[STAR] Penrose is Wrong

Drew McDermott
Department of Computer Science
Yale University
New Haven, CT
U.S.A.

mcdermott@cs.yale.edu

PSYCHE, 2(17), October, 1995
http://psyche.cs.monash.edu.au/v2/psyche-2-17-mcdermott.html

KEYWORDS: Consistency statements, Gödel incompleteness theorems, Turing machines, ordinal logics, Platonism, proof theory.

REVIEW OF: Roger Penrose (1994) *Shadows of the Mind*. New York: Oxford University Press. 457 pp. Price: $25 hbk. ISBN 0-19-853978-9.

[Note to the on-line reader: Penrose's hollow five-pointed star, the symbol of unassailability, I indicate with the string "STAR".]

1. Penrose vs AI - Again

1.1 Roger Penrose's new book, *Shadows of the Mind*, is strongly reminiscent of his previous work in the same vein, *The Emperor's New Mind*. This book restates the author's central line of argument about the place of consciousness in the material world. He has no sympathy at all for attempts to work out a computationalist theory of mind, and instead pins his hopes on a future theory that would allow large-scale quantum-mechanical effects in the brain to play a central role.

1.2 A broad outline of his argument goes like this:

- Because of Gödel's Incompleteness Theorem, mathematical insight cannot be mechanized.
- Mathematical insight depends on consciousness, and so it is doubtful that any part of consciousness can be mechanized.
- But then a physical system can be conscious only if it can't be simulated by a computer.
- That would be very strange; fortunately, the world as imagined in modern physics *is* very strange.

- The interaction between quantum mechanics and the general theory of relativity is poorly understood. Fundamental questions about time and causality seem to depend on how that interaction gets worked out.
- Perhaps the brain exploits some large-scale quantum coherence to achieve consciousness. Perhaps the site of this effect is in the cytoskeletons of neurons.

1.3 This argument, when put down in black and white, seems extraordinarily weak. The least speculative step is the first, but that's also the easiest to show is fallacious, as I will do shortly. But before I do, I want to raise the question, Why is Penrose bothering?

1.4 A clue might be this sentence on p. 373: "It is only the arrogance of our present age that leads so many to believe that we now know all the basic principles that can underlie all the subtleties of biological action." Penrose wants to do battle against the arrogance he perceives, especially in the AI community, regarding the problem of consciousness. It is true that AI has, from its inception, had the ambition to explain *everything* about the mind, including consciousness. But is this arrogance? Or merely the sincere adoption of a working hypothesis? If someone wants to work on the problem of mind, it seems to me that he must choose among three options: treat the brain as a computer, and study which parts compute what; study neurons, on the assumption that they might be doing something noncomputational; or work in a seemingly unrelated field, like physics, on the off chance that something relevant will turn up. In any case, no matter which tack is taken, one gets mighty few occasions to feel arrogant about one's success. Neuroscience and AI have made definite progress, and so has physics, for that matter, but their successes haven't resulted in a general theory of mind. If anything, AI seemed closer to such a theory thirty years ago than it seems now.

1.5 So if someone wants to believe that AI will never explain the mind, he might as well. The burden of proof is on whoever claims it ultimately will. Penrose isn't satisfied with this state of affairs, however, and wants to exhibit a proof that a computationalist theory of mind is impossible. I suppose he sees himself fighting for the hearts and minds of neutral parties, who are in danger of being fooled into thinking that AI is on the verge of such a theory by the breathless stories they read in the papers. I don't know; perhaps an argument like Penrose's will, once it has been filtered through the distorting lens of the TV camera, be a sort of homeopathic antidote to those breathless stories. But, I regret to say, the argument would still be wrong. And so those of us in a position to point out the flaws in it must sheepishly rise to do so, in the full knowledge that AI can't win the debate if it degenerates into Mutual Assured Destruction ("You can't prove AI is possible," "Oh yeah? Well, you can't prove it's not").

2. Gödel's Theorem or Bust

2.1 Penrose stakes everything on his analysis of Gödel's Theorem. This analysis is all wrong, but what's striking is how much he tries to hang on it. Penrose assumes that there is a single attribute called "consciousness" that accounts for insight, awareness, and free

will. Hence, if he can show that computers lack a certain sort of insight, they must also lack all awareness and free will. (One wonders where this leaves five-year-old children.)

2.2 In addition, all the plausibility of Penrose's theory of "quantum consciousness" in Part II of the book depends on the Gödel argument being sound. It certainly provides no plausibility by itself. There is a lot of material in the book about the mysteries of quantum mechanics. There is a much smaller amount about where in the brain quantum-mechanical effects might be important. But if you seek an account of the link between these hypothetical effects and insight, awareness, and free will, there isn't any. This nontheory gets all of its oomph from the pathetic weakness of the computational alternative, as described in Part I of the book. The slightest flaw in Part I would knock most of the stuffing out of Part II.

