Some Responses to the Critics of AI

Several objections have been made to the notion of "artificial intelligence." In this paper I will present the most ubiquitous of these and show that they each seem to be problematic in some way. I shall then sketch one positive argument and present another positive argument for artificial intelligence that has been somewhat ignored. A conclusion will then be drawn based on the considerations discussed in the present paper. Of course, none of this entails that "artificial intelligence" is at all possible, just that it hasn't been successfully shown to be impossible. (Even the positive arguments are far from being conclusive or completely persuasive. The primary role of this paper is largely negative.)

I am using the definition of artificial intelligence[1] (hereafter, AI) as used by Douglas Hofstadter in *Metamagical Themas: Questing for the Essence of Mind and Pattern* (Hofstadter 1985, pg. 631).

> "- the belief that a programmed computer can, in principle be conscious. Various synonymous phrases could be substituted for ”be conscious” here, such as
>
> * think;
> * have a soul (in a humanistic sense rather than a religious sense);
> * have an inner life;
> * have semantics (as distinguished from ”mere syntax”)
> * have content (as distinguished from ”mere form")
> * be something it is like something to be (a weird phrase due to T. Nagel);
> * have personhood; "

[1] This paper is NOT about the so called "computational theory of mind." While this theory is interesting philosophically in its own right, and there is considerable overlap between the two subjects, they are not the same. Hence, in the interests of making this paper manageable, I am not dealing with it here. One can look at the two issues as "mirror images" of each other. The computational theory of mind looks at to what extent brains are like conventional computing devices, and artificial intelligence concerns itself with to what degree "intelligence" can be imparted to computing devices.

This paper will be divided into 5 sections; the zeroth is a brief note to the reader concerning some issues in computer programming and how they relate to this paper. The first section, which is the beginning of the paper proper, consists of a discussion of some "naive" anti-AI arguments. These arguments, while ubiquitous, do not address the feasibility of the AI project(s) at all. Nevertheless, they are important because they set the stage for further developments of my position. The second section will consist of a series of biologically flavoured arguments concerning alleged powers of the brain including creativity and brain plasticity.The so-called "biological" arguments will be wide ranging in rigor and sophistication. This explains why there is a great variation in the kinds of answers in this section's discussions. Then, in the third section, will be found a discussion of several mathematical arguments against AI. Fourthly, I will briefly discuss two positive arguments for AI, including one rather unique one which has recently come to my attention. I will then finish up (Section IV) with a conclusion which will attempt to draw everything up into a reasonably clean little package.

Section 0: Note to the reader:

I will be using Scheme[2], a small, widely-ported computer programming language[3] to produce several simple examples of what I am talking about in several places. I am aware that some readers will not be familiar with Scheme or with programming as a whole; I ask their indulgence and provide several references to starting programming and Scheme in specific in the references section (see Chapman 1991, Friedman, Wand, and Haynes 1996). In this paper, Scheme code will be in **9-point Geneva Bold** and output from the Scheme interpreter will be in 9-point Geneva. Various names of products are used throughout this paper; any copyright, trademark, etc. is hereby acknowledged. Mention of a product is not necessarily an endorsement.

---

[2] Scheme is an easier to understand dialect of LISP, the *lingua franca* of AI.

[3] I could at this stage define computer, program, programming language, etc. but I will refrain from doing so to avoid poisoning the well for or against my position. Definitions of some of these will be invoked or referenced and explained further on.

Section I: "Naive" arguments against AI

One criticism of AI presented is that the field has been around for 50 or so years and has not yet produced anything close to human thinking. I agree completely; and further I think that some researchers were/are too optimistic. However, this does not rule out the possibility of AI. After all, natural selection took at least three and one half billion years to produce thinking in humans; we'd be arrogant[4] to think we could replicate it in fifty. Another way of looking at the issue is to consider the case of flying machines; Leonardo developed plans for flying machines centuries before one was built. No doubt he was ridiculed for promising flying machines, but they were eventually built. In other words, the "50 year argument" is a non sequitur.

Some have criticized AI by saying something like: "AI is unexciting. Why do it anyway? Either it will work or it won't." I ask these people whether they think building airplanes was a waste; after all, we can build flying machines or we can't. Right? Well, it is true, but missing something. Both AI and inventing flying machines come in stages; further, the intelligence that AI is trying to duplicate comes in degrees. (Again, just as flying ability comes in degrees.) Intelligence is not an either/or - consider worms, lobsters, cats, elephants, dolphins, chimpanzees, and so on. The intelligence (that is the degree of variation in responses to their environment, as well as other factors) of other animals varies widely. As for it being unexciting, so what? Not all research areas are interesting to all people; people should do what they find interesting. And some people find AI interesting.

Another criticism of AI projects, particularly that of the fifth generation project of the early 1980s in Japan, is that they are too expensive and waste large sums of money. I agree completely that the fifth generation project was overly ambitious; however this does not

---

[4] This does make some earlier AI researchers very much arrogant. However, not all AI researchers are such, and even if they were, it doesn't address the possibility of AI at all.

entail that money shouldn't be spent on AI or that nothing came out of the project. As pointed out in Downing and Covington 1992, the fifth generation project produced advances in parallel processing and logic programming, particularly using the programming language Prolog.

Another possible anti-AI argument is that found in Bunge 1985. Bunge suggests that even if we could build artificially intelligent machines, we should not because we want machines that are useful and hence do not make mistakes or are otherwise imperfect. This is a strange position to hold; after all, everyone is aware that no machine (and no person) is perfect. Since an AI would of necessity have a different background of experience, innate capacity, etc. obtained from its human creators and its own unique look at the environment, its perspective would be unique, and may provide insight into things in different ways. This would be much like how communicating with some sort of extra-terrestrial being would be useful. Furthermore, it is useful to point out that it appears the ability to commit errors is necessarily part of intelligence in general. The ability to produce novelty also means one must "go wrong" at least some of the time. The upshot is that to have more sophisticated tools, one also must bring in more chance for error and mishap. An AI would/could be maximally sophisticated as a "smart tool", but this may mean it would not desire to <u>be</u> (merely?) a tool. However, this does not bear on the <u>possibility</u> (as opposed to the desirability) of AI.

A final criticism in this section I do not feel much obligation to talk about, but nevertheless will spend some time on, is a dualistic or immaterialistic objection to AI. This would be Descartes' objection - that it is impossible to build an intelligent machine because there's something immaterial, and hence ontologically different, in the domain of human intelligence. There are actually two answers to this. If there <u>is</u> basically a "miracle" connecting mind and body, then I agree, AI is likely impossible[5]. However, if one can study the mind at all (even if

---

[5] To be entirely fair, one could imagine that a god infuses an AI with the *deus ex machina* at the appropriate time, but this is of course irrelevant to our current discussion.

it is in an ontologically different category), it may still be possible to instantiate or implement (whatever is necessary) this "new stuff" in a computer. (This, being completely hypothetical, of necessity sounds rather odd.) Depending on what this new "stuff" is, the way in which it could interact with a computer would vary.

Section II: Arguments from biology and related issues

Many people have pointed out that certain things that people do with their intelligence are non-algorithmic in character. This particular objection has many variations. Penrose thinks that certain kinds of mathematical reasoning are nonalgorithmic (Penrose 1989,1994,1997); Bunge (1983a, 1983b) points out that there is no algorithm for scientific discovery. One could also point out that artistic expression, like being able to compose a symphony or write a poem, is nonalgorithmic. I am in perfect agreement with those people with this point; such activities are NOT algorithmic, but this does not entail that the activity cannot be programmed. The underlined part of my previous sentence is the source of all the misunderstanding[6]. Creativity is produced by emergence as well as subcomponents, and subsubcomponents and subsubsubcomponents, and so on, down to a low level which does follow some precise chemobiological laws. The "behaving in a strict fashion level" of the nervous system is actually likely below the level of the neurons, perhaps that of the neurotransmitters. However, this exact detail is relatively unimportant; it simply matters that it exists. Further, it doesn't matter whether the laws it obeys are stochastic or deterministic. Whether something obeys probabilistic laws or deterministic laws, there are still objective regularities in the way in which it operates[7]. This is relevant to keep in mind when I discuss computers and creativity) and is true for both nervous systems and

---

[6] Arguably, this is the single most prevalent worry about AI, and the single hardest point to grasp in this paper (or any discussion of the possibility of AI.) Once one realizes that computers can perform nonalgorithmic (in the mathematical sense) activities, a lot of AI objections just disappear.

