passing of the Turing Test and Q to be the bearing of intelligence. Abduction then allows us to reason from an agent passing the Turing Test, along with the view that intelligence (at least of a certain sort) implies the ability to pass the Turing Test, to the conclusion that the agent is intelligent.

Stalker points out that abductive reasoning requires an argument that the particular $Q \rightarrow P$ that one chooses must be the *best* explanation, not just any one, and he thinks he has a better one, namely, that the machine is merely following a particular computational procedure. Moor's reply amounts to arguing that the intelligence view is, as an abductive explanation, just as good, if not better.

Abductive reasoning in general has the following problem: The explanation that is best may still be wrong. Moor implies as much when he talks about the possibility that new evidence can cause one to change one's conclusions. So the move to viewing the Turing Test as abductive evidence of intelligence probably won't satisfy those (like Searle) who believe themselves in possession of a priori arguments against the possibility of mechanical intelligence. No matter how much "evidence" of this sort accumulates, the deductive conclusion from the premise "machines can't think" will trump the abductive evidence to the contrary.

It may also not satisfy Block, as it is hard to see how to rate the relative quality of the explanation "the machine is intelligent" and "the machine is looking up the replies in a table" without begging the question.

Nonetheless, the attempt to salvage the Turing Test as a test for intelligence by changing the kind of demonstration that we take it to be is a promising one. In the next section, I argue that the Turing Test can serve as a proof of the antecedent in the compact conception, and therefore a sufficient condition for intelligence, under a notion of proof that is very slightly weakened. By going from a requirement of deductive proof to that of interactive proof, and adding a weak condition of physical realizability, we can resurrect the Turing Test as a criterion of intelligence.

5. The Interactive Proof Alternative

To review, there are two psychologistic promissory notes out in the compact conception of the Turing Test. First, we must ascertain a general *capacity*

to pass the test. Second, the manner of passing must satisfy a *compactness* limitation. I will pay the capacity promissory note first, and then turn to the compactness issue.

Certainly, there is no deductive move that allows one to go from observation of the passing of one or more Turing Tests to a conclusion of a general capacity; the monkeys and typewriters argument shows that. This is the Humean problem of induction. But it does not follow that there is no method of reasoning from the former to the latter. I will argue that the powerful notion of an *interactive proof*, taken from theoretical computer science, is exactly such a reasoning method. Furthermore, as I will show in Section 6, Turing Tests bear some of the tell-tale signs of interactive proofs that have been investigated in the computer science literature.

Although interactive proof is a mathematical notion, the argument I will provide is not a formal one. I do not propose that the Turing Test is an interactive proof in the mathematical sense, but rather, that interactive proofs provide the right metaphor or analogy for thinking about what Turing Tests provide.[7]

Interactive proofs are protocols designed to convince a verifier conventionally denoted V that a prover P has certain knowledge or abilities, which we will think of as being encapsulated in an assertion s.[8] In a classical (deductive) proof system, P would merely reveal a deductive proof of s, which V then verifies. This provides V with knowledge of s and perhaps other knowledge implicit in the proof. Interactive proofs augment classical proof systems by adding notions of *randomization* and *interaction* between prover and verifier. (The interaction implicit in classical proof systems—P's presenting V with the proof—is essentially trivial.) Interaction is added by allowing V and P to engage in rounds of message-passing. Randomization is introduced in two ways: First, the verifier may make use of random bits in constructing her messages. Second, she may be required to be satisfied with a probabilistic notion of proof. When we state that P proves s with an interactive proof, we mean (implicitly) that s has been proved but with a certain determinable residual probability of error. That is, the verifier may need to be satisfied with some small and quantifiable chance that the protocol indicates that s is true when in fact it is not, or vice versa. The residual error is the reason that moving to a notion of interactive proofs is a weakening relative to a view as a deductive proof. The fact that the residual error can rapidly be made vanishingly small through repeated protocols is the reason that the weakening is referred to as "very slight".[9]

The idea of interactive proofs has been absolutely revolutionary in computer science since their introduction by Goldwasser, Micali, and Rackoff (1985). It has had two major payoffs. First, there are efficient interactive proofs of assertions for which classical proofs are hopelessly inefficient. Second, there are interactive proofs of theorems that reveal to V much less

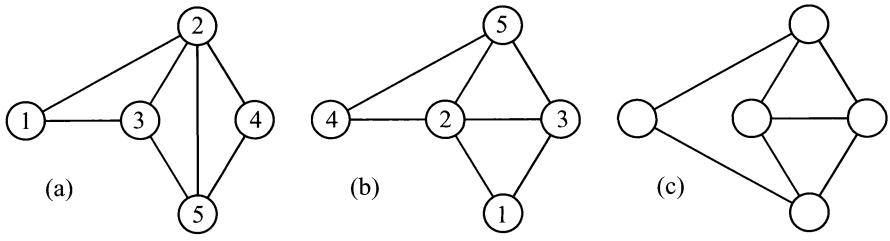

Figure 1. Example graphs

knowledge about *s*; in the case of so-called *zero-knowledge* proofs, they reveal nothing but the fact of the assertion's truth.

The idea is perhaps best grasped through an example, a variation on the GRAPH NONISOMORPHISM interactive proof system of Goldreich, Micali, and Wigderson (1991). A graph is a mathematical object consisting of a set of nodes and a set of edges connecting some of them. Two graphs are isomorphic if there is a one-to-one mapping from the nodes of one to the nodes of the other such that there is an edge between a pair of nodes in the one if and only if there is an edge between the pair of nodes in the other that they map to. Figure 1 presents a graphical depiction of some graphs. Although all have the same number of vertices and edges, only graphs (a) and (b) are isomorphic, under a mapping of the vertices given by the node numberings in the figure. Neither is isomorphic to (c). This is easily seen, as (c) has a minimal cycle (that is, a set of vertices connected in a cycle no subset of which forms a cycle) of four vertices, while neither (a) nor (b) do. In the general case, determining that two graphs are nonisomorphic is not so straightforward. It is important for the purposes of this example to understand that (given current assumptions in the foundations of computational complexity) the time required to determine if two graphs are isomorphic is exponential in the number of nodes in each graph, that is to say, the problem is very difficult.