2.3 Part I is, in fact, full of flaws. The basic argument is straightforward and convincing: Suppose that *all* a mathematician's reasoning techniques could be embodied in an algorithm A that was believed to be sound. For technical reasons, assume that when A is given a problem of the form "Will algorithm C_q(n) stop?," where C_q is the algorithm with code q, and n is input data, A signals that the algorithm will not stop by stopping. Soundness means that when A(q,n) stops we are guaranteed that C_q(n) will not. There is some k such that C_k(n) is the computation A itself deciding whether C_n(n) will halt. If C_k(k) stops, then C_k(k) does not stop (because A is sound). Therefore C_k(k) does not stop, and we *believe* it doesn't stop, because we *believe* that A is sound. But A fails to draw this conclusion (i.e., it fails to signal the conclusion by stopping), so it is unable to conclude something that the mathematician (i.e., we) can conclude. Therefore A does not in fact coincide with the algorithm used by the mathematician. But the only feature of A that we assumed was soundness. Therefore (to quote Penrose, p. 76) "Human mathematicians are not using a knowably sound algorithm in order to ascertain mathematical truth."

2.4 As I said, the argument is convincing. But it's also a bit anticlimactic, since no one in his right mind would suppose that that human mathematicians "use" (or embody) a sound algorithm, let alone a "knowably" sound one. To verify this point, you need merely to find a case where a mathematician made a mistake. Penrose acknowledges this problem, and devotes most of a chapter, Chapter 3, to trying fix it, by showing that in spite of appearances human mathematicians really are sound theorem provers. His attempts fail.

3. Patching the Proof

3.1 His first move (p. 98ff.) is to try to argue that it is reasonable to abstract away from an individual mathematician and talk about "mathematicians" as a team, all of whose members basically agree on mathematical truths. "Are mathematicians' judgements actually so subjective that they might disagree *in principle* as to whether a particular ... sentence has, or has not, been established as true?" The problem is the phrase "in

principle." Of course, in principle all mathematicians agree, and all nations want world peace. But just the other day I received a flyer for a journal called the "Mathematical Intelligencer" that airs mathematical controversies. The flyer contained some choice quotes from a recent article by mathematician A about why B's published proof of some supposed theorem was inadequate, and promised a reply from B in a future issue. The quotes made it clear that A was not objecting to B's logical system, but to some claims B had made that certain lemmas were obvious. What would Penrose say about such a case? I suppose he would say that as long as a public controversy was going on, all claims involved should be placed in escrow. But what about cases where B gets away with his claims, and no A has the time or inclination to notice gaps? Isn't this the usual situation? Isn't B usually right (but not always)? In view of all this, it seems as though if there is a body of mathematical claims endorsed by the mathematical community, the current set of claims is always inconsistent.

3.2 To take one famous example, cited by Ball and Coxeter (1939), in 1879 A.B. Kempe published a proof of the four-color theorem (Kempe 1879). According to Ball<1> and Coxeter, the bug in Kempe's proof was not disclosed until the publication of Heawood (1890). Hence for that eleven-year period the mathematical community was in a state of contradiction, and there is no reason to suppose any other period is immune.

3.3 Human mathematicians do not generate an answer to a problem and then stop thinking about it. In fact, human mathematicians never stop, except for reasons irrelevant to the kind of in-principle argument we're doing here. Consider, for example, this passage from Kaplan and Montague (1960), concerning the Hangman Paradox:

Before the appearance of Shaw's (1958) article, we had considered a form of the paradox essentially identical with his, and found it, contrary to his assertion, not to be paradoxical. At the same time we were successful in obtaining several versions which are indeed paradoxical. The present note is intended to report these observations.

3.4 In other words, they started thinking about the problem, derived an analysis of it, found flaws in someone else's analysis, then kept analyzing. And the result of all their cogitation (to date, that is) is a paradox, an elusive *inconsistency* in some seemingly straightforward postulates! It is difficult to see how thinkers like these could even be remotely approximated by an inference system that chugs to a certifiably sound conclusion, prints it out, then turns itself off.

3.5 For other examples, see (Lakatos 1976).

3.6 Penrose tries to solve this problem by distinguishing between "individual mistakes---or 'slips'" and "correctable" errors on the one hand, and "unassailable" conclusions on the other. He supposes that the unassailable conclusions are what we really care about, and that these must be the output of a sound reasoning system of some kind. "... We are not concerned with inidividual mistakes---or 'slips'---that a mathematician might happen to make while reasoning within a consistent overall scheme." (p. 138) Then we take as the output of our formal system the conclusions the computer takes to be unassailably true. The resulting system then must then suffer from the incompleteness described above.

If our robot is to behave like a genuine mathematician, although it will still make mistakes from time to time, these mistakes will be correctable --- and correctable, in principle, according to its own internal criteria of 'unassailable truth.' ... If we are supposing that our robot is to be capable of attaining (or surpassing) the level of mathematical capability that a human being is in principle capable of achieving, then its concept of unassailable mathematical truth must also be something that cannot be attained by any set of mechanical rules that can in principle be perceived as sound .... We are to assume that the robot ... possesses a ... secure level of unassailable mathematical 'belief,' so that some of its assertions---attested by some special imprimatur, which I denote here by a symbol 'STAR,' say---are to be unassailable, according to the robot's own criteria." (pp. 157--159)

3.7 This move raises two obvious questions: What is meant by "unassailability"? and, What in human practice corresponds to tagging assertions with STAR? The answer to the second question might be "publication in a reputable journal." But of course errors do occur in even the most carefully refereed journals, as in the example described above. Perhaps unassailability is flagged by placing a statement in a journal and then not having it be corrected over some period of time, say a hundred years. But of course there are errors in journals that have gone undetected for more than a hundred years.

