This preprint differs from the published version.

Do not quote or photocopy.

The Curious Case of the Chinese Gym

B. J. Copeland

ABSTRACT

Searle has recently used two adaptations of his Chinese room argument in an attack on connectionism. I show that these new forms of the argument are fallacious. First I give an exposition and rebuttal of the original Chinese room argument, and a brief introduction to the essentials of connectionism.

Searle launched his now famous Chinese room argument in 1980.[1] His target was traditional program-writing, symbol-crunching AI. Since that time the connectionist revolution has taken place, leaving AI and cognitive science considerably altered. Searle has recently used two adaptations of the Chinese room argument in an attack on connectionism.[2] I shall show that these new forms of the argument are fallacious. First, I will give an exposition and rebuttal of the original Chinese room argument, and a brief introduction to the essentials of connectionism.

The Chinese Room Argument

Consider a hypothetical AI program that responds intelligently, in written Chinese, to an input of Chinese sentences. Suppose the program performs indistinguishably from a native Chinese speaker under all input conditions. Searle's argument is designed to show that, performance notwithstanding, the program cannot actually understand Chinese. (In his 1980 paper, Searle's stalking horse was Roger Schank's SAM, a program sometimes described - inappropriately- as being able to understand simple English stories.[3])

The hero of the proceedings, call him Joe Soap, is locked in a room with a hard copy of the program. Joe understands no Chinese. His job is to 'handwork' the program - to carry out by hand all the bit manipulations that are performed by a computer running the program. (Conversion between the Chinese input/output and bit code is effected via a lookup table that pairs Chinese characters with, say, Pinyin ASCII.) Joe's only contact with the outside world is via a couple of slots in the

wall, labelled Input and Output. The experimenters push in a story followed by a sheet of questions (all in Chinese characters, of course) and then cluster eagerly around the Output slot to await results (imagine that Joe produces answers in minutes rather than months). To the experimenters, the symbols that Joe pushes through the Output slot are intelligent answers to their questions, but to Joe they are just so many squiggles, hard-won but perfectly meaningless.

Now for Searle's argument. In handworking the program, Joe has done everything done by a computer running the program; in effect the experimenters have run the program on a human computer. But running the program doesn't enable Joe to understand Chinese. It follows that running the program doesn't enable a computer to understand Chinese.

Since no mention is made of any specific features of the program, the argument generalises to all AI programs that there will ever be (and, indeed, to all forms of cognition).[4]

The Fallacy in the Argument

In Searle's telling of the tale of the Chinese room, as in my retelling, a crucial participant in the events receives far less attention than it deserves. Told fairly, the tale contains two principal characters - Joe Soap, the tireless labourer, and the System, whose exotic conversation emanates from the Output slot. The climax of Searle's tale comes when Joe is asked whether all his symbol-manipulating has enabled him to understand the input questions, and he (of course) says No. From this we are supposed to conclude that these symbol-

manipulations cannot be sufficient to produce understanding. But why ask Joe? He is, after all, nothing more than a cog in the machinery. What we want to know is whether the *System* understands. (If we ask it, it will naturally assure us that it does indeed understand Chinese. But ex hypothesi we are refusing to take its verbal output at face value.) Searle's argument in fact consists of an inference from a premiss about the man in the room to a conclusion about the System.

Premiss No amount of symbol-manipulation on Joe's part enables Joe to understand the Chinese input.

Conclusion No amount of symbol-manipulation on Joe's part enables the System to understand the Chinese input.

Presented in this stark form, the Chinese room argument is plainly invalid.This burlesque of it has just the same form:

Premiss Bill the cleaner has never sold pyjamas to Korea.

Conclusion The company for which Bill works has never sold pyjamas to Korea.

This response to the argument is superficially similar to a response dubbed by Searle the 'Systems Reply'. This is the claim that '[w]hile it is true that the individual person who is locked in the room does not understand the story, the fact is that he is merely part of a whole system, and the system does understand the story'.[5] As Searle correctly remarks, the Systems Reply begs the question.[6] My own point is simply that the Chinese room argument is invalid; and this does not, of course, involve me in claiming that the System understands. (Indeed, where the program in question is Schank's SAM, I happen to think that the conclusion of the argument is true.)

