

STABILITY AND PLASTICITY: CONSTRUCTING COGNITIVE AGENTS

A THESIS
SUBMITTED TO THE DEPARTMENT
OF COMPUTER ENGINEERING
AND THE INSTITUTE OF ENGINEERING AND SCIENCE
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE

By
Öge Bozyiğit
September, 2006

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Asst. Prof. Dr. David Davenport

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Prof. Dr. Fazli Can

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assoc. Prof. Dr. Ferda Alpaslan

Approved for the Institute of Engineering and Science:

Prof. Dr. Mehmet Baray
Director of Institute of Engineering and Science

ABSTRACT

STABILITY AND PLASTICITY: CONSTRUCTING COGNITIVE AGENTS

Öge Bozyiğit

M.S. in Computer Engineering

Supervisor: Asst. Professor Dr. David Davenport

September, 2006

The AI field is currently dominated by domain-specific approaches to intelligence and cognition instead of being driven by the aim of modeling general human intelligence and cognition. This is despite the fact that the work widely regarded as marking the birth of AI was the project of creating a general cognitive architecture by Newell and Simon 1959. This thesis aims to examine recently designed models and their various cognitive features and limitations in preparation for building our own comprehensive model that would aim to address their limitations and give a better account for human cognition. The models differ in the kind of cognitive capabilities they view as the most important. They also differ in whether their foundation is built on symbolic or sub-symbolic atomic structures. Furthermore, we will look at studies in the philosophy and cognitive psychology domain in order to better understand the requirements that need to be met in order for a system to emulate general human cognition.

Keywords: Cognitive architecture, human-level intelligence, knowledge representation, prediction, spreading activation.

ÖZET

SAĞLAMLIK VE ESNEKLİK: BİLİŞSEL AJANLAR YARATMAK

Öge Bozyiğit

Bilgisayar Mühendisliği, Yüksek Lisans

Danışman: Yrd. Doç. Dr. David Davenport

Eylül, 2006

1959'da Newell ve Simon'nin geliştirdiği genel insan zeka kabiliyeti modelleme projesi yapay zeka ile ilgili çalışmaların başlangıcı olarak biliniyor. Buna rağmen, günümüzde yapay zeka alanında genel beyin kabiliyeti modelleme yöntemleri değil, dar bir alanda spesifik zeka problemleri için özel çözüm üreten yöntemler hakim. Bu tezin amacı genel bilişsel fonksiyonlar için yeni tasarlanmış modellerin özelliklerini analiz edip, onların sınırlarını hedef alan bir modelin araştırma zeminini oluşturmak. Modelleri birbirinden ayıran özellik temel bilişsel kapasitelerinin hangisinin daha üstün veya yaygın sayıldığı. Temel farkların bir diğeri ise, bazılarının sembolik altyapıdan oluşturulmuş, bazılarının ise daha ilksel mekanizmalardan kurulmuş olmaları. Bunların yanında, genel bilişsel kapasitelerinin gerekçelerini daha da somutlaştırmak için bu alanda yapılan felsefik ve psikolojik araştırmalar da tezin konusu dahilinde.

Anahtar Sözcükler: beyin fonksiyon modelleme, bilişsel dizge, mantık modelleme, genel yapay zeka, tahmin kabiliyeti

ACKNOWLEDGMENTS

I would like to thank my supervisor Dr. David Davenport for his support and encouragement throughout my study and for the insightful back-and-forths we have had.

I would like to thank my parents and my brothers Emre and Arda for their encouragement during this arduous process and, occasionally, for some of the distractions. I also would like to thank Alper Tarimci for the helpful conversations we have had regarding the thesis.

Table of Contents

Abstract	iii
Özet	iv
Acknowledgements	v
Table of Contents	vi
List of Figures and Tables	ix
1 Introduction	1
1.1 Constructing Cognitive Agents Today.....	2
1.2 Required Characteristics of a Cognitive Agent.....	4
1.3 How to Build a Cognitive Agent.....	9
1.4 The Importance of Philosophy and Cognitive Psychology.....	11
1.5 Organization of Thesis.....	12
2. Philosophical Foundations for Cognitive Agents	13
2.1 Computationalism in General.....	14
2.2 Connectionism.....	16
2.4 Symbol Grounding.....	19
2.5 Modularity.....	21
3. Cognitive Psychology and Neuroscience	26

3.1 Mirroring the World.....	27
3.1.1 Mental Simulation and Mechanical Reasoning.....	27
3.1.2 Goals and Imitation.....	28
3.2 Dissociation Versus Association.....	29
3.2.1 Theory of Mind Module.....	29
3.2.2 Reward and Attention.....	30
3.2.3 Short- Term and Long-Term Memory.....	32
3.4 General Frameworks.....	33
3.4.1 Level of Consciousness Model.....	33
3.4.2 Connectionist Models of Biased Competition in Cognition.....	36

4. Computational Models of Cognition **40**

4.1 The Emotion Machine and Critic-Selector Model.....	40
4.2 Non-Axiomatic Reasoning System (NARS).....	46
4.3 SHRUTI.....	51
4.4 Novamente Engine (NE).....	57
4.5 Confabulation Theory (CT).....	61
4.6 SINBAD System.....	69

5. Analysis of Cognitive Architectures **76**

5.1 Non-Axiomatic Reasoning System (NARS).....	77
5.2 SHRUTI.....	80
5.3 Novamente Engine (NE).....	81
5.4 Confabulation Theory (CT).....	82
5.5 SINBAD System.....	84
5.6 Summary and Implementation Pointers	86

6. A Look at the Inscriptor Model **89**

7. Conclusion	98
References	100
Further Reading	103

List of Figures and Tables

3.1 Level of Consciousness Model	35
3.2 Model of Pre-frontal Cortex Control	38
4.1 Guiding Principle for the Emotion Machine	41
4.2 Levels of Critics in EM-One	42
4.3 Meta-Critic Control	44
4.4 A SHRUTI Query	52
4.5 Primitive Control Mechanisms in SHRUTI	55
4.6 A Map in a Novamente Engine	58
4.7 A Single Module in Confabulation Theory	63
4.8 A Winner-takes-all Process	64
4.9 A Lexicon Module	66
4.10 A Knowledge Link	67
4.11 Pyramidal Cell in SINBAD	71
4.12 A Layer of 32 Pyramidal Cells	74
5.1 Comparisons of Models	86
6.1 Inscriptor parent and child nodes	89
6.2 Mixing Abduction and Deduction	91
6.3 Lateral Repetition Testing	94
6.4 Simultaneous Lateral Repetition Testing	95
6.5 Parental Node Repetition Testing	96

Chapter 1

Introduction

“Have you ever stopped to think, and forgot to start again?”

--Winnie the Pooh

The ability we have to let our mind wander around without worrying about losing our memory, without finding ourselves unable to do simple arithmetic, or finding ourselves suddenly unable to tell the difference between a cat and a dog, is something many people take for granted. In reality, this is the sort of thing that happens to many AI systems that try to mimic human cognition. If one substitutes stability for plasticity, the system is unable to adapt to a rapidly changing environment. If one substitutes plasticity for stability, it is unable to achieve the stable states necessary to achieve goals and function as a consistent unit. Thus in exasperation many AI researchers chose to keep the two apart, some preferring to only deal with “low-level” plastic functions, while others preferring to only deal with “high-level” stable functions.

When someone is handed a paper with three dots that are drawn an equal distance apart from each other but not on the same axis, he will inevitably see a triangle. He will have mentally filled in the lines even though they are not there. What has happened appears to have been a result of visual processing, which would mean it was accomplished by a relatively “low-level” cognitive function. In other words, what has transpired has been automatic and appears not to involve “high-level” cognitive capabilities such as those involved in reasoning and active concentration. However, this “filling in” function is also present in high-level reasoning: your conscious mind is constantly filling in the blanks when faced with a dearth of information, whether it’s

trying to understand the main point of what someone is telling you, or trying to solve an incomplete equation. If the underlying mechanism is the same, what is the sense of keeping “low-level” and “high-level” cognitive functions independent of each other? The fact that one can simply add an extra dot randomly in the drawing and thus force conscious awareness into the process just shows how they are interlinked. There is a “low-level” plasticity that is able to extract various patterns from the sensory input, and there is a “high-level” stability that imposes completeness when faced with partial data. These cognitive aspects, it would seem, should ultimately be merged under a general approach for modeling human cognition.

1.1 Constructing General Cognitive Agents Today

Nowadays trends in AI focus on creating domain-specific models for specific human cognitive abilities or tasks and it seems that the aim of creating a general cognitive framework has been sidelined by much of the community. As a result new subfields are created for tools that often cannot singularly accomplish unified human cognition. The idea that these various independently-developed domain-specific models can be merged without any deficiencies also seems unrealistic. In order for a model to be cognitively plausible it must account for seemingly conflicting cognitive abilities, such as being goal-oriented while at the same time having a highly interruptible "attention span."

In 1959, the first attempt at building something resembling a cognitive architecture was made by Newell and Simon [1]. The system was called the General Problem Solver, and despite the fact the system was tuned to handling problems that did not involve interaction with a complex environment, like Towers of Hanoi or chess, the underlying motivation for the project was to model human cognitive capabilities. In fact, the work of Newell and Simon would turn out to have a significant impact on later studies of cognitive psychology even though it was not applied to a wide spectrum of tasks that were representative of the complexity of human cognition.

For many, Newell and Simon’s work marks the birth of the AI field. However, the fact that their own aim was to provide a *general* cognitive architecture is lost on many in field, or simply ignored. The reality is that domain-specific AI research dominates, and

the aim of providing general cognitive architectures either seems too vague for some or too demanding to be pursued by the majority of researchers.

However, the goal to achieve artificial general cognition (AGC) has been around for quite some time even if it hasn't dominated the spotlight, and the interest for general cognitive architectures that led to the birth of AI is still present to sustain it. Furthermore, in recent years there has been an increase in activity in this line of work. There now are several currently developing frameworks and models available that can be examined and reviewed. Among these there are a few academic labs dedicated to the project, such as the Emotion Machine (EM-One) at MIT and SHRUTI at Berkley, as well as models developed in the private sector such as the Novamente Engine. The guiding principles also vary considerably between various models and frameworks. There are some like the Optimal Ordered Problem Solver that have a heavily algorithmic approach, whereas many have a connectionist approach and differ in whether their building blocks are symbolic or sub-symbolic in nature.

One of the reasons that the development of AGC models have picked up speed is because of the recent substantial amount of data coming out of fields of cognitive psychology and neuroscience, and the fact that modeling certain cognitive capacities based on those observations have become possible in light of increasing computational power. However, these discoveries are also fueling domain-specific approaches to a large extent because of the difficulty of merging the vast amount of data from different cognitive domains and interpreting them under a unified framework.

Usually, the domain-specific approach to developing human intelligence is argued as follows: "This system is far more accurate for this domain of intelligence than a general system that attempts to work for all domains of intelligence. Furthermore, all one needs to do in order to create a general system from a collection of domain-specific systems is simply plug them together." The problem with this approach is that it undervalues the task of "plugging them together." In order for one to connect domain-specific modules together, one has to find a common, cognitively plausible way for them to communicate with each other. Since, however, domain-specific approaches are usually developed with integration with other domains low on the priority list, filtering and translating information that can be used by other modules becomes increasingly difficult.

On the other hand, general approaches are often criticized for not accounting for the apparent domain-specific or modular behavior of human cognition. For example, detractors point out that the vision and linguistic domains often perform without any interference from other domains or higher-level cognition. Hence, some claim a dogmatically general approach risks being irrelevant because general human cognition itself engages in domain-specific behavior.

It is the assumption of this thesis that domain-specific approaches are utilized best when they are pursued under the umbrella of a general approach to human cognition, for it is this umbrella that will help ensure that such approaches will develop with cognitive integration in mind. Furthermore, the long-term benefits of a viable general approach can also be of a practical nature. When a new domain is discovered, a lot of time and effort is put into developing domain-specific methods from scratch. If a general approach to a collection of domains existed, the relationship between existing domains and the new domain could be explored and exploited while the system is on-line.

1.2 Required Characteristics of a Cognitive Agent

The ability to interact with different environments, adapt to changing goals, and integrate a wide range of information, is as much a practical advantage over rigid domain-specific AI systems as it is an essential component of general cognition. Whereas domain-specific systems are reverse engineered by first determining a certain problem and then creating a tailor-made solution, a general cognitive architecture is developed by anticipating a wide range of domains and is built in such a way as to establish adaptive relationships between them and any new domain that the agent faces at a later time.

The Seed Concept In order to accomplish this, cognitive agents are born with a “seed.” This refers to the set of basic abilities, structures, and units that the agent is initialized with that essentially allows it to learn autonomously without having to be hardwired for a specific purpose later on. It is this essential step of bootstrapping that allows the agent the adaptability and flexibility that is required in order to accomplish human-order cognitive tasks, and it is also this critical component that makes or breaks the general intelligence endeavor. From this seed and continuous interaction with its

environment, the agent will incrementally learn and develop context-specific routines when it faces situations it has not encountered before. In domain-specific approaches, there is usually very little difference between the initial seed and the developed system. This is mainly because the seed is given everything it might require at the beginning and will not encounter many unforeseen situations by definition. The general approach has a relatively small seed compared to its developed state, and acquires most of its context-dependent knowledge through learning and self-improvement.

Goal-directedness and Adaptability Like its specialized counterpart, general cognitive agents have to be goal-directed. However, because it has to be adaptive in a way the other does not, it must also have a method of interrupting its current goals and replacing them with others as it responds to environmental output. This is yet another case of the stability-plasticity problem, where one desires the agent to be stable enough to take deterministic steps towards fulfilling a goal without being distracted by noise, but plastic enough to determine that a certain batch of input is not noise but relevant data which requires the agent to alter its current goals. Therefore adaptability refers to two things: the attention component related to establishing relevant goals given the context, and the ability to learn new problem-solving routines autonomously.

Basic Pattern Processing and Contextual Experience Since the system has to determine whether a certain collection of data is relevant not just for one context, but an entire range of contexts, it may have to develop different ways of representing the environment in order to prevent relevant data being mistaken for noise. Processing patterns is of course a vital component of establishing relevance, and it is usually an assumption of many general approaches that basic pattern processing methods fuel many aspects of high-level cognition such as abstract thought and logical reasoning.