3.8 Penrose is quite vague about what unassailability is. He comes close to endorsing the view that unassailability means provability. "Is it really plausible that our unassailable mathematical beliefs might rest on an unsound system...?" he asks rhetorically on p. 138, thus seeming to imply that unassailable beliefs rest on some kind of "system," or "scheme," as in the quote I cited above from the same page. But of course it can't be a *formal* system or scheme. That is, there can't be a formal mathematical theory such that everything unassailable is provable in that theory. That's what Gödel proved. So unassailability comes down to some kind of subjective feeling. On pages 138-141 he talks about the case of Frege's famous reaction to Russell's discovery of a paradox in his life's work, and seems to imply that mathematicians were sloppy then in ways we've learned to overcome. "Perhaps mathematicians have now become more cautious as to what they are prepared to regard as 'unassailably true'---after a period of excessive boldness (of which Frege's work indeed formed an important part) at the end of the nineteenth century." The way I read this is, if Frege had simply resolved not to be so reckless, he would never have been tempted to publish something that, deep down, he knew all along was not unassailable. I realize that I'm caricaturing Penrose's view, but it's awfully hard to get a precise reading about what he means. He wants unassailability to be both informal and guaranteed accurate, and I don't see how that's possible.

3.9 In anticipation of these problems, in Section 3.2, p. 131, Penrose tries to defuse the idea that human mathematicians might be modeled by an unsound algorithm, by shifting gears substantially:

...Unsoundness does not help at all for a known formal system F which ... is actually known---and thus believed---by any mathematician to underlie his or her mathematical understanding! For such a belief entails a (mistaken) belief in F's soundness. (It would be an unreasonable mathematical standpoint that allows for a disbelief in the very basis of its own unassailable belief system!). Whether or not F is actually sound, a belief that it is

sound entails a belief that G(F) [essentially the sentence stating that C_k(k) doesn't halt] ... is true, but since G(F) is now---by a belief in Gödel's theorem---believed to lie outside the scope of F, this contradicts the belief that F underlies *all* (relevant) mathematical understanding.

3.10 This is really a different argument than the one we started with. The second one starts from different premises, and arrives at a different conclusion, namely that no entity can believe that *any* given algorithm can be responsible for its thought processes, because it would have to believe that the algorithm is sound, and then would reach a conclusion that that algorithm doesn't reach. The earlier conclusion was correct but unthreatening; the new conclusion is just false, for a clear reason: believing that a certain formal system underlies my reasoning processes does *not* entail belief that the formal system is sound. The problem is in the meaning of the phrase "formal system." What AI is interested in is not formal deductive systems whose theorems are exactly the "unassailable" mathematical conclusions, but in formal state-transition systems, that is, computers and programs. Confusion between these two concepts is so common that a brief digression on the topic may be worthwhile.

4. Informality From Formal Systems

4.1 Digital computers are formal systems, but the formal systems they *are* are almost always distinct from the formal (or informal) systems that their computations *relate to*. To analyze a digital computer as a formal system is merely to express its laws of operations in the form of transition rules among discrete states. When we take the inputs and outputs of the computer to refer to various real states of affairs, thenit need not be the case that there exists a consistent or sound formal system C such that whenever the computer concludes Q from P, the conclusion is licensed by C. Nothing prevents me from writing a program that, given any input P, prints out "P and not-P." There is, of course, a formal system S that stipulates that exactly this event is to occur, but this formal system is not about the entities mentioned in sentence P. If it's about anything, it's about the states of the computer, and nothing more. To make this point vivid, note that S, even though it prints out self-contradictory sentences, is "consistent," considered as a formal system, because it never says that the computer is to be in two distinct states at the same time. Consistency is essential to a formal system, because in almost all formal logics anything at all follows from a contradiction. Consistency is *not*, however, essential to computers. To continue my trivial example, I could take my self-contradictory program, and alter its behavior slightly, so that in response to "The earth is flat," it would say "False," and in response to "The earth is not flat," it would say "True," whereas in response to all other inputs P it continued to respond "P and not-P." Computers are not committed to inconsistency on all issues after revealing an inconsistency on one issue, any more than people are.

4.2 Hence if someone were to show me a listing and claim it embodied me, I would have no reason at all to believe that its conclusions were always correct (quite the contrary!). So Penrose's second argument is just fallacious. He very much wants to believe that the existence of artificially intelligent mathematicians would entail the possibility of an all-

encompassing axiomatic mathematics ("the very basis of its own unassailable belief system"), but it just wouldn't.

4.3 Hence this second argument (whose conclusion I'll call Nontheorem 1) is wrong, and the first, "Theorem 1," that human mathematicians don't use a sound formal system to do mathematics, is correct but harmless to AI research. It provides no evidence at all against the proposition that someday we'll have algorithms that are just as good, and just as fallible, as human mathematicians.

5. Ensembles of Random Theorem Provers

5.1 In addition to the basic bug in Penrose's argument, there are lots of little bugs, having to do with various technicalities, and I fear that unless these are discussed, the impression will be left that somehow by hard work Penrose has succeeded in revising his theorem to the point where it's actually relevant and true.