Suppose *P* claims to know that the following assertion *s* is true:

> Graphs G_0 and G_1 are not isomorphic.

V wants to be convinced of this. We can imagine that the graphs G_0 and G_1 are quite large, say thousands of nodes. It would thus be impractical for *V* to determine by direct computation the truth of *s*.[10]

The following interactive proof protocol achieves this goal.

(1) *V* selects one of the two graphs G_0 or G_1 at random by choosing a random bit *b*, a 0 or 1; the selected graph is then G_b, the unselected graph G_{1-b}. *V* then computes a random permutation[11] G' of the chosen graph G_b. (If the assertion is true, this new graph is isomorphic to G_b but not G_{1-b}. If the

assertion is false, G' is isomorphic to both the original graphs.) V sends G' to P as a message.

(2) P checks if G' is isomorphic to G_0. If so, he sends the message "0", otherwise the message "1".

(3) V receives the bit b' that P sent. If $b' = b$, V accepts the proof; the assertion has been proved. If $b' \neq b$, V rejects the proof.

(We will call a series of messages generated according to a protocol such as this a *transcript*. An *accepting transcript* is one in which the verifier accepts the proof in the final step, and correspondingly for a *rejecting transcript*.)

The protocol is a bit like a mentalist's mind-reading trick. The verifier thinks of a number between 0 and 1, and the prover must guess that number. The prover gets a clue, namely the knowledge that if the verifier is thinking of the number 0, the graph she sent is isomorphic to G_0, and similarly for 1. If G_0 and G_1 are nonisomorphic (that is, the proposition is true), the clue is enough information to reconstruct the verifier's number. P will be able to reconstruct the bit b and the verifier will accept the proof. If the graphs are isomorphic (the proposition is false), the clue provides no help in guessing the verifier's number. In that case, the prover can do no better than guessing randomly, and will thus be wrong about half the time, causing V to reject the proof. The other half of the time, she will erroneously accept the proof; the prover "got lucky". It is in this sense that the interactive proof is probabilistic. If the verifier doesn't like these 50-50 odds of false positives, V can rerun the test several times. The more rounds, the less likely it is that the prover can guess right every time, unless the clues are actually helping (that is, unless the proposition to be proved is actually true). The probability of a false positive after k rounds of this protocol are 1 in 2^k, because the prover would have to get lucky k times in a row.

5.1. Interactive Proofs of Capacity

If a Turing Test is a kind of interactive proof, it needs to be a proof not of knowledge, but (as argued above) of a capacity. In the parlance of theoretical computer science, it is a proof of an *ability*. Bellare and Goldreich (1992) extend the notion of an interactive proof of knowledge to an interactive proof of an ability. Their method is quite sophisticated and general; in essence, they demonstrate that (with arbitrarily high probability) playing the role of P successfully in a proof system to compute some function is tantamount to computing the function itself. We don't require such a general setup. Rather, I present a simple mechanism for making statistical conclusions about a general capacity based on an interactive proof.

I start with an analogy. Suppose you are given a dartboard that is painted black, except for a single region of red. You throw some darts at the board uniformly at random, and note that 75% of them land in the red region. Intuitively, this should indicate to you that roughly 75% of the dartboard is

red, but of course this depends on how many darts you threw. If you threw only four darts, there is a reasonable chance that the red region is relatively small (say less than 50%), and yet three of the four happened to land in that region. By the time you have thrown one hundred darts with 75 landing in the red, the likelihood that the red region is less than 50% is, intuitively at least, much lower.

This line of reasoning can be made precise and quantified. In so doing, the intuition is proved correct, and in fact, the probability that the sample fails to represent the whole can be shown to be exponentially small in the number of samples. Crucially, this method allows us to reason from the behavior of a sample (the dart throws) to the space undergoing sampling (the entire dartboard). Identical reasoning can be applied to extrapolate from a sample of verbal behavior to a general capacity. Such reasoning is exactly what is needed to argue from the passing of a Turing Test to the attribution of a general capacity to do so.

I now describe how to use a rigorous form of the dartboard method to generate an interactive proof of a general capacity. Suppose that P claims a general capacity to compute a function f over possible inputs from a (presumably large and perhaps infinite) space S. How can V verify this claim by testing P in computing f on a sample of S? First, we must make more precise what we mean by a general capacity to compute a function. We may not (and, in the case at hand, do not) want to require perfect performance in computing the function; P may get the answer wrong on occasion for incidental reasons, even though in general P can compute the function. Let t_p be the fraction of the space for which P performs correctly. We can pick a threshold t_l to serve as a lower bound on the size of the subspace. We will say that if $t_p > t_l$, P has a general capacity to compute f. For the purpose of concreteness, we might take t_l to be 1/2. Then we are saying that if P computes f correctly on at least 50% of all possible inputs, it has a general capacity to compute f. Figure 2 depicts the general setup.

In this way, we can make precise a notion of having a general capacity. Nonetheless, this definition of general capacity still requires that we determine how P performs on all possible inputs, not just a subsample, in order to (deductively) verify that the subspace on which P performs correctly is larger than the threshold t_l. An interactive proof protocol can be used to prove this general capacity. V can sample k inputs $x_1, \ldots, x_k$ uniformly from S, and have P compute f on these inputs. V then verifies the correctness of each $f(x_i)$.