Automated Attention In most AI methods, environmental inputs are tagged by a supervisor in order to indicate the importance or relevance of the incoming information. Without a similar ability to what humans have when they are able to switch their attention from one object to the other by an established internal criteria, the general cognitive agent will be severely handicapped. Since it seems unlikely that the supervisor can tag all inputs in real-time, this method will essentially convert the agent's continuous

real-time processing to discrete-batch processing, and in turn limit its ability to make use of the most amount of relevant data.

Continuous Sensory Data Real-time processing exposes the system to large amounts of noise, but it is vital for the system to be exposed to an abundance of information in order to develop an adequate representation of various contexts and acquire the motivation for abstraction that is a characteristic of high-order cognition.

Top-Down and Bottom-up Supervision When agents interrupt their current behavior because of newly processed environmental behavior, the actions that are taken and goals that are established afterwards can be described as being data-driven since they have resulted from a bottom-up flow of information. Alternatively, high-level goals that are established by the agent and are directed toward lower-level structures represent the agent's ability to self-supervise its learning and "thinking" in general. A general cognitive agent must be able to utilize both the learning that results from data-driven bottom-up supervision, and top-down self-supervision.

Integrated Knowledge Structure Despite acquiring knowledge that is relevant to different contexts, it seems necessary for general cognitive agents to have a highly integrated knowledge structure that is capable of self-organization. An agent that is primed towards adaptive learning is going to be hindered if its knowledge representation structure does not share the same characteristic of flexibility. Having a seed knowledge structure that is incapable of altering *the way* it grows based on experience with a continuous environment will put this flexibility in danger.

Flexible Conceptual Understanding Concepts hold an important place in cognitive agents because they are the building blocks of high level learning. However domain-specific methods that aim to create logical reasoning agents, for example, often hardwire concepts into the agent and inject concepts that have rigid boundaries that do not change over time and experience. In reality, however, concepts are much more flexible, their boundaries often changed and dimensions are added and updated as they gain additional meanings in different contexts. Therefore, concepts in a general cognitive agent should exhibit this apparent characteristic of concepts in human cognition. They should be contextually grounded in experience, expanding or contracting as the agent learns the boundaries that facilitates its interaction with the environment.

A World Model The world model of a general cognitive agent is crucial to its development; the sensory, predictive, and decisive characteristics that connect the agent to the world join together to form its manipulative ability. The sensory aspect connects elements in the external reality to the agent's internal structures. The predictive aspect determines what future sensory inputs may be. The decisive aspect uses the two previous characteristics in order to select an action that can achieve the desired goals. The manipulative aspect of the agent is what predicts a possible future state and creates and implements a set of actions that bring about such a desirable state. The issues that are involved in these aspects of the agent's world model are issues regarding time, causality, goals, and invention.

A Model of Self Self-modeling allows agents to predict their own behavior and not just the environment or other agents' behaviors. In a way, however, the agent also has to realize that it is also part of the environment. Therefore by trying to model and predict the environment, it should realize that in order to do that accurately it may have to model and predict its own behavior. Introspection can then be initially represented as a utility function that can help achieve the agent's immediate goals. Thus structures in the agent's seed should allow the agent to develop the recursive modeling that would be needed in order to establish an environment/self distinction.

Learning with Sensory Absence In human cognition, absence of certain sensory data is often taken as positive evidence for a specific state. For example, when you are listening to music and your mind starts to wonder off you will be alerted if the music suddenly stops. Here the absence of certain sensory data (music) has alerted you to a new state in the environment. This poses a difficulty because it suggests that if a system is to account for such behavior, it must not only consider the sensory data it does receive but also the sensory data that it *doesn't* receive. Whereas the sensory data the system receives from the environment (the explicit presence of certain phenomena) is finite, the sensory information that it does not receive is *infinite*. In other words, the list of everything that is absent cannot be counted. However, it's apparent that human cognition does not engage in such infinite awareness, and therefore an AGC agent must likewise be alerted to absent phenomena, and likewise limit such awareness to the finite through self-supervision.

High-level and Low-level Temporal Understanding How time is represented in a general cognitive agent may also have some far-reaching consequences. Since human cognition appears to have an intuitive understanding of time, just strapping the agent with an internal explicit clock may not explain the nature of how humans come to determine the temporal values of abstract concepts such as “slow” and “fast.” If the brain does use a system clock, high-level cognition seems to have no access to it and instead attributes relative temporal values to events, whether an event came before or after—or was longer or shorter—than another event. Hence, one needs to determine what sort of relationship this understanding of time has with the understanding of time that is able to govern the movements of the physical body. Furthermore, the perception of synchronization also seems to be susceptible to change. Events that are perceived to occur at the same time, in reality may not have occurred at the same time. What sort of mechanisms determine whether one will categorize two things as occurring simultaneously, or whether he will take one event to have occurred before the other?

Adaptable Time Constraints In domain-specific AI, the goal often is to reduce the time it takes the system to solve one task because in a domain-specific environment tasks are generally of the same type, and therefore a reduction in the time cost of one task means that the reduction can be applied to all tasks. However, for a system that aims to simulate general cognition, the rule is that tasks are often *not* of the same type and therefore one can not simply focus on trying to reduce the time cost of a task without taking into account what type of task it is. Therefore, when the domain-specific AI agent moves out of its domain, the uniform time constraint it has put on addressing tasks may render it ill-suited for its new surroundings. Hence, the AGC agent, unlike its domain-specific AI counterpart, will be able to adjust its time allocations for goals based on an understanding of context that has resulted from interaction with a complex environment.

Different Levels of Access to Sensory Data One should also take note that certain characteristics of human cognition that seem like disadvantages may in fact be an important part of developing high-level cognition. Being unable to access certain raw sensory data may in fact be a result of the abstraction that is needed for high-level cognition. There is evidence that certain sensory data that cannot be consciously accessed is stored in the brain, thus perhaps creating a difference between functional forgetting and

the erosion of stored sensory data. When developing general cognitive agents then, one should be careful of determining whether certain surface limitations are in fact limitations, or whether they are factors that enhance general cognition. There has been much discussion on the difference between procedural knowledge and declarative knowledge, mostly concerning the largely unconscious nature of the former and the conscious nature of the latter. Infants, for example, have been observed passing certain number tests that they later fail at as preschoolers. This, it has been suggested, results because the knowledge that was acquired in the early learning phase was declarative and could be consciously accessed, and once the task could be actually performed successfully the required knowledge became procedural and the child could no longer have conscious access to it.

1.3 How Cognitive Agents are Built

There are several topics not related to the engineering aspect of general cognition that are crucial to the success of a general cognition system. The system will only be as good as the theory of knowledge and learning that it is built on. For example, a general cognitive system that holds that conceptual representation is vital to high-level cognition will most likely perform differently than a system that rejects such an assertion. Furthermore, one has to establish what types of structures allow concepts to become thoughts. It is difficult therefore to see how one can bypass such philosophical and psychological considerations when developing a general cognitive agent.

Reverse Engineering vs. Functional Requirements Reverse-engineering is at times an attractive way to build cognitive agents. For example, one can note that we already have an instrument that achieves general intelligence, that is, the brain. Thus if we try to reverse-engineer such a system we will be able to create something that does exactly what it does, at least that's how the intuition goes. The problem, of course, is that we do not know the functions of all the components in the brain, nor is it easy to determine whether a certain function that is discovered for a component of the brain is the only function for that component. Thus, it is sometimes more useful if one outlines what the functional requirements for general intelligence are, and builds a system that is

designed to meet those requirements, instead of depending on the incremental lighting of a black box (the brain). If one believes evolution is not the only path to cognition, nor the most optimal path, then it is common sense not to abide entirely to its design. The aim is to find heuristics that can establish relationships between high-level cognitive functions and low-level mechanisms, and also enable the agent to engage in meta-cognitive heuristics that can incrementally improve those existing relationships. The goal of a general artificial cognition should not depend on mirroring the time it takes a human to develop adult cognition, as much as it depends on mirroring the developmental stages.

Broad Cognitive Mechanisms vs. Raw Performance Another characteristic of domain-specific methods is their obsession with one-upping the performance of the previous system on a certain task. If, for example, a domain specific approach is able to perform a three or four percent optimization on the previous approach, it may be considered significant. A general approach should not be obsessed to such an extent with raw performance, and should focus on quality instead of quantity. For example, one quality measure would be how it is able to perform under limited sensory input. Even though a general cognitive agent should be primed for interaction in a real-time environment, reducing the complexity of sensory data at times may in fact give clues as to how well the agent can in fact learn autonomously and how well its internal mechanisms utilize the data. Just as deaf or blind people are able to be highly intelligent, for a general intelligence agent the reduction of raw sensory data should not completely prevent the agent from achieving a high degree of intelligence. Thus, on the one hand, a certain complexity of sensory data is necessary in order for the agent to improve its learning capabilities, on the other hand, this should not undervalue the importance of the internal mechanisms and structures that make use of this data.

Emphasis on “Low”-Level Cognition When one is building AGC agents the significance of low-level cognition should not be forgotten. In a certain sense, the structural difference between the cognitive instrument of animals that are not capable of high-level cognition and the human brain are not very significant. In the same way, a system that is very adept in the type of low-level cognition that can be found in animals may not have to undergo significant structural changes in order to be capable of high-

level cognition since goal-directedness and adaptability are vital characteristics of animal cognition as well.

Limitations of Specific AI Tools There are several approaches that are popular in narrow AI fields and one may be tempted to employ one of them in order establish a quick foundation for a AGC agent. These approaches include logic-based planning algorithms, neural nets, evolutionary programming, and reinforcement learning.

Though successful in robotics, logic-based planning algorithms are not very malleable and they would have difficulty in implementing a flexible knowledge base for detecting features from sensory input. Recurrent back-propagation has been suggested to provide a general neural-net approach to procedure learning, but the costs in efficiency are very steep. Much more efficient neural-net methods have been proposed that depend on clever heuristics, but improved efficiency has been achieved through narrowing the scope of the neural-nets. In the case of reinforcement learning, the difficulty in tuning the necessary parameters and problems with scalability make it ill-equipped to serve as a primary foundation for cognition. The problems with evolutionary programming, an approach that mimics certain general evolutionary concepts such as crossovers and mutations, are similar to the problems that occur with neural nets and also include scalability issues. The learning process for evolutionary programming, as the name suggests, is indeed also very slow.

Despite their specific flaws, however, the integration of some of these tools may in fact yield a workable foundation for an AGC system. It is important to stress though, this does not refer to the process of utilizing these tools in different domains and then merging them after the fact. Rather, it is integration first *then* learning.

1.4 The Importance of Philosophy and Cognitive Psychology

The words *mind* and *consciousness* often lead one to think of philosophy, and yet many AI researchers run into these terms due to the proximity of these concepts to their ultimate goal. However, a reluctance to clarify these concepts and the studies that surround them leads to a poor understanding of what one is actually trying to accomplish. Furthermore, philosophical studies also aim to explore the limitations of certain AI

frameworks with regard to their cognitive plausibility. The reason why such studies are specifically important for AGC systems is because these systems take a considerable time to develop. Furthermore, incremental testing during the development process is often difficult to judge.¹ In the past, these characteristics have led the philosophical analysis of cognition to have a significant impact on the path AI research has taken.

Likewise with cognitive psychology, we do not want to misunderstand the cognitive functions that are actually being performed by the biological machines we want to emulate. This kind of misunderstanding may frequently occur when AI researchers depend on casual adult intuitions regarding cognition without taking into account how cognition develops in infants. On the other hand, noisy landmark developments in this field have a way of biasing AI research towards certain paradigms, for better or worse, and therefore also deserve attention for this reason. With these considerations in mind, a significant portion of this thesis is dedicated to examining philosophical and psychological analysis regarding cognition.

1.5 Organization of Thesis

In the next chapter, certain philosophical concepts relating to cognitive agents will be discussed. In Chapter 3, we will look at work done in the fields of cognitive psychology and neuroscience that should be taken into account when formulating a cognitive architecture. In Chapter 4, we will describe recent models that aim to account for human cognition. In Chapter 5, we will analyze their features and limitations. In Chapter 6, we will introduce our own Inscriptor Model which we hope to fully develop in our future research.

¹ Incremental testing is best suited for systems that can be sequentially built module-by-module. However, since a general approach to cognition mandates that various cognitive modules depend on each other and develop in parallel, this is difficult to achieve.

Chapter 2

Philosophical Foundations for Cognitive Agents

It may be beneficial perhaps to start off this section with a rather obvious point: that the philosophy of mind has been around long before the science of cognition. This by no means is to suggest that such an endeavor has always borne fruit, but such studies do at least provide a background with which to consider this relatively new science. One characteristic that a philosophy of cognition also exhibits more of than its scientific counterpart is *holism*. In other words, it is currently easier to find studies concerned with a unified approach to cognition in philosophy than it is in science. This, as mentioned in previous sections, is most likely fueled by the fact that the latter depends and is overwhelmed with vast amounts of experimental data. Therefore it may not come as too much of a surprise that a holistic scientific approach to cognition is often viewed to be more “philosophical” compared to other scientific approaches. The reason is that the science of cognition is yet to determine how cognition is in fact unified in the human mind. Hence, holistic approaches to cognition tend to involve a greater number of assumptions than their rigidly experimental counterparts. Philosophical studies in this area can be said to be concerned with which set of assumptions are in fact “cognitively plausible” and which are not by determining whether they run contrary to an earlier set of assumptions that have been made, or whether they run contrary to experimental conclusions that have been established beyond a reasonable doubt.

The philosophy of mind is much more significant with respect to modeling human cognition than it is to psychiatry or neuroscience. Since the workings of the mind have not been revealed in their entirety, creating a model of human cognition at this stage requires one to make a set of assumptions in order to fill the gaps of experimental

knowledge. However, if a system built on such assumptions is able to successfully perform a variety of cognitive tasks at a human level, then such a system may in turn give clues to as to how the mind works since it has been shown to be equivalent with the human mind at a certain functional level.

Several key philosophical topics that are important in modeling human cognition should be discussed. These include computationalism, connectionism, symbol grounding, and modularity of mind. In many of these topics the arguments for certain contrarian views can be traced back to Jerry Fodor and his reasoning on these matters will surface frequently in the following sections.

2.1 Computationalism in General

As mentioned previously, a unified theory of cognition requires robust explanations of many cognitive subfields. However, there are currently no dominant theories in cognitive subfields such as perception, memory, concepts, semantics, and the rest. Eric Dietrich, editor of JETAI², believes that when there are no well-developed theories for those components of cognition, people begin to look for scapegoats: “When ‘theoreticians’... wish to indict something for the ‘failures’ of cognitive science, they descend on computationalism precisely because it is the only component of a unified theory that is robust enough to attack.”