What Searle needs to render his argument valid are some bridging premisses. Two suitable premisses emerge in the course of his discussion of the Systems Reply. He says:

> My response to the systems theory is quite simple: Let the individual ... memoriz[e] the rules in the ledger and the data banks of Chinese symbols, and [do] all the calculations in his head. The individual then incorporates the entire system. ... We can even get rid of the room and suppose he works outdoors. All the same, he understands nothing of the Chinese, and a fortiori neither does the system, because there isn't anything in the system that isn't in him. If he doesn't understand, then there is no way the system could understand, because the system is just a part of him.[7]

The Chinese room argument becomes valid, then, if it includes the premisses:

(a) The system is part of Joe.

(b) If Joe [in general, X] cannot understand Chinese [in general, cannot Ø] then no part of Joe can understand Chinese [can Ø].

I will refer to the generalised form of (b) as Searle's 'Part-Of' principle.

Searle makes no mention of why he thinks the Part-Of principle is true. Yet the principle is certainly not self-evident. Indeed, the principle is easy to counterexample. X, let us imagine, has been kidnapped by a group of fanatical AI researchers. This group believes that the best way to achieve AI's ultimate goal of superhuman intelligence is to run 'neuronic programs' on human brain tissue. Official backing for their project has not been forthcoming and they have resorted to

clandestine methods, with the result that X now lies strapped to a surgical couch in a cellar beneath the AI lab. He gazes apprehensively at the web of wire connecting his shaven skull to a keyboard and visual display unit. Without removing or damaging any tissue the team have imprinted their 'neuronic program' on a small area of cortex (thanks to the vast amount of redundancy in the cortex they have been able to do this without any impairment of X's faculties). The trial 'program' that the team have devised for the experiment is one designed to prove theorems of tense logic. It works very successfully, and X stares uncomprehendingly at the input and output as they are displayed on the screen. X can't prove the formulae which the experimenters enter as input (to him they are just meaningless symbols); but a part of X can.

Could Searle insist on the truth of the Part-Of principle and say that since a part of X can now prove theorems of tense logic it follows that X can now prove theorems of tense logic? Presumably not. If X's denial that he can do the proofs were to count for nothing, the same would have to go for Joe when he operates the memorised program. It is a cornerstone of Searle's argument that Joe's saying 'I do not understand these symbols' is acid proof that handworking the program is insufficient to give Joe the ability to understand Chinese. (One might call this Searle's Incorrigibility Thesis. It, like the Part-Of principle, is left completely unsupported by Searle.[8])

Could Searle try searching for an argument to show that, although the Part-Of principle is false in general, it is true for predicates of a particular type ('understand' being of this type)? This strategy does not seem at all promising. The general principle to be extracted from the

foregoing counterexample is that the principle is false in cases where some subsystem of X can Ø - for any predicate Ø - and the subsystem's output is caused to pass directly into the outside world. If our imaginary research team could somehow induce X's liver to emulate a brain, the liver remaining in situ, and input/output passing from/to a suitable array of transducers in the fashion of the previous scenario, then the Part-Of principle would have counterinstances aplenty, involving every form of cognition. (Searle has no reservations concerning the application of predicates like 'understand' to sub-personal systems. He writes (against Dennett):

> I find nothing at all odd about saying that my brain understands English. ... I find [the contrary] claim as implausible as insisting 'I digest pizza; my stomach and digestive tract don't'.[9])

To summarise my objections: the original version of the Chinese room argument is invalid, and the second, more complicated version is valid but has a thoroughly unacceptable premiss. Searle has recently complained - and not without justice - that the stock responses to his 1980 paper 'fail to come to grips with the actual Chinese room argument'.[10] Rather than debate Searle's conclusion, I have focussed directly upon his argument for it. The argument is in fact unsound and has no useful part to play in the discussion of whether a symbol-processor can have intentionality.[11]

Parallel Distributed Processing

This micro-guide aims to provide just enough detail to make the Chinese gym argument intelligible.