Suppose that P generates correct answers on some percentage t_s of samples greater than t_l; in the case at hand, we might take t_s to be 75% of the samples. Can we conclude that P has a general capacity to compute f (in the sense of computing f correctly on at least 50% of all inputs)? Such a conclusion does not logically follow. Perhaps P computes f correctly on less than 50% of all inputs, but V happened to select 75% of the samples from this smaller

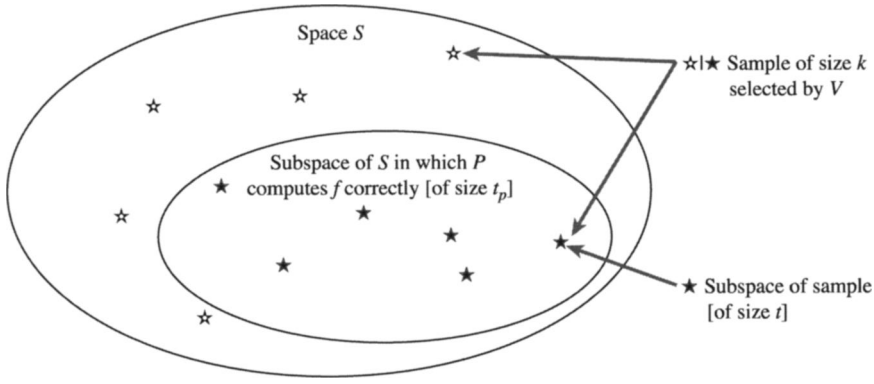

Figure 2. Sampling inputs to a function to test a general capacity for a prover to compute the function correctly on a given fraction of inputs.

subspace. This would constitute a "false positive", reasoning incorrectly from the sample to the space as a whole.

A false positive occurs when a sample of k inputs is selected for which f is computed correctly on t of these, where $t > t_s$ (the prover outperforms the sample threshold on the sample), yet $t_p < t_I$ (the subspace is smaller than the definitional threshold). Using the method of Chernoff bounds (see, e.g., Chapter 5 of the text by Motwani (1995)), it can be shown that the probability of a false positive is

$$Pr[t > t_s] < e^{-\mu\delta^2/2}$$

where $\mu = (1 - t_I)k$ and $\delta = 1 - \frac{1-t_s}{1-t_I}$. For the example in which $t_I = 1/2$ and $t_s = 3/4$, we have $\mu = k/2$ and $\delta = 1/2$, so

$$Pr[t > 3/4] < e^{-(k/2)(1/2)^2/2} = e^{-k/16}.$$

In general,

$$Pr[t > t_s] < e^{-ck}$$

where $c = \frac{(t_I - t_s)^2}{2(1-t_I)}$. Thus, it has the behavior of an interactive proof: As the number of samples k increases, the probability of a false positive decreases exponentially.

It is important to realize that the probabilities of error that we are talking about can be literally astronomically small. For the bounds that we have been talking about, if we let k be, say, 100, we are already in the realm of false positive probabilities on the order of 1 in 500. At $k = 300$, the false positive probability is on the order of 1 in 10^{10}; at that rate, if a population the size of

all humanity were tested, one would expect to see *no* false positives. At $k = 1000$, the false positive rate of some 1 in 10^{27} is truly astronomically small.

A similar argument shows that the probability of false negatives decreases exponentially in sample size as well. We suppose there to be another bound $t_u > t_s$ such that there is an agent that performs correctly on a fraction of the space given by t_u. A false negative occurs if $t < t_s$ for such an agent. For example, suppose t_u to be 90%, and suppose P computes f correctly on more than 90% of inputs, yet V happens to choose more than 25% of the samples on the less than 10% of the subspace that P fails to compute f on. Then for this sample, the apparent performance t will be less than the sample bound $t_s = 3/4$, a false negative. Again, the probability of such an occurrence can be shown to be exponentially small in k.

5.2. The Turing Test as an Interactive Proof of Capacity

I view the Turing Test as an interactive proof for the antecedent of the capacity conception of intelligence, that is, it is a proof that P "has the capacity to produce a sensible sequence of verbal responses to a sequence of verbal stimuli, whatever they may be." Consider the (perhaps infinite) space of all possible sequences of verbal stimuli. An agent without a general capacity to produce sensible sequences of responses would fail to do so on some nontrivial fraction of this space. Block notes that a 100% criterion is neither necessary nor appropriate. One wants to be able to "ask of a system that fails the test whether the failure *really does* indicate that the system lacks the disposition to pass the test." Indeed, people put under similar tests would at least occasionally perform in such a way that a judge might deem the responses not sensible. So there is some percentage, less than 100%, such that if an agent produced sensible sequences of responses on that percentage of the space, we can attribute a general capacity, sufficient for the antecedent of the capacity conception. Let us say, for the sake of argument that this threshold is 50%. That is, if an agent produces sensible responses to 50% of the space of possible verbal stimuli, we will consider it to have a general capacity to produce such responses. Importantly, we are not saying that the agent must merely produce sensible responses to 50% of some subsample of possible stimuli that we confront it with, but with 50% of all possible stimuli, in a counterfactual sense, whether we ever test it with these stimuli or not. The interactive proof approach of the previous section is directly applicable to this problem, with $t_l = 1/2$.

Suppose we sample k sequences of verbal stimuli uniformly from this space, and test some agent as to whether it generates sensible sequences of responses to them. Suppose further that the agent does so on 75% of these stimuli (that is, $t_s > 3/4$). On this sample, then, the agent performs well above our 50% cut-off. The analysis of the previous section shows that the probability that the agent does not have a general capacity at the 50% level is exponentially small.[12]

What about false negatives? If we assume that people generate sensible responses on, say, 99% of the space (recalling that 100% is not required here), then again the odds of a sample showing a performance of less than 75% is exponentially small in k.

In summary, a protocol in which we run k Turing Tests and receive sensible responses on at least, say, 75% provides exponentially strong evidence that the agent satisfies the antecedent of the capacity conception, that is, has a general capacity to produce sensible responses to verbal stimuli, whatever they may be.

Of course, one might think that a 50% capacity is insufficient to characterize a general capacity for verbal fluency. Perhaps 80% would be better. (We had better not get too greedy, though, as people don't deliver 100% performance.) Or one might think that some people, and even intelligent ones, don't approach 99% performance; maybe 85% is all we can guarantee. As long as there is a differential between the two bounds, we can place the threshold t_s between them and still achieve an exponentially small rate of both false positives and negatives. The difference between the two bounds merely determines a constant in the exponent. One can think of this as adjusting the number of samples needed before the knee in the exponential curve. (If one doesn't think there is a differential between the two bounds, one is denying the capacity conception itself.)

