CONNECTIONISM, CHINESE ROOMS, AND INTUITION PUMPS

By

MICHAEL CARVER

Bachelor of Arts in Philosophy

Oklahoma State University

Stillwater, OK

2012

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTER OF ARTS
May, 2014

CONNECTIONISM, CHINESE ROOMS, AND

INTUITION PUMPS

Thesis Approved:

Dr. James Cain

Thesis Adviser

Dr. Doren Recker

Dr. Shannon Spaulding

ACKNOWLEDGEMENTS

I'd like to thank the Oklahoma State Philosophy Department and The Stonewall Tavern for providing encouraging environments in which to read, think and write. I'd also like to the Plumb family for their continued support of my endeavors.

Name: Michael Carver

Date of Degree: May, 2014

Title of Study: CONNECTIONISM, CHINESE ROOMS, AND INTUITION PUMPS

Major Field: Philosophy

Abstract:

John Searle's famous Chinese Room argument is perhaps the most well-known attack on computational views of mind. At the center of this argument is a thought experiment in which the reader (thinker) is lead to an intuition that computational models of mind are deeply flawed due to their syntactic (or formal) nature. In this paper, I argue that the resulting intuition of this thought experiment is dampened when the 'Classical' program contained in the original thought experiment is replaced with a 'Connectionist' program. The resulting thought experiment – The Korean Room – helps show that the intuitive results of Searle's 'intuition pump' can change as a result of relatively small changes in what we're asked to imagine.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

Classical and Connectionist Paradigms of AI

> The view that machines cannot give rise to surprises is due, I believe, to a fallacy to which philosophers and mathematicians are particularly subject. This is the assumption that as soon as a fact is presented to a mind all consequences of that fact spring into the mind simultaneously with it. It is a very useful assumption under many circumstances, but one too easily forgets that it is false.
>
> —Alan Turing, "Computing Machinery and Intelligence"

John Searle's famous Chinese Room argument is perhaps the most well-known attack on computational views of mind. At the center of this argument is a thought experiment in which the reader (thinker) is lead to an intuition that computational models of mind are deeply flawed due to their syntactic (or formal) nature. In this paper, I'll argue that the resulting intuition of this thought experiment is dampened when the 'Classical' program contained in the original thought experiment is replaced with a 'Connectionist' program. My analysis of Searle's thought experiment will closely follow that of Hofstadter and Dennett. After showing a number of variations on Searle's theme (those Searle gives in response to various objections), I'll give a variation of my own – The Korean Room. My purpose in this is to show that an understanding of different computational architectures can lead to differing intuitions concerning the prospects of computational theories of mind and 'Strong AI.' The first chapter will contrast the traditional

(or 'Classical') computational models of mind with Connectionist (or 'PDP', for 'Parallel Distributed Processing') computational models. With these contrasts in place, I'll end the chapter with an attempt to sidestep the traditional arguments between the 'Classical' and 'Connectionist' camps concerning which model is the correct model for human cognition. I'll draw on Andy Clark's 'Multiplicity of Mind' hypothesis to do this. The second chapter is then devoted to Searle's original thought experiment and some of its variations. My ultimate goal, again, is to show that the intuitions arising from the original Chinese Room thought experiment are not as conclusive as they first might appear. This is due to both the semantic status of the computational pieces in 'Connectionist' programs as well as how one decides to present the thought experiment itself.

What is the Computational Theory of Mind?

The Computational Theory of Mind (CTM), as defined by Horst, is the combination of two theses. The first thesis is the Representational Theory of Mind (RTM), which states "that intentional states such as beliefs and desires are relations between a thinker and symbolic representations of the content of the states."[1] For example, to believe that it is raining is to be in a certain functional relation to a mental representation of the content 'it is raining'. The particular functional relation determines the intentional state (i.e. belief is one type of functional relation, desire is another, and so on), and the content is given by the mental representation, which is

[1] Horst, Steven, "The Computational Theory of Mind", *The Stanford Encyclopedia of Philosophy* (Spring 2011 Edition), Edward N. Zalta (ed.), URL=http://plato.stanford.edu/archives/spr2011/entries/computational-mind/.

then manipulated in ways characteristic of the particular intentional state (e.g., belief). The second thesis is the Computational Account of Reasoning (CAR). This states that the representations referred to in the RTM "have both semantic and syntactic properties, and processes of reasoning are performed in ways responsive only to the syntax of the symbols".[2] The idea here is that mental representations can retain their semantic content while only being manipulated according to their syntactic properties. That is, reasoning consists of syntactic, symbol manipulation that at the same time preserves semantic relations and values. In order to fully draw out what this all means and involves, it will be helpful to discuss (1) what it is to manipulate symbols formally (i.e. what a formal system is) and (2) how this relates to the Computational Theory of Mind. Then we can compare this with how connectionist systems work.

What is a Formal System?

According to Haugeland, every formal system has three essential properties. First, they are "token manipulation" games. Second, they are "digital." And third, they are "finitely playable."[3] A "token manipulation" game essentially consists of (1) a set of tokens and (2) a set of rules according to which these tokens are manipulated. Games such as Chess provide easy examples for formal systems. So for example, the tokens for chess are the board's pieces, and likewise for checkers and tic-tac-toe. These pieces need not be standard physical pieces, manipulated simply by moving one's hand. They can be marks on a sheet of paper, lights, or what have you. This is to say that the system is neutral on the details of its implementation. So long as the formal nature of the

[2] Ibid.
[3] John Haugeland, *Artificial Intelligence: The Very Idea* (Cambridge, MA: MIT, 1985), p. 48.

system remains, the system can be made of anything. Formal systems are *medium-independent*. "Texas millionaires, for instance, could play chess from their opposing penthouses, using thirty-two radio-controlled helicopters and sixty-four local rooftops. Or, if they owned an eight-story hotel with eight rooms per floor, they might use brightly marked window shades."[4] Non-formal systems do not share this property. For example, if the field on which American football is played is halved or made out of ice, the result may be hilarious, but it wouldn't be football in the normal sense. American football is not medium independent. Formal systems are.

These tokens are then manipulated according to a set of rules. These rules can include moving, adding, deleting, or altering tokens (by changing what *type* of token is present). In playing Chess, I can move a bishop diagonally across the board, turn a "pawn" piece into a "queen" piece if it's in the right circumstance, or remove certain pieces (when they've been "captured"). Though there is no circumstance in which I can add a piece in chess, another example, tic-tac-toe, consists solely of adding pieces.

Second, formal games are "digital." According to Haugeland, "a digital system is a set of positive and reliable techniques (methods, devices) for producing and reidentifying tokens, or configurations of tokens, from some prespecified set of types."[5] This means there is a method of producing tokens according to the rules of the system (or "writing") that can work perfectly (the process is "positive") and does not typically fail (the process is "reliable"). Further, the method of identifying (or "reading") the tokens of the system shares these same characteristics (positivity and reliability). This

[4] Ibid, p. 58.
[5] Haugeland, *Artificial Intelligence*, p. 53.

method of identifying "implies nothing about understanding (or even recognition) but only differentiation by type and position."[6] If the method of producing tokens can be labeled "writing" and the method of reidentifying tokens can be called "reading", then a digital system amounts to a process of reading and writing tokens that can work perfectly and reliably. Of course, some mediums will be better suited to digital processes than others. To distinguish digital mediums from non-digital mediums, Haugeland compares Rembrandt's paintings with Shakespeare's sonnets.

> Even given the finest care, the paintings are slowly deteriorating; by no means are they the same now as when they were new. The poems, by contrast, may well have been preserved perfectly. Of course a few may have been lost and others miscopied, but we probably have most of them *exactly* the way Shakespeare wrote them—absolutely without a flaw. The difference, obviously, is that the alphabet is digital (with the standard read/write cycle), whereas paint colors and textures are not.[7]

The method that is used in the Shakespeare case has the possibility to both read and write perfectly, whereas with Rembrandt's paintings, the best one can do is approximate both what is read and what is written. Third, formal systems are "finitely playable." To be finitely playable, a system must be able to be played by a finite being, with a finite (typically small) set of primitive abilities (e.g., reading and writing tokens).