One can find many definitions given for computationalism in literature. For Dietrich, computationalism is the theory that “cognition is best *explained* as an algorithmic execution of Turing-computable function....we want to explain how some system does what it does by interpreting it as executing some computation.” In other words, cognitive state transitions can be viewed as recursive read-write operations on a string of symbols with syntax and semantics. Within computationalism there is a further divide between classical, or symbolic, computationalism and connectionist computationalism. In the latter, “Subsymbols are not operated upon with symbolic manipulation: They participate in numerical—not symbolic—computation,” as Smolensky puts it in [2].

² *Journal of Experiment and Theoretical Artificial Intelligence*

The aim of computationalism is merely to provide a framework within which to conduct further study. Since the hypothesis itself does not put forth specific functions with which to explain cognition, nor what type of models to build it cannot be proven false simply by pointing to computational models that have thus far failed. In fact computationalism leaves most of the work to theories regarding specific cognitive components and functions. However, because of being a broad and general hypothesis, computationalism is often accused of being vacuous and unfalsifiable. Supporters, though, argue that the hypothesis has a clear falsifiability criteria: If a cognitive agent can be shown to compute a non-Turing computable function, then the hypothesis is false.

The computationalism hypothesis, many insist, shouldn't be confused with *computerism*. Computerism holds that the architecture for cognition is structured like a von Neumann machine, in other words, a digital and serial computer. A standard computer is simply an engine for computation, and the computationalism hypothesis does not make any claims to what type of engine should be used.

There are also a number of people who believe that computationalism is necessary but not sufficient. Among these is Jerry Fodor, who views the computation hypothesis necessary in order to understand modular and relatively low-level cognitive ability such as perception and syntactic language processing. He doesn't, however, believe it can account for certain high-level cognitive ability such as abductive reasoning (inference-to-the-best-explanation). This is where a system has to decide on whether a certain set of information is relevant to the problem at hand given all the information available to the system. The only way, one can posit, that the field of AI has of attaining abductive reasoning is through the use of heuristics. That is, the relevance measures such a system has are dependent on heuristic guesses about what is or is not relevant. At this stage, Fodor claims that heuristics are incapable of providing abductive reasoning and therefore something other than a computational approach is necessary.

Computationalists such as Dietrich [3] take issue with Fodor's rejection of heuristics as adequate tools for abductive reasoning. Fodor's core argument is that "it is circular if the inferences that are required to figure out which local heuristic to employ are themselves often abductive." In other words, adhering to computationalism for this task will trap one in a vicious cycle of using abductive reasoning in order to solve the

problem of abductive reasoning. Dietrich, on the other hand, points out that humans and machines cut through this infinite loop by “immediate inference.” For him, Fodor’s point is the same point raised by Lewis Carroll in “What the Tortoise said to Achilles,” which Dietrich summarizes as: “To decide, one has to decide to decide, but to do that, one has to decide to decide to decide, and so on...Hence, one can never make a decision.” Dietrich’s view is that humans don’t fall into this infinite regress, not because there is something other than computational components to this act, but because the cognitive architecture prevents such a loop from taking place. In AI, the system doesn’t need a heuristic to execute a heuristic; it just executes it. When such immediate inference is hardwired into the system, the infinite loop is avoided. Dietrich draws a human analogy for this point: “Standing in the Jackson Hole valley in western Wyoming, I don’t need to justify that I see the Grand Teytons beyond just noting that I see them.”

2.2 Connectionism

The term “connectionist model” refers to models that at some critical level are structurally similar to neural networks and represent certain important cognitive types in a distributed fashion. Within the computationalism hypothesis, there has been a debate between the symbolic classical approach and the connectionist approach. With tangible advances in neuroscience, however, the philosophical arguments against connectionist models at times appear to be practically irrelevant. Nevertheless, human cognition at some level seems to process symbols and therefore arguments that suggest that certain connectionist approaches are not cognitively plausible should be looked at carefully.

The main difference between the classical symbolic approach and the connectionist approach with respect to the computationalism hypothesis is that the latter places an emphasis on subsymbolic features that are not engaged in symbolic manipulation, whereas the former holds that the atomic types in cognition are symbolic.

Jerry Fodor’s challenge [4] to connectionists is stated as follows: explain the existence of systematic relations between cognitive capacities without assuming that cognitive processes are causally sensitive to the constituent structure of mental representations. Fodor claims that if connectionism can’t account for systematicity, it

thus fails to give an adequate basis for a theory of cognition; but if its approach to systematicity requires mental processes that are sensitive to the constituent structure of mental representations then their theory is at best an alternative implementation of classical or traditional architectures of cognition.

Systematicity is the idea that “cognition comes in clumps.” In other words, it seems that there are groups of semantically related mental states such that there is a psychological law that dictates that a cognitive agent is able to be in one of those states belonging to the group only if it is able to be in many others: “Thus you don’t find organisms that can learn to prefer the green triangle to the red square, but cannot learn to prefer to the red triangle to the green square...[or one] that can think the thought ‘the girl loves John’ but can’t think the thought ‘John loves the girl.’” In the classical approach, the mental representation that is entertained when one thinks the thought “John loves the girl” is a complex symbol, “of which the classical constituents include representations of John, the girl, and loving.”

Paul Smolensky [2] responds to Fodor’s challenge by giving two alternatives to the classical approach. The two ways correspond to the ways in which complex mental representation can be distributed. One is where the distribution yields “weak compositional structure,” the other yields “strong compositional structure.” Fodor, addressing Smolensky’s reply, claims that weak compositional structure and strong compositional structure do not account for systematicity.

In Smolensky’s weakly compositional connectionism the concept CUP WITH COFFEE is represented as a vector of binary values for constituent properties, e.g, 1/0 values for such things as “hot liquid,” “upright container”, “burnt odor” etc. Fodor says that intuitively one would think that the concept “coffee” is a constituent of the concept “cup with coffee”. But Smolensky asserts the way one obtains the concept “coffee” from such a representation of the concept “cup with coffee” is to subtract “cup without coffee” from it (vector subtraction). However, it is not intuitive that “cup without coffee” is a constituent of “cup with coffee.” The reason is because “there is no single vector that counts as the ‘coffee’ representation, hence no single vector is a component of all the representations which, in a classical system, would have ‘coffee’ as a classical

constituent.” The reason why systematicity is not satisfied is because this representation of ‘coffee’ is context dependent—in this case, the context provided by ‘cup’.

For strong compositional structure, Smolensky refers to the idea of “a true constituent” that “can move around and fill any of a number of different roles,” and claims that this can be achieved with vectors encoding distributed representations without implementating symbolic syntax constituency as in classical approaches. However, Fodor remarks that the vectors simply do not form a constituent structure. In other words, in cognition when one thinks of the concept “cup with coffee” he has to token or think about the concept “coffee” and the concept “cup”. In the vector representation, when one thinks of “cup with coffee” there is no structural law in place to force him to think of “cup” and “coffee” in order to do so. Hence, Fodor deems such method of representation as cognitively implausible.

Although Fodor’s argument that the presented model may be cognitively inadequate may have some merit, there are many who insist that his generalized argument presented with Pylyshn fails to refute connectionism as a whole. To repeat, Fodor and Pylyshn’s argument is that connectionism cannot account for the compositional semantics in human cognition (unless it implements a classical structure), which consists of compositionality (the meaning of “the girl loves john” is derived from the meaning of its parts, “the girl”, “loves”, and john) and systematicity (the ability to think “the girl loves John” is connected to the ability to think “John loves the girl”).

Chalmers in [5] argues that Fodor’s argument against the compositionality of connectionism cannot be made without applying it to connectionist implementations of classical architectures as well. However, since the classical architectures can be implemented by Turing machines, and connectionist machines can implement Turing machines, the argument cannot be valid. Wallis in [6] argues that classical architectures are not necessarily systematic, that human cognition contains many unsystematic cognitive capacities, and that Fodor’s systematicity premise is false: “For example, people who will pay \$100 to reduce a risk from one in one million to zero will only pay at most \$1 to reduce the very same risk from two in one million to one in one million...Such reasoning, although widespread, looks *prima facie* unsystematic. That is, it treats the same objective risk differently in different cases.”

2.4 Symbol Grounding

One of the often-cited claims against computationalism is that it fails to account for the presence of meaning in its manipulation of symbols. Stevan Harnad [7] calls this the “symbol grounding problem.” However, it’s far from obvious why symbol grounding cannot be attained under the computationalist paradigm. For one would assume that if symbol grounding can be achieved using computational means, it is compatible with computationalism. Harnad, on the other hand, seems to shrink the broad framework of computationalism in order to suggest that computationalism does not suffice, and that his symbol grounding approach should be seen as a required extension of computationalism, instead of a required branch of the computationalist tree (which is my view of it). However, Harnad’s insistence on the former does not prevent his work being very useful in this latter respect.

Harnad posits that symbols are grounded in sensorimotor projections. In effect, the sensorimotor system categorizes objects of the real world in the mind of the individual. However, once such basic categories are established language can come steal these categories and their inherent meanings and quickly form new categories without necessitating interaction with the outside world. Such new categories may or may not exist in the real world, and they account for a prevalent feature of human cognition: the ability of believing something that one cannot see.

Harnad states that others, like Jerry Fodor, have attempted to dodge the grounding problem by saying that the solution for it is to connect the symbol systems to the world in the right way. This, he posits, is not of much use because the suggestion doesn’t tell you anything about how to connect the symbols to the outside world. If it were to be done this would result in a “wide theory of meaning” where the internal is directly connected to the external with causal connections. Harnad wants to stay narrow and hold onto symbols and ground them using only internal resources. Instead of looking for a connection between the symbols and the “wide world”, the narrow view looks for a connection between symbols and mental representations--the sensorimotor projections of the categories the symbols designate: “it is a connection between symbols and the proximal “shadows” that the distal objects cast on the system’s sensorimotor surfaces.” In all

fairness, I'm not sure Fodor was ruling this sort of thing out when he talked about connecting the internal to the external.

At the root of Harnad's work is the idea that cognition is categorization, and that categorization is at heart a sensorimotor capacity. For example, the art of separating male chicks from female chicks cannot be learned without some degree of trial and error: A person who is a master in such a regard cannot simply tell a learner, "a male chick has xyz features, and a female chick has uvz etc," or draw what he sees for that matter. There seems to be things available to the sensorimotor system that cannot propagate upward to the realm of higher cognition.

If the cognitive system interacts with only its own representations (Harnad's "proximal projections") of the external world, how can one know whether the basic categories are right or wrong? In other words, does meaning again slip through our fingers—do we have to match our internal representations with outside objects all over again? Harnad's response is that categories are learned "on the basis of corrective feedback from the sensorimotor consequences of *miscategorization*." If you eat a toadstool it will make you sick, Harnad says. Presumably toadstools are then removed from the edible category and placed under the inedible category.

However, if two mushrooms look the same, taste the same etc., but one creates defects years later, one cannot categorically distinguish them with his sensorimotor system. In order to learn such categories--using Harnad's lingo--one engages in "linguistic theft," where the subject is taught a certain category using already grounded categories or symbols. The intuitive aim is to establish an architecture for meaning that is "recursive, though not infinitely regressive." Hence, using "linguistic theft" you can learn categories and concepts that may be invisible to you.

Harnad, though, senses he has a lot of philosophical weight he has to get off his shoulders and confesses that he hasn't "really provided a theory of meaning at all, but merely a theory of symbol grounding." He states that the difference between symbol grounding and meaning may be illustrated by noting what a robot engaging in symbol grounding might be missing when comparing its ability to human cognition. The symbol grounding that they share, however, puts both of them beyond computerized encyclopedias.

I think, however, one needs to be cautious of Harnad's zeal in taking sensorimotor systems to be the heart of categorization. Harnad points out how linguistic theft can give one categories that sensorimotor systems cannot sense. He does not, though, examine cases in which categories gained by linguistic theft go *directly against* one's senses. For example, a color-blind person knows bricks are red even though he sees them to be green, because he is told so and bricks can remain under the RED category in his mind. Or that, to one's eye, the moon appears to be larger when it's closer to the horizon but one is able to believe that the moon is actually the same size. Our ability to reject certain categories and beliefs fueled by our senses by using higher cognition makes the endeavor of category-building much more difficult than the idea that the senses provide the basic categories (less controversial) and linguistic theft merely fills in the gaps (more problematic). A robot built on Harnad's model still needs to figure out when not to believe what it sees and when not to believe other people.

2.5 Modularity

A common belief in modern cognitive science is that aspects of cognition are modularized. Meaning that linguistics, perception and such are independent of each other. Jerry Fodor outlines four cases of modularity:

Encapsulation: Information flow between modules is constrained by mental architecture. Knowing you're seeing an optical illusion doesn't make it disappear.

Inaccessability: The inverse of encapsulation. Seeing an optical illusion doesn't necessitate that you think it is real.

Domain specificity: Concepts that are available for language learning may not be available for things like face recognition or math, and vice versa.

Innateness: Modules are “genetically preprogrammed” to an extent. An English learning/speaking infant may make a lot of mistakes, but he never seems to utter sentences like “loves mommy me.” Infants seem to misspeak in specific ways.

Fodor believes in all four cases of modularity and takes the task of examining arguments presented by Karmiloff-Smith [4] with respect to these cases. Karmiloff-Smith contends that mental processes become encapsulated in the course of cognitive development, which Fodor calls “modularization.” She also argues that modularized information becomes increasingly accessible over time as a result of a “epigenetic” process of representational redescription (RR theory); this Fodor calls “demodularization.” Karmiloff-Smith believes that only some domain-specific information is innate and that encapsulation and accessibility are not.

Fodor finds Karmiloff-Smith’s account of modularization weak, and summarizable to the following: “the plasticity of the infant’s brain militates against the thesis that its cognitive architecture is innately performed.” It seems to Fodor then, that modularization is simply a process of maturing, and the proposal gets nowhere if such a maturing process is itself genetically determined. For Fodor, to say that the neural plasticity of the infantile brain determines cognitive architecture (as opposed to only cognitive content) is to stipulate. If genetics determines the way in which neurons are plastic, then they are not really plastic in those ways. Karmiloff-Smith’s modularization hypothesis has no evidence that can refute it, but neither does it seem to have any evidence to support it. These do not look like promising beginnings for a valid hypothesis in Fodor’s eyes.