5.2 Penrose raises the possibility that randomness plays a key role in humans' mathematical abilities, and that such randomness might account for the errors people make. "It would be reasonable to suppose that whenever the robot does make an error in one of its STAR-assertions, then this error can be attributed, at least in part, to some chance factors in its environment or its internal workings." (p. 169) So we would have to include a random input to our robot mathematician, and this would apparently vitiate the Gödelian argument. Not so, according to Penrose: The robot's "environment can also be provided as some kind of digital input," and if we can take as our computational entity the ensemble of all possible robot + environment combinations, then "there will be a finite number of ... possible alternatives" (p. 169). I am a little unsure of exactly what Penrose means by "alternatives," but I think he means possible A + environment pairs. "Thus, the entire ensemble of all possible robots ... will itself constitute a computational system.... One could see how to build a ....Turing machine ... that could carry out the simulation, even though it would be out of the question *actually* to carry it out" (still p. 169). Now we can detect that a STAR-assertion is in error by letting the robots vote. Since errors in STAR-assertions are rare, random, and statistically independent, it will be essentially impossible for a majority of robots to be wrong, so the ensemble will serve as the A necessary to get the argument moving.

5.3 There are two huge problems with this idea. The first is that it seems to assume that A is a distillation of a human mathematician that leaves out absolutely every human motivation or interest except mathematics, and even within mathematics leaves out everything except a single problem we've told it to work on. Hence if several copies of A are told to work on a problem, and are also given an "environment" to move around in (simulated, of course), then they will all generate outputs on about the same time scale and then stop. But what if it's the case that human mathematicians can't get far without collaborating with other human mathematicians? Won't the environment have to include

them? What if A develops an interest in mathematics because of a beloved third-grade math teacher? We'll have to throw the teacher in, too. What if some A's become devotees of category theory, and others can't stand it? How will we cajole the second group into solving problems in category theory?

5.4 It seems to me that we are led to the idea that the only way to implement A is to simulate billions of copies of the entire universe on a Turing machine, and hope that a significant number develop a community of mathematicians that find our problem interesting. Okay, we're talking "in principle" here, so I'll grant that. What I won't grant (and this is the other huge problem with the idea) is that this ensemble of universes implements a sound inference algorithm that we believe is sound (which is required for Theorem 1). The computation is dominated by a simulation of the physics of the world. It's not clear how we're even going to *find* the mathematicians, more of which can presumably evolve as the simulation progresses, let alone be sure that they obey our STAR convention.

5.5 The situation is even worse with respect to Nontheorem 1, which requires us to postulate that the ensemble of universes hypothesizes itself to be a particular sound inference algorithm. Even if we grant, only for the sake of argument, that each element of the ensemble contains one or more pieces that hypothesize that they are sound inference algorithms, that doesn't mean the ensemble entertains this hypothesis or any other hypothesis.

5.6 The sheer scope of the simulations required to run the argument bothers even Penrose. "The reader may still have the uneasy feeling that no matter how careful we have been, there may still be some erroneous ... STAR-assertions that could slip through the net.... Soundness requires that absolutely *no* erroneous STAR-assertions<2> are included.... This may still seem to us, and perhaps to the robots themselves, to fall short of certainty---if only for the reason that the number of possible such assertions is *infinite*." (p. 173) To circumvent this problem, he develops an argument in Section 3.20 that only a finite number of STAR-assertions need to be considered. This argument is intricate, and seems at first to contradict the following elementary theorem of computability theory: For all c, there is a Turing machine that can deduce exactly the true mathematical statements of length c. The Turing machine merely contains a table of all theorems and nontheorems. Of course, this machine cannot actually be constructed without knowing which statements are theorems, which may serve as a warning about the exact status of existence claims in computability theory.

5.7 As I said, Penrose's argument seems at first to contradict this fact about bounded-length theorems. But his argument avoids this problem because it says that for any *given* putative theorem prover, there *exists* a bound c such that we can find a provable statement of that length or less that the prover can't prove. The argument is quite dense and hard to follow, and it seems to vacillate between trying to be a variant of Theorem 1 and trying to be a variant of Nontheorem 1. I think, though, that I can extract the essence of the argument, and possibly even strengthen Penrose's conclusion so that it applies to any of a

certain class of probabilistic inference algorithms, not just the somewhat bogus ensemble of universes that I discussed above. I will put the argument in the style of Theorem 1, so that we don't have to use the dubious postulate that if an inference system entertains the hypothesis that it is implemented by a given algorithm, it must assume that the algorithm is sound.

6. Random Ensembles and Gödel Gaps

6.1 We start by assuming we have an algorithm Q that is T(c)-reliable, in the following sense: We give Q the problem of deciding whether computation q would halt on input d. I'll write Q(q,d) to refer to the running of algorithm Q on inputs q and d. Q is supposed to compute for a while, then print out Y if q would halt on d, else N. Suppose that for all problems such that size(q) + size(d) =< c, there is a time T(c) such that if the algorithm answers Y or N within this time, it is always right. To be concrete, we'll say that the algorithm is "T(c)-reliable" if, whenever it says Y or N before time T(c)/2, and then doesn't change its mind before time T(c), then what it says is actually true. Q is a probabilistic Turing machine, which means that it has access to an extra input tape containing an infinite sequence of completely random bits. With different input tapes, it might come to a conclusion about different <q,d> pairs, but it's never wrong about a pair it comes to a conclusion about.