Thus, under the notion of proof provided by interactive proofs, the Turing Test can provide a proof of a general capacity to produce a sensible sequence of verbal responses to a sequence of verbal stimuli, whatever they may be. It can therefore unmask the monkeys on typewriters.

5.3. The Turing Test as an Interactive Proof of Compactness

Now to the question of compactness. First, I rephrase the compact conception of intelligence; rather than placing an upper limit on the size of the agent, I place an equivalent lower limit on the length of the test.

> **The modified compact conception:** If an agent has the capacity to produce a sensible sequence of verbal responses to a sequence of verbal stimuli that is at least logarithmic in the storage capacity of the agent, whatever they may be, then the agent is intelligent.

The modified compact conception is logically equivalent to the compact conception; the difference is just in the phraseology.

The interactive proof approach provides leverage for demonstrating this compactness as well. When all we know is the system's performance on a fixed sample of stimuli, the storage requirements to generate these responses is linear in the length of the stimuli. But the size of any fixed *fraction* of the space of possible stimuli is exponential in their length. By being able to reason from the sample to the fraction of the space as a whole—as the interactive

proof approach allows—we can conclude that an agent using a memorization strategy (as the Aunt Bertha machine) would require exponential storage capacity to achieve this performance. Conversely, any agent not possessing exponential storage capacity would fail the interactive proof.

Nonetheless, how can a Turing Test reveal that the machine *doesn't* have exponential storage capacity? Recall that the modified compact conception requires that the agent pass Turing tests at least logarithmic in its storage capacity. Thus, without bounding its storage capacity, we can't bound the length of the Test we would need. There is no purely logical argument against this possibility; the Aunt Bertha argument shows this. Some further assumption must be made to pay the compactness promissory note. I now turn to how weak an assumption is required.

Suppose we could bound the amount of information in the universe. Then any physically realizable agent that could pass Turing Tests whose length exceeded the logarithm of this amount would satisfy the compact conception. We would be able to bound the length of the Turing Test required under the compact conception, at least for any agent that is *no larger than the universe*. (And of course, no agent is larger than the universe.) We will call this length bound the *critical Turing Test length*. One might worry that the critical Turing Test length might be centuries or millennia long. In the paragraphs that follow, we show that the critical Turing Test length is actually quite short.

A crude upper bound on the information capacity of the universe can be constructed by examining the fine structure of space-time itself. (We will refine the estimate shortly.) Quantum theory predicts that the basic structure of space-time is strongly fluctuating on length scales of order 10^{-35} meters (the Planck scale). Any attempt to resolve phenomena below this scale, as would be necessary to store information, would require so much energy that the region being resolved would collapse into a black hole. It is thus reasonable to take a volume of this linear size as the smallest region in which one could store a bit. Let us assume one could actually store bits uniformly at this level of granularity. (We return to this assumption below.) This gives a volume at this primitive level of 10^{105} bits per cubic meter.

To obtain a value for the information capacity of the universe as viewed from a given location, we need a further estimate of the volume that could in principle affect that location. For any given distance in the past, the volume of the accessible universe is a sphere corresponding to a three-dimensional slice through the four-dimensional space-time light cone. For example, the volume of the universe of a minute ago accessible to a location now is a sphere centered on the location with radius of one light-minute (approximately 17 million kilometers). As we look farther backwards in time, the volume grows, but not without limit. The big bang serves as a second point of reference. At that point, the universe volume was effectively zero. The accessible volume, then, starts at zero with the big bang, grows to some maximal

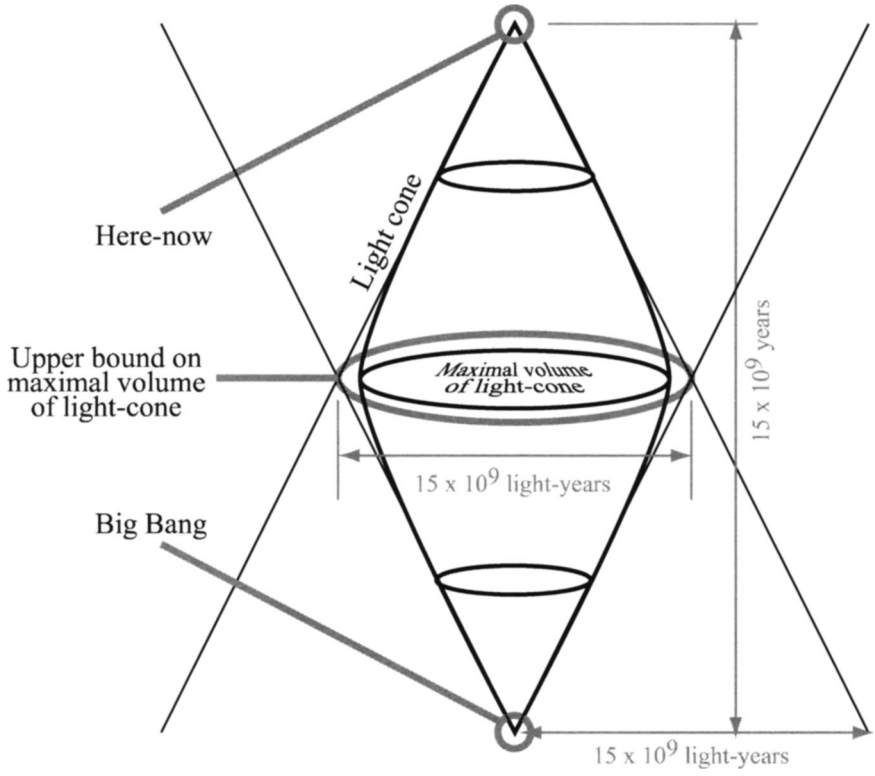

Figure 3. The volume of the universe accessible to a point in space time (here-now). The volume grows starting with the big bang, reaches some maximal volume, and then shrinks back to zero at here-now. The size of the maximal volume is bounded by a sphere whose diameter is given by the time since the big bang.

volume, and then shrinks again to zero as we approach the given location in space-time. A two-dimensional depiction of the situation is given in Figure 3. The point in question is how large this maximal volume is.