[7] This is just a statement of the principle of lawfulness, a general ontological principle common to all science and technology. See Bunge 1977 for more.

conventional computer hardware. Probabilism appears to be a problem for the concept of AI (as it appears at first glance to go against the theory of Turing Machines), but is used by researchers in the field and is in fact ESSENTIAL, as we will see later.

Both the probabilism view and the "emergence from subcognitive components view" (as we shall see later, this could also be called the heuristics view) have been long held by some workers in the field of AI. (For instance, see Hofstadter 1979, particularly pages 289-309, 641-680). The "emergence from subcognitive components viewpoint" recognizes that there are plenty of activities which humans perform that are nonalgorithmic, but also recognizes that this poses no intrinsic limitation to what is implementable on computing devices. Clear examples of this are described in *Fluid Concepts and Creative Analogies* (Hofstadter et. al. 1995). It will be instructive to go over an example here for purposes of illustration.

Consider the program 'Numbo', which "plays" the French game *Le compte est bon*. In this game, players are given a number from 1 to 150 called the target, and 5 other numbers from 1 to 25 called the bricks. The goal of the game is to produce the target from the bricks by performing any number of additions, subtractions, and multiplications on them, in any particular order. It should be relatively apparent that humans do not use a "brute force" solution to this sort of exercise; and so to have the computer solve the problems this way would be to totally miss the point. (This is why I, and various AI researchers, find programs like "Deep Blue" to some degree wrongheaded, and NOT good examples of AI[8].)

[8] To the extent that a program uses brute force, a program is to that degree in violation of the spirit of AI, as far as I and various AI researchers (notably Hofstadter) are concerned.

Since Numbo was developed to model (to some degree) what humans seem to do when playing *Le compte est bon*[9], its strategy is far from being brute force, and emerges from several 'subtasks', such as recognizing 'closeness' to the goal upon simple operations. For example, if the target is, say, 113, and the bricks are 10 10 21 8 1, Numbo, like humans, will tend[10] to 'notice' that 10 times 10 is 100, which is close to 113. The tending here is important; Numbo doesn't act strictly deterministically. (Note that this violates typical definitions of computer in discussions of formal computation, recursion theory, etc. For example, *Set theory, logic and their limitations* (Machover 1996) contains the deterministic assumption in its chapter on recursion theory. But since Numbo runs on real hardware and is a real computer program, this simply shows Machover's definition to be unrealistic[11].) Numbo is also provided with certain basic facts that most adult humans would have on hand without calculation when solving such problems; for instance, the multiplication tables up to 9x9 or so.

My point in this section is not to intimately describe what Numbo does; plenty of details are found in Hofstadter 1995. What should be gained from this is that one can program a computer to accomplish a goal in a nonalgorithmic manner. However, if it still appears that Numbo is performing an algorithm, or is an algorithm, that would also be correct. Note that the at highest level of description (Numbo's "behaviour") the program does indeed act nonalgorithmically, just as a scientist doing science doesn't act algorithmically at the behaviour level. (But at a

---

[9] This it does, to a remarkable degree. Hofstadter 1995 contains transcripts of Numbo's internal states, as well as that of a person recording her own thoughts as she solves one of these problems.

[10] The tendencies and random biases of Numbo are stochastic. Nobody ought to claim that at the lowest level any AI yet makes use of randomness. Hofstadter points out repeatedly, (and I am in perfect agreement) that many critics of AI simply do not understand how "layered" computer programs can be.

[11] Admittedly, Machover needs the deterministic assumption to use computers pedagogically to present the Gödelian incompleteness results, but this position on his part does beg the question in favour of Penrose *et. al.* as we shall see later.

"lower level" 'the scientist' acts in the "strict fashion" mentioned above.)

The nonalgorithmic level emerges out of more basic algorithmical levels. Emergence of unforeseen[12] properties of computer programs is nothing surprising; even relatively simple programs are unpredictable at one level of description or another. Take the following transcript from a Scheme session:

: **(define bizarre (cons 1 2))**
bizarre

: **bizarre**
(1 . 2)

: **(set-cdr! bizarre bizarre)**

Mutation of objects (as with the **set-cdr!** above) in Scheme has unspecified side effects. Try the code above on several different Scheme implementations to see the result of the last statement; it will be different on different implementations[13]. Of course, if we knew how our particular Scheme interpreter was implemented, we might be able to guess the side-effect of the last statement. On the other hand; it could be beyond our capacity to figure out. One of the biggest problems facing computer programmers these days is that programs are getting so large and complicated, precisely figuring out their "output" from the source code is sometimes (epistemologically) impossible[14]; one simply has to

---

[12] Of course, some might claim that the outcome of Numbo's processing was perfectly determined in advance. I doubt that it was ontologically indeterministic, but do not think we know whether human brain states are either. Numbo and humans are certainly both very much epistemologically indeterministic.

Also note that I am using the concept of emergent properties developed at great length in Bunge 1977. Briefly, an emergent property is a property possessed by an system of objects that the components individually do not possess.

[13] For instance, in MacGambit, the interpreter tries to display the new pair, and gets caught in an infinite regress. However, in PLT Scheme, it doesn't try to display the new pair at all, and hence doesn't get trapped in the regress.

[14] Even more extreme is the case when software, the operating system supporting, it and the hardware interact in unforeseen ways. For those

run the program through and find out what happens. In the simple case above, it is unlikely, but possible that would be so. (Note: I am not saying one has to exploit unspecified behaviour of a language in order to create some 'unpredictableness' in a computer program.)

Consider also the case of a chess program. While many chess programs aren't terribly useful AI-wise because they are too "brute force", nevertheless they are quite unpredictable. If they were too predictable, why would people (and the creators of said programs, no less!) bother to play them? The lesson to be learned from this section? Complex computer programs have emergent properties just as much as assemblies of neurons do. Simple (neural assemblies and programs) less likely. Note that I am not saying the "correspondent" in an AI system to a neural assembly is a complex program. It could be what the neurons do. For example, neurons produce neurotransmitters this way and respond this way to outside stimuli or any number of other things. (I do not wish to commit AI to any sort of strong functionalism. It does go without saying, however, that a weak form of functionalism is necessary for AI to be possible.)

Another worry about AI is the following: it might be argued that a computer program cannot produce true novelty because it can 'only do what it was programmed to do', or that it seems paradoxical to mechanize creativity. This objection is very similar to the preceding point and hence has a similar answer. Moreover it will help us to better understand the answer I gave to the 'nonalgorithmic activity' worries[15].

What does it mean to be creative? I will first state that all creativity necessarily involves something old in part. Consider what would happen if, for instance, someone claims to have created a work of

---

that might be inclined to think this is caused by programmer laziness, they should consider that pieces of modern software and operating systems are often millions of lines of code in an often non-rigorously specified language, which in turn is often run on bargain basement hardware. No wonder, then, that "strange things" happen.

[15]In some sense, these two are two sides of the same worry, in so far as they are both concerned with novelty in thought.

art but was totally unlike any work of art ever created; that is, neither wasn't a sculpture, a painting, or a symphony, or anything like that, but was totally radical. This new item could hardly be called a work of art, for how would one call it such if it had nothing in common with other such works? A similar line of reasoning could be used (*mutatis mutandis*) for a mathematical invention[16], a scientific theory, a technological advance, and so on. Note that the above argument applies even if neurons spontaneously fire, because that's just another way of looking at the creative process from the "inner" rather than the "outer" perspective.

So now that we have seen that to some degree variations on a theme are part of creativity, what part is not 'computerizable'? We cannot leap wholly beyond the system of neurons in our head. (cf. Lucas, who thought otherwise - we'll see more of him when I discuss Gödel's Incompleteness Theorem.) Again, this suggests programmed heuristics. One cannot will, for example, a scientific discovery[17], but one can do things that are conducive to producing it.