The basic building blocks of a PDP network are simple switch-like units, each of which is either on or off (a form of unit with more than two activity levels is described later). These are the artificial neurons. A network consists of a densely interconnected mass of units. Figure 1 shows a unit with, for simplicity, just three input connections. Call them AB, AC, AD. Connections have differing weights. Suppose the weights of AB, AC and AD are 1,2 and 3 respectively. This means that from the point of view of unit A, the effect of unit C (D) turning on is twice (thrice) that of B turning on. Thus when B, C and D are all on, the total input to A is 6. Connections whose weights are positive are called excitatory; those whose weights are negative are called inhibitory. If AC were an inhibitory connection of weight -2, the total input to A when B, C and D are all on would be (1+3)-2. Finally, the threshold of a unit is the minimum total input that will cause it to turn on.

In a sense, PDP networks operate in an extremely simple way. All that happens is that units switch themselves on and off in response to the stimulation they receive from their neighbours. This simple principle of operation leads to overall behaviour that is grotesquely complicated. Since all the units are interconnected, either directly or via some number of intervening units, they all influence one another, and the patterns of interaction are as complex as the pathways of connections between them. The network is a buzzing hive of parallel

interaction, with the units causing each other to switch on and off at a furious rate.

Input and Output

Conceptually, the units are arranged in the network in layers. The units in the input layer are such that the operator can 'clamp' them on or off, thereby overriding their tendency to change state in response to the activity of their neighbours. To compute with a network, one clamps the input units into some pattern of ons and offs. The repercussions of this disturbance rebound through the network in cycle after cycle of parallel activity. This violent reaction gradually subsides and eventually the network settles down into a stable, quiescent state. To put it metaphorically, the network gradually relaxes as it discovers how to live ever more harmoniously with the input, until finally it crystallises into a fixed configuration. Once the network has accommodated itself to the input in this way, the output can be read off the bottom layer. The output is, so to speak, one 'edge' of the stable pattern into which the network falls. (In one famous experiment each input pattern was an encoding of the root form of an English verb and the corresponding output pattern an encoding of the past tense form.[12])

In practice, a given input pattern is usually capable of producing a number of different stable states: the 'most relaxed' one, and a number of 'uneasy truces' - states that are just sufficiently stable to prevent the network from looking for a more harmonious way of accommodating the input. If a network settles down into one of these

compromise states it will stick there and never produce the desired output. To prevent this from happening units can be set up to operate probabilistically: a unit may or may not switch on when the total input it receives exceeds its threshold, and the probability that it will do so depends on the amount by which the input exceeds the threshold. This 'background noise' has the effect of shaking the network out of any compromise states into which it may fall.

Training a Network

Networks store information in a distributed fashion, with each connection participating in the storage of everything the network 'knows'. 'Programming' a network is a matter of getting the weight of each individual connection just right. This is normally achieved by a cyclical process of adjustment known as training.

Consider, for illustration, the task of generating the input pattern in reverse order on the output units (inversion). For training purposes one might choose 50 input patterns at random. The input units are clamped and the network is allowed to settle into a stable state. The output units are compared one at a time with the desired output pattern. If an output unit which ought to be on is off, the weights of the excitatory (inhibitory) connections leading to that unit from other active units are incremented (decremented). This means that the next time the network is given the same input pattern, the unit in question will be more likely to turn on. Similarly if an output unit which ought to be off is on, the weights of excitatory connections are decremented and of inhibitory connections incremented.[13] (All this is usually done

by computer.) The process is repeated with the remaining patterns in the sample. This entire cycle is iterated as many as several hundred times (using the same sample of input patterns).

Each of the inversions in the training sample makes its own individual demands on the network's connections, and so each step of the training cycle pulls the connection weights in slightly different directions. The effect of repeating the training cycle a large number of times is to forge a system of weights that suits all equally. So long as the training sample is diverse enough to represent all the demands that inversion can make on the connections, this pattern of weights will enable the network to invert patterns that it has not previously encountered.

The processing that takes place within a network consists of units exciting and inhibiting one another - not of the manipulation of stored bit-strings. (As McClelland and Rumelhart put it, 'The currency of our system is not symbols, but excitation and inhibition'.[14]) Networks just 'squirm' until they 'feel comfortable' with the input, and have more in common with a globule of molten metal cooling into a solid lump than with a VAX or an IBM stepping through a program. They are best viewed as devices that perform transformations on tuples of real numbers (*not* numerals!).[15] (I say real numbers because it is, in fact, only in the case of a restricted class of networks that the input/output vectors consist exclusively of 0s and 1s. In the general case, a unit can adopt levels of activity between 0 and 1, the level at any moment depending on the amount of input being received from other units.)