For demodularization, Karmiloff-Smith concentrates on the reorganization of cognitive domains that usually occur after a child has achieved a behavioral mastery of the domain. The claim is that there is an increase at this stage in the accessibility of a module. Fodor highlights one of her main examples: “Once young children are beyond the very initial stage of language acquisition and are consistently producing open-class and close-class words...there can be no question that at some level these are internally represented as words. [But, on the other hand] when asked to count words in a sentence young children neglect to count the closed-class items.” The three-year olds recognize

“table” as a word but reject that “the” is a word, but nonetheless they are able to parse “the” as if it were a word. According to Karmiloff-Smith, “The RR model posits this developmental progression can be explained only by invoking, not one representation of linguistic knowledge, to which one either has or does not have access, but several re-representations of the same knowledge, allowing for increasing accessibility.”

Fodor claims that there is no evidence that the accessibility of modularized information increases as cognition develops, and that even if it did in Karmiloff-Smith’s RR model, the redescription of the modularized information wouldn’t explain why it does. The crucial question, to him, is whether what is accessed at a later stage in cognitive development is information *inside* the module (intramodular). He mentions that what could have changed between three-year olds and six-year olds may have simply been their understanding of “word”. The former may have understood it as meaning “open-class word,” whereas the latter may have learned that’s not what was meant. But Fodor gives Karmiloff-Smith the benefit of the doubt and tries to argue the point while accepting it has something to do with accessibility.

The problem is “whether the three-year old who has achieved behavioral mastery (e.g. who is able to fluently parse utterances of the sentence ‘the boy ran’) has off-line access to the information that “the” is a word.” When “the boy ran” is parsed, “the” is part of the structural description of the parse. Fodor contrasts this with the “the” lexicon that is stored in the child’s language module. If it turns out that the child is learning that the word “the” in the structural description is a word, then he is not learning anything more from inside the module than before and Karmiloff-Smith’s account looks to be incorrect. The data that Karmiloff-Smith uses does not rule out that the child is learning something outside of the module. However, neither account seems to be able to explain why the problem only occurs for closed-class words.

The structural description of “the window was broken by the rock” tells one the sentence is passive, but it does not tell one what operations are done in order to construct the passive sentence. The latter would be module internal information, and Fodor argues that such information only becomes accessible through a linguistics course, not through mere cognitive development.

Fodor invites the reader to suppose that module internal information does become accessible through cognitive development and argues that Karmiloff-Smith's representational redescription does not seem to account for it. Her account posits that "the human mind exploits its representational complexity by re-representing its implicit knowledge into explicit form." Fodor takes this as meaning that "it's the child's changing of his representational formats—in particular his changing from formats that are less accessible to formats that are more so" that accounts for the cognitive development. The idea that differences in format explain differences in accessibility troubles Fodor. The sentence "the cat is on the mat" when said in French is more accessible if you speak French, less so if you speak English, and equally accessible if you speak both: "No 'format' is either accessible or inaccessible as such. So no story about changing formats is in and of itself, an explanation of changes in accessibility." If a child changes representational formats during cognitive development, it doesn't mean that intramodular information becomes more available: "it just raises the question *why* the information is more accessible in the new format than the older one."

Fodor moves onto the ways in which Karmiloff-Smith thinks of redescribing representations. She believes that in early cognitive development, representations are procedural representations, "in the form of procedures for analyzing and responding to stimuli in the external environment." According to her, representations become less procedural as cognition advances. For example, one starts off with a parser and then ends up with a grammar instead of the parsing instructions of old. However, Fodor opposes the idea that "deproceduralized" representations make the representations any more accessible, for the reasons stated above.

Another use of redescription is inductive generalization. In this case, "the child starts with representations of a variety of discrete cases; subsequently, he generalizes over these, arriving at rules that apply to all the cases of a certain kind." Although Fodor does not object to the existence of inductive generalization, he doesn't think it will help Karmiloff-Smith's representation redescription theory. He's inclined to think that inductive generalization is not "redescription," but the addition of new information. Furthermore, even though adults may know more generalizations than children, like children, they don't seem to be able to articulate these generalizations, e.g. they can't tell

you why “walked” ends with a ‘t’ sound and why “vended” doesn’t. So then that would mean that the generalizations stay inside the module, and thus don’t “demodualize” as Karmiloff-Smith supposes: “what children theorize about is not what’s represented in their modules, but rather what’s represented in the outputs in their modules.” Fodor thinks if adults could theorize about what’s inside the modules, then the world would not need linguists.

Chapter 3

Cognitive Psychology and Neuroscience

If one views the human brain as a black box, then experimental cognitive science aims to “shed light” into what is inside the black box and what the range of abilities of the black box is. Those interested in building a computational model of human cognition, however, are attempting to build their own box out of different materials that can do the same things as the black box. The latter can get clues as to what type of new box to build by the discoveries of the former; and the former can get clues as to what type of things are inside the black box if the latter can successfully build a box that accomplishes the same tasks.

In exploring the relationship between connectionist models of human cognition and developmental science, David Klahr comes across some words of complaint written by Allen Newell in 1990, one of the first AI scientists to attempt to build a unified model of human cognition:

I have asked some of my developmental friends where the issue stands on transitional mechanisms [that guide advance in cognitive ability]. Mostly, they say developmental psychologists don't have good answers. Moreover, they haven't had the answer for so long now that they don't very often ask the question anymore. [8]

This gives one an idea as to how little help developmental science and neuroscience were to those attempting to build computational models of human cognition some fifteen years ago. Now, however, the tables have turned somewhat. The AI community is having trouble keeping up with the advances in developmental psychology and neuroscience.

Thus, for those interested in building a unified computational model of human cognition it may be helpful to look at some recent findings from experimental cognitive science.

3.1 Mirroring the World

Humans accomplish a great deal of learning by observing actions of others and their own actions occurring in the environment. In order to survive humans must thus engage in imitation learning. In fact it has been argued that many aspects of what we consider “culture” is a consequence of this type of learning. Recently, neurophysiological studies have noted the existence of “mirror neurons” in the brain. This has led many to speculate that imitation learning has a larger role in human cognition than previously thought. However, to what extent this type of learning governs cognitive development is yet to be determined.

3.1.1 Mental Simulation and Mechanical Reasoning

Mechanical reasoning is one of the areas where it is thought the mind engages in “mirroring” external phenomena. One of the issues that can arise is how people mentally represent the physical system and the mechanical rules that govern them. In a recent analysis [9], Hegarty notes that when solving mechanical reasoning problems, participants appear to simulate the physical processes in question, rather than apply known abstract rules to them. Though there is anecdotal evidence of this phenomenon, recent research appears to support the idea that one-to-one mental simulations of physical phenomena take place in the mind.

People with higher degrees of spatial ability seem to solve mechanical reasoning problems more easily, whereas verbal ability seems to have no effect on how well a person solves the same class of problems. Such initial evidence leads some to believe that mechanical inference “involves transformations of spatial representations and depends less on verbal representations.” In experiments where subjects were asked to think aloud

when solving problems, they used gestures that mimicked the behavior of the mechanical systems before they gave a verbal account of the process.

Initially, it may be argued that even though people do not indicate explicit knowledge of a physical process, the correct decisions they take while simulating the event may indicate the presence of implicit knowledge. However, the key aspect of this implicit knowledge is that it only seems to surface while one is engaged in mental simulation. Therefore, if such implicit knowledge cannot be decoupled from mental simulation then such knowledge does not pose as a part of a rival account of mechanical reasoning. In conclusion, mechanical reasoning seems to depend more on perceptual representation and mechanic simulation than on rule-based abstract representation.

3.1.2 Goals and Imitation

There have been many studies in developmental psychology that have focused on the ability of infants to learn by copying the actions of those around them. The debate, in this respect, has been about the degree of flexibility of infants who exercise in copying actions. Whereas one view has held that infants mimic adults without engaging high-level cognitive function, another view holds that infants as young as 12 months old show high-level cognitive ability in being able to pick and choose what adult actions they mimic.

In one study that supports such a view [10], infants observed adults hop a toy mouse across a mat with sound effects. The adults also were put in situations where they placed the mouse in a toy house at the end (House scenario), and situations where there was no house (No House Scenerio). In the House scenario, the infants simply put the mouse in the house and didn't imitate the hopping, whereas in the No House scenario the children imitate all the hopping and the sound effects. The hypothesis is that the infants didn't imitate the hopping in the House scenario because they established that the goal of the adult was to put the mouse in the house. In the No House scenario, the infants determined that the goal of the adult was the hopping and the sound effects and that's what they imitated. Thus, this study shows experimentally that infants who are a year old decide which actions to imitate and copy based on the goals of others.

Studies such as these indicate that imitation in infants that was previously interpreted as being low-level automated mimicry does in fact utilize high-level cognitive ability such as recognizing the goals of others. The observation that learning in one-year old infants is a combination of low-level and high-level cognitive ability indicates the advantages that a general approach in cognition would have with its emphasis on interaction between different cognitive levels.

3.2 Dissociation versus Association

In neuroscience research, there has been a phase in which many studies have provided evidence that appear to suggest the dissociation (or domain specificity) of cognitive abilities that were previously thought to be interlinked. However, in many cases certain dissociations are assumed without much thought given to the empirical basis for such an assumption. As a result, some neuroscientists are now examining whether the collected evidence does convincingly show that certain cognitive properties are in fact not associated with each other, or whether there is reason to doubt the assumed domain-specificity of some cognitive phenomena.

3.2.1 The Theory of Mind Module

The ability to reason about mental states is sometimes referred to as the brain's "theory of mind" (ToM). As in other areas, evidence from functional imaging and neuropsychology has been used to support the view that the mechanisms responsible for it are domain-specific and created by a modular architecture. However, a recent systematic analysis [11] of the collected evidence [12,13] has suggested that there is no clear evidence that supports the domain specificity or modularity of ToM representation.

One aspect of ToM is belief reasoning and the ability to pass false-belief tasks. In such one task, participants are told a story where there is a box and a basket in a room. A girl enters and puts a toy in the basket and leaves, followed by a boy who takes the toy out of the basket and puts it in the box. The participants are told that the girl comes back into the room and are asked whether they think the girl will look in the box or the basket.

Adults will correctly give the basket as the answer, whereas children under four have been shown to incorrectly give the box as the answer. Thus the participants that pass the task are said to be able entertain false beliefs.

In order to pass false-belief tasks, the knowledge item, “[the girl thinks] the toy is in the basket,” has to override the knowledge item “[I know] the toy is in the box.” Therefore, there is an inhibition of the self-perspective. However, if one runs false-belief tasks on a population with impaired executive function (autism), the evidence may not be valuable if one cannot determine whether the error results in obtaining the knowledge item about what the girl thinks *or* the knowledge item about the truth of the matter. For example, a schizophrenic may answer correctly, but only because he thinks he is the girl and not because he entertains a false belief. It is argued by recent analysis that most studies have failed to separate these processes, and therefore one cannot claim there is an independent ToM module at work. In the only study that investigates the knowledge of reality on belief reasoning, those that couldn’t inhibit the self-perspective also lacked a certain amount of inhibitory control in non-social tasks. Unless future studies show dissociation between the inhibition of self-perspective and inhibition in non-social tasks, it currently seems that false-belief tasks depend on general inhibitory processes and not specialized ToM function.

3.2.2 Reward and Attention

Neuroscientific studies of reward and attention have usually been researched separately. The question some have raised, however, is whether this separation has been continued out of convenience, or whether there is in fact a cognitively plausible basis for such a separation. There has been progress made in studying neuronal signals that are related to cognitive processes such as attention, memory, and motivation, and this allows researchers to distance themselves from our own subjective understanding of cognitive processing, which might often be misleading. In a recent examination of research in these areas, Maunsell [14] notes that studies have been conducted in order to observe the effects of attention and reward at the level of single neurons [15,16]. Studies on neural representations of reward have looked into immediate detection of rewards received and

their effect on expectations of future reward. Research on attention has focused on selective stimuli processing, alertness, and supervisory systems that exhibit control of some kind. However, since these studies have been occurring independently, Maunsell believes that the studies may have been focusing on the exact same neuronal signals, since the functional relationship between attention and reward may be hard to separate.

Many such studies have been conducted with monkeys, where researchers focused on a certain type of neuron group (specifically, in the lateral intraparietal area) that were found to be more active when it was more likely that a response would be rewarded. Such conclusions have been supported by a number of studies, where the neuronal modulations were to be associated with reward parameters and the animal's expectation on how much reward it would receive.

However, Maunsell notes that it is easy to reformulate these conclusions in terms of attention. Studies of attention have come to conclusions that are rather intuitive. For example, increased attention to a stimulus in the visual cortex ends up increasing the response of the neurons that represent the stimulus. The reason why studies in attention and reward have remained separate probably has to do with the fact that attention studies have focused on responses to stimuli without considering why attention is directed toward certain stimuli. Maunsell points out, however, that “the only tool used to control attention is the manipulation of rewards.” This expectation of reward may be in terms of physical gains, or it may be in terms of information gains (e.g. you may turn attention to things you think will give you more information). Since the neurofunctional and behavioral consequences of attention and reward appear to be the same, considering them as different phenomena seems to have little benefit. Even though Maunsell points out that some researchers have tried to describe the phenomena of being one or the other, it is because positive reward is usually not thought of in terms of information gained (which attention is thought to cause), that such a desire to distinguish exists. If one broadens the definition of reward to include such motivating factors, then any known difference between attention and expectation of reward is diminished.

This seems to illustrate the advantage of building a general connectionist framework that does not have predetermined separations between certain aspects of cognition. Analysis such as Maunsell's indicates the perils of submitting to professional

convenience when separating the study of seemingly different cognitive characteristics without providing a cognitively plausible account for their dissociation.

3.2.3 Short-term and Long-term Memory

Several neurological case studies dating back to 1957 have suggested that the brain has different mechanisms or regions for dealing with short-term (STM) and long-term (LTM) memory. The former is thought to deal with memory in the order of seconds, whereas the latter deals with memory in the order, of minutes, hours, days etc. These studies [17,18] mainly revolve around cases where a neurological impairment has affected one type of memory but not the other, thus leading scientists to believe that there is dissociation between STM and LTM functions [19].