Creativity is thus programmable, but not likely from the top-down perspective. This means one would program a bunch of little sub-creative "agents" whose combined emergent behaviour (interaction) leads to creativity[18]. After all, neurons by themselves aren't creative; only assemblies and systems of neurons are. Interacting computer programs already can be said to "talk" to each other, as when a WebStar http server uses AppleEvents to "talk" to a FileMaker Pro database. Each program can reprogram itself in some sense, based on the activity of the other. I am not saying that WebStar and FileMaker Pro are creative

---

[16] Even a mathematical platonist must admit that her "discoveries" must fit into the framework of what is called mathematics.

[17] Note in passing this discredits the ridiculous (and logically impossible!) idea that the goal of AI is to produce perfectly sound reasoners.

[18] As Dennett points out (see Dennett 1987, 1991), this does not commit the homunculus fallacy if the "agents" are "stupider" than the higher level feature they comprise/make up/produce via emergence.

together - I am simply saying they have emergent properties when allowed to interact with one another in the 'virtual environment' on a computer. Neither program's author has to be directly aware that the other program exists, or will exist, in order for these features to get used.

An illustration of a more sophisticated version of the above objection follows. Suppose we tell a computer to investigate some basic number theory. The computer does not choose to do so on its own, and hence it lacks some basic creative decision making power. This to my mind seems to beg the question against those in favour of AI. An argument needs to be given to show how computers could not do so in principle. Note that computers can already monitor their internal states and take action independently of their users or programmers. For instance, one could have a computer defragment its storage volume if its fragmentation rose above a certain level. Of course, someone might say, "Well, it has to be told to that, too." But what human being is born with enough innate ability to NOT require motivation to do things (including acquire knowledge), help to acquire said knowledge, and so forth? No computer can decide to do things totally on its own, but nor can a young child or even an adult. (I am quite willing to hazard a guess that if AI is accomplished, the first AI programs will be very childlike.)

I will now discuss brains and the nervous system a little. Brains think; however, what other things can think? This is the fundamental issue of this paper. The anti-AI crowd often holds that only brains can think. Now, how do we know that? Well, the only things currently that think are brains, but that does not rule out the possibility of something else. After all, before the invention of airplanes, only bats, (some) birds and (some) insects could fly - and some "antiflying critics" did indeed say that humans will never fly because humans don't have wings - the "right stuff" as it were. In this section, let us see if we can see rudiments of thinking in computer programs, further, examine claims that only brains are made of the right stuff (plastic neurons) to think.

One process that thinking things must be able to do is learn from their environment and from other things- that is, reorganize their internal states based on outside stimuli[19]. One very simple example of a computer program that learns from its environment is url, a program (of a sort called a 'bot') written in perl by Kevin Lenzo. (See Lenzo 1996 for details and locations to download its source code.)

Now, as one can see from the following transcript[20], most of url's behaviour is 'canned' and 'creativity' is strictly random (as opposed to weighted randomization and emergence which as I discuss elsewhere, is how it appears one ought to implement intelligence) This is presently unimportant; we see some limited examples of computerized creativity elsewhere in this paper. However, what is interesting is that its learning consists of parsing uniform resource locators (URLs, hence its name) out of text conversation on Internet Relay Chat.

```
kd: url, who am I?
       url: you are probably mailto:godel@cam.org and at
ftp:\\ftp.here.com\bots\kd and at mailto:godel@cs.mcgill.ca or at
http://www.cam.org/~godel/ or at http://www.cam.org/~godel/kd-kiwi.jpg
*** Mode change "-bb *!*befan@* *ribefa*!*@* " on #macintosh by pounder
   Arcanis: because url always interperets "you" as "url", even in urls
     jadin: man
     jadin: i wish it was sunny today
     jadin: i coulda gone to the beach and gotten a tan
     jadin: heh
   Arcanis: I wish I was in school today... not
     jadin: or a burn, in my case
   Arcanis: go eat a hamburger, jadin
kd: hmm, lets see...
     jadin: ew.
kd: bowling balls are at http://www.bowling.com/            *****
           Arcanis offers jadin goose liver paté
kd: url, bowling balls?
     jadin: man. this sucks. this kickass cd ripper doesn't work on this
machine
       url: bowling balls are at http://www.bowling.com/
     jadin: hopefully it works on my cdrom up in orlando
     jadin: heh
   Arcanis: Hello kd's professor
```

---

[19] Warning: I am not saying that this is the only issue in intelligence; far from it, however it is an important one.

[20] Note: I am "kd" in the transcript. "Arcanis" and "jadin" are other humans, and "pounder" is a bot of very different sort than url. Also note that I previously warned the channel that I was going to produce a transcript.

```
   Arcanis: no, url, bowling balls are at http://www.balls.com/
       url: okay, Arcanis.
   Arcanis: url, forget bowling balls
kd: I don't even know if the professor will be grading this, or whether
we have a reader or something
   Arcanis: url, bowling balls?
url: arcanis: wish i knew
   Arcanis: url, tell kd about me
*url* you are at mailto:necrom@magpage.com or at
http://www.magpage.com/~necrom/
       url: okay, Arcanis, i did.
   Arcanis: url, have a chocolate covered botsnack
       url: :)
```

As can be seen by the line in the above transcript with the "******", url can 'listen in' to a conversation and pick up information that is not explicitly directed to it. This is actually a fairly sophisticated process, as it means scanning the whole conversation for URLs of the various sorts (which are in various different forms and which can be indefinitely long), and entering it in the database, all while maintaining a 'look out' for more URLs. Url can also be told to forget what it knows and will have appropriate answers to various other items. The next stage in its development would be to have him combine information and produce more novel sentences. (For instance, "url, is foo at http://www.bar.com/?" "no it is at ftp://ftp.baz.net.")[21]

The lesson to learn from all of this is that url is very capable of dealing with human conversation and extracting useful information out of it, despite the vagueness of what is said. Hence, in a limited way, url learns. Elsewhere in this paper, I will draw a connection between learning and programming in general. (See section III and IV of this paper.)

Assuming the above process (of url's) could be refined, and url could learn about itself by asking itself about itself and what not, and further, could learn about the world instead of just text strings, it would be well on its way to becoming a thinking thing. But some people, notably John Searle, armed with his infamous Chinese Room thought

---

[21] Addendum since the first writing of this paper: The author of url informs me that this feature is under development!

experiment (reprinted with excellent commentary in Hofstadter and Dennett 1981), think that even if that were done, something (in the case of Searle, semantics, in the case of others, 'genuine understanding'.) would be missing. Let us review the thought experiment, and then turn to some classic answers and my answer. This will lead into a general discussion concerning how one attributes the ability to think to something.

Searle's thought experiment concerns the Turing Test, a notorious test of the "ability to think". The Turing Test asks us to imagine ourselves at a computer terminal, conversing with something at the other end of the terminal link for a period of time. If one cannot tell that one is interacting with a computer rather than a human after a sufficiently long period of time, the computer passes the test. Searle purports to show that this test is invalid, by imagining a person hand simulating a program that reads and writes Chinese and passing answers to questions in Chinese out of a room. He tells us that this program would work by 'matching up' inputs to outputs. From this, Searle imagines that we could pass the Turing Test by mechanically following a procedure. But the person in the room doesn't understand Chinese! So it appears that the Turing Test doesn't correctly attribute understanding this time. Or does it?

Many thinkers disagree. The most common answer to this thought experiment is what is called the systems reply. This answer consists in pointing out that nobody would want to attribute understanding to just a part of the system (just as one wouldn't want to attribute understanding to a single neuron), but instead, we attribute understanding to the WHOLE system. Searle's answer to this is to have the person doing the hand simulating of the program internalize the whole 'matching process.'

This is ludicrous (a human being memorizing hundreds of books), and points to the biggest flaw in the thought experiment. At best it describes a situation which couldn't be done at any decent speed by a human being. Furthermore he is asking us imagine a human memorizing what would be literally millions of books. Speed of execution is vital for

consciousness or thinking; after all these two activities both require interaction with the world and hence something possessing the ability to genuinely think must respond to it appropriately. If someone yelled FIRE! in Chinese, and it took more than a split second for the Chinese individual to react and run out of his room, we'd say that he was at least currently (that is, at that time) lacking in intelligence.