These devices can be used to support symbolic computation - even to simulate a von Neumann machine - but they can also be used in a totally different way.[16]

It is possible that Searle does not fully appreciate the difference between PDP and conventional computation. He sometimes writes as if he thinks the only difference is parallel vs serial. He says:

> Strong AI claims that thinking is merely the manipulation of formal symbols. ... The Churchlands are correct in saying that the original Chinese room argument was designed with traditional AI in mind but wrong in thinking that connectionism is immune to the argument. . . [T]he connectionist system is subject even on its own terms to a variant of the objection presented by the original Chinese room argument. . . [The argument] applies to any computational system ... whether [the computations] are done in serial or parallel; that is why the Chinese room argument refutes strong AI in any form.[17]

In fact, Searle's two anti-connectionist arguments are more powerful than he seems to think. If either works, it refutes the general claim that some form of connectionist architecture is capable of cognition, and not just the narrower claim which Searle refers to as 'Strong AI'.

The Anti-Connectionist Arguments

> Imagine that instead of a Chinese room, I have a Chinese gym: a hall containing many monolingual English-speaking men. These men would carry out the same operations as the nodes and synapses [i.e. units and connections] in a connectionist architecture. ... [T]he outcome would be the same as having one man manipulate symbols according to a rule book. No one in the gym speaks a word of Chinese, and there is no way for the system as a whole to learn the meanings of any Chinese words. Yet with appropriate adjustments, the system could give the correct answers to Chinese questions.[18]

To supply some detail: the people in the gym might simulate the behaviour of units by passing each other plastic tokens, green tokens representing input along an excitatory connection and red tokens along an inhibitory connection. The number of tokens passed from one player to another (in a single transaction) represents the weight of the connection. Since a very(!) large number of people will be involved in the simulation, it is no doubt a good idea to give players lists detailing to whom they must pass their tokens, and how many should be handed over. During the training phase of the simulation, the players make changes to their lists in accordance with the shouted instructions of the trainer.

One can agree with Searle that no amount of handing around tokens and fiddling with these lists will enable the individual players to

learn Chinese. Yet one should surely decline to conclude from this that the set-up as a whole cannot learn Chinese.[19] The fallacy involved in moving from part to whole is even more glaring here than in the original version of the room argument.

Searle's second development of the room argument runs as follows.

> Because parallel machines are still rare, connectionist programs [sic] are usually run on traditional serial machines. ... Computationally, serial and parallel systems are equivalent.... If the man in the room is computationally equivalent to both, then if he does not understand Chinese solely by doing the computations, neither do they.[20]

When Searle loosely says that serial and parallel systems are equivalent, I assume he is referring to the fact that, given unbounded resources, any connectionist architecture can be *simulated* by a von Neumann machine. Expressed rigorously, the argument seems to be this. Let C be a connectionist architecture purportedly capable of understanding Chinese. Since the man in the Chinese room is ex hypothesi capable of simulating a von Neumann machine, he is capable of simulating C (by the above fact and the transitivity of 'simulates'). Carrying out the simulation will not enable him to understand Chinese. Therefore C cannot understand Chinese. (Notice that the Chinese gym and its occupants - another simulation of C - are not required in this second argument.) Once again we have the part-whole fallacy. There is

no entailment from 'The man does not understand' to 'The von-Neumann-simulation of which he is a part does not understand'.

Searle has issued frequent warnings on the perils of confusing a computer simulation with the thing being simulated. Here are some characteristic passages.