6.2 Penrose has in mind that his ensembles of computers are such a system, which he parameterizes with several parameters, not just the T(c) I am using. But I think the theorem works just as well for this somewhat broader class of inference algorithms. Let's use the term "short_c-theorems" for formulas of the form $halts(q,d)$ and $not halts(q,d)$, for which size(q) + size(c) =< c and Q(q,d) prints Y or N reliably within time T(c), as described.

6.3 Here's what it would mean for the community of human mathematicians to be T(c)-reliable in this sense: Suppose we give the mathematicians a problem, Does q halt for input d?, to work on, where size(q) + size(d) =< 100. After 500 years, if they don't have a solution, we just forget about this problem. Otherwise, they'll say Yes or No, so we give them another 500 years to try to find an error in their solution. If they stick by it after that time, we label it "proved." Let's suppose that no buggy solution survives this filtering process, and that, given 1000 years, the mathematical community would never let a buggy solution to a problem of size 100 remain STAR-flagged. And it might be the case that for all c, we could take T(c)=10c and the human mathematical community would never make a mistake about a size-c problem given 5c years to solve it and 5c years to check the solution. If you don't buy that, perhaps it will help to let T(c)=100^c, or take T(c) to be any other computable function whose algorithm has size O(log c), an easy requirement to satisfy.

6.4 Now what Penrose does in essence is to define a derived computational system

Q_c(a) that takes as input an algorithm description a, and runs Q(q,d) for all inputs q and d such that size(q) + size(d) =< c. It runs Q for only T(c) time units per <q,d> pair, and collects all the short_c-theorems. It then enumerates all deductive conseqences of these theorems (each of which is a formula of the form $halts(q,d)$ or $not halts(q,d)$). If $not halts(a,a)$ ever appears in this enumeration, then Q_c(a) stops. Otherwise, it goes on forever. Clearly Q_c is sound for all c, in the sense that if it halts for input a then machine a actually runs forever given a copy of itself. What we now show is that it has the usual blind spot.

6.5 Penrose's key observation is that the size of Q_c, considered as an algorithm, grows only slowly with c. That's because c occurs in it only as a loop limit and as the argument to T(c), which itself (i.e., size(code(T))) grows only slowly with c. Hence it is easy to pick a c* such that 2.size(code(Q_c*)) =< c*. Define Q* to be Q_c* . If we let k=code(Q*), then consider what happens when we run Q* with argument k, a computation we call Q*(k). The conclusion $halts(k,k)$ or $not halts(k,k)$, if derived, will follow from a Q computation of size =< c* (because k+k =< c*), so if that conclusion is included in the short_c*-theorems it will be correct. Now if Q*(k) halts, then it says Q*(k) does not halt, so by soundness it must not halt, but Q* cannot infer it (the usual Gödel gap). Every short_c*-theorem of Q is a theorem of Q*, by construction, so Q does not give an answer on the input <k,k>.

6.6 So far the argument is unproblematical (and quite ingenious), and shows that any T(c)-reliable algorithm must be incomplete. We can call this Theorem 2. The only trouble is that Penrose can't quite get from that conclusion to the one he wants, which is that the incompleteness occurs at a point where humans have no trouble drawing the correct conclusion. And at first blush this looks like such a case. Just take c* to be greater than the number of characters in the conclusion, and you have a short_{c*}-theorem for people that isn't a short_{c*}-theorem for Q. Unfortunately, that isn't quite as easy as it sounds. In the proof of Theorem 1, we were asked to picture a situation where we had a listing of an algorithm that was claimed to embody us. We were then given a theorem that the algorithm couldn't prove, except that we weren't really given it --- we were given a way of constructing it from the listing. Suppose that AI triumphs completely, and you hold in your hand a CD-ROM containing a listing of a computerized Gauss (call it G). Can you then apply the construction described above to derive a theorem that mathematicians find easy to prove and that G cannot prove? No, because G is not the Q we need to start the construction. To create Q, we would need to simulate lots of mathematicians (including a von Neumann and maybe even a Penrose as well as our Gauss), plus a large chunk of their environment. It's not at all clear that AI research would ever get to the point where it could take a stand on the existence or nature of Q. Furthermore, suppose that a candidate for Q were suggested. How would we evaluate it? In particular, how would we ever prove that it was T(c)-reliable? We would have to show somehow that no matter what random bits were input to the algorithm, it would never make a mistake. I conjecture that the possibility would always remain open that both the algorithm and the human mathematical community are not T(c)-reliable. Even worse, there's no way even in principle that we could determine that Q duplicated exactly the

conditions prevailing in our universe. The best we could hope for is that Q be indistinguishable from our universe, that it apparently yield "typical" behaviors. But it could be the case that arbitarily small physical differences could change a T(c)-reliable universe into a non-T(c)-reliable one.