However, the alternative “associational” view suggests that STM and LTM can be explained by the differences in the dynamics of activation inside the same memory network. In this view, both LTM and STM are governed by reactivation of memory network by environmental stimuli, and internal top-down processes of memory recall. Moreover, STM is further supported by continuously shifting stimulus-specific activity and repetition-related activity dims its activation. This results from having to deal with situations where one has to actively maintain information across short delays.

With this approach, the difference in these two types of memory is not motivated by a difference in the time intervals that they are concerned with, but rather by difference in the type of information that they handle or represent. The hypothesis is that there are multiple memory systems (more than two) that have the same fundamental architecture but handle different types of information across short and long time intervals in order to respond to various cognitive tasks.

3.4 General Frameworks

3.4.1 Levels of Consciousness Model

In psychology, the “levels of consciousness” (LOC) model asserts that high-level degrees of consciousness develop in children sequentially through time and not in parallel and primarily depend on the presence of lower-levels of consciousness. Zelazo in [20] observes that there are two main models of consciousness in this area: an information processing model and a developmental model. In the *information processing* model, consciousness is represented as a single module where information is either available to it or it is not, in a zero-sum fashion. In the *developmental* model of consciousness, there are several layers (referred to as levels of consciousness, or LOCs) where information available to one may not be available to the other.

The need for several LOCs comes from child development examples, such as the Dimensional Change Card Sort scenario. Such a scenario is meant to show that children of a certain age exhibit “a curious dissociation between knowing and doing.” In this task, a set of cards are shown to the child consisting of either blue rabbits or red cars. Then the child is given a set of cards where the objects match to the opposing colors (red rabbits and blue cars), and is told to sort the cards based on one dimension (e.g. separate red from blue), then the other (separate rabbit from car). However, despite being told the rules at each turn, 3-year-old children insist on sorting the cards based the dimension that they sorted in first. More curiously, when asked at each turn where the different cards should go they answer correctly even though they then proceed to sort with the wrong dimension. The account Zelazo gives for the LOC model is as follows: The rules for the different dimensions are stored at the same level of consciousness, however the children are unable to reflect on these two different rules at a higher LOC, and therefore are unable to make a decision to switch to sorting with the other set of rules. The child is able to answer the question correctly because both rules are on the same LOC. The reason why the initial rule is used for sorting even though it is on the same LOC with the other rule is because the initial rule has been associated with the act and there is no higher LOC

available to switch to the other rule. If there were a higher LOC, the child would notice that he was breaking the rules.

One of the core claims of the LOC model is that a higher LOC forms through a process of reflection that allows objects at one level of consciousness to be seen in relation to other objects on the same level. The model contends that increases in levels of consciousness correspond to increases in executive function in child behavior. Initially, the child is said to possess minC—the most simple, but conceptually coherent level of consciousness. Zelazo says that such consciousness must also have intentionality as well, and he agrees with the sentiment that “if one is conscious in *any* sense then one must be conscious *of* something.” Despite being intentional, minC is unreflective, only aware of the present, and does not have a concept of self. Therefore, if a child with minC sees a bird, he is only conscious of the concept of the bird and not the concept of seeing the bird. Zelazo also notes that minC is also the level of consciousness that is responsible for seemingly automatic behavior in adults, such as driving a car while not fully aware. However, even at this level the person is conscious of something (e.g., external stimuli), so to refer to such behavior as “unconscious” would be disingenuous.

In order for minC to produce behavior in the child, an object from the environment, objA, triggers a description from semantic long-term memory, and then becomes an intentional object, IobjA, of minC. In Zelazo’s example, a rattle triggers the description of “a small thing” in memory, which in turn triggers an associated action such as sucking. At this stage, the development of minC accounts for the behavior of a one-year-old, after which the child develops many new abilities within months (first words, using objects functionally etc). According to the model, the next level of consciousness that is developed is deemed to be recursive consciousness, or recC.

At this level of consciousness, experience gains labels, the child is able to label an object of minC. When a child points and says “That is That,” what he means is “that thing I see is the thing I have in my mind.” Without labels, the objects of minC cannot be retained in anyway. By having labels for experiences, experience can be stored in memory and can be called upon without having to occur. In the recC level, when objA triggers IobjA in minC, instead of triggering an immediate action, IobjA is fed into minC and is associated with a label descA from semantic long-term memory. This label can

then be used in working memory as a goal that triggers an action in the absence of objA. The action brought about by such a process may conflict and override the action that the minC enacts. In other words, even when objA is present, instead of responding immediately with the minC action, the child may delay and respond with an action *triggered by the label*.

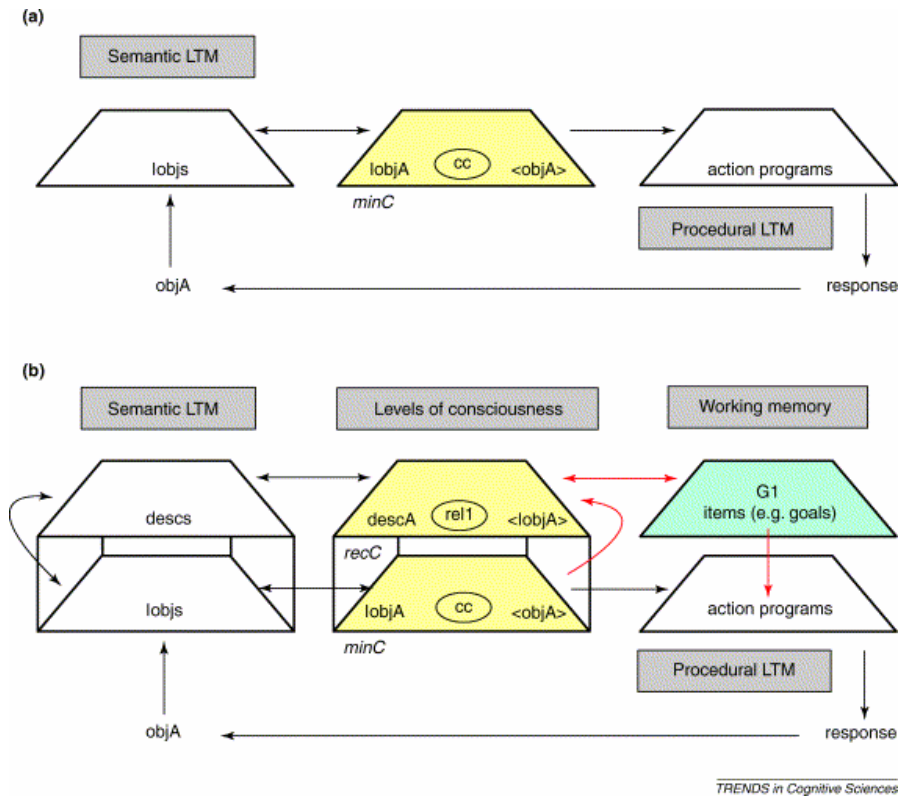


Figure 3.1 (a) A process model of minimal consciousness (*minC*). An object in the environment (*objA*) triggers a representation of that object (*IobjA*) in semantic long term memory (*LTM*); this *IobjA*, which is causally connected (*cc*) to a bracketed *objA*, becomes the content of *minC*, it triggers an associated action program stored in procedural *LTM*, and a response is generated to *objA*. **(b)** Recursive consciousness (*recC*). The contents of *minC* are fed back into *minC* via a recurrent feedback process (curved red arrow), producing *recC*. The contents of *recC* can be related (*rel1*) to a corresponding description (*descA*) or label, which can then be deposited into working memory (right) where it can serve as a goal (*G1*) to trigger an action program in a top-down fashion from procedural *LTM*, adopted from [20]

With this bootstrapping, new LOCs are created with higher degrees of complexity. The idea, insists Zelazo, is that such development allows flexibility, where with the addition of each new level, the child moves away from stimulus control and towards conscious control. The development of self-consciousness, or selfC, allows the child to consider means in order to reach an end. The labels at this level are meant to act as a description of the child's own capacities. These labels can then be used in working memory as a rule in order to achieve a goal. The selfC level occurs at the end of the child's second year and 3-year-olds exhibit an even higher LOC called reflexive consciousness (ref1C), where they are able to consider two labels in selfC in relation to each other. At the age of four, the age that one is able to perform the Dimensional Change Card Sort, the child develops the refC2 LOC, where the child is able to unify two incompatible rules with a single structure. Therefore at certain ages, there seems to be a maximum complexity to the rules children can formulate in order to solve problems. The LOC model aims to give a framework as to how the maximum complexity of such rules increase.

If the LOC model is sufficiently accurate, then in the future it should be shown that conscious processing can occur at one level without having to occur in all the levels, and that there are demonstrable differences between levels of consciousness.

3.4.2 Connectionist Models of Biased Competition in Cognition

Maia and Cleeremans believe that neuroscience and connectionist models of the mind are converging on several important points, namely, the idea of widely distributed representations competing with each other and the idea of top-down bias in representation selection. They highlight several mechanisms that they believe are necessary for consciousness and cognition [21].

- *Active representation*: neuronal firing is a necessary aspect for cognition (though not sufficient)
- *Global competition biased by top-down control*: Cognition depends on the global competition of representations. This competition achieves a “global constraint

satisfaction,” which can be described as the large-scale application of the brain’s knowledge to the current context.

- *Recurrent connections*: Allow high-level processes to influence low-level processes.
- *Meta-representation*: Representations that are developed are fed back into the same network as input. Allows mind to form thoughts about its own thought processes.

One such connectionist model dependent on these factors is a model of *global workspace theory* that has been developed in order to simulate observed “attentional blinks” in subjects. The attentional blink refers to a phenomenon in which subjects observe two successive stimuli, and if the time in between them is short, their ability to notice the second stimulus decreases (thus described as a “blink”).

The model is composed of several levels for representation. Simply put, when the model observes the first stimulus S1, it represents S1 “globally” in all levels of the hierarchy. The recurrent connections allow this stimulus to be represented in the network for a certain amount of time. If the second stimulus S2 is observed by the model in a short time, then it has to compete with the representation of S1 still lingering in the higher level of the hierarchy, and thus the model fails to process S2 properly.

The factors at the forefront in this model are lateral inhibition and recurrent connections. The model’s implementation of a top-down bias is limited to the effect of the lingering representation of previous stimuli on the processing of new stimuli that enters into the lower levels. However, the model lacks a continuous form of top down bias that is present in other models in the field.

In recent neuroscience studies, connectionist models [22,23] have been developed that have aimed to implement the function of working memory and cognitive control that is observed in the prefrontal cortex (PFC). The PFC seems to be able to switch between representations when needed. The basic gist of the model is as follows: recurrent excitation maintains the activity of PFC neurons, and a gating mechanism is used by the basal ganglia to prevent access to the PFC when it is necessary to keep PFC away from

interference and to allow access when it is necessary to modify information in the PFC. The representations that are maintained in the PFC work in order to bias competition between representations in other areas of the brain. The representations that are maintained in the PFC usually have to do with the current task, and therefore are able to provide “the context” for representation selection.

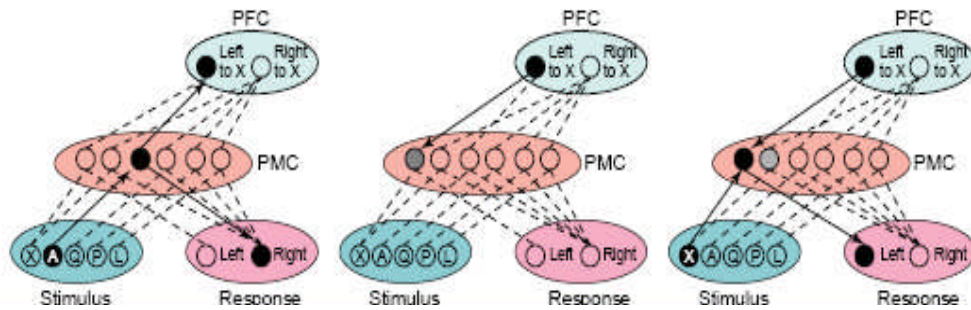


Figure 3.2, adopted from [21]

In a task that involves streaming letters, participants have to press the left button if the letter X occurs after an A, and the right button otherwise. This is considered a simple working memory task because participants have to retain the last stimulus in memory in order to respond correctly. The PMC stands for the posterior and motor cortex that maps the stimulus (A, X, Q, P, L) to the response (Left, Right). The cue is the first stimulus, followed by the delay, and then followed by the probe in the form of the second stimulus.

In the model, the PFC is the module that remembers whether the last stimulus was an A. Hence, in the model the response is dependent on both environmental stimuli (the next letter after A) and top-down bias from the PFC (if A occurred right before). With every observed stimulus two things happen: a response is given and the information in the PFC with regard to the last stimulus is updated. Within each layer, there is lateral inhibition. In the first layer, when A is active, the other representations of stimulus are prevented from firing, and similarly in the layers above.

Thus, when the model observes an A, during the delay the PFC biases the PMC layer toward pressing Left. If X then follows, the further activation of the “Press Left” nodes results in exceeding the threshold required to fire the Left response. Had the last

stimulus not been an A, then the PFC would bias the PMC towards Right, and since the stimulus X is also connected to the “Press Right” nodes of the PMC, the bias from the PFC would result in a Right response.

Hence, such a model is able to explain situations involving working memory and cognitive control by applying biased competition in a connectionist framework.

Chapter 4

Computational Models of Cognition

The previous chapters have dealt with some of the necessary attributes cognitive agents need to have and some of the requirements they need to fulfill. In this chapter, our aim is to describe specific large-scale computational models that have been proposed with the purpose of satisfying the demands of general cognition. Many of the models presented in this chapter have been developed in the last few years. Among them, most adhere to the seed concept and are designed to require minimal off-line additions in later stages. They are designed not only to acquire most of their knowledge on-line, but also most of their *know-how*.

Despite many differences, these models are all connectionist models and share a fundamental attachment to distributed information processing, employing centralized algorithms as little as possible in order to maintain plasticity. Unlike algorithmic approaches to high-level intelligence, an algorithm is not tasked to do the heavy cognitive lifting itself. Furthermore, these models are much more sensitive to philosophical and psychological considerations, and many of the models presented here place a higher emphasis on meaningfully organizing sensory input and relating that organization to other aspects of cognition.

4.1 The Emotion Machine and Critic-Selector Model

In proposing his Emotion Machine [24], Marvin Minsky attacks the notion of there being a great disconnect between rational thought and emotional states. He believes

rather in the opposite: that emotional states are vital in the development of rational thought. According to Minsky, cognition revolves around the act of turning a set of resources on and turning another set of resources off. Emotional states merely act as components in the cognitive resource control process.