> No one supposes that a computer simulation of a storm will leave us all wet .... Why on earth would anyone in his right mind suppose a computer simulation of mental processes actually had mental processes?[21]

> Barring miracles, you could not run your car by doing a computer simulation of the oxidation of gasoline, and you could not digest pizza by running the program that simulates such digestion. It seems obvious that a simulation of cognition will similarly not produce the effects of the neurobiology of cognition.[22]

> From the fact that a system can be simulated by symbol manipulation and the fact that [the system] is thinking, it does not follow that thinking is equivalent to formal symbol manipulation.[23]

Searle's examples show forcefully that it is in general invalid to argue in the following way: S is a simulation of X; X can Ø; therefore S can Ø. Let me refer to this form of argument as Searle's Beastie. Searle's second argument is nothing other than a contraposed form of his own Beastie: Joe is simulating C; Joe cannot understand Chinese;

therefore C cannot understand Chinese.[24] The second argument and the Beastie stand or fall together. (Notice that the move from 'the simulation of X can't understand Chinese' to 'X can't understand Chinese' is not required in the original version of the Chinese room argument. This is because the room set-up is not a simulation of a symbol manipulator - it *is* a symbol manipulator. At bottom, what gets Searle into trouble here is the fact that connectionist computation is not necessarily - and in most current research projects is not - symbolic computation.)

The same problem affects Searle's first anti-connectionist argument. The argument in fact consists of two inferences, one from a premiss about the individual players to a conclusion about the simulation as a whole, and the second from this intermediate conclusion to a claim about the network being simulated.

No individual player can understand Chinese.
∴ The simulation as a whole cannot understand Chinese.
∴ The network being simulated cannot understand Chinese.

Suppose one rejects Searle's argument for the intermediate proposition (on the ground that the argument commits the part-whole fallacy) but nevertheless feels disposed to agree that the cranky simulation of C contained in the gym is not a candidate for a Chinese-understander. I am sure this will be a common intuition. Why should one be at all tempted to infer from this that C itself cannot understand Chinese? Searle is the last person who should be advocating this inference, for its form is as before: the gym set-up is a simulation of C; the simulation cannot understand Chinese; therefore C cannot

understand Chinese. There is certainly no reason why those who claim that mentation is a computational process should accept the inference. A simulation will lie somewhere on a scale from poor to exact, and the claim that mentation is a computational process does not entail that any and every simulation of that process itself mentates. What the claim does entail is that if a computational device understands Chinese then a sufficiently exact simulation of the device understands Chinese (for an exact simulation of a computational process simply is that process: here we have an instance of Searle's Beastie that is valid). Contraposing gives a tightened form of the second part of the gym argument: if a sufficiently exact simulation of a computational device does not understand Chinese then nor does the device itself. But there is, of course, no reason to agree that the gym contains such a simulation of C.

The idea that the people in the gym can give an exact simulation of a network complex enough to process and respond appropriately to an input of Chinese sentences is otherworldly. Indeed, a number of Searle's points against cognitive science involve a studied refusal to take physical and biological realities seriously. For example: 'if we are trying to take seriously the idea that the brain is a digital computer, we get the uncomfortable result that we could make a system that does just what the brain does out of pretty much anything . . . cats and mice and cheese or levers or water pipes or pigeons or anything else . . . '.[25] The theory that the brain is a computer does not (need I say) imply that an artificial brain can *really* be made out of pigeons. (Only an absurd theory could imply this - Searle is surely right about that much.) In fairyland small boys are

made of frogs and snails and puppy dogs' tails and computers are made of cats and mice and cheese, but in the real world the structure and properties of matter place severe constraints on what children, computers, and exact simulations of large connectionist networks can be made from.

It would be a tactical error for a supporter of the second part of the gym argument to cry 'Thought experiment!' at this point. To make it true by fiat that the people in the gym are capable of giving an exact simulation of C is to chase away the intuition we began with. The argument's protagonist would be asking us to consider a world that differs radically enough from the real world as to enable a band of humans with their pockets full of coloured tokens to enact an exact simulation of a network containing maybe as many as one thousand million million connections. I have no firm intuitions about fairyland, save that one should expect the bizarre.

NOTES

1Searle 1980a; see also Searle 1989.

2Searle 1990.

3Schank and Abelson 1977.

4Some writers take Searle's set up to involve a program consisting simply of a giant lookup table that pairs Chinese characters with Chinese characters (for example Kim Sterelny, 1990, p.220ff.) This is a misunderstanding. Searle makes it clear that the details of the program make no difference to the argument (see 1980a, p.417 and 1990, p.20). Indeed, the Sam program, which Searle uses to illustrate the argument, does not have a lookup table architecture. This misinterpretation makes the Chinese room argument look weaker than it is, and lays it open to the mistaken objection that since no one believes that a lookup table architecture could qualify as a Chinese understander, 'Searle tells us only what we already know' (Sterelny, p.222).