In order to define a given formal system (or token manipulation game), you need three key pieces of information: what the tokens are (including what the types of tokens are – e.g. a queen vs. a pawn piece), a starting position for the tokens, and a method of

[6] Ibid.
[7] ibid, p. 55.

determining what manipulations of tokens are allowed in any given position.[8] Some formal games have only one starting position, whereas other games may have many possible starting positions. To be able to determine what token manipulations are allowed in any given position is simply to know the rules of the game. That is, if I know the rules of the game, I can accurately deduce what moves are legal from any given position. A formal game then proceeds by (1) setting up the starting position of the tokens and then (2) manipulating those tokens through the legal moves set out by the rules of the game. Many games of this kind have an end goal (obviously chess and checkers are examples of this), but this isn't strictly necessary.

Another essential feature of Formal Systems is that the *meanings* of the tokens aren't implicated in any way in the rules of the system. When we define a formal system, nothing rides on any potential meanings of the tokens. The rules of the game only rely on or reference the syntactical or formal nature (type and position) of each token. Keeping with the chess example, the rules of chess don't in any way reference the role of the queen piece as representative of 'Queenhood' or in any way implicate the pawn's subservience to the more royal pieces. The only relevant feature of the 'Queen' piece is the type of token it is. It is a type of token in the game that can be moved in certain ways relative to the other tokens, as per the rules of the game. Another way to say this is to say that "formal systems are *self-contained*; the "outside world" (anything not included in the current position) is strictly irrelevant."[9] This includes the history of the game/system. If we were to completely define chess

[8] ibid, p. 49.
[9] ibid, p. 50.

formally, we would need a couple tokens outside of the board to indicate any relevant historical facts about the game (e.g., whether the king has been moved prior to castling).[10]

However, formal systems can be (and often are) organized such that the moves within the game respect and preserve the meanings of their constituent tokens. The pieces of the game are then, not only tokens, but *symbols*. A token that has an assigned or interpreted meaning is often called a *symbol*.[11] Further, an *automatic* formal system is a device in which the moves of the game are carried out without any direct guidance from an outside mind. Haugeland gives two essential characteristics of such devices. First, some parts of the device are identified as the tokens of the formal system. Second, token manipulation according to the rules of the system is carried out automatically.[12]

What is the Computational Theory of Mind? (Part 2)

The Computational Theory of Mind, then, is the claim that humans *are* (to a significant extent) very complex automatic formal systems. Our mental states are a combination of contentful representations that are tokened, and the rules by which those representations (the tokens) are created, manipulated, and used. The manipulation and usage of tokens only relies on the syntax (or form) of the representations while at the same time respecting the content (or meaning) of those tokens. Our minds turn out to be a system of symbol shunting, a formal game in which the tokens and their process of manipulation drives (intelligent) behavior. These

[10] ibid, p. 257 note 2.
[11] John Haugeland, "What is Mind Design?," in *Mind Design II: Philosophy, Psychology, Artificial Intelligence*, ed. John Haugeland (Cambridge, MA: MIT, 1997), p. 16.
[12] Haugeland, What is Mind Design?, p. 11.

operations can be carried out by a system in which the "drivers" of the process are "physically simple" mechanisms. These mechanisms instantiate and carry out the operations of the formal system by doing nothing other than obeying the laws of physics.

Semantic Transparency

A key feature of this view concerns the fundamental elements on which the computations are performed. The units that are manipulated by computational means are *semantically transparent*. Andy Clark defines a semantically transparent system as one in which the formal tokens of the system can be given a direct (one-to-one) mapping onto categories of ordinary day-to-day discourse. As Clark states, "what this means is that the computational operations specified by the algorithm [the set of token manipulations that comprises mental activity on the CTM hypothesis] are applied to internal representations that are projectibly interpretable as standing for conceptual-level entities."[13] This means that the tokens of the mental formal system can be given a one-to-one mapping to the ordinary categories of discourse. These tokens will then *represent* the very concepts of this mapping. These tokens can be both simple and complex. They constitute a 'language of thought' or 'Mentalese' in which the system thinks.[14] By contrast, connectionist systems are neither semantically transparent nor

[13] Andy Clark, *Microcognition: Philosophy, Cognitive Science, and Parallel Distributed Processing* (Cambridge, MA: Bradford Books/MIT, 1989), p. 18.
[14] Following Clark (*Microcognition*, p. 19-20), I claim that all Classical models are committed to both the hypothesis that these systems are semantically transparent and well as the existence of the language of thought.

posit a language of thought. This will be due to the nature of the fundamental units of computation in connectionist systems (more on this later).

We should also note that the Computational Theory of Mind not only states that the input-output profile is of a certain type or that we could simulate our thinking through such a formal system, but that the inner-workings of the mind – what ultimately creates that input-output profile – is formal symbol manipulation. It is one thing to say that an input-output function can be simulated or otherwise duplicated on a computer (through *some* computational means). It's another to positively claim the method by which the mind transforms that input into output. The Computational Theory of Mind claims that thinking *just is* this process of moving around representational tokens according to certain syntactical rules.[15]

A couple of merits of this theory follow from this. First, this theory (if correct) allows for minds to be explained purely in terms of physical systems. Practically all of the parties to the debates surrounding the Computational Theory of Mind agree that the mind (or at least intelligence) is to be explained in physical terms. That is, they're all psychological materialists.[16] Second, this theory is an empirical hypothesis about how minds work. As a result, it implies a broad set of research projects, rising out of a varied group of disciplines, some of which are created in direct response to this hypothesis. I'm referring, of course, to cognitive psychology and artificial intelligence specifically, which

[15] Haugeland, What is Mind Design?, p. 16.
[16] ibid, p. 2.

along with neuroscience and linguistics form the larger project of Cognitive Science.[17] Thus the Computational Theory of Mind gives rise to an explicit theory about how minds work.

GOFAI as a Research Project (the logical extension of CTM)

The Computational Theory of Mind is often discussed in reference to a set of projects in Artificial Intelligence (AI) that developed out of this view of mind. Haugeland characterizes the project of Good Old Fashioned Artificial Intelligence (GOFAI) – the Classical AI Project – as taking its cue directly from the idea that intelligence and reasoning essentially involve formal symbol manipulation. He claims that two ideas are essential to all projects within the GOFAI movement. First, "our ability to deal with things intelligently is due to our capacity to think about them reasonably (including subconscious thinking)." Second, "our capacity to think about things reasonably amounts to a faculty for internal "automatic" symbol manipulation."[18] These theses about intelligent systems point to the Computational Theory of Mind as the heart of GOFAI.

The Physical-Symbol-System Hypothesis

The GOFAI tradition was given an explicit formulation by Herbert Simon and Allen Newell's discussion of Physical Symbol Systems (automatic formal systems). In their 1976 paper, "Computer Science as Empirical Inquiry: Symbols and Search", Simon and Newell put forward an admirably explicit thesis about how intelligent systems work:

---

[17] Gardner, Howard. *The Mind's New Science: A History of the Cognitive Revolution* (New York: Basic, 1985), p. 37.
[18] Haugeland, *Artificial Intelligence*, p. 113.

> The Physical Symbol System Hypothesis: A physical symbol system has the necessary and sufficient means for general intelligent action.
>
> By "necessary" we mean that any system that exhibits general intelligence will prove upon analysis to be a physical symbol system. By "sufficient" we mean that any physical symbol system of sufficient size can be organized further to exhibit general intelligence. By "general intelligent action" we wish to indicate the same scope of intelligence as we see in human action: that in any real situation, behavior appropriate to the ends of the system and adaptive to the demands of the environment can occur, within some limits of speed and complexity.[19]

In this definition, not only have Simon and Newell stated that any physical symbol system, sufficiently complex and organized, will be intelligent (the sufficient condition), but that any intelligent system will turn out to be a physical symbol system, organized in some specified way (the necessary condition). This is very strong, for it not only applies to human minds (our own signpost for what intelligence is or could be) but this seems to apply to *all minds* or (perhaps more accurately) all intelligent systems. With this explicit hypothesis about how intelligence works, one which Simon and Newell describe as an empirical thesis of computer science about the nature of intelligence, similar to the germ theory in medicine, Simon and Newell started a tradition of research in Artificial Intelligence – the GOFAI tradition.

For Simon and Newell, the project of A.I. is straight-forward. The first step of trying to confirm the Physical Symbol System Hypothesis is to construct computer programs (i.e., physical symbol systems) that can do intelligent tasks. Simon and Newell compare their project with germ theory:

---

[19] Allen Newell and Herbert Simon, "Computer Science as Empirical Inquiry: Symbols and Search," *Communications of the ACM* 19, no. 4 (1976): p. 116.