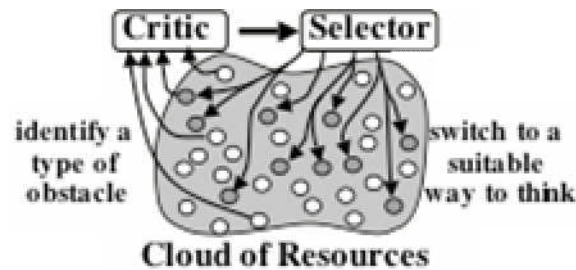


Figure 4.1 Guiding Principle for the Emotion Machine, adopted from [24]

The framework for his approach to cognition is the Critic-Selector model. The basic gist of it is that the objects called *critics* are called on to recognize problems and *select* a Way to Think about the state of affairs, before the system actually thinks about the state of affairs in a specific way. For example, the feeling of frustration may function as a Critic that Selects a way of thinking which aims to avoid any relation to a previous line of action. Hence, after that decision is made all thinking from that point becomes specifically concerned about lines of action that are substantially different from those previously employed. Thus the Critic-Selector model implicitly requires a range of routines, and knowledge about the contexts in which to invoke such them. It doesn't have a fundamental single-purpose routine that is invoked for all problems the system encounters, instead it is composed of levels of critics. For example, reactive critics may propose courses of action in the world, deliberative critics may analyze hypothetical world events, and reflective critics may notice the deficiencies of reactive and deliberative critics.

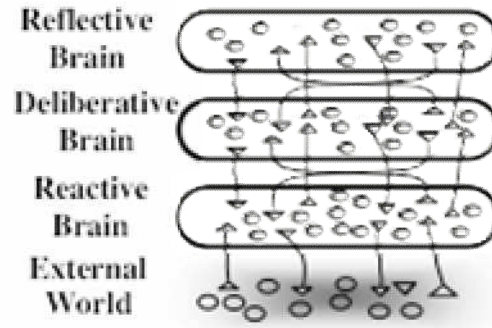


Figure 4.2 Levels of Critics in EM-One, adopted from [24]

A recent implementation of these three-level critics called EM-ONE [25,26] has been developed by Push Singh at MIT Labs. In the reaction stage, the reactive critics try to match the current situation to situations experienced in the past. Once a similar situation is found, the critics propose a sequence of actions that had accomplished the goals of those past situations. In the deliberation stage, deliberative critics reason about the consequences of the actions proposed by the reactive critics. They try to formulate future hypothetical situations in order to determine whether the actions proposed by the bottom layer will succeed or fail. The reflective critics are then used to judge the predictive powers of the deliberative critics and fix any shortcomings inherent in the critics themselves.

Many of the other critics, in both high and low levels, affect the behavior of the deliberative critics, so it would seem appropriate to look into how deliberation is conducted. The hypotheses that are constructed by these critics are analyzed according to several factors. The *plausibility* of the hypothesis is measured by how closely related the hypotheses is to the original scenario; the *importance* of a hypotheses depends on whether it predicts a high degree of success or failure, and is kept in the system longer the closer the conclusion is to the two extremes; *cohesiveness* is measured by the chain of causally connected events the hypotheses encapsulates; the *informativeness* of the hypotheses is determined by how much useful information is missing within the construct; and if a hypotheses contains a situation that has been observed to be both true and false, then it is not considered *consistent*.

Minsky and Singh believe that other higher-level critics are necessary. Other types of critics they mention are the self-reflective, self-conscious, and self-ideals critics.

Self-reflective critics are those that analyze the shortcomings of the system itself. Self-conscious critics are able to make judgments regarding the difference between how the system views itself and how others view the system. The Self-ideals critics look for conflict between the system's action and the high-level values the system has created for itself (breaking the golden rule, for example).

For EM-ONE, Singh does not use logical rules in order to formulate a knowledge database but instead uses what he calls “narratives,” which are in the form of a string of sentences. Singh believes their advantage lies in being able to handle several different realms relating to self-world interaction. For example, a social narrative is described as follows:

Pink desires to hold the stick. Pink grasps the stick. Pink observes that Green observes that it grasped the stick. Pink believes that Green believes that Pink desires to hold the stick.

The idea is to be able to store context information that would otherwise be lost, and in so doing being able to connect events to intentions and goals of the agent, as well as access multiple meanings of situations. Instead of trying to deconstruct complex events into logical constructs to be digested, the complex events remain as is, in order to make sure relations that would otherwise be lost remain intact. Not only is it beneficial to represent some knowledge in multiple ways, the claim is that it's necessary for cognition because each form of representation brings with it its own limitations. Since these various representations may conflict with each other, it's best to leave the knowledge in its most expressive form.

The critics that provide the control mechanism in the system are called *metacritics*. The function of the metacritic is to decide which critics should be activated in the next time step. In light of what critics have been described, the metacritic ends up deciding whether the system should be reacting, deliberating, or reflecting. Therefore, such critics are associated with notions of goals and planning, since questions like “What should I be doing right now?” describes problems that are designated for metacritics. The critic-selector begins to look like a network of if-then statements, where at the high meta-level we have statements such as:

If too many Critics are aroused then describe the problem in more detail.

If too few Critics are aroused then make the description more abstract.

If important resources conflict then try to discover the deadlock.

If there has been a series of failures then switch to a different set of Critics.

Learning in such a system is encapsulated in the changing of the critics' behavior.

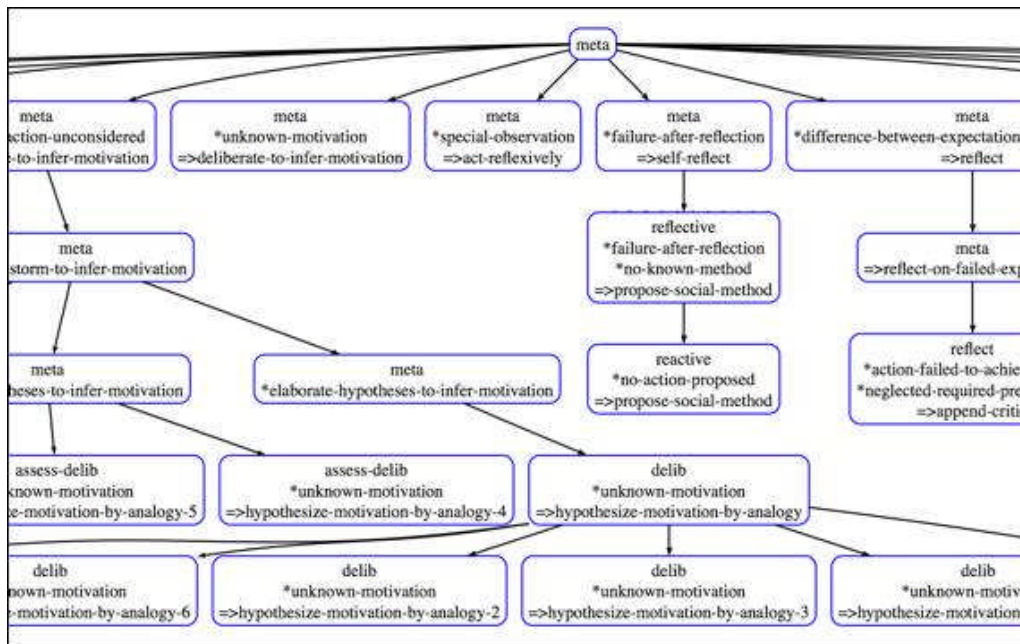


Figure 4.3 An example of a metacritic attempting to react to an agent's action by directing other critics, adopted from [25]

Since there are critics involved in many cognitive layers, this means that learning occurs in the system by altering goals and planning procedures, changing description of events, and making new hypotheses and generalizations among other alterations. The system retains what the critics select depending on whether that selection was used successfully. Minsky states that if a system solves problem P, this is partly the result of the resources that were active during that time. The Selector would be a net that connects the recent resources that were used. Therefore, whenever the Selector is activated, the resources that are associated with it are also activated. These net of resources are intended to be used as “snapshots of the mental state” that solved Problem P. The system

will be unlike an if-then network in a certain way, because if a problem activates several selectors that share resources, a cascade of conflicts and compromises might have to be worked out first. Minsky believes that at any given moment, one is always using a different combination of resources than at any other moment, and this means that new problems end up forcing the system into new ways of thinking where new “snapshots” of the mental processes are taken to in order to be used for similar future tasks. However, Minsky notes that the snapshot might contain resources that were not relevant to the problem at hand, and human cognition seems to only use the relevant resources in the snapshot when faced with similar instances in the future. Therefore, critics are needed in order to test the relation between the snapshot resources and the new similar problem the system faces.

Minsky also suggests the use of what he calls a Difference Engine in order to decide the similarity of two separate situations. First, the engine creates a list of differences that have to be ordered from “serious” to “less serious,” and at every cycle in the engine a method is used in order to alleviate the most serious difference. The Difference Engine is not only believed to be required for abilities such as analogous reasoning, but Minsky finds it essential for reaching goals and planning actions. In order to know if one has reached a goal, the system has to be able to analyze the difference between the desired situation and the current situation, and the desired situation might have a complex array of satisfaction conditions (for example, sometimes people feel that they want something but have trouble describing what that thing is, or how they would know when they actually possess it). Therefore, the Difference Engine also aids in creating subgoals for the system, because the method to reduce the present most serious difference between two situations itself becomes a subgoal. A difference network is to be created for objects as well. The system realizes A is different from B when the environment or the system itself forces it to notice a difference D between objects A and B. For example, if A is successfully used for some task and B fails, this fact becomes a difference item between these two objects. Such a difference D then becomes the link in the network between A and B.

In proposing the Critic-Selector model along with his Emotional Machine, Minsky defines several principles that have to be confronted when attempting to create

any general intelligence system. The Optimization Paradox states that the more able a system is, the greater chance that a change can adversely affect the system. The Investment Principle states that if a certain process functions better in relation to other processes, one is less likely to focus one's energy in improving it. The Parallel Paradox states that when there are more parts of a system interacting, the more chance there is that the parallel processing will adversely affect the system. The basic idea is that as the system improves in functionality, the harder it is to improve the system. For Minsky, though, the solution is to create a system of parts that can evolve separately where the selection of Ways to Think by the Critics vary greatly within the system. In doing so, one would prevent the improvement rate of the system from being curtailed.

4.2 Non-Axiomatic Reasoning System (NARS)

By creating NARS [27,28], Pei Wang aims to propose a model that adapts to the environment and functions under the limitations of insufficient knowledge and resources. In order to achieve this, NARS is designed to have experience-grounded semantics in which concepts are defined in their relation to existing concepts in the system. The guiding principle is that all relations can be converted into inheritance relations and that all knowledge can be stored as such. The backbone of NARS is constructed by inheritance relations formulated in term logic, and within the system these logical sentences are referred to as “Narese.” For example the term logic sentence $\langle \text{bird} \rightarrow \text{animal} \rangle$ (“Bird is an animal”) means that bird is a specialization or sub-category of animal, while animal is a generalization or super-category of bird. Experience in NARS is taken in by the system in the form of this logic, otherwise known as categorical logic, where there is a categorical relation between a subject term and a predicate term, and where the terms denote concepts.

In effect, Wang intends for knowledge in NARS to be viewed as an “inheritance network”. For its reasoning procedures, the system uses the notions of *intension* and *extension*, where in the example above, $\langle \text{bird} \rangle$ would be in the extension of $\langle \text{animal} \rangle$ and $\langle \text{animal} \rangle$ would be in the intension of $\langle \text{bird} \rangle$. In more formal terms, for a concept T, the intension and extension of T is expressed as follows:

$$T^{\text{Intension}} = \{x: T \rightarrow x\} \quad T^{\text{Extension}} = \{x: x \rightarrow T\}$$

Hence, when a new inheritance relation $\langle S \rightarrow P \rangle$ is experienced, S inherits the intension of P while P inherits the extension of S . If the system is faced with the task of determining whether or not a Narese statement is true, then it has to collect evidence of its validity using the present knowledge in the system.

For a Narese statement $\langle S \rightarrow P \rangle$:

- I. if a concept is in the intention of P and S it counts as positive evidence*
- II. If it is in the intention of P but not S it counts as negative evidence.*
- III. If it is in the extension of S and P it counts as positive evidence*
- IV. If it is in the extension of S and not in the extension of P it counts as negative evidence.*
- V. If it is neither in the extension of S or in the intention of P it is an irrelevant concept for evidence.*

For example, if the statement happens to be $\langle \text{birds} \rightarrow \text{swimmers} \rangle$, swans would count as positive evidence, hummingbirds as negative evidence, and tigers would be irrelevant. In this way, the meaning of a concept in NARS consists of the union of its intensional concepts and extensional concepts. For Wang, NARS aims to unite Prototype Theory and Exemplar Theory, where in the former a concept is determined by its properties, and in the latter, its instances. In NARS, the properties of a concept are represented by its intension, and its instances are represented by its extension. Therefore both instances and properties of concepts are taken into account in the system. The prevalence of one type of evidence determines the truth value, whereas the positive evidence to negative evidence ratio brings the necessary notion of uncertainty to a system that is required to function under finite knowledge and resources. For each Narese statement the truth value consists of a pair of values $\langle f, c \rangle$ where f is the *frequency*, or ratio of positive evidence to total evidence, and c , the *confidence*, is the ratio of current evidence to expected future evidence. Therefore when f is 1 then all evidence is positive

and when f is 0 then all evidence is negative. When no evidence exists, *confidence* is 0. Since NARS is experienced based, all true statements are of the form “true to a degree.” An idealized truth value $\langle 1,1 \rangle$ denotes absolute truth, which means all the evidence that could affect the truth value of the statement has been processed (which never really happens in a model based on experience).

Such an inheritance network allows for the deduction, induction, and abduction needed for various cognitive tasks. Wang also defines term logic temporal and procedural statements to be used in the system, in order for the system to reason about time and action sequences. An “event,” for example, is considered to be statement that is true only within a given time frame. The system also allows for variable terms and sentences to be used as terms.

The system receives information from the environment via an input stream of Narese sentences in the following manner:

$$S_1 N_1 S_2 N_2 \dots S_i N_i$$

Where S_i is a Narese statement, and N_i , is the time step difference between the arrival of S_i and S_{i+1} .