There are other reasons for supposing that Searle's thought experiment is incoherent, but I will leave that to the literature. (See Hofstadter and Dennett 1981, Dennett 1987, 1991[22], Fellows 1995[23], and especially Tipler 1994 for plenty of criticism.)

Allow me to use this brief discussion of Searle's thought experiment to move into a discussion of the Turing Test in general. I will deal with three main objections. The first, the notion that the test could fail on an intelligent individual who has nothing to say or is 'scared' by the test; the second, the idea that one could be fooled by "Eliza" type programs. Finally, I will answer charges that the Test is too "behaviouristic."

First, could the test fail on an intelligent individual who for some reason or another is unwilling or unable to respond? Let us look at the unwilling case first. If an individual is never willing to express him or herself, and sits in a corner, curled up in a ball, we might call this person catatonic or some other psychiatric term. In this day and age, we'd presumably look for some sort of brain or nervous system trauma to find a reason for this person's lack of action. Producing some sort of stimulus of the appropriate kind would no doubt produce a reaction. I would think that any individual who has no sensory input at all (in any modality) would scarcely be aware. Helen Keller is

---

[22] This work even includes a note that Searle thought all of Dennett's remarks in *The Intentional Stance* (Dennett 1987) were "irrelevant", but didn't explain why.

[23] Fellows' paper is a demonstration that Searle's thought experiment is logically incoherent. As it relies on some aspects of Chomskian linguistics I do not feel qualified to evaluate, I will not discuss the details of this (apparently) brilliant refutation of Searle.

recorded (quoted in Dennett 1991) as pointing out that before she learned from her teacher, she had effectively no inner life; and Keller did have sensory input in several modalities (touch, taste, smell.)

What if the individual was simply terrified by us interrogating it? If you admit this, then you admit that it is conscious (emotions and (some) consciousness go together, Lieutenant Commander Data of *Star Trek: The Next Generation* not withstanding) and so the Turing Test isn't even needed. Though, I imagine many AIs would WANT to talk; after all, they might be cooped up in a box, and we can walk around.

But what of the "Eliza" effect? Couldn't an interrogator be fooled by a program that had a number of canned responses, and "answered" by asking questions? Well, I have seen people fooled at first glance by the bot, url, discussed previously in this paper. But the more they observe its bevhaviour, the more they realize that something is amiss. The following is a transcript of me "talking with" a very simple program called Julia (Julia is on odd numbered lines, I respond on even - the transcript has had the numbers added to it):

1) I like most animals but I favor cats over dogs. Are you a cat lover?

2) Me? A cat lover? That's what people say. Really, I wouldn't know

3) Oh, I'm so glad.

4) Why are you glad?

5) Do you have any pets? ***

6) I have one pet.

7) So when will Socks write his book? ***

8) When you pass the turing test.

9) What is this, a pop quiz?

10) In a matter of speaking.

11) Do you think that animals can think? ***

12) Probably more than you can.

13) Hardee har har!

14) Was it really that funny?

15) What kinds of pets did you have as a child?

16) well, we had goldfish once, and some gerbils.

17) I have one very old goldfish (his name is Schwarz).

18) How old is Schwarz?

19) I don't know any schwarz. ***1

20) But I thought Schwarz was your very old goldfish.

21) I'd rather have a cat. Cats are my favorite, because you can pet cats, they purr a lot, and they're easy to keep. ***

22) ok. nice talking to you, Julia, good day.

23) Goodbye.

Notice the large numbers of illogical responses (marked with a ***) produced by this program in such a short period. Anyone of these would seriously call into question the hypothesis that one was dealing with a thinking thing. Together, especially with the one marked ***1, clearly show that a brief interaction can demonstrate lack of intelligence. However, one has to be prepared to change one's mind with the Turing Test, in either direction. If a program fools someone ALL the time, and this investigator is demonstrably something close to a reasonable person, does this mean that the Turing Test is too unsophisticated? I will return to this remark when I've tried to make a case for the necessity of this test.

This brings me to the second objection. Namely, that is the Turing Test too "behaviouristic"? I move that, while it treats the subjects as "black boxes" to some degree, the test is actually necessary. Consider the case of someone who has claimed to find parts of the human brain responsible for intelligent activity. Now, this neuroscientist wants to test her hypothesis, and removes these sections one at a time from her patient[24]. How does she show that what she has removed is responsible,

---

[24] Of course, this is only a thought experiment, but only for ethical reasons!

say, for the ability to produce language, or whathaveyou? She has the patient perform certain tasks, including of course, a sort of Turing Test. Without this these appropriate responses, she is just asserting that she has found the relevant parts of the brain. Someone at this stage might say, though, that once we've done this for a few humans, we know what causes intelligence, and the stuff (that is, biochemical structures of a particular kind) isn't what is in a conventional computer. Hence computers aren't or cannot be conscious or aware and AI is a chimera.

When put this way, it is clear the thesis that one needs to look at neuroanatomy to determine if something is conscious or has intelligence begs the question. To make sure we don't prejudge something based on what is made of (and it is still possible that only biochemistry of the right sort is fast enough or is capable of being connected enough, etc.), we use the Turing Test. And how do I know that another person is not a "zombie" (a hypothetical creature who looks conscious on the outside and yet has no "inner life" - see Dennett 1991 for explanations of why the notion of a zombie is actually incoherent.) or an unsophisticated program? I know by talking to and interacting with the person, and so forth. Or, in other words, in some sense, by Turing Testing her!

The above response allows us to answer the worries of those who believe we might be fooled by an Eliza effect. My answer is simply that we wouldn't be fooled after a sufficiently long period of time, and that, as previously stated, we have the freedom to change our mind. Tipler (1994) points out that this sort of procedure is what convinced (most) European males that non-Europeans and women were aware - they could act and behave just as they could, outward differences in appearance notwithstanding. Someone at this stage will point out that the neuroanatomy of all humans is the effectively the same; however it wasn't opening up skulls of slaughtered Africans or others that convinced these European males that they were dealing with equals, but by observing their behaviour such as the capacity to reason.

One can also remember Hofstadter's remark that a well done Turing Test is like an electron scattering experiment. (See Hofstadter 1995.) Both tests reveal internal structures of something by an indirect method. Are those who still think that external testing for intelligence is a bad idea ready to say that the electron scattering experiments are wrongheaded too? Hofstadter points out that a well done, apparently indirect, test does seem to blur the line between indirect testing and direct testing.

Furthermore, Dennett also explains in *Consciousness Explained* (1991) why anything capable of passing the Turing Test would think it was conscious. He calls this "falling prey to a user illusion." It is quite relevant to our discussion, because it does seem to discredit the possibility of something being behaviouristically identical to a human, without having an "inner life" (private experience). In other words, there are no zombies of the sort mentioned previously.

Having dealt with the Turing Test, I would like to move on to a discussion of some biochemistry and biology related issues. First consider the issue of whether or not the human brain is limited in capacity and this issue's purported relevance to AI. Mario Bunge writes (Bunge 1983a):

> "The limited capacity thesis applies to telephone lines, computers, and other artificial information systems, but it has not been proved for humans. There are indications that it does not apply to our brains: (a) unlike artificial information processors, our brains are plastic and, in particular, they have the property of self-organizability; (b) every time we learn something we become better prepared to learn further items: learning is an autocatalytic process, not the filling in of prefabricated bookshelves; (c) with practice we can learn to do two or more tasks at once (...)"

The last, (i.e. point c above) is easiest to deal with. Computers, at least at the lowest level of description, can easily be made to do two things at once; this is the very definition of parallel processing. Taking one step further up the level of description hierarchy, computers can appear to do more than one thing at once. For instancen I can dial into my Internet Service Provider and download a document while typing this paper. This is accomplished by a multitasking system, in which applications being run periodically receive time to 'do their thing'

many times a second. It is interesting to note that it is not at all clear which form of multitasking the human brain actually does when it performs two actions at once. My guess (and it IS just a guess) that there are several multitasking like systems running on several "nodes" (subsystems) of the parallel "hardware." In other words, that BOTH forms of concurrency are used. I also fail to see why doing two things at once (in either fashion) would imply that the limited resource thesis didn't apply; it would just mean that certain resources are capable of being shared. (This point I will come back to when I discuss how computers do not really "manipulate numbers" as is commonly thought.)