> The basic paradigm for the initial testing of the germ theory of disease was: identify a disease, then look for the germ. An analogous paradigm has inspired much of the research in artificial intelligence: identify a task domain calling for intelligence, then construct a program for a digital computer that can handle tasks in that domain.[20]

This project (the project of Classical A.I.) works on the sufficiency side of their hypothesis. If these A.I. researchers can show that certain types of physical symbol systems can carry out tasks requiring intelligence, then they've shown that those systems will suffice for intelligence. On the other side of the biconditional is the project of cognitive science, which examines man's intellectual capacities and "attempts to discover whether his cognitive activity can be explained as the working of a physical symbol system."[21] The basic story here is that Simon and Newell take the Computational Theory of Mind and begin to create a research project. They lay out an explicit thesis about how intelligent systems work then begin a broad research program for to show the sufficiency side of their thesis. This program is GOFAI. Again, this program is essentially linked to the view of minds as automatic formal systems (CTM). Throughout this paper, I will sometimes refer to the larger project of cognitive science that takes the Computational Theory of Mind as its foundation as "Classical Cognitivism."

On the more philosophical side of Classical Cognitivism, Jerry Fodor (one of the first to put forward the Computational Theory of Mind and the Language of Thought hypothesis specifically) claims that the hypothesis of the Computational Theory of Mind (with its requisite language of thought) is the only good hypothesis that we have about

[20] Simon and Newell, Computer Science, p. 118-119.
[21] Ibid.

how (at least a significant portion of) minds work. Fodor's claim, essentially, is that Classical Cognitivism is "the only game in town."[22] However, many point to Connectionist models of cognition as a promising alternative to the Computational Theory of Mind and Classical Cognitivism.

What is Connectionism?

While more classical models of cognition are abstracted away from the details of implementation, Connectionist models of cognition seek to model intelligent behavior with an eye toward those details. The connectionist approach to cognitive models is "neurally-inspired."[23] These types of models look at the basic properties of neurons and neural networks to create a model that more shares some of those basic properties. Such models are variously called "connectionist", "PDP" (Parallel Distributed Processing), or "neural networks". These models are able to deal with and act on information that is incomplete, ambiguous, and error-prone in a flexible way. This is largely due to their brand of computational architecture.

While there are numerous species of Connectionist / PDP models, they all share some key traits. In a Connectionist model a set of fundamental computational pieces (often called 'nodes' or 'units') are arranged and connected in a way strongly analogous to how neurons connect to one another. These nodes are given some input from other

[22] Jerry Fodor, *The Language of* Thought (New York: Crowell, 1975).
[23] David Rumelhart, "The Architecture of Mind: A Connectionist Approach," in *Foundations of Cognitive Science*, ed. Michael Posner (Cambridge, MA: MIT, 1989), p. 206. Reprinted in *Mind Design II: Philosophy, Psychology, Artificial Intelligence*, ed. John Haugeland (Cambridge, MA: MIT, 1997). – All page references are from the reprint.

nodes and then they send some output to other nodes. Each node has an activation value which determines whether or not the node "fires". If this activation value surpasses a certain threshold (the activation threshold) then it will activate (or "fire"), sending an output signal to the connecting nodes down the line. Further, each individual connection between nodes has a "weight." This weight determines both the strength of the connection between nodes as well as whether the connection is excitatory or inhibitory. A connectionist network, then, consists of a number of these units, each working in parallel to the others and each influencing the behavior of the others. Once some of the nodes in the network are activated, their activation spreads around the network, and, depending on the arrangement of the connections (along with their weights), some nodes will tend to be excited and therefore activated while some will tend to be inhibited and therefore remain dormant (or unactivated).

Further, connectionist networks give rise to additional advantages in that they are have the following properties: "content addressable memory, graceful degradation, default assignment, and generalization."[24] In order to explain these properties and show how they come out of a connectionist model, an extended example is in order.[25] As will become clear, representations can emerge out of the global properties of Connectionist networks.

[24] Clark, *Microcognition*, p. 88.

[25] The following example and illustration are taken from Clark *Microcognition*, p. 86-92. Clark borrowed this from David Rumelhart and James McClelland, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition* (Cambridge, MA: MIT, 1986).

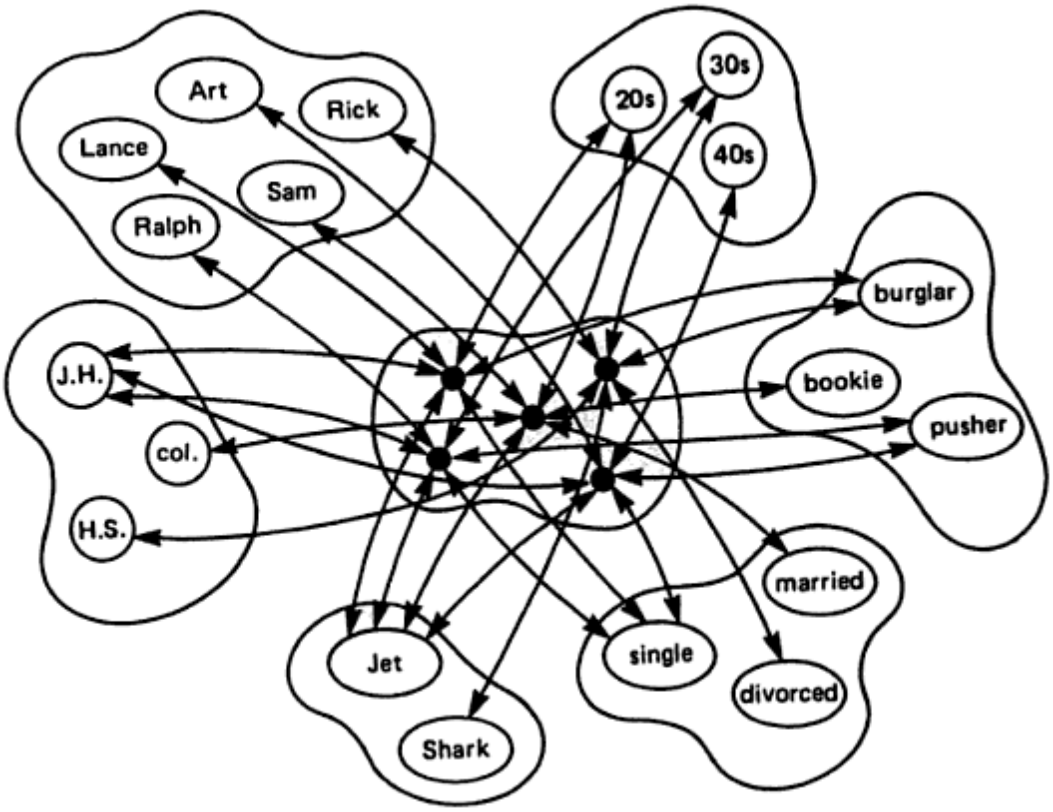

Figure 1

In Figure 1, a number of gang members are represented in a connectionist network. Each person represented has the following properties: name, gang, age, education level, marital status, and occupation. In order to prevent too many overlapping lines from being drawn, a couple of conventions will hold for this model. First, the clouds represent inhibitory groups. That is, there is an inhibitory connection from any given member of a cloud to any different member of that cloud. So, if one member in a given cloud is activated, then the other members of that cloud will receive an inhibitory signal. Second, the lines with arrows at either end indicate that two connected nodes are mutually excitatory. In this example, the units are properties of people, but this need not be the case.

Name	**Gang**	**Age**	**Education**	**Marital Status**	**Occupation**
Art	Jet	40s	J.H.	Single	Pusher
Lance	Jet	20s	J.H.	Single	Burglar
Ralph	Jet	30s	J.H.	Single	Pusher
Rick	Shark	30s	H.S.	Divorced	Burglar
Sam	Jet	20s	Col.	Married	Bookie

Table 1

If one were to organize the data pertaining to these five gang members into a traditional (GOFAI) data structure, each person would likely be given their own structure, with separate memory stores for each property of that person (something structurally similar to Table 1). "In a more conventional approach this information would be stored at one or several addresses, with retrieval dependent upon knowing the address. But a designer may want to make all this information accessible by *any* reasonable route."[26] Connectionist models have just this ability. To see this, note that, in the Connectionist model, we can "pull up" the name of a given member in a number of ways. For example, if I wanted to find the name of the Single gang member is his 40s, the system could easily find that information. First, as input, the nodes for Single and 40s are activated. This activation would spread to a couple central nodes, which would activate three nodes in the central cloud, one of which would be doubly activated because it was sent signals by both the node labeled 'Single' and the node labeled '40s'.