The incoming Narese statements in the stream are all seen as tasks that NARS has to perform. Tasks can either be *judgments*, *questions*, or *goals*. In this regard, judgments are statements about the world accompanied by a truth value that the system must add to its knowledge network, after which point it becomes a belief. A question is a Narese statement without a truth value that the system must try to find an answer to. A goal is a Narese statement that the system must make true (find the statements that would make it true, and convert them into actions).

The operating procedure for NARS is as follows:

1. *Select a concept to work on based on priority.*
2. *Within the context of the concept, pick an inference task based on evaluated priority.*

3. *Within this further context, select a piece of previous knowledge to be used to solve the task, again, based on evaluated priority.*
4. *Derive new tasks and beliefs by applying inference rules to the selected tasks and knowledge.*
5. *Return the selected tasks and knowledge into memory.*
6. *Insert new tasks and knowledge into memory.*

Bags are the priority queue data structures that are used for these operations. The need for a priority queue is extenuated by the limitations of finite time and resources. The top-level bag consists of a prioritized list of concept bags. A concept bag for concept *T* contains a knowledge bag and a task bag that each contains Narese statements that have *T* in the subject or predicate. Long-term and short-term factors affect the priority of an item. These long-term and short-term factors handle the differences between past experience and current context. In the knowledge bag, the long-term priority of one item of knowledge (proportional to the probability of selection) is determined by its successful use in solving tasks. In the short-term, relevancy of an item to the current task at hand is what affects its priority and selection. While the item is not used in the inference reasoning of the system it decays in priority. In all bags, low-priority items are removed if the memory is full.

The priority of an item can be summarized to be affected by the following factors:

1. The *origin* of the item: how and what process it was generated by. This value remains constant, and is available from the moment of its creation.
2. The *quality* of the item: the richness of its attribute values. This value may or may not change and is also available initially.
3. The *usefulness* of the item: based on past contributions. The value is acquired gradually.
4. The *relevance* of the item: based on the current selected concept or task. The value is revaluated at intervals.

The first three are long-term factors, and the last is a short-term factor. New, non-derived tasks that come in from the environment already have priorities attributed to them (this is because, in many cases, a user is involved with the environmental input).

The priority of a task can also get affected by the quality of the inference results it gives. If a new task gives a confident conclusion its priority is increased. If a derived task gives a good answer for its parent task to process, its priority is also increased. Furthermore if a derived task has conflicting premises, its priority is increased (the conflict needs to be resolved).

Long-term endurance is measured by a concept's *durability*, and this measure does not have to be directly proportional to the concept's priority. When a task is completed, it is removed from the bags of the concepts that were involved. Every time a task is removed from a concept, the priority of the concept decreases, and decreases further if the priority of task was high. However, if the concept was related to a high-priority task that was solved, its durability is increased.

If a certain high priority task is concerned with concept M, it may be removed from M's bag while a new task is derived that involves concepts S and P. This derivation implies that S and P were related in some way to M. The new derived task having been added into the concept bags of S and P increases the priority of each of these concepts while the priority of M is decreased by the removal of the parent task. In effect, what occurs looks like an *activation phenomenon*, where activation appears to flow from one concept outward towards others.

When dealing with continuous interaction with the environment, the system may think that executing an operation will achieve a goal but this does not guarantee that the execution of the operation will in fact achieve that goal. Wang notes that there is also the possibility that the operation may negatively affect other goals, and make them more difficult to accomplish regardless of whether the targeted goal itself is actually reached. Therefore in the NARS system other goals get to have a say in the operation intended to reach a given goal. Depending on the contribution of the operation to their ends, other goals can increase or decrease the operation's *utility value*. This is affected by the goal's importance, as well as the strength of the belief that links the operation to the goal.

When the system is given a task, because of its time and resources constraints, it has to somehow avoid getting stuck on that task. If the task is a question, for example, the system returns the first answer it finds and lowers the priority of the task. This way it can still work to find a better answer to the question and address other tasks in the queue.

In NARS, the system executes a single operation (action) at a given time. The execution of operations can be regarded to be fairly different than processing tasks. The decision to execute an operation is considered binary, whereas the decision to process a task is done in degrees. There are many steps in processing a task, and they can be done in parallel using different speeds. However, the execution of an operation is a yes/no affair, and thus is done sequentially.

Hence in NARS, the “meaning” of a concept is determined by all its knowledge items and associated tasks *along* with their priority distributions. Since both external experience via the environment and internal processes such as inference reasoning affect the priority, links, and the truth values in the knowledge system, this implicitly causes the meaning of the concept to change through experience. In this way, concepts are meant to be fluid with flexible boundaries that can be stretched.

4.3 SHRUTI

One of the problems arising with connectionist models of cognition has always been how to control the spreading of activations among the nodes so as to duplicate the seemingly decisive and sequential behavior (in thought and action) of human cognition that is often better imitated in logic-rule-based classical systems, and to do this without sacrificing the dynamic advantages of a connectionist system.

This is precisely the problem that the authors of SHRUTI [29], a connectionist model of cognition, have attempted to tackle. SHRUTI is composed of a network of *focal clusters*, which are themselves a group of nodes each with different functions. The focal clusters account for the representation of concepts, objects, rules, and relations in the system. For example, if one focal cluster represents “buy” and another represents “own”, and a third focal cluster links those two together then this linkage represents the knowledge rule “buying results in owning”.

The nodes in the system are meant to be analogous to a collection of neurons, in that they can contain a limited amount of information and can have varying ranges of activity functions. A collector node, referred to as “+”, when active indicates that something is true or false (can be positive or negative); an enabler, “?”, signals a query; a utility node, “\$”, indicates a reward or punishment associated with a relation. Binding two nodes depends on their coordinated activity. For example, such activity with respect to a role node (“buys-buyer”) and a type object node (“Tom”) is taken to represent the type object as the filler for that role (“Tom is the buyer”).

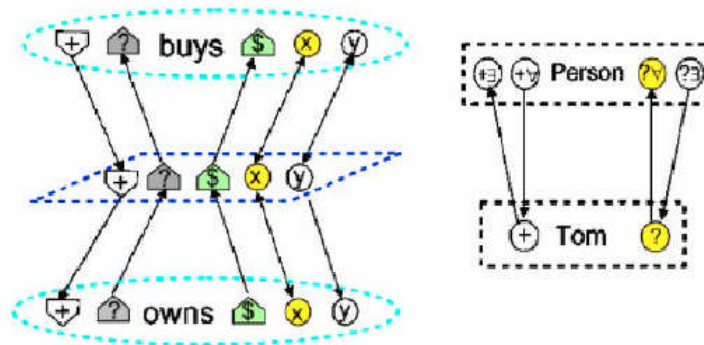


Figure 4.4 If SHRUTI is given the query “Does Tom own something?” represented by the <owns> box, activation spreading activates relevant query, “Did Tom buy something?” represented by the <buys> box in order to give an answer. As one can see the “?” query nodes activate in the opposite direction of the “+” collector/verifier nodes. The square in between the two query relations is expanded on the right. “Tom” is the filler for the <buys> and <owns> relation in this case. If something is true for all <Person> types then one can deduce that it is true for Tom, which is why the “for all” symbol in the <Person> type connects to the + node in <Tom>. Likewise, if Tom has an attribute then one can deduce that there exists a <Person> type with that attribute, hence the connection in the other direction. A similar situation is the case with the query node as well. Adopted from [29]

Currently, the system becomes active when an assertion or query is received as input. The activation of the nodes that represent this input results in the spreading of the activation across various focal clusters. The activations proceed from collector to collector node in a forward causal traversal, whereas backward traversals occur from enabler node to enabler node. In other words, forward causal engineering occurs when

facts are entered into the system and reverse engineering occurs when queries are entered. Activations pass from role nodes to role nodes via the rule nodes that link them together. Activations proceeds from goals to subgoals via utility nodes (e.g. the want of ownership results in the want of purchase).

The authors believe that actions need to be handled differently than nodal relations because they need to be under the direct control of the system or agent, for patterns of reward and punishment may have serious consequences. They believe that without a set of controls imposed on the system, the connectionist model will not be able to handle a large set of complex decision-making problems.

In a system based on the spreading activation of nodes, it becomes difficult to maintain the causal binding between two nodes when the number of nodes between them increases (because of the activation propagation going through many clusters). Therefore, if the activation propagation is extensive, there will be a certain amount of information loss that would need to be addressed. Hence, the authors attempt to define a set of control mechanisms that can channel the spreading of activation toward complex decision-making ability. However, one first needs to determine what kind of relationship the control mechanisms are to have with the existing connectionist structure of the model. In contrast to such a structure, the control mechanisms in SHRUTI are to behave in a serial manner and not be automatically activated whenever a node it is connected to is activated. Also unlike the elements of the focal clusters in the system, the control mechanisms do not partake in the representation of any kind of knowledge items or relations. The authors aim to avoid the method of creating a few but very complex control mechanisms, and opt to create instead a set of simple control mechanisms that are highly distributed throughout the system.

The function of control mechanisms in the connectionist system will be to manipulate a sequence of actions, highlight certain subgoals, and maintain a set of active items. Each control mechanism will be directed toward one specific failure of the underlying connectionist architecture. The “subgoal focus control” would end up restricting the activation propagation to a certain subnetwork of the system; a “hypothesis testing control” would allow complex scenarios to be assessed that feature the possibility of early actions preventing the occurrence of later actions, in other words, they would

help avoid deadlock. If the activation propagation has become overburdened by the depth of the causal node chains, a “search control” would decide which paths to rule out, and which paths to continue propagating in. The authors contend that such relatively complex control mechanisms can be created from a small set of control primitives. The primitives that were created for SHRUTI are used for the process of monitoring, filtering, maintenance, organization, and manipulation.

The monitoring control primitive is among the simplest, and its function is to receive input from a set of targets and then signal if a certain property is satisfied by the target set. An example of a simple property that a monitoring control should detect could be “all nodes firing above a certain threshold.” An important monitoring control would be conflict detection. When there is an occurrence of a high level of disbelief and a high level of belief about a relational item, for example, the conflict control would receive both input from both positive and negative collectors and when their product exceeds a threshold this control would become active. A higher level conflict control may be connected to a set of such controls across the system and fire if their product in turn exceeds a given threshold. Such a simple monitoring control circuit is currently implemented in SHRUTI.

Another primitive is a filter control. A filter in effect selectively highlights the activations of certain nodes while simultaneously dimming other activations. Such a filter may dim a set of activated goal nodes, while another lights up a set of queries. Thus, a filter can be simply implemented by a node with excitatory or inhibitory connections to a set of nodes, or with excitatory connection to one set of nodes while having an inhibitory connection to another set of nodes. Such a node differs from regular nodes in the system because its own activation depends on another higher-level control mechanism.

A selection control primitive is used to single out a node for further processing from an existing activated target set. Such primitives can be made to select randomly or based their selection on the varying activity level of the target set. The selection primitive must be able to enforce its choice on the subsequent processing that occurs in the system. Such a primitive can be used to create a “winner-takes-all” (WTA) type of selection that is very useful for certain cognitive tasks. This allows a set of nodes that are connected amongst themselves to inhibit each other until only one remains active. A trigger allows

the activation to proceed to the WTA control, which then sends a signal back to the input nodes in order to enforce its selection. One can also combine the WTA control with a filter so that the WTA control is connected to many input node sets and decides only to communicate with one set at a certain time.

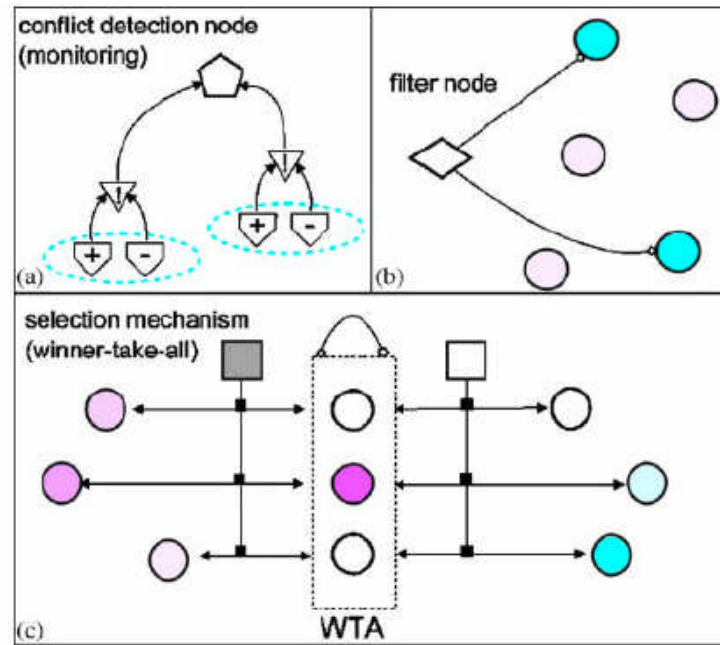


Figure 4.5 Various primitive control mechanisms in SHRUTI, adopted from [29]

A maintenance control primitive can simply be a single connection from a maintainer node to a target node. Maintainer nodes are activated or “locked” when a higher-level control mechanism needs to maintain the activation of the target node. If both nodes are activated at the same time, the target node remains active from that point on in coordination with the maintainer node whether or not it happens to be receiving any activity from its regular input connections. Furthermore, the resulting activations from the maintained state remain identical had the target node continued to be activated by its input connections. There are no consequences at all in the system however, if the maintainer node is activated without the co-occurring activation of the target node. In other words, an activated maintainer never maintains the inactivity of a target node (which would be preventing its activation from regular sources), but instead maintains the maximum activity of the target node during the period the maintainer itself is active.

In a connectionist model, being able to organize activated elements across the system is also necessary. The authors suggest that organizing the simpler maintenance control modes instead of having to organize the regular representational nodes in the system, might be a more efficient way of implementing the organization. Manipulation control primitives are also required if there needs to be a “will” of the system directing the spreading of activation. Such control primitives may alter existing connections of their target nodes or change the set of target nodes when they are activated by a higher-level control mechanism.

The authors illustrate how such control mechanisms could be put in use in a connectionist system by using an example of a military commander agent trying to preserve his unit. In such an example, the commander can consider a set of subgoals that can help him succeed: destroying the enemy, avoiding the enemy, and preventing an attack. The analysis of these subgoals happen in parallel through the inference activation spreading in the system. However, since two subgoals can diverge rapidly in relation to the set of knowledge items that are relevant to each completion (e.g. the concepts involved in avoiding the enemy can be very different from the concepts involved in destroying the enemy), the inferences involved may become too detailed and exceed the capacity of the system. Thus, with the addition of a subgoal focus unit in the system, when such an activation complexity capacity is reached, the subgoal focus control will suppress all subgoal clusters except one (either by random selection or by some other criteria) in order to ensure that all relevant nodes to that subgoal can be explored. Such a subgoal focus unit would require monitor, filter, and selection primitives.