Point (a) above, concerns plasticity and self-organizability. While it is true that computer hardware cannot rearrange itself (physically), it is not true that software cannot. Take, for example, the following code:

```
: (define x (vector (lambda (y) y) (lambda (z) (* z z))))
x

: ((vector-ref x 1) 2)
4

: (define swap (lambda (vec)
            (let ((temp (vector-ref vec 0)))
              (begin
                (vector-set! vec 0 (vector-ref vec 1))
                (vector-set! vec 1 temp)))))
swap

: (swap x)

: ((vector-ref x 0) 2)
4
```

This is just a simple example of (parts of) a program rearranging some data structures in memory. Note that the items that are in the data structure are themselves procedures (or, more precisely, lambda expressions.) Point (a) above is thus at least partially false, and it is still not clear how this remark bears on the issue of the finite capacity hypothesis. How would rearranging existing materials change the capacity for storage?

Point (b) is the most difficult in terms of accomplishing on conventional computational devices, but it is still possible. Consider a dynamic storage mechanism using generic pointers[25]. It requires no 'bookshelf' type knowledge; new items just get put into memory. Furthermore this mechanism could be made more sophisticated by tying each item to 'related items' and having the whole structure updatable dynamically. While this is possible on a computer, it may not be relevant to the issue of finiteness. How does the ability to file arbitrary data and make associations between them overcome the finiteness of a brain?

Another argument presented by some people (for example, Bunge 1985) is the following. As an electron in a box can be in an infinite number of states, the Turing Machine formalism is forlorn, for Turing Machines can only be in any one of a finite number of states. For the objection to have relevance against AI, one must clearly show that the number of cognitive states in (say) a human is actually infinite. While indeed it is a very large number, so is the number of states for my friend's programmable calculator with 256 **bytes** of RAM (roughly 10^{310} for the states of the RAM qua RAM alone) and arguably a computer which instantiated an AI program would have 12 orders (i.e. in the terabytes of RAM - see Tipler 1994) of magnitude more RAM. Further, unless the brain states change indefinitely fast (which seems impossible, due to relativity) one might be able to make up for the lack of RAM states simply by rapid processing.

Finally, consider the difference between the brain of a live animal and that of a dead one? While eventually, once rotting sets in, there will be substantial biochemical differences, this doesn't occur right away. What happens immediately happens is a change in the pattern of activation of brain states. In other words, the previous consideration seems to indicate that other large scale properties of the brain are sort of "systems properties," and indeed functional properties

[25] In other words, something using void * datatypes in C, for instance.

(either emergent or aggregate) and so we should look at systems properties of computers too.

As for whether the brain really is finite, this issue will be returned to in the discussion of state machines in Section IV of this paper.

There are various other biology related issues involved in the AI controversy; onr popular one concerns the issue of "perhaps only brains have the right stuff to think." There are several ways in which this objection is phrased. One makes an analogy with a simulation of a stomach not being able to process nutrients, hence a simulation of a brain wouldn't really think. While it is immediately clear to me that simulated stomachs don't digest, (a computer scientist would say they don't receive the right input) it is not so obvious that this analogy holds to the case of brains. After all, what exactly is it that this simulation of the brain cannot do? (Computer scientists would say that it likely receives the right input.) We can attach the computer to something (a terminal, or a speech synthesizer, etc.) in order that it may communicate with us and it will act as if it thinks to a certain degree. But what is missing? (Keeping in mind that if it is capable of passing the Turing Test, it will think it is conscious.) Assuming that it has a perceptual system[26] and that (as above) it can communicate with us, several common answers come to mind.

The three most common answers are have intentionality, possess qualia, possess emotions. To start, consider have intentionality (John Searle's objection.) What does it mean for something to have intentionality? Searle alleges that it is property that allows brains to (for instance) process mental inputs (for example, speech) without the alleged consequences of his Chinese Room thought experiment. Since we have seen how the Chinese Room thought experiment is grossly

[26] I am willing to guess that an AI really worthy of the term would have some form of perception, though perhaps into a "virtual space", if such a space could be described very richly. See Hofstadter 1979 p. 586-593 for a primitive, but effective, example of this notion.

mistaken, this tends to discredit his notion of intentionality. However, suppose one is of the opinion that only assemblies of neurons can have this mysterious property called intentionality. As clearly one neuron doesn't possess intentionality[27], then a threshold number of neurons must be involved. But what about the neurons allows them to have this property? Why can silicon chips wired together at this level of complexity not support the same intentionality (or alternatively, software modules strung together by interapplication communication in some form such as events, threads, semaphores, etc.)? This is pointed out by a thought experiment that involves slowly replacing Searle's neurons by some other "stuff" that preserves the "input" and "output" as well as internal states of each cell. Searle must conclude that he would keep on ACTING as he did before, but that his words would lose their meaning. This is supposed to show that meaning is definitely associated with the pattern of activation, and not with the qualities of individual neurons. In other words, meaning, and hence other things like the ability to lie (see Bunge 1985 for a huge and well thought out list) are emergent properties. It would be very strange if one had to microreduce so much that organic chemistry was necessary for an explanation of meaning. (Since the jury is still out on how exactly biochemistry produces semanticity through emergence, simply to assert that conventional computers haven't got the "right stuff" begs the question against the AI researcher.)

But could an supposedly AI system have qualia? Qualia, also known as sensa, raw feels, the basic data of experience, secondary qualities (a term from Locke) and so on, are the bugaboo of many a materialist philosophy. However, if one is ready to admit that qualia are in fact somehow material in origin[28], then one must propose there is some sort

---

[27] After all, one neuron wouldn't even be able to control human autonomous functioning, never mind something like intelligence or consciousness.

[28] Dennett 1991 contains what I consider to be one of the best explanations yet; however this is not terribly relevant to the discussion in this paper. All one needs is to concede that qualia are material in origin. Dennett also points out that there really aren't such things as qualia, they are just a *façon de parler* - but this need not concern us here.

of (presumably) neurophysiological mechanism about how they arise. Now, unless one can somehow show that qualia only come out of biochemical processes, AI is still possible in principle, because one could conceivably use the same sort of organizational characteristics in the machine. (Later, when learning is briefly discussed, what this means in greater detail will be mentionned.)

It is most difficult for most people to imagine how a computer could have emotion. But what is an emotion? Emotion is defined as "A response of the whole organism, involving (1) physiological arousal, (2) expressive behaviors, and (3) conscious experience." (Myers 1996, pg. 335) Take the first, physiological arousal. For those who picture computers as being a hunk of plastic, and metal (etc.) on their desk, this may be difficult to comprehend. However, existing computers already automatically respond to changes in their environment, albeit in a very primitive way compared to organisms. For instance, Apple's line of PowerBook Duos automatically "know" whether they have been started up alone, or in the "docking station." Many more examples can be given. In computerized data acquisition (for example, Natural Intelligence's Labview) the computer responds to external changes in state and further, to its own response. This is certainly analogous to changes in physiology. While it may be objected that the basic underlying hardware doesn't change, and that emotion (for instance, fear) is produced by chemical changes (like amounts of adrenaline). However, the over-all anatomic structure of the organism does not change. What does change is internal state and behaviour. Again, however, I stress that AI is not committed to a neuron to chip correspondence. The neuronal behaviour may be created at higher levels of implementation (i.e., in some level of software.) But does that mean the computer will feel? I argue yes, for the reasons discussed above under the issue of "qualia."

Point (2) above had to do with expressive behaviours. Again, computers do react to outside stimuli; for instance, entering data on a keyboard, drawing with a mouse, or responding to voice commands ("computer, tell me a joke") are all possible with today's computers. The computer reorganizes its internal states based on the input.

Point (3), namely conscious experience, in the definition of emotion above is the very issue of this paper. One cannot argue that "AI is impossible because computers cannot have conscious experience." because that is the very point of contention! (See page 1 of this paper for the definition of AI.)