[26] Clark, *Microcognition*, p. 88.

Further, this particular node would have the effect of inhibiting the other nodes in the cloud. Eventually, through this process of spreading activation, we get the nodes labeled 'Art', 'Jet', '40s', 'J.H.', 'Single', and 'Pusher' activated significantly more than the other nodes. This same process would occur with varying amounts of input information. In a Classical system, the program would have to somehow find the memory address of the desired member. In a Connectionist system, all that's needed is some information that uniquely picks out that member. Connectionist systems have a *content addressable memory* that's typically missing from Classical systems.

With this example, in which we have a couple pieces of correct information, a GOFAI system could perform just as well (using a technique called hash coding). However, if we were to add in errors to our data, the GOFAI system would need to use a costly 'best-match' search in order to cope with the errors.[27] With a connectionist structure, the weeding out of errors is done without any extra processes, through the use of existent inhibitory connections. Suppose we wanted to find the name of the married pusher in his thirties. As it happens, this description doesn't match anyone. There is pusher in his thirties, but he isn't married. If we tried to activate the three nodes labeled 'Married', 'Pusher', and '30s', a process of spreading activation would occur in which the node labeled 'Married' would eventually be inhibited, since the activation from the node 'Single' would be larger and thus inhibit the 'Married' node. Thus the nodes labeled 'Ralph', 'Jet', 'Thirties', 'J.H.', 'Single', and 'Pusher' would end up

[27] Ibid.

activated. This trait – the ability for the system to remain resilient and give sensible responses in the face of errors in input – is often referred to a *graceful degradation*.

Scripts, Schemas, and Frames

Schank and Abelson gave evidence that we often deploy scripts or schemas for dealing with situations or events.[28] These scripts or schemas are data structures that contain paradigmatic traits of common events or objects. Since we don't typically have the full information about some particular event or object when first exposed, we assume 'default values' based on the category to which the event or object belongs. All of these default values, which comprise what we might call a *paradigm case* of that category, need not be instantiated in any one instance. For example, your schema for a kitchen (the 'ideal' kitchen) may contain a conventional stove, microwave, two-sink station, and dishwasher though you have never encountered a kitchen containing all of those appliances. Nevertheless, we use this sort of script or schema in order to fill in missing information with what we take to most likely be the case.

In 1974, Marvin Minsky presented a Classical model through which we can understand how such schemas are structured. He defines a 'Frame' as a knowledge structure in which typical items, events, and situations can be stored with values pertaining to the typical properties of those items, events, and situations. As Minsky states, "a *Frame* is a data structure for representing a stereotyped situation, like being in

[28] Roger Schank and Robert Abelson, *Scripts, Plans, Goals and Understanding* (Hillsdale, NJ: L. Erlbaum Associates, 1977).

a certain kind of living room, or going to a child's birthday party."[29] These data structures consist of a number of memory stores that contain the stereotyped situation and give the knower something to expect when various situations arise. Further, the Frames contain instructions about what to do when an event doesn't conform to the stereotypical expectations. These structures are also flexible in that the 'default values' can be changed as needed. Frames start out as very imperfect notions of what is typical and are further perfected as the knower gathers more information. A vague, and mostly incorrect, idea of what a 'party' consists in is improved as one goes to more parties.

Two key aspects of Minsky's notion of Frames serve to highlight the character of the Classical Cognitivist approach to knowledge structure, storage and retrieval. First, the knowledge structure of Frames is explicit. There is a separate area of memory which contains the properties of the stereotyped event. Second, since these 'paradigms' are stored explicitly, they are changed through explicit methods. There are operations performed on that structure that serve to assign default values, change values, decide when enough evidence is present to change a value, recognize that a situation is occurring (and thus that a particular Frame applies), and so on. These operations are done explicitly within the system. Connectionist systems don't have any of these properties. Continuing with Jets/Sharks example, if I wanted to pull up an "ideal" member of the Jet gang in the connectionist model, I could simply activate the node labeled 'Jet'. At the end of a sequence of spreading activation, the nodes corresponding to 'Jet', '20s', 'Single', 'J.H.', and 'Pusher' would all be activated, though there is no

[29] Marvin Minsky, "A Framework for Representing Knowledge," in *Mind Design II: Philosophy, Psychology, Artificial Intelligence*, ed. John Haugeland (Cambridge, MA: MIT, 1997), 111-2.

single member of the Jets with all of those traits. Unlike the standard organization for Classical models, in which the schema would be set up in advance and then stored in a local address alongside knowledge of particulars, the representation for the schema in Connectionist systems is not only *distributed* throughout the system, but was never explicitly stored. As Clark notes, "One striking feature of the PDP version [of representation storage] is its capacity to generalize in a very flexible way with no need for any explicit storage or prior decisions concerning the form of required generalizations. The network can give you a typical completion of any pattern you care to name if there *is* some pattern in the data."[30] Not only does this model benefit from this distributed (and implicit) storage of schemata, but it also allows for the recognition of unpredicted patterns to be found within the data itself.

The Subsymbolic Paradigm

As Smolensky emphasizes, connectionist systems do without one of the key features of Classical models: semantic transparency. The fundamental units of connectionist systems aren't semantically transparent and do not straight-forwardly represent concepts in the one-to-one way that Classical models do. Rather, representations occur as a result of the global behavior of the connections between nodes and weights. In the Jets/Sharks example, the representation of an 'ideal' Jet isn't contained in any of the nodes. Rather, it's contained in the global arrangement of the system's weights. In Classical models, the fundamental units are straightforwardly symbolic, but Connectionist models are in what Smolensky calls the 'Subsymbolic

[30] Clark, *Microcognition*, p. 92.

Paradigm.' Smolensky notes that one of the key problems with the "symbolic paradigm is quite simply…that it has provided precious little insight into the computational organization of the brain."[31] In Connectionist models, the "*subsymbolic level* is supposed to be closer to each of the neural and symbolic levels than they are to each other."[32] That is, the fundamental units of computation in Connectionist models take place below the level of ordinary concepts and above the level of the neuron. The Connectionist's node is a gross simplification of how a neuron functions and is meant to be so.

The nature of the difference between the 'symbolic' paradigm (e.g., Classical models) and the 'subsymbolic' paradigm (e.g., Connectionist models) concerns the semantic status of the fundamental units that are operated on. Clark highlights the divide with a question: "The essential difference between the subsymbolic and the symbolic approach, as Smolensky paints it, concerns the question, Are the semantically interpretable entities the very same objects as those governed by the rules of computational manipulation that define the system?"[33] For classical models, the answer is yes. The entities that are computationally manipulated are those very concepts that are manipulated. The units are semantically transparent. For connectionist models, the answer is no. The Connectionist modeler "urges that the entities whose behavior is governed by the rules of computational manipulation that define the system need not share the semantics of the task description. For what is so governed is just the activation

[31] Paul Smolensky, "Connectionist Modeling: Neural Computation / Mental Connections," in *Mind Design II: Philosophy, Psychology, Artificial Intelligence*, ed. John Haugeland (Cambridge, MA: MIT, 1997), p. 237.
[32] Ibid.
[33] Clark, *Microcognition*, p. 112.

profiles of individual units in a network. And in a highly distributed model these units in the end will have no individual semantic interpretation, or at least none that maps neatly and projectibly onto our ordinary concepts of the entities to be treated in a model of the processing involved."[34]

This apparently contradicts the Jets/Sharks example, in which there were semantically transparent nodes (those representing the properties of gang members). This is true. However, I used the Jets/Sharks example only to help ease the explanation of some key features of connectionist systems. Other models can do without such a crutch, as Clark notes above. The Connectionist program promises that all such semantically transparent features can be dispersed throughout the system's nodes and system of weights in models that are sufficiently complex. In sufficiently complex Connectionist models, all representations are highly-distributed throughout the system.

Of course, the debate between Classical and Connectionist models of cognition rides on more than just the symbolic nature of each program's computational pieces and the implications for representation. However, I need not weigh in on this debate. I only need to suppose that an ideal program that's Connectionist "at bottom" can pass the Turing Test. It may be the case, of course, that parts of this fundamentally Connectionist system are instantiations of Classical programs. My purposes here will be served with a plausible story about how the two paradigms could come together such that the system is Connectionist "at bottom." Andy Clark's 'Multiplicity of Mind'

[34] Ibid.

provides one such story. The rest of this chapter will be devoted to explicating just what he has in mind.