However large this maximal volume is, it can be no larger than a volume of diameter given by the time since the big bang some 15×10^9 years ago, which is the maximal volume if no contraction in the accessible volume occurs as we look back in time beyond that governed by the expansion of the universe since the big bang. Thus, the maximal accessible volume of the universe—which we can think of for our purposes, talking loosely, as *the* volume of the universe—must be less than $(15 \times 10^9)^3$ cubic light years. Recalling that a light year is about 10^{16} meters, the volume of the universe is thus bounded by 10^{79} cubic meters, and the total storage capacity is bounded by 10^{184} bits. Call it 10^{200} (thereby increasing the estimate by 16 orders of magnitude).

Descending now from these ethereal considerations to the concrete goal of analyzing the Turing Test conceptions of intelligence, under the modified compact conception, we would require an agent with this literally astronomical storage capacity to have a capacity to pass Turing Tests of on the order of $\log_2 10^{200} \approx 670$ bits. The entropy of English is about one bit per character or five bits per word (Shannon 1951), so we require a critical Turing Test length of around 670 characters or 140 words. At a natural speaking rate of some 200 words per minute, a conversation of less than a minute would therefore unmask a Turing-Test subject whose performance, like that of the Aunt Bertha machine, is based on memorization.

Current results in quantum gravity yield even smaller estimates of the information-storage capacity of the universe. Work on the so-called *holographic principle* (regarding which see the survey by Bousso (2002) for a review) limits the information stored in a volume based on its surface area rather than volume. Thus, the exponent in our estimate is off by a factor of 3/2; a more accurate estimate would be some 10^{120}. An important property of this result is that (unlike the estimate of the previous paragraph,[13] in which we assumed that we could store only one bit per Planck volume) it does not depend on any assumptions about the fine structure of physical theory. It is a pure principle of physics, like relativity; regardless of future discoveries of more and more finely differentiated particles, say, this limit on information content will hold.

As a side note, Block claims that

> Nothing in contemporary physics prohibits the possibility of matter in some part of the universe that is infinitely divisible. ... Suppose there is a part of the universe (possibly this one) in which matter is infinitely divisible. In that part of the universe there need be no upper bound on the amount of information storable in a given finite space. So my machine could perhaps exist, its tapes stored in a volume the size of, e.g., a human head.

Current physics shows that this claim is incorrect—the holographic bound on the information content of the universe holds regardless of the divisibility of matter—and limitations on the information-carrying capacity of the universe can allow us to draw conclusions from the fact of passing a Turing Test of sufficient (supercritical) length. Further, this length is a perfectly practical one.

An even smaller bound on the informational capacity of the universe has been developed by Seth Lloyd (2002), based on the total number of distinct quantum states in the universe. His estimate based on this methodology (which ignores the ability to store bits using gravitational degrees of freedom) is 10^{90}. (He notes as well that adding in the gravitational degrees of freedom gives a limit of 10^{120}, in agreement with the estimate derived above.) Thus, the critical Turing Test length might be even smaller than the one minute we estimated initially.

Nonetheless, the skeptic might wonder how sensitive our estimates of the critical Turing Test length are to these numbers. Suppose our estimates of the information content of the universe were off by, say, 1000 orders of magnitude, and there might be as many as 10^{1200} bits in the universe. Even then, the required Turing Test length would be around 4000 characters or 800 words, the size of a very short essay and far less than a five-minute conversation.

Thus, under extraordinarily conservative (but admittedly not logically necessary) assumptions, even quite short Turing Tests are sufficient to pay the compactness promissory note. It might seem counterintuitive (especially to those familiar with toy programs designed to engage in conversation on this or that topic) that the critical Turing Test length should be so short. Keep in mind that the Test here is the unrestricted Turing Test—any and all queries on any topic of any sort are allowed—and that the machine that we are trying to unmask is of a particular sort, namely one that has memorized answers to every possible such query. As it turns out, the combinatorics of language are such that only a short time is required to generate a vast number of possible queries, and the design of the Aunt Bertha Machine is such that we need only find one that is unhandled to unmask it.

It is important to understand that the physical calculations performed here, by themselves, are insufficient to provide grounding to the compact conception. The argument requires the interactive proof notion as well, for it is the interactive proof view of the Turing Test that lets us go from a conclusion about a sample of verbal behavior to a conclusion about possible untested behaviors, and the storage capacity required by a sample is merely linear in the sample size, but the storage capacity required by the possible untested behaviors is exponential in the sample size.

What I have argued is not that one can *deduce* from an agent passing a Turing Test the agent's intelligence, but rather, that one can prove this under a conception of proof that admits false positives with astronomically small probability, and that makes physical assumptions of an astronomically weak nature. Further, the proof is of the strong antecedent to the compact conception of intelligence, including the capacity requirement and the compactness limitations on the agent. In essence, I have argued for the following recasting of the basic syllogism supporting the sufficiency of the Turing Test:

> **Premise 1:** If an agent passes k rounds of a Turing Test of at least one minute in length, then (with probability of error exponentially small in k) it has the capacity to produce a sensible sequence of verbal responses to a sequence of verbal stimuli that is logarithmic in the storage capacity of the agent, whatever they may be.
>
> **Premise 2:** (= Modified Compact Conception) If an agent has the capacity to produce a sensible sequence of verbal responses to a sequence of verbal stimuli that is logarithmic in the storage capacity of the agent, whatever they may be, then it is intelligent.

Conclusion: Therefore, if an agent passes k rounds of a Turing Test of at least one minute in length, then (with probability of error exponentially small in k) it is intelligent.

6. Interactive Proof Properties Shared by the Turing Test

Several previously unexplored properties of Turing Tests follow from properties of interactive proofs that have been demonstrated in the computer science literature. Such properties inhere to interactive proofs as opposed to classical proofs. Their applicability to Turing Tests thus provides a further argument for the view of Turing Tests as interactive proofs. I discuss two such properties here: nontransferability and lack of closure under composition.