7. The Infallibly Fallible Robot

7.1 Penrose ends his treatment of Gödel's Theorem with the strange fantasy of Section 3.23, in which a robot mathematician (MJC, for "Mathematically Justified Cybersystem") gets into an argument with its human creator (Albert Imperator, or AI). AI convinces MJC that if MJC and its fellow robot mathematicians even entertain the possibility that they embody any algorithm Q, then there is a sentence (which I'll call Omega(Q), simplifying Penrose's notation a bit) that is true but unprovable by MJC and its robotic colleagues if they are infallible. MJC sees the logic of this argument, and, refusing to abandon belief in its infallibility, goes insane. AI is forced to destroy MJC and all its fellows.

7.2 This fantasy is incoherent at several levels. It seems to assume that infallibility is such an integral part of the AI research program that the robots can not even conceive of not possessing it. Yet MJC demonstrates spectacular fallibility in concluding at the end of the dialogue that its initials actually stand for Messiah Jesus Christ and that it is divinely guided to its mathematical conclusions. It seems to me that it would be much less traumatic for MJC just to say, "I guess we must very occasionally make mistakes; in fact, my impetuous assertion of infallibility was just such a mistake!"

7.3 The dialogue has MJC hearing and agreeing with the argument for Omega(Q). "Yet .... it's impossible that [we] can accept Omega(Q), because, by its very nature of your Gödel's construction, is something that lies outside what can be STAR-asserted by us.... It must be the case that ... the procedures incorporated into Q are *not* after all the ones you used." (p. 183) Surely MJC has a pretty good case here. It is agreeing with the argument for Omega(Q); it even shows that it understands several implications of it. It sounds odd for AI and Penrose to continue to talk as if MJC really is unable to conclude Omega(Q). If MJC were to affix a STAR to Omega(Q), on what grounds would we quibble?

7.4 Of course, in imagining MJC's possible behavior, I'm just coasting on the anthropomorphic fuel Penrose provides by painting such an extravagant picture of what MJC can do. And that brings me to what is really incoherent about this dialogue. It seems to knock the keystone out of Penrose's whole argument, which is that finding one tiny gap in the ability of robots to do mathematics would destroy any hope that they could ever really understand anything. If that's the case, then he would presumably believe that nothing like the dialogue between AI and MJC, in which the robot seems to understand every nuance of the conversation, could ever actually take place. The casual reader, who is urged by Penrose to skip all the hard stuff in Chapter 3, and go right to Section 3.23, is surely going to draw the conclusion that Penrose thinks that robots can do almost any

task, except prove a certain theorem. He titles the section "Reductio ad absurdum--- a fantasy dialogue," and I suppose it could be taken as trying to show that no matter what powers of understanding we imagine we could give to robots, we will also have to imagine them having strange lapses (but strange lapses that are consistent with infallibility in some way), and that therefore we mustn't impute those powers. But it's as if I presented a proof that all toasters were useless by hypothesizing a talking toaster and showing that it must burn at least one slice of toast.

8. What Does Penrose Think AI Is?

8.1 Now that I've torn Penrose's argument to shreds, it's time for a spirited rebuttal of his critique of the computationalist theory of consciousness. Unfortunately, Penrose has no critique. Indeed, he says almost nothing about points of view different from his. The two and a half pages of Section 1.14, "Some Difficulties with the Computational Model," are almost all there is. There's a brief reference on p. 149 to what he supposes is the computationalist view of mathematical ability, a somewhat odd discussion of "top-down" vs. "bottom-up" programs in Section 1.5, and a few other remarks in passing throughout the book. One might conclude from this silence that AI has had nothing in particular to say about consciousness, but in fact there has been quite a bit of theorizing. In particular, between the publication of Penrose's previous volume and "Shadows of the Mind" appeared Daniel Dennett's "Consciousness Explained," which provides a rich set of ideas for thinking about computation and consciousness. I would have been quite interested in seeing Penrose's critique of that set of ideas. But there are (by my count) exactly two references to Dennett in "Shadows of the Mind," both in passing.

8.2 Let me deal with his observations, what there is of them, in reverse order. Section 8.2, "Things that computers do well---or badly," distinguishes problems on which we would expect computers to do better than people from problems on which we would expect people to do better. The analysis is "a little crude," as Penrose admits, but basically correct. Suppose a problem can be analyzed as a search space with a branching factor of p. Then a computer might examine on the order of T=t.p^n search states if the solution is m moves away and it takes time t to explore a state. "It follows ... that games for which p is large, but can effectively be cut down significantly by the use of understanding and judgement, are relatively to the advantage of the human player." (p. 397) One might wonder what this has to with consciousness, but Penrose, as I said before, assumes that awareness and judgement are two manifestations of the same underlying property. "...The essential point is that the quality of human *judgement*, which is based on human *understanding*, is an essential thing that computers lack, and this is generally supported by the above remarks...." But nothing of the sort follows from the formula T=t.p^n. AI practitioners routinely think in terms of this formula when they look for heuristics to cut down p. Furthermore, there is no reason in principle why the computer needs to stay in one search space. The problem of finding the right search space can sometimes be phrased as a search problem itself.

8.3 Finally we work our way back to Section 1.14, which is a review, at a sketchy and

shallow level, of "difficulties" with the computational model. At the risk of stating the obvious several times, let me review these "difficulties."