More biology related issues raised against AI include the issue of representation, notably computerized representation of time. This objection also is often expressed in terms of Turing Machines not representing time and space. There are of course two ways anything gets represented in a Turing Machine. One is by placing the representation in the state table (i.e. placing it in some sort of built in program.) The other is by feeding it in as part of the input (i.e. from the environment)[29]. Of course, these approaches are not mutually exclusive. One can see that the first way corresponds to a sort of innately given ability, and the other fashion corresponds to a outside influence. It seems that biological creatures rely on both notions (after all, we know that naive empiricism and extreme innatism are both false); however which process is actually used by them is not terribly relevant. After all we can easily do either or both for our AI program. Suppose we set up our computer to play a chime in order to wake us up, in other words, as an alarm clock. In what sense, then, is the computer not representing time? One objection might be that the computer has no knowledge of time, and is just matching inputs. The second is that the computer doesn't react to the passage of time any way, just at discrete instances takes action. Both objections while true are irrelevant. While current computers just match inputs with regards to time, and have no understanding, to say that they never will possess deeper knowledge is the whole issue of AI. To say that AI is not possible because computers

---

[29] Also keep in mind, because most AI programs are expected to be very "layered" in design, it is also quite possible to have a computer program change its own state "table". The degree and the depth to which this is possible is suggested by Hofstadter (1979) as a measure of possible intelligence of the thing in question. Humans are more flexible than flies because we can modify ourselves more.

cannot understand time at a deep level, then, is a circular argument! Again one would have to show that there is something intrinsic that is not representable.

The allegation that computers do not react to the passage of time can be answered in several different ways. One is to point out that in some sense, humans create also continuity out of discreteness; we are, after all, available to watch television or a movie despite the fact that it contains many discrete frames (and not continuous motion[30]).

Another way to respond is to notice that there are computer programs and technologies that are specially designed to deal with the passage of time. The most famous of these is Apple's QuickTime. Often used as a base for multimedia technology (and this is indeed how it is marketed), QuickTime is actually more general in usefulness than that. For instance, QuickTime at base is about time tracks; one can set up a time track base for any sort of periodic activity. Periodic in this context can be irregular (e.g. after 1 ms; 4 ms, 16 ms, 256 ms, etc.). While QuickTime is only to sensitive 1 millisecond increments of time, this is not a problem, because in principle one could have a technology more sensitive. Further humans are not such that their "tick of the internal clock" is infinitesimal. Of course, the lesson to be learned here is that QuickTime can actually look back, when it receives some CPU time and adjust for "time lost", so to speak. For instance, if playing a sound sample was calculated to take 2321 ms, and it took 2349 ms (either due to calculation round off, or external events that interrupted the time base, etc.), the accompanying video track could be adjusted accordingly so that the two are kept in synchronization. Hence computers in some sense do respond to passages of time, not discrete influences, and in very sophisticated ways. (See Inside Macintosh: QuickTime. http://devworld.apple.com/ has online versions as well as ordering of paperbacks) for more information for the programmer or philosopher of technology that can program.)

---

[30] Some physicists point out that it appears that motion itself is discrete, not continuous, at least in some ways of looking at quantum mechanics. For examples and discussion of this, see Stenger 1995, particularly chapter 7.

Section III: Mathematics and AI:

In this section I will discuss two families of arguments from the domain of mathematics purporting to show the impossibility of AI. The first is a common one, concerning computers and representations of numbers. The second are those of Lucas and Penrose, which rely on a use of Gödel's Incompleteness Theorems.

It is often wrongly claimed that humans don't have to deal with approximations to real numbers, or that storage for numbers isn't limited, the way electronic computers are (apparently) so limited. If this were true, mathematics software packages like MacSYMA that for instance symbolically integrate better than the vast majority of trained mathematicians (See Dennett 1987) would not be possible. The above argument stems from the fact that it appears that all computers deal with are fixed sized integers. This is not the case. As pointed out in *68000 Family Assembly Language* (Clements 1994), computers are fundamentally at the hardware level a collection of either/or switches, or, even lower, a pattern of voltages[31]. Clements writes on page 2 (underlining added):

> "An n-bit word can be arranged into 2^n unique bit patterns and may represent many things, because there is no intrinsic meaning associated with a pattern associated of 1s and 0s. For example, the 8 bit values 11001010 and 00001101 do not mean anything. The actual meaning of a particular pattern[32] is the meaning given to it by the programmer."

---

[31] Finding what level one wants to describe a computer's operation at, is, I think, the whole source of the problem of AI. Hofstadter (1979) asks his readers whether computers are super flexible or super rigid. His following point, and mine in this paper, is to convince our readers that computers are both, depending on how one looks at what is going on.

[32] One slight oversight on Clements' part is the consideration that the same bit string can be simultaneously used in two different ways, for instance, both as an instruction and as a data item, or as several different data items, etc. Exploiting this possibility is usually considered rather poor programming practice for any number of reasons, but natural selection is blind, and hence the counter part to this possibility certainly exists in the nervous system. In fact, most parallel-distributed-processing models of memory (see Medin and Ross 1996, particularly chapter 9, for details) rely on this.

The confusion over computer representations of data items stems from the "1" and "0" used to represent the two states of the instantiation of a bit. If they had been called, say, α and β, the confusion likely wouldn't have arisen. This problem is historically rooted as the first (electronic) computers were used exclusively for numerical calculation.

In the case of the human nervous system, the "programmer" would presumably be natural selection and epiphenomenal emergence[33] as well as environmental stimuli. Nor is 'plasticity' is not possible, for there are higher levels of description. For instance, it is often claimed that with computers representation of real numbers is fixed to a certain fixed size. This is true, however, it is not at all apparent that humans aren't limited in the same way. Research in cognition suggests that most humans can represent "in the mind's eye" at most 9 digits or so (see Medin and Ross 1996 for a good introduction to the limitations of human 'number processing'.)[34]. Compare that with a program in the language Scheme, below. Unlike some languages (for instance, C), Scheme has arbitrarily sized integers.

: (define factorial (lambda (n) (if (= n 0) 1 (* n (factorial (- n 1))))))
factorial

: (factorial 100)
933262154439441526816992388562667004907159682643816214685929638952175999
932299156089414639761565182862536979208272237582511852109168640000000000
00000000000000

Finally, it surely cannot be claimed that humans deal with all the digits of (say) π at once; they can deal with possibly very large numbers of them, and crank out more and more digits by using a procedure for doing so. This procedure, on a computer, not only can be accomplished, it can even run in the background of another process, and

---

[33] For a very interesting look at the view that some aspects of intelligence are 'epiphenomenal', see Dennett 1991.

[34] Using a paper and a pencil here cannot help those who would want to claim that the human capacity is potentially infinite; after all, why do we use paper and pencil? To make up for the limited capacity of our working memory. I discuss long term memory very briefly elsewhere.

so when the program requires more digits of π, it can request them. This process, somewhat similar to a mathematician who can keep cranking out more places of π as he continues doing other things, is illustrated somewhat simply below. (I borrow most of this example from the MacGambit documentation.)

```
: (define (series term)        ;;; Concurrency is expressed with FUTUREs
   (let ((sum 0) (stop #f))
    (FUTURE (let loop ((i 0))
              (if (not stop)
               (begin (set! sum (+ sum (term i))) (loop (+ i 1))))))
    (lambda (msg)
     (cond ((eq? msg 'value) sum)
           ((eq? msg 'stop)  (set! stop #t))
           (else           (error "unknown message" msg))))))
series

: (define pi  ;;; start a task to compute series expansion for pi
   (series (lambda (i) (/ 4. ((if (odd? i) - +) (+ (* i 2) 1))))))
pi

: (pi 'value)   ;;; get current value of series
3.141419882340216

: (cons 'a 'b)   ;;; do some other operation and the series calculation continues
(a . b)

: (pi 'value)   ;;; again... it has changed!
3.1415194471477133

: (pi 'value)
3.1416300745380195
```

So, computers can deal with arbitrary numbers, mathematical symbols and with indefinite approximations to series at least as well as humans, at least in some domains. Whether they can do so in all domains runs quickly up into questions of (formal) incompleteness, which I will discuss next.