6.1. Nontransferability

Interactive proofs typically are *nontransferable*; they provide proof only to the verifier and no one else. In a classical proof system, an accepting transcript includes a record of the prover submitting to the verifier an independently checkable proof of the proposition. Thus, the transcript can serve as a proof to a third party as well. But a third party who is given an accepting transcript of an interactive proof protocol does not thereby gain proof of the truth of the proposition being proved. This follows from the fact that accepting transcripts can be generated *without knowledge of the truth of the proposition being proved*. For the graph nonisomorphism case, the only message from the prover to the verifier in an accepting transcript is a bit the identity of which the verifier already knows. Such a transcript is therefore trivial to generate.

Another reason that third parties do not gain evidence from an accepting transcript, even if they watch it being generated, is that they do not have access to crucial facts concerning whether the protocol is being accurately followed. The evidence provided by an interactive proof depends, for instance, on certain bits being generated randomly by the verifier and kept private from the prover. Only the verifier knows that the bits were random and secret, as only the verifier generated them and held them. If the bits were generated nonrandomly (for instance, according to a prior collusive agreement with a "prover"[14]) or were not kept secret (communicated to a "prover" after being generated randomly), the recipient could use the knowledge of the bits to complete an accepting protocol instance without knowledge of the proposition being proved.

Turing Tests are also nontransferable in this sense. Accepting transcripts (that is, sensible conversations) can be generated by the verifier (judge) independently and without knowledge that the prover (subject-under-test) possesses the ability in question. Similarly, if a "verifier" fails to obey the randomness or secrecy requirements of the protocol, an accepting transcript can be generated even though the "prover" lacks the general capacity to respond sensibly to verbal stimuli. For example, if V restricts her questioning

to a particular line of conversation that she knows P has been programmed to handle well, the transcript will appear to an outside observer to be an accepting one, even though it provides no information about the general capacities of P to respond sensibly. This is the phenomenon of the "cooked demo", which can appear very convincing to an observer while of course being completely unconvincing to the participants. The observer lacks the crucial knowledge possessed by the participants that the protocol was apparently, but not actually, being followed.[15] The only way for you to know that a demo hasn't been cooked is to act as the verifier yourself.[16]

Of course, nontransferability is an intended property of typical interactive proof protocols—the typical cryptographic applications of interactive proofs make nontransferability a desirable property—whereas nontransferability is an inadvertent property of Turing Tests. Nonetheless, the similarity is real.

6.2. Lack of Closure under Composition

It can easily be shown that interactive proofs are not closed under composition. In particular, if agents participate in multiple interactive proofs at the same time, the conclusions that can be drawn from the set of proofs can be much weaker than those that could have been drawn by similar proofs generated independently (that is, without shared participation). Block alludes to this issue with his example of the simultaneous chess player.

> Jones plays brilliant chess against two of the world's foremost grandmasters at once. You think him a genius until you find out that his method is as follows. He goes second against grandmaster G_1 and first against G_2. He notes G_1's first move against him, and then makes the same move against G_2. He awaits G_2's response, and makes the same move against G_1, and so on. Since Jones's method itself was one he read about in a comic book, Jones's performance is no evidence of his intelligence. (Block 1981)

Failures of interactive proofs of this sort have been noted in the computer science literature as well. Desmedt, Goutier, and Bengio (1987), for instance, describe what they term the "mafia fraud", which is a failure of the Fiat-Shamir interactive proof method for authentication (Fiat and Shamir 1986). In an authentication protocol, P proves his identity to V; the Fiat-Shamir protocol does so by proving (via interaction, and in the normal probabilistic sense) to V that P has knowledge of a certain private key known only to P, without revealing that key to V (or anyone else).

In the "mafia fraud", P', who has no knowledge of P's private key succeeds in authenticating himself to V by carrying out a separate authentication protocol with P. Whatever messages V sends to P', P' sends on to P; whatever responses P' receives, he sends on to V. The two instances of the protocol being carried out are accepting ones, hence P is authenticated to P' and P' is authenticated to V. The composition of the two protocol instances thus

fails to ensure the correctness of the conclusions (at least in the case of the authentication of P').

This technique of composing interactive proofs and playing one participant off against another can trip up Turing Tests as well. Here is a six-line program, clearly unintelligent, that can pass two simultaneous Turing Tests. (Here, *query*(*judge*$_i$) returns the next query sent by the i-th judge, and *respond*(*judge*$_i$, r) sends a given response r to the i-th judge.)

repeat

$i_1 :=$ *query*(*judge*$_1$);
respond(*judge*$_2$, i_1);
$i_2 :=$ *query*(*judge*$_2$);
respond(*judge*$_1$, i_2);

until finished

It merely shuttles the responses of each of the judges to the other, just as the chess player shuttles the moves of the two grandmasters to each other and as the mafia defrauder shuttles protocol messages from P to V. Engaging in two Turing Tests at once does not necessarily provide twice the evidence generated by a single Turing Test, and may provide no evidence at all, just as simultaneous Fiat-Shamir proofs fail to provide the authentication guarantee that single Fiat-Shamir proofs do.

7. Conclusion

I have argued that the Turing Test is appropriately viewed not as a deductive or inductive proof but as an interactive proof of the intelligence of a subject-under-test. This view is evidenced both by the similarity in form between Turing Tests and interactive proof protocols and by the sharing of important properties between Turing Tests and interactive proofs.

In so doing, I provide a counterargument against Block's demonstration that the Turing Test is not a sufficient criterion of intelligence. Our counterargument requires a (very slight) weakening of the conditions required of the Turing Test—weakening the notion of proof (from classical deductive proof to interactive proof with its exponentially small residual error probability) and strengthening the notion of possible agent (from one of logical possibility to one with a trivial realizability requirement essentially of nomological possibility). These weakenings are sufficiently mild that they can be seen as providing foundation for the view that the Turing Test is a sound sufficient condition for intelligence. Block is right, yet Dennett may be too.

It merits pointing out that this view of the Turing Test is consonant with (though by no means implicit in) Turing's view of the Test as presented in his writings. His view of the Test as being statistical in nature and his pragmatic

orientation toward its efficacy are of a piece with its status as an interactive rather than classical proof.