8.4 On p. 42, he says,

....It is the mere 'carrying out' or enaction of appropriate algorithms that is supposed to evoke awareness. But what does this actually mean? Does 'enaction' mean that bits of physical material must be moved around in accordance with the successive operations of the algorithm? Suppose we imagine these successive operations to be written line by line in a massive book. Would the act of writing or printing these lines constitute 'enaction'?

8.5 Presumably awareness will not be "evoked" by some computation; it will be *constituted* by some computation, and not just any computation. (See below.) And "enaction" does not mean recital of a sequence of operations; it means taking part in a certain interaction with the environment. It's as if someone objected:

It is the mere 'carrying out' or enaction of appropriate switch transitions that is supposed to control a furnace. But what does this actually mean? Does 'enaction' means that bits of metal must be moved around in accordance with the successive operations of the thermostat? Suppose we imagine these successive switch transitions to be written line by line in a massive book. Would the act of writing or printing these lines constitute 'enaction'?

8.6 The same objection has been made, with slightly more subtlety, by John Searle (1992) and Hilary Putnam (1988). In each case it rests on a perverse identification of a computer program with a trace of its execution (a *particular* trace of a *particular* execution), which is simply absurd.

8.7 "In any case," continues Penrose,

it would presumably not be the case, according to [computationalism], that just any complicated algorithm could evoke ... awareness. It would be expected that some special features of the algorithm such as 'higher-level organization', or 'universality', or 'self-reference', or 'algorithmic simplicity/complexity', or some such, would be needed before significant awareness could be considered to be evoked. Moreover, there is the sticky issue of what particular qualities of an algorithm would be supposed to be responsible for the various different 'qualia' that constitute our awareness. What kind of computation evokes the sensation 'red', for example? What computations constitute the sensations of 'pain', 'sweetness', 'harmoniousness,' 'pungency', or whatever? Attempts have been sometimes been made by proponents of [computationalism] to address issues of this nature (cf. Dennett 1991, for example), but so far these attempts do not strike me as at all persuasive. (p. 42)

8.8 It may be that Penrose finds the computationalist theory unpersuasive, and in a sense he's surely right. *No one* has a completely worked out theory of consciousness, Penrose least of all. But it would have been sporting of him to tell the reader what he takes the computationalist position to be before dismissing it. Since he didn't, I will. What follows is my interpretation of a theory due to Minsky (1968, 1986) and Dennett (1991). (See also Gelernter 1994.) I make no claim, however, that I am representing their views or the views of a majority of the AI community.

9. The Computationalist Alternative

9.1 The basic idea is that a computational system can often be said to have a model or theory of some part of its environment. I hesitate to use either the word "model" or "theory" here, because of the danger that some will assume I mean to use these words in the senses they have in mathematical logic, and I emphatically do not. Perhaps "simulacrum" is the right word; some computational systems maintain simulacra of some part of their surroundings. A simulacrum allows the system to explain and predict the behavior of the world around it. It's very important at the beginning of the exegesis to understand that when I use words like "explain" and "predict" I mean them in the least anthropomorphic way possible, as when one might say that an antiaircraft missile predicts the future locations of its target.

9.2 Simple systems can get by with simple simulacra, but the more complex the organism, the broader must its skills be in relating one part of its environment to others, so that at some point it becomes legitimate to talk of the organism's simulacrum of the world. And at some point the organism must include *itself* in the model. This is not meant to be a mystical step. A computer taking an inventory of office furniture will include itself in its simulacrum. Of course, nowadays the computer will not distinguish itself from other workstations, or hat racks for that matter. But if the same computer is used to control the movements of the office furniture (using robotic sensors and effectors), then some interesting singularities arise. Some items of furniture will, as they are moved, give rise to moving patches of pixels in the images the computer's camera produces. But at least one item, the camera itself, will cause quite different sensory events when it is moved. The computer's world simulacrum must, to be accurate, reflect the asymmetry between these different kinds of physical objects.

9.3 So far, no consciousness, and nothing out of the ordinary either. We have robots in our lab that watch their arms move toward targets, and they use different models for the arm and the target (Grunwald et al. 1994). The point where consciousness arises is where an agent requires a model of itself as a behaving agent, and even there consciousness does not depend on the agent having just any model of itself; it must have a model of itself as a being with free will, a transcendental ego, sensations with certain qualia, and so forth. This model is based on attributes that the being really does have. Free will is based on the fact that the computations the agent carries out really do influence its behavior. The transcendental ego is based on the fact that the agent must behave as a single entity. Qualia are based on the fact that sensory information really is processed. The model goes beyond the truth, but it's not really a lie; it's a self-fulfilling fiction.

9.4 One pleasant (perhaps suspiciously pleasant) aspect of this theory is that it explains so nicely why the theory seems incredible. Our self-models deny that things like qualia are computational entities. Of course, they also deny that qualia have to do with large-scale quantum coherence, or any other physical phenomenon. That's why qualia seem so mysterious: any explanation of consciousness in terms of nonmysterious entities is ruled out as if by reflex.

9.5 This theory has plenty of difficulties. To my mind, its biggest problem is that it raises a question that it has yet not answered, which is: *How* do we tell when a computational system X has a simulacrum of entity Y? The answer cannot depend on whether it's convenient for outside observers to impute this property to X. We have to start from an observerless universe and infer observers. But I don't think these problems are insurmountable, and they suggest some interesting lines of research.