Classical v. Connectionist Task Domains

Connectionist models are simply better for a number of tasks. For example, they provide an ability to "shade" the meanings of words in various contexts, as demonstrated by McClelland and Kawamoto's model of language comprehension.[35] With these models, we can better understand how "words seem to take on different shades of meaning in a continuously varying fashion, one that seems unspecifiable in advance."[36] For example, Clark considers three sentences:

(1) The boy kicked the ball.
(2) The ball broke the window.
(3) He felt a ball in his stomach.

Each instance of the word 'ball' in the first two sentences of course signifies a physical object, though the characteristics of each ball may vary widely (e.g. softness, size, color, typical use, and material). The third use is of course metaphorical – there's no physical ball in his stomach. A Classical model of language comprehension would likely have to allot separate areas of memory for each meaning, assign values to various properties, and so on. A Connectionist model, on the other hand, can do all of this with one localized set of nodes. Depending on context, a large number of different global

[35] James McClelland and Alan Kawamoto, "Mechanisms of Sentence Processing: Assigning Roles to Constituents of Sentences." In David Rumelhart, James McClelland and the PDP Research Group, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition* (Cambridge, MA: MIT, 1986), p. 272-325.
[36] Clark, *Microcognition*, p. 110.

patterns of activation can emerge and vary with the meaning of any one word. Further, "there would be no firm, God-given line between literal and metaphorical meaning; the metaphorical cases would simply occupy far-flung corners of a semantic-state space…Metaphorical understanding, on the present model, is just a limiting case of the flexible, organic kind of understanding involved in normal sentence comprehension."[37] Where the Classical models would have to do this in a serial, linear, and explicit way, Connectionist models get all the flexibility of metaphor and language ambiguity without paying a heavy computational cost. Further, Connectionist models are typically adept at relatively basic cognitive tasks, such as vision and sensorimotor control.[38]

However, there remain tasks at which the Classical model remains the best model. Clark points to such tasks as "the serial-reasoning tasks of logical inference, the temporal-reasoning tasks of conscious planning, and perhaps the systematic-generative tasks of language production."[39] The common thread of these sorts of tasks is that they involve explicit rule-following of just the type that Classical models are built around. Activities that are traditionally in the purview of work in A.I. are just these sorts of activities. Playing chess, doing logic, and even conscious attempts to drive a car all involve this sort of explicit reasoning.[40]

There is a thematic difference between the types of tasks that work best with these two models. Clark notes that the tasks that have seen the most success with

[37] ibid, p. 111.
[38] ibid, p. 127.
[39] Ibid.
[40] Ibid.

connectionist modeling are tasks that tend to be evolutionarily prior to those tasks in which connectionist system perform poorly. That is, if we were to look at the likely order in which cognitive abilities developed in the course of evolution, those tasks at which connectionist systems excel come early in the list. Clark gives a list (partially drawn from a work on animal cognition) of cognitive tasks that are prior. The list includes general abilities like locomotor and manipulative skills, spatial skills, perceptual skills, general analogical reasoning, and basic social skills.[41] Clark emphasizes that, if we look at what he calls the "functional phylogeny of mind," we notice that early abilities are just those at which the connectionist program excels. And just the opposite holds for Classical models.

Clark's 'Multiplicity of Mind'

Further, Clark notes that most of the talk between the two camps (Classical & PDP) assumes that the uniformity assumption is correct. The Uniformity Assumption states that "*every* cognitive achievement is psychologically explicable using only the formal apparatus of a single computational architecture."[42] And of course each camp believes that their brand of the uniformity assumption is the correct one. The Classical Cognitivist believes that all cognitive abilities are explicable with Classical models, and the Connectionist believes that those same abilities are only explicable with Connectionist models. Clark, however, explores the neglected middle, the notion that

[41] ibid, p. 73.
[42] Ibid, p. 128.

Classical models are more accurate for some task domains while Connectionist models are right for others.

Clark, noting the evolutionary relevance of how the task domains are split between the two camps, further speculates on the way in which these two models can be related. He takes a suggestion from Rumelhart that "three capacities combine to allow human beings (who are assumed to be PDP devices at root) to perform complex, sequential, symbol-processing tasks. These are:

(1) a basic PDP pattern-matching capacity,
(2) a capacity to mentally model our environment
(3) a capacity to physically manipulate our real environment, and to perceive the effects of such manipulations (adapted from Rumelhart, Smolensky, et al. 1986, 44)"[43]

The idea here is that we used our ability to model the environment to model the definite, serial manipulation of external objects and symbols. This ability then was taken on-board as the serial symbol-processing method that Classical Cognitivism represents. Our ability to perform very logical and serial kinds of tasks is a result of the recent adopting of certain properties of object manipulation. We've taken our initial ability to manipulate the environment in a serial manner and adapted that ability to our already on-board PDP architecture. Further, Clark notes that, if this is the case, our basic cognitive apparatus remains a PDP system at its root. Now, it's just a system that has been adapted to use the advantages of doing certain cognitive tasks in a linear, systematic way. This is just what we would expect from this sort of evolutionary adaptation. As Clark states:

[43] ibid, p. 134.

> Notice, however, that even if some mental models have evolved, our basic architecture would remain a PDP system, though cunningly configured to make possible certain kinds of sequential, symbolic thought. We would not have an architecture purposely built for such reasoning. The historical snowball effect…works to kludge an architecture chosen for speedy perceptual and sensorimotor processing into something capable of some kinds of sequential, conscious reasoning.[44]

A cognitive system that evolved to carry out certain basic tasks is snowballed into a system that can carry out a fundamentally different kind of task. Recently, we've evolved to carry out explicit and logical kinds of tasks, but we can't do so whole cloth. This must be taken on-board through the use of what we already have. Thus, we remain Connectionist or PDP systems at bottom.

Throughout this paper, I'll assume that something like Clark's hypothesis is right, that a connectionist system can run Classical Virtual Machines and classical models will remain useful for modeling certain processes. However, if this is true, then connectionism will still be true "at bottom." That is, wherever Classical Cognitivist models hold, they will only hold because they are being instantiated by connectionist models. With this, we can get onto the main point of this paper.

[44] ibid, p. 134.

CHAPTER II

Chinese Rooms and Intuition Pumps

The primary purpose of this chapter is to show that whatever intuitive appeal Searle's Chinese Room thought experiment (CR) had begins to fall apart when it's modified such that the room contains a connectionist system, rather than the Classical GOFAI system that is contained in Searle's original Chinese Room. My method will be to use Douglas Hofstadter and Daniel Dennett's analysis of CR to explore a number of alternate thought experiments that may shed light on what Searle's own thought experiment does. One of these thought experiments (The Korean Room) will be my own. I will argue that modifying the Chinese Room such that it contains a Connectionist system will go far toward taking the intuitive appeal out of Searle's original thought experiment.

The Chinese Room

With the Chinese Room, the primary argument that Searle takes himself to give states that 'Strong AI' – the thesis that a suitably programmed computer would, in virtue of such programming, have a mind in the very sense that we have minds – is deeply flawed and mistaken. Specifically, Searle argues that the suitable arrangement of

a system with only syntactical properties (a formal system) isn't sufficient to create semantics. Or, applying the system to minds, the system would fail to have intentionality (the mental analogue of semantics). His argument, more formally, looks like this (from Dennett).

Proposition 1. Programs are purely formal (i.e., syntactical).
Proposition 2. Syntax is neither equivalent to nor sufficient by itself for semantics.
Proposition 3. Minds have mental contents (i.e., semantic contents).
Conclusion 1. Having a program—any program by itself—is neither sufficient for nor equivalent to having a mind.[45]

Propositions 1 and 3 are largely non-controversial.[46] Searle's primary task then is to make the case for Proposition 2. He does this by way of his 'Chinese Room' thought experiment.