Acknowledgments

I would like to thank Ned Block, Raphael Bousso, Daniel Fisher, David Israel, Michael Rabin, Ken Shan, Andrew Strominger, Salil Vadhan, and the members of the Artificial Intelligence Research Group at Harvard University for valuable discussions and insights regarding the issues discussed in this paper. I also thank the anonymous reviewers for their comments that led to several clarifications.

Much of the work on this paper was done while visiting the Centro per la Ricerca Scientifica e Tecnologica (itc-IRST), Trento, Italy during the spring of 2002. My deep appreciation goes to Oliviero Stock and itc-IRST for space and support to work on this material during my visit.

Notes

[1] I take the term "sensible sequence of verbal responses" directly from Block to mean whatever criterion of human indistinguishability that the judge in a Turing Test is verifying. It may be that the term is not entirely felicitous for that purpose. For instance, there may be sequences of responses that are sensible in the informal sense of the term, yet reveal the non-human character of the generator by being stilted in some way. Under certain circumstances, even clearly nonsensical responses are appropriate in a Turing Test, as in Block's example of a judge requesting "Let's see you talk nonsense." (Block 1981 pages 19–20) Nonetheless, for consistency hereafter I will follow Block in using the phrase, with the request that the reader interpret it in the intended manner.

[2] At the end of "Psychologism and Behaviorism", Block presents claims that an agent that exhibits intelligent behavior on the basis of exact emulation of the neurological processes of a person would arguably still not be intelligent.

> Consider a device that simulates you by using a theory of your psychological processes. It is a robot that looks and acts as you would in any stimulus situation. Instead of a brain it has a computer equipped with a description of your psychological mechanisms. You receive a certain input, cogitate about it, and emit a certain output. If your robot doppelganger receives that input, a transducer converts the input into a description of the input. The computer uses its description of your cognitive mechanisms to deduce the product of your cogitations; it then transmits a description of your output to a mechanism that causes the robot body to execute the output. It is hardly obvious that the robot's process of manipulation of descriptions of your cogitation is *itself* cogitation. It is still less obvious that the robot's manipulation of descriptions of your experiential and emotional processes are themselves experiential and emotional processes.

It is hard to know how this claim could be distinguished in spirit from Searle's, and Block (personal communication 2002) has since stated that, though the various hedges make it possibly literally true, it goes too far.

[3] The conceptions highlighted here correspond roughly to Block's "operationalist proposal", "neo-Turing Test conception", and "amended neo-Turing Test conception", respectively, except that crucially they are phrased as conditionals to better accord with the view of the Test as an ostensible sufficient condition, not an ostensible definition. In particular, Block states his conceptions in the form of definitions, e.g., "Intelligence (or more accurately, conversational intelligence) is the capacity to produce a sensible sequence of verbal responses to a sequence of verbal stimuli, whatever they may be." Compare this to my capacity conception below.

[4] This anti-behaviorist argument was apparently first proposed in sketch form by Shannon and McCarthy (1956, page vi): "A disadvantage of the Turing definition of thinking is that it is possible, in principle, to design a machine with a complete set of arbitrarily chosen responses to all possible input stimuli.... With a suitable dictionary such a machine would surely satisfy Turing's definition but does not reflect our usual intuitive concept of thinking."

[5] For this reason, adding this extra condition to the conception of intelligence is not ad hoc. It amounts to saying, in a precise way, that the agent must have the capacity to produce sensible responses without having memorized them.

[6] Hobbs, Stickel, Martin, and Edwards (1988) present a similar symmetrical view of the three types of reasoning.

[7] In independent work, Bradford and Wollowski (1994) do attempt to provide a mathematical argument relating interactive proofs and the Turing Test, but of a quite different flavor. For instance, they assume that the subject-under-test is polynomially bounded, and take the subject-under-test and confederate to be the verifiers, and the judge to be the prover. It is safe to say that the upshot of their analysis is unclear given the strong assumptions that they make.

[8] The roles of verifier and prover are analogous to those of Victoria and Peter/Pietro above. For convenience in reference, we will therefore refer to them using gendered pronouns "her" and "him" respectively.

[9] The probabilistic nature of interactive proof conclusions constitutes a very important distinction between interactive proofs and general inductive evidence (as appealed to by Moor). Inductive arguments may, like interactive proofs, be thought of as statistically founded, but they end in a step of *acceptance* of the general conclusion of which the instances form the inductive evidence. (We select many marbles from a jar and note that all are red. Statistics and independence assumptions allow us to compute a probability distribution over proportions of red marbles in the jar, with 100 percent being the most likely. By inductive acceptance, we conclude that all of the marbles in the jar are red.) Pollock (1990, Chapter 5) reviews problems with the statistical foundations of induction, and in particular, the acceptance step. But by virtue of yielding probabilistic conclusions, interactive proofs have no acceptance step, and thus do not fall prey to these problems.

[10] I digress to discuss a technical issue in order to forestall confusion about the graph nonisomorphism protocol. In order for the interactive proof of graph nonisomorphism to be of interest, we must assume that the verifier is computationally limited. Otherwise, the verifier could check whether G_0 is isomorphic to G_1 herself. It is standard, therefore, to restrict V to amounts of computation polynomial in the size of the graph. Under this assumption (and, again, current assumptions in the foundations of computational complexity), the verifier cannot herself determine if the two graphs are nonisomorphic. There is, of course, no reason to assume such computational limitations on the prover, and it is standard not to do so. For this reason, the interactive proof protocol can involve the prover carrying out actions for which no such computationally limited method is known, such as the computation of graph isomorphism in Step 2. The issue is discussed in detail by Goldreich et al. (1991). Interactive proof protocols for other problems, such as GRAPH 3-COLORABILITY, are known for which computationally limited provers are sufficient.

[11] Informally speaking, a random permutation of a graph is just an isomorphic copy of the graph whose relation to the original has been lost.

Formally, a random permutation G' of the graph G_b is a graph isomorphic to G_b constructed as follows: The nodes of G' are a set of the same cardinality as the set of nodes of G_b. A one-to-one mapping π from the nodes of G_b to the nodes of G' is chosen at random from all possible such mappings. For each edge in G_b connecting nodes n_1 and n_2, there is an edge of G' connecting nodes $\pi(n_1)$ and $\pi(n_2)$, and there are no other edges in G'.