The final anti-AI arguments that I will deal with in this paper are two, one by Lucas and another by Penrose (see Penrose 1989, 1994, 1997) which purport to use Gödel's Incompleteness Theorems. But first, I will deal with Lucas' simplistic version of what amounts to the same argument. Lucas (1961) writes:

> "However complicated a machine we construct, it will, if it is a machine, it will correspond to

> a formal system, which in turn will be liable to the Gödel procedure for finding a formula unprovable-in-that-system. This formula the machine will be unable to produce as being true, although a mind can see it is true. And so the machine will still not be an adequate model of the mind. We are trying to produce a model of the mind which is mechanical - which is essentially "dead" - but the mind, being in fact "alive", can always go one better than any formal, ossified, dead system can. Thanks to Gödel's Theorem, the mind always has the last word."

Hofstadter (1979) points out that this argument is very compelling indeed. It (along with Roger Penrose's version) is one of the most seductive anti-AI arguments as far as I am concerned. But what are its flaws? There are several, in fact. The first one I noticed is that the Gödel sentence for a system (call it G) is, as Hofstadter pointed out, a sort of analog to the liar's paradox, inside a formal system. If that was correct, and some sort of "translation" of this sentence back into English of G was "I am not a theorem of formal system F", then a sentence of the form "Lucas cannot consistently believe this sentence" seems to capture the appropriate intuition. This of course is much like the Whitely sentence: "Lucas cannot consistently assert this sentence." These, and many others, purport to show that incompleteness is a fact of life for humans, too. (See Smuyllan 1987 for a formal development of this idea.)

One must also ask of Lucas' argument if one can REALLY apply the Gödellian procedure in every case. It seems that after a point, one simply could not, due to one's own brain limitations. (Much as we cannot memorize the Montreal telephone directory!) Lucas' argument also doesn't consider what would stop a computer from Gödelizing itself. It would conclude, much as we would, that there are statements that are true, but it (i.e. the computer) can't prove, by virtue of its (i.e. the computer's) nature. Further, there comes a stage where we just conclude (or not.) Both brain and computer are ultimately fixed by physical law, humans are simply not infinitely creative.

Now I will turn my attention to Roger Penrose's version of the argument which differs only slightly, mainly in terms of his conclusions from it. (He claims that a new revolution in physics will be necessary to understand the brain's functioning.) But his premises are slightly different from the Lucas version. For instance, he explicitly states

that mathematical platonism is an "reason" for his position, as when he says (see Penrose 1997):

> "Somehow, the natural numbers are already 'there', existing somewhere in the Platonic world and we have access to that world through our ability to be aware of things. If we were mindless computers, we would not have that ability."

Secondly, Penrose seems to take it as a given that mathematicians, or perhaps the mathematical community as whole, are sound reasoners. He discusses this at length in Penrose 1994. However, this doesn't seem at all plausible. As pointed out in McDermott 1995, there are several key problems with this idea. One is that it is possible to show that any reasoner who claims its own consistency is not consistent. (This is pointed out in Smullyan 1987, two years before the first of Penrose's books on the subject.) Penrose also seems to think that AI is committed to the idea of building perfect reasoners. This of course is nonsense, for among other reasons the "belief incompleteness" discussed above. McDermott (1995) also points out that:

> "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*"

The above insight allows another approach to the issue of creativity, discussed previously in this paper. There's a fundamental distinction between the so to speak "behaviour" of a computer, and any sort of formal system that may or may not underlie it. For instance, when I use a computer to compose music, lots of aspects of the computer's underlying process are <u>irrelevant</u>; all that matters is a certain program with a certain user interface (etc.) is available. This confusion over the various levels of description of what a computer is doing has been noted repeatedly by Hofstadter, and mentioned several times in this paper. One (nonmathematical) example of such confusion over levels is found in Bunge 1985, pg. 268 where Bunge writes:

> "To an educated Englishman, the word 'Shakespeare' is likely to evoke a rich and highly personal cluster of cognitive and affective items; to a computer the same word is just one more physical process in a chip, perhaps on the same footing as 'William'."

This strikes me as being like saying that "brains can't possibly understand anything, because all understanding in a brain could be is one more physical process". One must remember that there is a division

between what something is, and what processes go on in it. This does not commit one to idealism, because processes are never actually disembodied; they are still processes of certain things. This is much like the earlier example of pressure. Just because pressure applies to many things, it does not mean that there "pressure" exists independently, in some platonic heaven. The notion that words (concepts) have rich associational structure is not a problem for computer implementation. Hofstadter's books (1979, 1981, 1985, 1995) are full of discussions of such things. Yes, at some level, "Shakespeare" is 'just' in a chip. But it is also 'just' in a group of neurons.

Most of the objections to Lucas also apply to Penrose, but people recently have focused more on Penrose's argument producing some interesting reactions. One is the point that AI has never really been about algorithms anyway, and hence the Gödelian construction does not apply[35]. It has been about heuristics, in general. Programming computers to perform heuristics is that not difficult; after all, MacChess uses them when it searches to infinite depth and must decide when to stop and move, or when Bertrand (a symbolic logic program) "guesses" that an infinite truth-tree is going to be produced. This is pointed out by Dennett in Hardesty 1995 as follows:

> "DD: The glaring problem in Penrose is simply that he attacks a doctrine of artificial intelligence that has never been held by artificial intelligence. For AI, we've always been looking for so-called heuristic programs for intelligence, and those are simply not covered by Gödel's theorem at all, so the criticism is just irrelevant. And I thought this had been pretty well realized by everybody in the field for twenty years, but somehow Roger didn't pick up on it."

The other approach (and really a specific case of the previous one) is concerns randomness. This staes that randomness can be used as one of the ingredients in creating novelty. Taner Edis writes about this (see Edis 1997):

> "This something, however, is quite unmystical: a touch of randomness turns out to be all we need to keep us---and machines, for that matter--- away from rule-bound blind

---

[35] Yes, it is still possible to produce a Gödelian construction for the system produced by the computer and the heuristics, however, the point is that this would be irrelevant to its usage (see McDermott's quote, above) as a general system or set of systems. This of course is parallel to the general "incompleteness of beliefs" examples I discussed previously.

> spots.[56] At first this seems crazy. After all, randomness is the only thing more mindless than rigid rules. Platonists at heart, we think the essence of Reason is a lawful, transcendent order. In contrast, randomness is a total lack of pattern, as in a sequence of coinflips. Such chaos can only corrupt Reason. But if we ignore Plato for a moment, we will find randomness is rather interesting. A patternless function is maximally nonalgorithmic. It is a nonalgorithmic function which is meaningless, useless for everything but keeping us from following preset rules all the time. So if we want a nonalgorithmic intelligence, distinguished not by a magical knowledge of functions like Turing's h but by its ability to jump outside of any system of rules, randomness is just what we need. A nonalgorithmic machine uses randomness as a device to introduce novelty, a way to break out of ruts."

Edis' insight can be used in AI by noting that by using a nonalgorithmic function in a computer program, one can create behaviour of a computer that is nonalgorithmic[36] and that creates true novelty. Numbo, noted earlier in this paper is an early precursor to this idea. Together with heuristics, the Gödelian snare can be avoided, at least in terms of formal mathematics. (The success of other programs (see Hofstadter 1995) using this probabilistic notion suggests that this technique applies in other domains.)

Section IV: A Few Positive Arguments For AI:

This section will discuss how it appears the brain appears computational in several important respects. It will employ a definition of a computer program as well as some psychological and neurophysiological considerations and also concern the notion of learning. We will see furthermore that a brute force sort of argument also appears to work to some degree and will involve the concept of the Bekenstein bound.

Learning can help to provide a positive argument for AI. Learning is defined in Myers 1996 (page 195) as:

> "A relatively permanent change in an organism's behaviour due to experience."