The thought experiment runs like this. In a room there is a monolingual English speaker, who I'll typically refer to as 'the demon.' The demon has in the room with him sets of pieces of paper. On these pieces of paper are sets of Chinese symbols and rules for the manipulation of those symbols (these rules for manipulation are in English). The demon, being monolingual, knows nothing of what the Chinese symbols mean. To the demon, "Chinese writing is just so many meaningless squiggles."[47] The demon can only distinguish the different types of symbols. The demon is able to manipulate the symbols according to the rules he is given. Sheets of paper with Chinese symbols on them enter the room. The demon manipulates the symbols and outputs other Chinese symbols, all

[45] Daniel Dennett, *The Intentional Stance* (Cambridge, MA: MIT, 1987), p. 324.
[46] Though Dennett explicitly rejects all three (Intentional Stance, p. 336-7).
[47] Douglas Hofstadter and Daniel Dennett, *The Mind's I: Fantasies and Reflections on Self and Soul* (New York: Basic, 1981), p. 355.

according to the (English) rules he is given. Unknown to the demon, the input-output profile of the room is such that it passes the Chinese version of the Turing Test. Searle's move then is to look for the understanding (or semantics) of the system. Surely the man doesn't know what any of the Chinese symbols mean. To illustrate what's missing from the mind of the demon, Searle considers the demon's ability with regard to English. The demon would surely pass a Turing Test in English, for the very reason that the demon, by hypothesis, is a native English speaker and thus *understands* English just as well as any other native speaker. However, the demon also passes a Chinese Turing Test, but only because of the program that he runs. With respect to Chinese, the demon "simply behave[s] like a computer; [he] perform[s] computational operations on formally specified elements. For the purposes of the Chinese [he is] simply an instantiation of the computer program."[48] Thus it is claimed that "syntax is neither equivalent to nor sufficient by itself for semantics" (i.e. proposition 2 of Searle's argument).

The Systems Reply

Admirably, Searle includes his own answers to a number of objections to his original argument and thought experiment. I'll present the four most common and relevant objections and Searle's replies to give context to my own remarks later on. First, the Systems Reply claims that Searle's thought experiment relies on the failure to make a distinction between the man in the room and entire system that the man is a part of. When we consider that the man only manipulates symbols according to rules, we recognize that the pieces of paper and even the room itself help the system to function

[48] John Searle, "Minds, Brains, and Programs", *Behavioral and Brain Sciences* 3, (1980): 418.

properly. It's further claimed that the man may not understand any of the Chinese symbols, but we're asking the wrong question. It's the system that should be asked about. The correct question then becomes, "Does the system as a whole understand Chinese?" The claim then is that the system understands Chinese, although the man clearly does not.

Searle's reply is simple. Let's change the thought experiment such that the man internalizes the rules and "bits of paper." The man will then have all of the rules of the program internally. With this change, the system is fully contained within the demon. "There isn't anything at all to the system that he does not encompass. 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."[49]

The Robot Reply

The Robot Reply attempts to add something to the system. The problem, according to this reply, is that the computer doesn't actually have the right kind of causal commerce with the world. If we adjust the thought experiment to take this into account, the intuition that there's no understanding wouldn't seem so plausible. So let's put the Chinese Room into a robot that can move around the world and interact with

[49] Searle, *Minds, Brains, and Programs*, p. 419.

the world in much the same way we do, by taking in sensory data, walking, moving about, hammering nails, eating, drinking, and so on.[50]

Searle's response to the Robot Reply is much like his response to the Systems Reply: change the experiment again. This time, let's imagine that, as before, the man in the room manipulates symbols according to the rules that constitute the given program. However, this time the input comes from whatever the robot uses to perceive its environment and the outputs go to whatever the robot uses to move around in the world. Further, the man transforming these inputs to outputs has no idea that this is what the inputs and outputs are for. Searle then claims, predictably, that the primary result of the thought experiment remains unchanged. The man *still* doesn't understand the Chinese symbols. The man in the room is in effect a homunculus of the moving, walking, talking robot, yet remains oblivious to the import of his task. He doesn't understand what these Chinese symbols are supposed to refer to. He still lacks intentionality. Further, Searle claims that the Robot Reply is admitting that something more than simply being the instantiation of a program is sufficient for understanding. The Robot Reply is an implicit admission that minds require a certain causal contact with the world. This is more than what's traditionally proposed by advocates of the Computational Theory of Mind.

[50] ibid, p. 420.

The Brain-Simulator Reply

The Brain-Simulator Reply proposes a more radical change in Searle's original thought experiment. Instead of the man in the room manipulating symbols by the use of rules, the room serves to simulate the brain of a Chinese-speaking woman. The brain gets inputs, simulates the neural activity of some this native Chinese speaker, and outputs the appropriate responses. Surely, if a native Chinese speaker has true understanding, this simulation of her brain would have the same understanding.

Searle takes the suggestion and runs with it. We're to imagine that this brain-simulation is occurring by way of a series of water pipes with attached valves. The man in the room, then, turns these valves according to some English instructions he is given and in this way derives output from input. This series of water pipes, obviously, serves as a simulation of a Chinese woman's brain (the movement of water serving as "firing of neurons" and so on). And, as is the theme, Searle claims that there remains no understanding within the man or even the system as a whole. Anticipating a Systems Reply to go along with the Brain-Simulator Reply, Searle claims that it would do no good to make this move even here. He claims that "if we are tempted to adopt what I think is the absurd view that somehow the *conjunction* of man *and* water pipes understands, remember that in principle the man can internalize the formal structure of the water pipes and do all the "neuron firings" in his imagination."[51]

[51] ibid, p. 421.

The Combination Reply

The final "traditional" reply to consider here is the Combination Reply. Searle here anticipates a response that essentially puts together the Systems Reply, the Robot Reply, and the Brains-Simulator Reply into one thought experiment. We're to

> Imagine a robot with a brain-shaped computer lodged in its cranial cavity; imagine the computer programmed with all the synapses of a human brain; imagine that the whole behavior of the robot is indistinguishable from human behavior; and now think of the whole thing as a unified system and not just as a computer with inputs and outputs. Surely in such a case we would have to ascribe intentionality to a system.[52]

Searle claims that we might be warranted in ascribing intentionality to such a robot, yet as soon as we're made aware of the innards of the robot and no longer *need* intentional terms to describe its behavior, we would abandon any such intentional talk. "If we knew independently how to account for its behavior without such [intentional] assumptions, we should not attribute intentionality to it, especially if we knew it had a formal program."[53]

Hofstadter & Dennett's Analysis of the Chinese Room

In *The Mind's I*, Douglas Hofstadter and Daniel Dennett give an analysis of Searle's original thought experiment. They analyze the Chinese Room as an "intuition pump," a story designed to elicit a certain intuition. In this case the intuition concerns how the human mind works (or rather, doesn't work). They find at least five different variables of

[52] Ibid.
[53] Ibid.

this thought experiment that can be changed to render varying intuitions. Here are the five variables:

> 1. The material, or "stuff", out of which the calculations are to be performed. In varying thought experiments, Searle has used water and pipes (against the brain-simulator reply), cups and balls, and toilet paper and stones. In the original Chinese Room, he uses "bits of paper".
> 2. The level at which the program simulates the human brain, from the level of the atom to the level of psychological representations.
> 3. The size of the entire simulation. It could take place in the size of a regular room (a la the original CR), or perhaps inside a human head.
> 4. The size of the demon (or agent) carrying out the simulation. The original CR uses a normal-sized monolingual English speaker with presumably normal intelligence (but perhaps with superhuman diligence).
> 5. The speed at which the demon works. Of course, this could be the speed of light or as a slow as you like.[54]

And here are Hofstadter and Dennett's knob settings for Searle's original thought experiment:

> Knob 1: papers and symbols
> Knob 2: concepts and ideas
> Knob 3: room size
> Knob 4: human-sized demon
> Knob 5: slow setting (one operation every few seconds)[55]

Most importantly, Hofstadter and Dennett also claim that there's one other variable that's not quite a perfectly-tweakable "knob." They note that, in the original thought experiment, we're asked to take a certain point of view, that of the man in the Chinese Room, performing simple symbolic manipulations without any sense of the meaning of his task. They also claim that Searle, in the original CR, "is insistent…that we see this

[54] Modified from Hofstadter and Dennett, Mind's I, p. 376.
[55] Hofstadter and Dennett, Mind's I, p. 376.

experiment only from the point of view of the demon."[56] Essentially, Searle's move is to convince the reader that there's one, and only one, good viewpoint to take when trying to find understanding of Chinese – the demon's viewpoint. Hofstadter and Dennett, however, insist that the point of view of the system is also a legitimate point of view, and further it's not so obvious that the system as a whole doesn't understand Chinese. This amounts, of course, to the Systems Reply.