[12] The bounds presented here showing exponentially vanishing probabilities of error in adjudging capacities are predicated on the k samples being taken uniformly and independently. In the case of repeated Turing Tests, of course, the judge is free, and apt, to construct new

Tests based on the behavior noted in previous Tests so as to maximize the information received. Such nonindependent sampling can lead to dramatically smaller rates of error, in theory, for randomized tests of this sort.

[13] The estimate based on the holographic principle is far lower than our previous estimate because it respects the fact that any attempt to store bits as densely by volume as the previous estimate would have energy requirements that would cause the system to collapse under gravitational forces.

[14] The quotes are used to indicate that the entity playing the prover role is not acting as a true prover.

[15] For this reason, many of us in the natural-language-processing field have come to be healthily skeptical of published transcripts of the behavior of natural-language-processing systems.

[16] An anonymous reviewer urges correctly that we not make too much of the nontransferability of Turing Tests, noting that the same could be said about scientific proofs as documented in research papers. "A scientist can publish a result that is incorrect or even cooked. How can published results become knowledge for others who haven't run the experiment?" The answer, of course, is trust in the scholarly publishing system, a trust founded in large part on statistical evidence; incorrect results fail to replicate and cooked results occasionally come to light, and we can note empirically the rarity of both types of failures. In fact, the nontransferability of scientific proof reminds us of the degree to which the scientific enterprise is founded on an interactive notion of proof as well.

References

Bellare, Mihir and Oded Goldreich. 1992. Proving computational ability. Available at http://www.wisdom.weizmann.ac.il/~oded/PS/poa.ps.

Block, Ned. 1980. What intuitions about homunculi don't show. *Behavioral and Brain Sciences* 3: 425–426.

———. 1981. Psychologism and behaviorism. *Philosophical Review* XC(1): 5–43.

Bousso, Raphael. 2002. The holographic principle. *Reviews of Modern Physics* 74: 825–874, Available as hep-th/0203101.

Bradford, Phillip G., and Michael Wollowski. 1994. A formalization of the Turing test. Tech. Rep. 399, Department of Computer Science, Indiana University. Available at http://www.cs.indiana.edu/pub/techreports/TR399.html. An extended abstract appeared in the *Proceedings of the 5th Midwest Artificial Intelligence and Cognitive Science Conference*, ed. T. E. Ahlswede, pages 83–87, April 1993.

Chomsky, Noam. 2004. Turing on the "imitation game". In *The Turing Test*, ed. Stuart M. Shieber, chap. 20, Cambridge, MA: MIT Press.

Davidson, Donald. 1990. Turing's test. In *Modelling the Mind*, ed. K. A. Mohyeldin Said, W. H. Newton-Smith, R. Viale, and K. V. Wilkes, chap. 1, 1–11, Oxford, England: Clarendon Press.

Dennett, Daniel. 1985. Can machines think? In *How We Know*, ed. M. Shafto, 121–145, San Francisco, CA: Harper and Row.

———. 1987. Fast thinking. In *The Intentional Stance*, chap. 9, 323–337, Cambridge, MA: MIT Press.

Desmedt, Yvo, Claude Goutier, and Samy Bengio. 1987. Special uses and abuses of the Fiat-Shamir passport protocol. In *Advances in Cryptology—CRYPTO '87*, ed. Carl Pomerance, vol. 293 of *Lecture Notes in Computer Science*, 21–39, Berlin, Germany: Springer-Verlag.

Fiat, Amos, and Adi Shamir. 1986. How to prove yourself: Practical solutions to identification and signature problems. In *Advances in Cryptology—CRYPTO '86*, ed. A. M. Odlyzko,

vol. 263 of *Lecture Notes in Computer Science*, 171–185, Berlin, Germany: Springer-Verlag.

French, Robert. 1990. Subcognition and the limits of the Turing test. *Mind* 99(393): 53–65.

Goldreich, Oded, Silvio Micali, and Avi Wigderson. 1991. Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proofs. *Journal of the Association for Computing Machinery* 38(3): 691–729.

Goldwasser, Shafi, Silvio Micali, and Charles Rackoff. 1985. The knowledge complexity of interactive proof-systems (extended abstract). In *Proceedings of the 17th ACM Symposium on the Theory of Computing*, 291–304, Providence, RI.

Gunderson, Keith. 1964. The imitation game. *Mind* 73(290): 234–245.

Hobbs, Jerry R., Mark Stickel, Paul Martin, and Douglas D. Edwards. 1988. Interpretation as abduction. In *Proceedings of the 26th Annual Meeting of the Association for Computational Linguistics*, 95–103, Buffalo, New York.

Lloyd, Seth. 2002. Computational capacity of the universe. *Physical Review Letters* 88(23).

Moor, James H. 1976. An analysis of the Turing test. *Philosophical Studies* 30: 249–257.

———. 1978. Explaining computer behavior. *Philosophical Studies* 34: 325–327.

Motwani, Rajeev. 1995. *Randomized Algorithms*. Cambridge, England: Cambridge University Press.

Newman, M. H. A., Alan M. Turing, Sir Geoffrey Jefferson, and R. B. Braithwaite. 1952. Can automatic calculating machines be said to think? Radio interview, recorded 10 January 1952 and broadcast 14 and 23 January 1952. Turing Archives reference number B.6.

Pollock, John L. 1990. *Nomic Probability and the Foundations of Induction*, Oxford, England: Oxford University Press.

Searle, John R. 1980. Minds, brains, and programs. *Behavioral and Brain Sciences* 3: 417–457.

Shannon, C. E. 1951. Prediction and entropy of printed English. *Bell Systems Technical Journal* 30(1): 50–64.

Shannon, Claude E., and John McCarthy, eds. 1956. *Automata Studies*. Princeton, NJ: Princeton University Press.

Stalker, Douglas F. 1978. Why machines can't think: A reply to James Moor. *Philosophical Studies* 34: 317–320.

Turing, Alan M. 1950. Computing machinery and intelligence. *Mind* LIX(236): 433–460.