9.6 The theory also makes a prediction, which Penrose anticipates on page 42:
"...Any clear-cut and reasonably simple algorithmic suggestion [for a theory of qualia] ... would suffer from the drawback that it could be implemented without great difficulty on a present-day electronic computer. Such an implementation would...have to evoke the actual experience of the intended [quale]. It would be hard ... to accept seriously that such a computation ... could actually experience mentality .... It would therefore appear to be the case that proponents of such suggestions must resort to the belief that it is the sheer *complication* of the computations ... that are involved in the activities of our own brains that allow us to have appreciable mental experiences.
9.7 The first half of this paragraph is correct; the second half is wrong. It does seem to be the case that consciousness is no big deal. I believe I could program a computer to be conscious; it may have already been done by accident. The reason why it's so hard to detect is because computers are so stupid and clumsy. It's child's play to program a computer to perceive its own sense-event descriptors, but if it can't actually see anything interesting, and can't really carry on a conversation, then it won't have much to say about its introspections. Hence the bottleneck in getting computers to be conscious is getting them to be smart. Intelligence is a prerequisite for (recognizable) consciousness, not the other way around, as Penrose would have it. "Sheer complication" is a red herring. The cerebrum is conscious, and the cerebellum is not, because it uses a certain kind of model of itself, and the cerebellum doesn't. The kind of intelligence that I am talking about here is not what is measured by IQ tests, but a general ability to integrate information about the world. I'm quite sure that mammals have enough intelligence to be nontrivially conscious, and quite sure that existing computer programs do not.

9.8 Curiously, the idea that consciousness will turn out to be quite simple is in harmony with Penrose's ideas. If we flip back to page 149, we find him expressing much the same conclusion in his framework: "[Understanding] need not be something so complicated that it is unknowable or incomprehensible.... Understanding has the appearance of being a simple and common-sense quality."

9.9 This is not the only place where Penrose's views run parallel to the computationalist view. The second half of the book is taken up with the problem of the observer in quantum mechanics, the same problem he wrestled with in "The Emperor's New Mind." As I mentioned above, for computationalism the problem arises in finding an objective way to draw lines around systems that model themselves. In quantum mechanics the problem arises at a more fundamental level, when we try to find macroscopic objects in a

world of wave functions. But it's likely a solution to the quantum-mechanical observer problem would shed light on the computational observer problem.

9.10 To summarize: Computationalism is scarcely examined, let alone refuted, by this book, which stakes all its marbles on the Gödelian-gap argument, and loses. A computational theory of consciousness has many problems, but is better worked out than any alternative, including especially Penrose's. It is not arrogance, but a humble desire for truth, that leads some researchers to pursue the computational theory as a working hypothesis. The biggest obstacle to the success of this theory is not the absence of an account of conscious awareness *per se*, but the fact that AI has as yet made little progress on the problem of general intelligence, and has decided to focus on a more modest strategy of studying individual cognitive skills. The burden is on AI to show that this research program ever will lead to a theory of general intelligence. People like Penrose should declare victory and withdraw.

Acknowledgements

Thanks to Richmond Thomason for helping me with bibliographical searches.

Notes

<1> Roger Penrose is the Rouse Ball Professor of Mathematics at the University of Oxford. Same Ball.

<2> Penrose actually has STAR_M-assertions here and in a couple of my later quotes. I don't think the distinction between these and STAR-assertions simpliciter is important for my discussion.

References

Ball, W. W. R. & Coxeter, H. S. M. (1939) *Mathematical Recreations and Essays.* 11th Edition. New York: The Macmillan Company

Dennett, D. (1991) *Consciousness Explained.* Boston: Little, Brown and Company

Gelernter, D. H. (1994) *The Muse in the Machine.* New York: Free Press

Grunwald, G. Hager, G. & Hirzinger, G. (1994) Feature-Based Visual Servoing and its Application to Telerobotics (with and). *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 164--171. IEEE Computer Society Press

Heawood, P.J. (1890) *Quarterly Journal of Mathematics, 24,* 332-338.

Kaplan, D. & Montague, R. (1960) A paradox regained. *Notre Dame Journal of Formal Logic, 1,* 79-90. Reprinted in *Formal Philosophy: Selected Papers of Richard Montague*, Richmond Thomason (Ed). New Haven: Yale University Press

Kempe, A.B. (1879) *American Journal of Mathematics, 2,* 193-200.

Lakatos, I. (1976) *Proofs and refutations: the logic of mathematical discovery.* John Worrall and Elie Zahar (Eds). Cambridge: Cambridge University Press

Minsky, M. (1968) Matter, mind, and models. In *Semantic Information Processing.*, Marvin Minsky (Ed). Cambridge, Mass: The MIT Press

Minsky, M. (1986) *The Society of Mind*. New York: Simon & Schuster.

Putnam, H. (1988) *Representation and reality.* Cambridge, Mass.: MIT Press

Searle, J. R. (1992) *The Rediscovery of the Mind.* Cambridge, Mass.: MIT Press

Shaw, R. (1958) The paradox of the unexpected examination. *Mind, 67,* 382-384.