I'd like to extend this analysis further and claim that Hofstadter and Dennett's five knobs of the Chinese Room serve as something like independent variables of this thought experiment, whereas the point of view that one is likely to take is something like a dependent variable. That is, the point of view that one is inclined to take in trying to find understanding largely depends on just what aspects of the thought experiment are emphasized and just what aspects are deemphasized or glossed over. When we turn the knobs, other aspects can come to the fore and change where we're likely to look for understanding. In the original Chinese room, the agent (or demon) carrying out the simulation is a normal-sized human being. We're obviously inclined (somehow) to take the perspective of those similar to us, and there's nothing in the original thought experiment as familiar to the imaginer as another typical human being. So the person within the room is emphasized and brought to the foreground. On the other side, the way the program itself is presented serves to mask, or deemphasize, the crucial role that the set of formal manipulations plays in the context of the room. This program is complex enough to pass the Turing Test, and we're told that this immense complexity

[56] ibid, p. 377.

and organization is contained in some "pieces of paper." Hofstadter and Dennett echo this concern:

> At the outset, the reader is invited to identify with [the demon] as he hand-simulates an existing AI program that can, in a limited way, answer questions of a limited sort, in a few limited domains [Schank's program]. Now, for a person to hand-simulate this, or any currently existing AI program—that is, to step through it at the level of detail that the computer does—would involve days, if not weeks or months, of arduous, horrendous boredom. But instead of pointing this out, Searle—as deft at distracting the reader's attention as a practiced magician—switches the reader's image to a hypothetical program that passes the Turing test! He has jumped up many levels of competency without so much as a passing mention. The reader is again invited to put himself or herself in the shoes of the person carrying out the step-by-step simulation, and to "feel the lack of understanding" of Chinese. This is the crux of Searle's argument.[57]

The point of the view that the reader is invited to take is formed by both the inclusion of a conscious human agent and the lack of emphasis in just what is involved in a program that can pass the Turing Test. When this point of view (the demon's) becomes the natural and obvious point of view to take, Searle's work is mostly done. The final step, looking for any understanding of Chinese, is bound to succeed. The man clearly doesn't understand Chinese because that's part of what Searle has the reader imagine in the first place. So, Searle's thought experiment relies crucially having the reader only take the viewpoint of the demon. My claim (with Hofstadter and Dennett) is that this is done by tweaking the knobs in a particular way (the way just outlined).

So, in light of this, let's look at the Systems Reply and Searle's response to it. The Systems Reply asks us to consider that the man in the room isn't the only candidate for

[57] ibid, p. 373-4.

understanding. If we realize the complexity of the program, and if we understand what all would be involved in passing the Turing Test, we would understand that the system itself, as instantiated by the man in the room *plus* the (vast number of) "bits of paper" *plus* whatever serves to deliver input and output, must have a wide range of abilities in virtue of its linguistic ability. This program must appear to (for example) remember previous bits of conversation, sense sarcasm, and be able to correctly use concepts like 'gender,' 'hole in one,' and 'virtual machine.' This program would likely even be able to give some response to the Chinese Room thought experiment itself. Further, once we notice that the demon's consciousness is largely irrelevant, all of this combines to undo exactly what Searle has done. It deemphasizes the role of the demon and emphasizes the complex nature of the program (and thus the entire system) itself. So, in response to this, what does Searle do? He keeps the demon as central to the thought experiment and proceeds to *further* deemphasize the role of the other parts of the system. Now the reader is invited to imagine that this program, complex as it is, is not only hand-computed by a single person, but is now perfectly contained within the memory of a single human agent. Again, here is emphasis on the mental nature of the demon and a large pushing-aside of a hypothetical program that passes the Turing Test. Searle, in response to the Systems Reply, keeps the focus on the demon and off the program.

The Chinese Gym

In light of this analysis, let's see what if anything is changed by replacing the Classical-style program with a Connectionist program. Searle has his own answer to Connectionist responses to the Chinese Room. In his 1990 article *Is the Brain's Mind a*

Computer Program, Searle modifies his original thought experiment again to take Connectionist models into account. His new thought experiment, the Chinese Gym, claims to add what Connectionists (in this case, Paul and Patricia Churchland) want:

> 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 in a connectionist architecture as described by the Churchlands, and *the 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.[58]

Searle goes on to claim that, while Connectionist models may provide certain advantages with respect to the prospects for Weak AI[59], the change in architecture makes no difference to his own argument against Strong AI. He claims that there's no difference between a parallel processor and its simulation on a serial processor. Since a parallel processor can be simulated with a serial program, both programs are computationally equivalent, and therefore there isn't any relevant difference between the two types of models.

So, what has Searle done here? Most obviously, Searle has us imagine a large number of typical humans (how many?) instead of the singular demon contained in the previous two setups. The Chinese Gym could backfire on this very point. If we try to imagine the number of people it would take to instantiate a Connectionist network that

---

[58] John Searle, "Is the Brain's Mind a Computer Program?," *Scientific American* 262, no. 1 (1990): p. 28. – italics are mine.

[59] Weak AI, according to Searle, is the notion that computers are merely a tool to use in the study of the mind (Searle, *Minds, Brains, and Programs*, p. 419).

passes the Turing Test, we *might* be inclined to take the point of view of the system, seeing the individual people as just so many automated pieces of a larger whole. Our inclination then would then be to look for understanding in the (behavior of) the system. This may result in an intuition that the system itself does have mental properties like understanding. However, the Chinese Gym, as Searle presents it, discourages such behavior. We're told that "there is no way for the system as a whole to learn the meanings of any Chinese words." In other words, don't look at the system. We're encouraged to take the point of view of any single person in the system. Searle again deemphasizes the role that program plays by (1) glossing over the fact that the Connectionist network in question needs to be complex enough to pass the Turing Test and (2) explicitly stating that the system doesn't understand the meanings of Chinese symbols. So, in response to a Connectionist critique of his original thought experiment, Searle multiplies the number of beings with which we naturally empathize. However, he states that "the outcome would be the same as having one man manipulate symbols according to a rule book." Why doesn't Searle use *this* thought experiment? Might this alternate thought experiment alter our point of view and thus the resulting intuition we're left with? My claim is that it will.

The Korean Room

Consider then The Korean Room. In this room, the demon (another monolingual English speaker) begins with a large book containing all the necessary information about a large number of nodes (activation thresholds, the placement of connections between

nodes, the weight values of those connections, and so on). This information constitutes the initial state of a Connectionist program. In addition to this, the demon is given a set of directions (in English) to determine when and in what way to change (or create, or remove) the weights between nodes. In other words, the demon has access to the learning rules of the Connectionist system. The demon receives input in the form of a set of initially-activated nodes. He then hand-simulates the spreading activation of the system resulting from the input (adding up the values of excitation and inhibitions, determining when a nodes has reached the threshold level, and so on). The demon also modifies the weights between nodes on the basis of the learning rules, also removing and adding nodes when necessary. Questions are then given as input to the room and output is given. In this way, the room passes the Korean version of the Turing Test.

As is probably clear, this is just the original Chinese Room with a Connectionist network replacing the Classical-style program. Both pass a version of the Turing Test and both feature a single, normal-sized human as the manipulator of the program. Also, both feature the program in the form of a book, or many "bits of paper." All the knobs of this thought experiment remain intact with the exception of one. Now the simulation is carried out in terms of the Connectionist's nodes instead of the semantically transparent symbols that characterize the traditional approach to AI. In other words, I've only tweaked knob 2 a bit.[60]

[60] Of course, what's written on the "bits of paper" must change too. Thus knob 1 is also moved somewhat.

How does this move from a Classical system to a Connectionist system change the results of our thought experiment? In the Chinese Room, the imaginer's task is to look for any understanding of Chinese. Naturally, the first and most obvious place to look is in the demon (given that the demon is a normal-sized person and the complexity of the program is adequately glossed over). The demon receives sets of Chinese symbols on pieces of paper. The demon looks at the symbols and clearly doesn't understand their meanings (though he does differentially respond to syntax). Thus, we ask "Does the man in the Chinese Room understand the symbols that come into the room?" Clearly not. Does something similar occur in the Korean Room?