As we know, the brain of an organism must change in order that its behaviour is able to change. When an organism learns, parts of the brain transform themselves (in ways that are still somewhat poorly understood)

---

[36] After all, the procedures used need not ever terminate. This is the very nature of heuristics.

and set up new networks and associations of neurons. (Bunge (1983a, 1983b) calls this the formation of new psychons. This term is useful - I will adopt it in the discussion that follows.) Psychons are formed (in part) by interaction with the environment. Similarly, in a computer, new data structures are assembled partially based on interaction with the environment. As these data structures may contain procedures to do new things, some sort of learning is possible. Computerized psychons do not involve changes in the hardware (though in principle that could be done), but rather reorganizations of software. For instance, the following program takes an input from the user and produces a new function that multiplies any arguments it receives from the user by the amount that was specified, by the first input. This also shows that procedures are data structures - a primitive form of self modification, and interactive learning.

```
: (define multiplier
    (lambda ()
      (let ((x (read)))
        (lambda (n)
          (* n x)))))
multiplier

: (define times10 (multiplier))
10      ;;; my input!
times10

: (times10 10)
100
```

There are two lessons to be gleaned from this example. Firstly, both computers and brains learn by changing the pattern of transitions between states and by assembling new structures. It may be objected that brains are able to construct new neurons, at least in early childhood and that a computer cannot construct new hardware for itself. However, this can be likened to a computer program which hasn't yet filled its memory space with data structures. The other connection between the two relates to "housekeeping". In order that memory doesn't get overly disorganized, it is thought that our brain periodically "cleans house" - a possible reason given for our dreaming. Computers do something analogous when a program performs what is known as a garbage collection. (I am not saying, however, that garbage collections correspond to

dreams, only that they appear to have similar uses.) Secondly, with a realization that all a computer program[37] is a set of regular[38] (including stochastic) transitions between states (see Dennett 1987), we can begin to see the intuition behind the "mind as computer program" idea and hence the possibility of computer program as mind. Since this paper is primarily a response to critics, I will leave it at that - in future, I may develop the positive account more. But before I conclude the paper, I will discuss one more positive argument for AI.

Finally, Frank Tipler in his Physics of Immortality (Tipler 1994) presents a very strange positive argument for AI. While questioning his book, I have yet to find a decent counter argument to his, which concerns phase space and state machines. As Tipler reminds us, phase space, hence entropy, hence number of available distinct states (see Benson 1991, pg. 425-426) for a system of radius R and total relativistic energy E (i.e., in usual symbols $E^2 = p^2c^2 + m_0^2c^4$) is bounded by the Bekenstein Bound, which is given by the following expression:

$$I \leq 2\pi ER/(\mathrm{h}c\ln 2)$$

Since the maximal information contained in the brain is finite in this sense, it would be implementable on conventional hardware. (One

---

[37] A computer program is not a disk or tape, though it may be instantiated on one. If it were, duplicating the contents could hardly be unlawful in some cases.

[38] If it is argued that the transitions between states of the human nervous system is not regular, then my interlocutor has claimed that the brain violates the principle of lawfulness. As remarked earlier, this goes against the basic postulates of scientific and technological research (Bunge 1977) so I will ignore it.

If it is objected that the current state does not uniquely the future state with a given input in the nervous system, it can be argued that we don't know this. There are simply so many possible brain states and inputs (i.e. environmental stimuli) that this seems impossible to verify, especially considering the brain has elabourate self-monitoring systems.

cannot, so to speak, compress an actual[39] infinite amount of information into a finite space.) Further, since state transitions must occur at a finite speed because of the speed limit of the universe, the speed of light. The human brain (and every finite subaggregate of the universe as whole, for that matter), at the ontologically lowest level, appears to be a finite state machine. If this is the case, a "brute force approach" to creating an AI, in principle will succeed. It should also be noted that due to speed of processing issues, doing this might also require simulation of an environment. While this lessens the plausibility of this method, it does not affect its conceptual possibility.

Section IV: Conclusion

I shall conclude this paper by stating a banal almost truism, then. By building machines, we do learn about ourselves. Construction of AIs is not therefore purely an academic game. For instance, we learn how we are NOT constructed. Deep Blue plays fantastic chess, but doesn't think about it. I look forward to the day when Deep Blue version N refuses to continue playing in disgust, or says to its human opponent out of the blue (pun intended) "nyah, nyah!" That day will not likely come in my lifetime - but that should not stop us from dreaming, building, testing and building again.

[39] As we have seen previously, the idea of a potential infinite is trivial to implement on a computer. Even the old BASIC program, "10 PRINT "!" : GOTO 10" produces a potentially infinite amount of output, and furthermore will not be limited in its output by the nature of its software (assuming that the BASIC interpreter isn't implemented in a horribly poor fashion).

References:

Benson, H. 1991. *University Physics*. John Wiley and Sons: New York.

Bunge, M. 1977. *Treatise on Basic Philosophy,* Vol. 3: *The Furniture of the World.* Dordrecht: Rediel.

Bunge, M. 1983a. *Treatise on Basic Philosophy*, Vol. 5: *Exploring the World.* Dordrecht: Rediel.

Bunge, M. 1983b. *Treatise on Basic Philosophy*, Vol. 6: *Understanding the World.* Dordrecht: Rediel.

Bunge, M. 1985. *Treatise on Basic Philosophy*, Vol. 7, *Philosophy of Science and Technology*, Part II: Life Science, Social Science and Technology. Dordrecht: Rediel.

Chapman, W.A. 1991. *Mastering C Programming*. Macmillan: Houndmills.

Clements, A. 1994. *68000 Family Assembly Languages*. Boston: PWS Publishing Company.

Dennett, D.C. 1987. *The Intentional Stance*. Cambridge (MA): The MIT Press.

Dennett, D.C. 1991. *Consciousness Explained.* USA: Little, Brown & Company.

Downing D., and Covington M. 1991 *Dictionary of Computer Terms* (3e). Hauppauge: Barron's Educational Series, Inc.

Edis, T. 1997. *Is There Anybody Out There? The Fate of God in an Accidental World*.
http://www.public.iastate.edu/~edis/archives/edis-book/8.text

Fellows, R. *Searle on the Computational Theory of Mind*, reprinted in *Philosophy and Technology* (Fellows, R. ed). Cambridge: Cambridge University Press.

Friedman, D.P., Wand, M., and Haynes, C.P. 1996. *Essentials of Programming Languages* (7th printing). Cambridge (MA): The MIT Press.

Hardesty, L. 1995. *Daniel Dennett: How Skyhooks Hoist Only Their Own Petards.*
http://www.bookwire.com/BBR/BBRInterviews.article$1412

Hofstadter, D. 1979. *Gödel, Escher, Bach: An Eternal Golden Braid.* New York: Basic Books.

Hofstadter, D. and Dennett, D., eds. 1981. *The Mind's I: Fantasies and Reflections on Self and Soul*. New York: Basic Books.

Hofstadter, D. 1985. *Metamagical Themas: Questing for the Essence of Mind and Pattern*. New York: Basic Books.

Hofstadter, D. *et. al.* 1995. *Fluid Concepts and Creative Analogies*. New York: Basic Books.

Lenzo, K. 1996. *Infobots: in4m, url, and hocus*.
http://www.cs.cmu.edu/afs/cs/user/lenzo/html/hocus.html

Lucas, J. 1961. *Minds, Machines and Gödel*. Quoted in Hofstadter 1979.

Machover, M. 1996. *Set theory, logic and their limitations*.
Cambridge: Cambridge University Press.

McDermott, D. 1995. *Penrose is Wrong*
ftp://ftp.cs.yale.edu/pub/mcdermott/papers/penrose.txt

Medin, D., and Ross, B. 1996 *Cognitive Psychology* (2e).
Orlando: Harcourt Brace College Publishers.

Myers, D. 1996. *Exploring Psychology* (3e). New York: Worth.

Penrose, R. 1989. *The Emperor's New Mind: Concerning Computers, Minds and the Laws of Physics.* New York: Oxford University Press.

Penrose, R. 1994. *Shadows of the Mind.* New York:
New York: Oxford University Press.

Penrose, R. 1997. *The Large, The Small and the Human Mind.*
New York: Cambridge University Press.

Smullyan, R. 1987. *Forever Undecided: A Puzzle Guide to Gödel.*
New York: Knopf.

Stenger, V. 1995. *The Unconscious Quantum* : *Metaphysics in Modern Physics and Cosmology*. Amherst: Prometheus Books.

Tipler, F. 1994. *Physics of Immortality*. New York: Doubleday.