5Searle 1980a, p.419.

6Searle 1980a, p.419. Block is the most recent writer to make this question-begging response (1990, p.282ff).

7Searle 1980a, p.419.

8Compare Dennett 1980, p.429.

9Searle 1980b, p.451.

10Searle 1990, p.24.

11For my views on that issue see chapters 3 and 6 of my *Minds, Brains, Computers*.

12McClelland and Rumelhart 1986, ch. 18.

13This particular method of adjusting the weights of connections is the classic perceptron convergence procedure of Rosenblatt (1962).

14Rumelhart and McClelland 1986, p.132.

15See Smolensky 1988.

16Although the difference between a network that is executing symbolic computations (for example a network with subnetworks whose input/output functions are identical to the characteristic functions of the primitive operations of some von Neumann machine) and one that is not is clear enough in given cases, we do not presently have a general criterion of when a computation counts as symbolic. I suspect the problem is not one of ignorance but absence: there are no necessary and sufficient conditions to be had. The best we can do - and this is quite enough - is cite paradigm examples of symbol-structures (wffs of the predicate calculus; recursive, compositional bit-code; . . .) and paradigm examples of primitive operations on these structures (compare two structures and place a marker in a given storage location if they are formally identical; replace a bound variable with a constant; shift the string in a given storage location one bit to the left; . . .) and judge a novel functional architecture in terms of its degree of similarity (if any) to the paradigm. I am suggesting that 'symbolic computation' is a family-resemblance concept.

17Searle 1990, pp.20, 22.

18Searle 1990, p.22.

19The Churchlands make this point. (Churchland and Churchland 1990, p.31.)

20Searle 1990, p. 22.

21Searle 1989, pp.37-38.

22Searle 1990, p.23.

23Searle 1990, p.21.

24To contrapose an argument one swaps the conclusion with one of the premisses and negates each. Thus A, -C ⇒ -B is a contrapositive of A, B ⇒ C. If an argument is invalid then so are all its contrapositives.

25Searle 1991, p25.

References

Block, N.: 1990, 'The Computer Model of the Mind', in Osherson, D.N., Lasnik, H. (eds.): 1990, *An Invitation to Cognitive Science*, vol.3: *Thinking*, MIT Press, Cambridge, Mass, pp.247-289.

Churchland,P.M., Churchland,P.S.: 1990, 'Could a Machine Think?', *Scientific American*, 262, no.1, 26-31.

Copeland, B.J.: (forthcoming), *Minds, Brains, Computers: An Introduction to the Philosophy of Artificial Intelligence*, Blackwell, Oxford.

Dennett,D.: 1980, 'The Milk of Human Intentionality', *The Behavioural and Brain Sciences*, 3, 428-430.

McClelland,J.L., Rumelhart,D.E. and the PDP Research Group: 1986, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, vol.2: *Psychological and Biological Models*, Bradford Books, Cambridge, Mass.

Rumelhart,D.E., McClelland,J.L. and the PDP Research Group: 1986, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, vol.1: *Foundations*, Bradford Books, Cambridge, Mass.

Rosenblatt,F.: 1962, *Principles of Neurodynamics*, Spartan Books, New York.

Schank,R., Abelson,R.: 1977, *Scripts, Plans, Goals and Understanding*, Lawrence Erlbaum Associates, New Jersey.

Searle,J.: 1980a, 'Minds, Brains, and Programs', *The Behavioural and Brain Sciences*, 3, 417-424.

Searle,J.: 1980b, 'Author's Response', *The Behavioural and Brain Sciences*, 3, 450-456.

Searle,J.: 1989, *Minds, Brains and Science: the 1984 Reith Lectures*, Penguin Books, London.

Searle,J.: 1990, 'Is the Brain's Mind a Computer Program?', *Scientific American*, 262, no.1, 20-25.

Searle,J.: 1991, 'Is the Brain a Digital Computer?', *Proceedings and Addresses of the American Philosophical Association*, 64, 21-37.

Smolensky,P.: 1988, 'On the Proper Treatment of Connectionism', *The Behavioural and Brain Sciences*, 11, 1-23.

Sterelny, K.: 1990, *The Representational Theory of Mind*, Blackwell, Oxford.