No, such a question cannot be asked. The demon doesn't receive semantically transparent elements, like those discrete elements that make up Chinese or Korean. The demon only learns which nodes in his Connectionist network are activated from the outside. Nothing within the input, the calculations, or the output is composed of semantically transparent symbols. The entire process, as from the point of view of the demon, occurs within the subsymbolic paradigm. Thus the question of whether or not the demon understands the relevant language (in this case, Korean) becomes irrelevant. Where the demon in the Chinese Room sees just some apparently meaningless marks, the demon in the Korean Room doesn't even get that. It's of no use to take the point of view of the man in the room if we're looking for understanding. With the change from a symbolic program (the Classical program) to a subsymbolic program (the Connectionist program), the point of view that Searle wants us to adopt immediately becomes less appealing. Of course, Searle's original thought experiment could have the demon

implementing a Turing Machine or some other computational device that doesn't have semantically transparent units. However, Searle's original thought experiment *does* contain semantically transparent units (at least in the input and output of the system). This fact plays to Searle's desired outcome.

David Chalmers makes this point in a slightly different way. He rightly notes that Searle's argument applies to both Classical and Connectionist systems because "Searle's argument is aimed at computational systems in general, and connectionist systems are computational systems."[61] Both kinds of systems perform computational procedures on formally specified elements. However, the difference comes when we notice that, in Classical systems, computations take place on elements that are also the elements of representation. This is what it means to say that the system is semantically transparent. In Connectionist systems, the level of representation takes place above the level of the fundamental units of computation. As Chalmers states, "In a symbolic system, the computational level coincides with the representational level. In a subsymbolic system, the computational level lies beneath the representational level."[62] With the distinction in place, Chalmers claims that we give up nothing when we see that the computational elements that the demon works on have no semantics. This is built into, and defines, the subsymbolic paradigm. In the Korean Room, the fact that the demon doesn't know what

[61] David Chalmers, "Subsymbolic Computation and the Chinese Room," in *The Symbolic and Connectionist Paradigms: Closing the Gap*, ed. John Dinsmore (Hillsdale, NJ: L. Erlbaum Associates, 1992), p. 37.

[62] Chalmers, Subsymbolic Computation, p. 35 – Chalmers proceeds to reword the distinction such that it only talks about "semantically interpretable" elements, so as not to beg the question against Searle's argument that Semantics cannot arise out of Syntax alone. The original formulation will serve my purposes here.

the nodes mean isn't a problem. We're inclined to look elsewhere for understanding (semantics) to emerge. This is only plausible on the level of the system (or some proper subset of the system that includes more than just the demon).[63]

Of course, this is only the first step in emphasizing the program and deemphasizing the demon (i.e. undoing Searle's work). We could go much further. We could start by removing the mentality of the demon. For the system to work correctly, all that's needed in the role of the demon is the ability to read, write, and manipulate the elements of the system properly.[64] With a connectionist system, this is largely a matter of calculating when nodes fire and adjusting weight values. We could also emphasize the fact that a connectionist program that passes the Turing Test would in some sense need to have a "personality." This output would consist in a possibly unique style of answering questions. It may claim that it experienced qualia or even give its own intuitions about the Chinese Room. These claims might be given more plausibility if this system were imbedded in a robot.[65] Tweaking the knobs changes where (and how) we're likely to look for mental features or properties. Doing so in one way (the way I've suggested) adds more emphasis to the proper topic of conversation – the program(s). Doing so the other way (Searle's way) emphasizes the most irrelevant feature of the thought experiment: the mental nature of the demon(s). By simply replacing the

[63] I should note here that the brand of Connectionism I've assumed in this paper is one in which talk of emergent representations and Connectionism as computational is taken as legitimate. A brand of Connectionism in which these do not hold would strictly serve my purposes. However, using a Connectionism with these commitments serves to highlight the claim that we only need to change Searle's thought experiment slightly in order to modify the resulting intuitions.
[64] I'm using 'read' and 'write' here in Haugeland's formal sense, explained in Chapter 1.
[65] Could the "sensory data" be cleanly split from the rest of the system in the new Connectionist room?

Classical system with a Connectionist system, we've gone a long way towards shifting our point of view away from the demon.

What Have I Shown?

None of the above goes to show either the promise or shortcomings of *any* model of mind. Rather, the point is that Searle's 'intuition pump' doesn't generate similar intuitions across different computational architectures. This is largely the result of how the original thought experiment works: through a convenient emphasis and de-emphasis of what the reader is asked to imagine. The Korean Room is merely one way to tweak the knobs such that Searle's thought experiment loses its strong intuitive appeal. Other, more radical, changes to the Chinese Room do this to a larger extent. The Korean Room helps us see that the original Chinese Room represents a very particular arrangement of the relevant variables – one peak on the landscape of possible intuition pumps. Instead of highlighting places far from the Chinese Room on the landscape, I've chosen to map out a place close by, in order to survey the local area and see how quickly our intuitions can change with relatively small changes in what we're asked to imagine. We have reason for pause when our judgments can be manipulated so easily.

REFERENCES

Chalmers, David. "Subsymbolic Computation and the Chinese Room," in *The Symbolic and Connectionist Paradigms: Closing the Gap*, ed. John Dinsmore (Hillsdale, NJ: L. Erlbaum Associates, 1992), p. 25-48.

Clark, Andy. *Microcognition: Philosophy, Cognitive Science, and Parallel Distributed Processing.* Cambridge, MA: Bradford Books/MIT, 1989.

Daniel Dennett, *The Intentional Stance* (Cambridge, MA: MIT, 1987).

Fodor, Jerry. *The Language of Thought.* New York: Crowell, 1975.

Gardner, Howard. *The Mind's New Science: A History of the Cognitive Revolution* (New York: Basic, 1985.

Haugeland, John. *Artificial Intelligence: The Very Idea.* Cambridge, MA: MIT, 1985.

Haugeland, John. "What is Mind Design?." In *Mind Design II: Philosophy, Psychology, Artificial Intelligence*, edited by John Haugeland, 1-28. Cambridge, MA: MIT, 1997.

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

Horst, Steven, "The Computational Theory of Mind", *The Stanford Encyclopedia of Philosophy* (Spring 2011 Edition), Edward N. Zalta (ed.), URL=http://plato.stanford.edu/archives/spr2011/entries/computational-mind/.

Marvin Minsky, "A Framework for Representing Knowledge," in *Mind Design II: Philosophy, Psychology, Artificial Intelligence*, ed. John Haugeland (Cambridge, MA: MIT, 1997), 111-142.

McClelland, James and Kawamoto, Alan. "Mechanisms of Sentence Processing: Assigning Roles to Constituents of Sentences." In David Rumelhart, James McClelland and the PDP Research Group, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition* (Cambridge, MA: MIT, 1986), p. 272-325.

Newell, Allen and Simon, Herbert. "Computer Science as Empirical Inquiry: Symbols and Search." *Communications of the ACM* 19, no. 4 (1976): 113-26.

Rumelhart, David. "The Architecture of Mind: A Connectionist Approach," in *Foundations of Cognitive Science*, ed. Michael Posner (Cambridge, MA: MIT, 1989). Reprinted in *Mind Design II: Philosophy, Psychology, Artificial Intelligence*, ed. John Haugeland (Cambridge, MA: MIT, 1997).

Rumelhart, David and McClelland, James. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition.* Cambridge, MA: MIT, 1986.

Searle, John. "Minds, Brains, and Programs." *Behavioral and Brain Sciences* 3, (1980): 417-457.

John Searle, "Is the Brain's Mind a Computer Program?," *Scientific American* 262, no. 1 (1990): p. 26-31.

Schank, Roger and Abelson, Robert. *Scripts, Plans, Goals and Understanding.* Hillsdale, NJ: L. Erlbaum Associates, 1977.

Smolensky, Paul. "Connectionist Modeling: Neural Computation / Mental Connections," in *Mind Design II: Philosophy, Psychology, Artificial Intelligence*, ed. John Haugeland (Cambridge, MA: MIT, 1997), p. 233-250.

VITA

Michael Wayne Carver

Candidate for the Degree of

Master of Arts

Thesis: CONNECTIONISM, CHINESE ROOMS, AND INTUITION PUMPS

Major Field: Philosophy

Biographical:

Education:

Completed the requirements for the Master of Arts in Philosophy at Oklahoma State University, Stillwater, Oklahoma in May, 2014

Completed the requirements for the Bachelor of Arts in Philosophy at Oklahoma State University, Stillwater, Oklahoma in 2012

Experience:

Graduate Teaching Assistant for Oklahoma State Philosophy Department, August 2012 - May 2014

Professional Memberships: None.