A monitor node would need to detect when the number of activations in the system exceeded the maximum threshold, a situation which would require the system to focus on a subgoal. Such a monitor node would be connected to every node in a set that has activation limitations. Once the monitor node becomes activated, it in turn activates a WTA selection control unit that is connected to utility nodes associated with the subgoals. If *destroy(enemyforce)* has the highest utility among the subgoals, then its activation is heightened while the activations of the other subgoals are inhibited. In order to focus on the *destroy(enemyforce)*, a filter is needed in order to enhance its subnetwork and suppress the activity of nodes not connected to the clusters involved.

In conclusion, in an uncontrolled connectionist system, the authors of the SHRUTI system believe that the possible deadlock and loss of information that would result would prevent the agent to exhibit complex decision-making behavior or reach its goals in an intuitive manner. As a result they suggest one should build a set of control mechanisms partially independent from the underlying connectionist framework in order to direct the spreading of activation.

4.4 The Novamente Engine

Goertzel's and Pennachin's Novamente Engine [30] aims to meaningfully bring together existing narrow AI methodologies in order to create a single general intelligence engine. However, it does not combine these approaches in so far as to leave them as separate interacting modules, but instead aims to meld them together as an interwoven architecture that will be greater than the sum of its parts. The idea of "the self" in Novamente is embodied in a collection of patterns the system recognizes in itself as a whole, in its external actions, and in its relationship with other intelligent systems. The overview of the system is as follows:

Nodes: *Nodes may symbolize entities in the external world, they may embody simple executable processes, they may symbolize abstract concepts, or they may serve as components in relationship-webs signifying complex concepts or procedures.*

Links: *Links may be n-ary, and may point to nodes or links; they embody various types of relationships between concepts, percepts or actions. The network of links is a web of relationships.*

MindAgents: *A MindAgent is a software object embodying a dynamical process such as importance updating, concept creation, or first order logical inference. It acts directly on individual Atoms, but is intended to induce and guide system-wide dynamical patterns.*

Mind OS: *The Mind OS, living within the DINI framework, enables diverse MindAgents to act efficiently on large populations of Nodes and Links distributed across multiple machines.*

Maps: *A map represents declarative or procedural knowledge, as a pattern distributed across many Nodes and Links.*

Units: A Unit is a collection of Nodes, Links and MindAgents, living on a cluster of machines, collectively devoted to carrying out a particular function such as: vision processing, language generation, highly-focused concentration.

Novamente has two different ways in which it represents knowledge in the system. There is the Atom level, consisting of single software objects that are either Nodes or Links; and there is the Map level, which consists of sets of Atoms being activated together, or activated according to a pattern. The content on both levels is the same, what differs is the breadth and depth of the knowledge. At the Atom level the knowledge representation is more specific and more reliable, but is devoid of breadth. The map level, due to its flexibility, is able to stretch out and react to augmentations and edits that are more conducive to learning and real-time behavior. For example, the concept “New York” at the Map level would be linked to related concepts with varying activity levels. A map in Novamente is part of the *Atomspace*, composed of all nodes and links and wherein each link spans more than two nodes (in other words, a hypergraph).

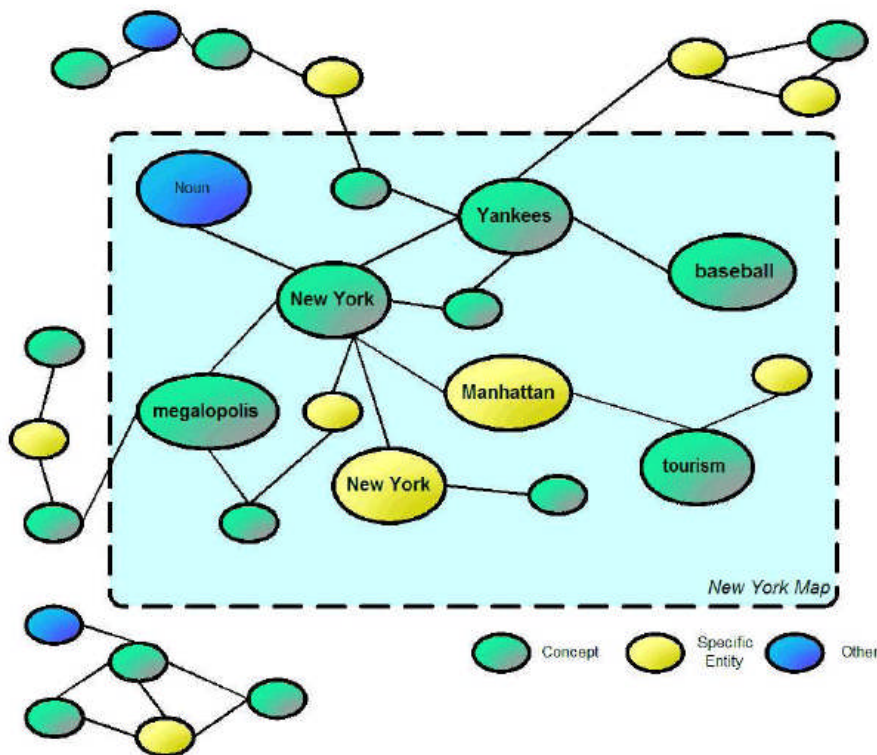


Figure 4.6 A Map in a Novamente Engine, adopted from [30]

In the Atomspace:

- The nodes and links are weighted with complex weight structures (Truth- Value and AttentionValue objects)
- Some of the nodes can contain data objects (characters, numbers, color values, etc.)
- Some of the nodes can contain small hypergraphs internally.

The weight structuring in Novamente aims to embody the two main realms of AI research. The Truth-Value objects represent the logic paradigm, and the AttentionValue objects represent the neural network paradigm.

The Truth-Value of an Atom (either a node or a link) aims to ascertain how reliable the Atom is in describing the world. This is composed of a probability value, and a confidence value that reflects the amount of evidence used to arrive at the probability. It may also include a probability distribution function, or the probability of the Atom occurring in a procedural or perceptual stream.

The Attention-Value is used to determine how much priority or attention should be given to the Atom. The Atom's long term importance (LTI) is measured by analyzing the value of keeping it in memory instead of paging it to disk. A Recent Utility value is assigned to the atom based on how much has been gained by the recent processing of the atom. An Atom also has a general Importance value, determined by its activation, LTI, and recent utility, which is in turn used to decide how much CPU time the Atom should receive.

All nodes and links in the Atomspace hypergraph are to be interpreted logically with probabilistic logic. Certain nodes and links can be observed to reflect causal processes, which lead to assigning credit to other Atoms, and is part of the system's adaptive attention allocation. Also, some nodes and links can be viewed as executable programs.

Nodes that are executable are called Procedure Nodes, which can be small programs, or representation of complex patterns, or action nodes that carry out logical evaluations. ConceptNodes may refer to specifics objects or portions of a map. Psyche Nodes consist of GoalNodes and FeelingNodes which aid in adjusting overall system

behavior. These nodes aim to reflect the idea that feelings go beyond momentary stimulations, for example by simulating expectation of satisfaction across a time range, and adjusting behavior according to such expectations.

There are also various types of links in the system. Logical Links consist of InheritanceLinks, SimilarityLinks, ImplicationLinks, and EquivalenceLinks. AssociativeLinks are links that are formed through activation and nonlogical means. Action-Concept links aim to form concepts around actions taken by Procedure Nodes. While different types of nodes are interacting at the Atom level, at the Map level one can keep track of same node types by observing maps consisting mostly of GoalNodes, FeelingNodes, or ConceptNodes.

The authors state the correlation between Novamente structures and brain structures are indirect. Neurons and synapses are not represented in the Novamente architecture, but the Novamente nodes aim to be analogous to “neuronal groups,” and respective links are to be analogous to bundles of synapses. Here the authors cite neuroscientist Gerald Edelman’s reference to “neuronal maps,” which they believe is similar to the notion of the map structure in Novamente.

The Mind-OS of Novamente is a distributed processing framework that operates throughout a cluster of machines so that a node on one machine can have links to a node on another machine. The OS executes a circular queue of MindAgents objects which, besides a few “system maintenance” objects, are supposed to embody a series of cognitive processes. MindAgents that the authors have already implemented in the system include agents for first-order inference, concept formation, importance value updating, and forming logical links from previously nonlogical links. The authors aim to expand the types of MindAgent and create such objects that find new patterns among Atoms in the Atomspace, modify existing maps, and perform higher-ordered tasks such as logical unification, and formation of procedural nodes.

The authors find that the map dynamics in the Novamenta system are difficult to describe in detail because unlike Atoms, where you can tell what Atoms are present in the system by checking the AtomTable, maps “emerge” from a set of Atoms that begin to act as a coordinated whole. However, the advantage is that the maps don’t need to be explicitly defined, and one can observe new maps due to different activation patterns, and

in turn use those patterns to tackle different cognitive processes. At a certain level of development, for example, the system should in fact have created a self-image map of some kind.

In short, the Novamente system aims to handle attention allocation fairly effectively by using an activation-based neural network approach. The system is also well-suited for pattern mining because its additional Map level of knowledge representation in addition to a concrete atomic representation, such an approach will also aid in language processing, the authors believe, because of the importance of context. Furthermore, the combination of both logical inference mechanisms and nonlogical associative relationships learned from experience enhances the reasoning ability of the system.

4.5 Confabulation Theory

Robert Hehct-Nielsen proposes a *Theory of Cerebral Cortex* (CT) [31,32] using a computational neuroscientific approach to hypothesize about the “underlying mathematics of cognition.” He contends that the essence of all cognition, including animal cognition revolves around the concept of *confabulation*.

In a language processing example, suppose one had a set of symbols “the train was going” and had to come up with an answer for the symbol that was to follow. Suppose one candidate was the symbol “south”, how would one measure its worth? One way is to measure the probability of the occurrence “the train was going” given “south” in the fifth position of the phrase, in other words, $p(\text{the train was going} \mid _ _ _ _ \text{south})$. However, it might be the case that one has never witnessed the set of symbols “the train is going south” before (had one, then the task would be trivial), therefore such a calculation would be of little worth. Instead one can substitute $p(\text{the train was going} \mid _ _ _ _ \text{south})$ with the proportional product:

$$p(\text{the } _ _ _ _ \mid _ _ _ _ \text{south}) * p(_ \text{train} _ \mid _ _ _ _ \text{south}) * p(_ \text{was} _ \mid _ _ _ _ \text{south}) * p(_ _ _ _ \text{going} \mid _ _ _ _ \text{south})$$

These pairwise probabilities can be calculated using previous experience. If this product is high compared to that of other symbols, then the symbol “south” is included in the set of answers. This is the essence of confabulation.

Confabulation involves a set of assumed facts and the set of answers that are derived by implication from these assumed facts. With regard to cognition, if $abcd$ are assumed facts and if $abcd _ e$ is a known rule, then one wants to find a value of e that maximizes the probability $p(abcd|e)$. However in nontrivial situations $p(abcd|e)$ cannot be calculated. Confabulation is therefore the act of substituting $p(abcd|e)$ with the product $p(a|e)p(b|e)p(c|e)p(d|e)$.

In a test run on language processing, for the set of symbols “her responsibility for making” the answer set returned by the confabulation system was [*erros*, *matters*, *sure*, *references*, *choices*, *lethal*]. Hecht-Nielsen states that none of the complete phrases appeared anywhere in the training corpus. Therefore, he claims that confabulation in this language processing example allows the system to implicitly abide by the rules of grammar and semantics.

In his computational neuroscientific model, all human cognition is taken care of by thousands of separate feature attractor modules. These modules contain a stable list of possibly thousands of symbols which can be augmented. Each symbol is encoded by a certain collection of neurons, where each neuron may play a part in representing more than one symbol.

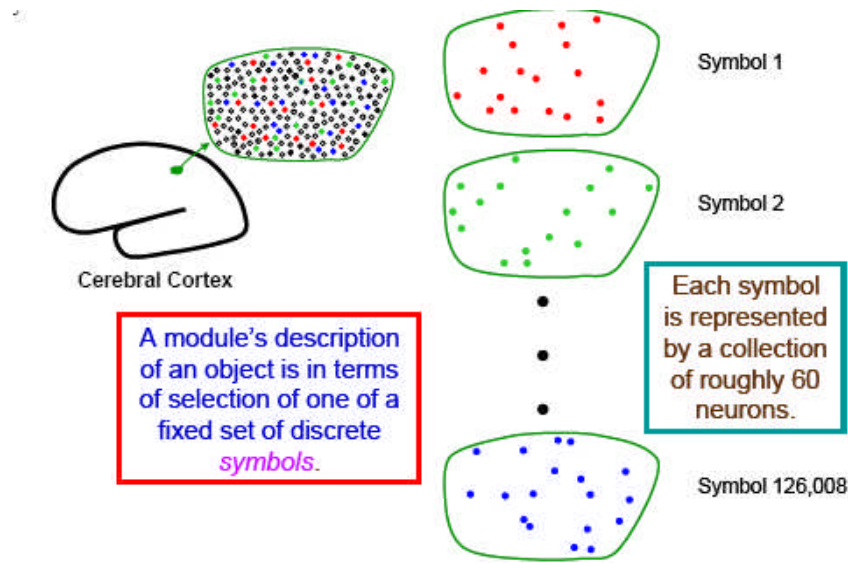


Figure 4.7 A single module contains many neurons, subsets of which represent different symbols, adopted from [32]

Confabulation is implemented by neurons in a feature attractor module working in parallel. The states of the neurons that participate during the process vary considerably due to the densely parallel interactions of a large number of neurons. The final state of a neuron can be active (momentary and intense), excited (lengthy communicating state but not as intense), or inactive. When the outcome of confabulation is only one symbol, the neurons encoding that symbol will be made active, *and the rest of the neurons in the module inactive*. If more than one symbol results from confabulation, then the few neurons that end in an excited or active state are the victors, and the rest of the neurons are inactive (“winner takes all”). If there are no excited or active neurons, this conclusion is referred to as the *null symbol*, and can occur frequently with confabulation. Hecht-Nielson states that such a symbol represents the “I don’t know” response that is often vital in human cognition.

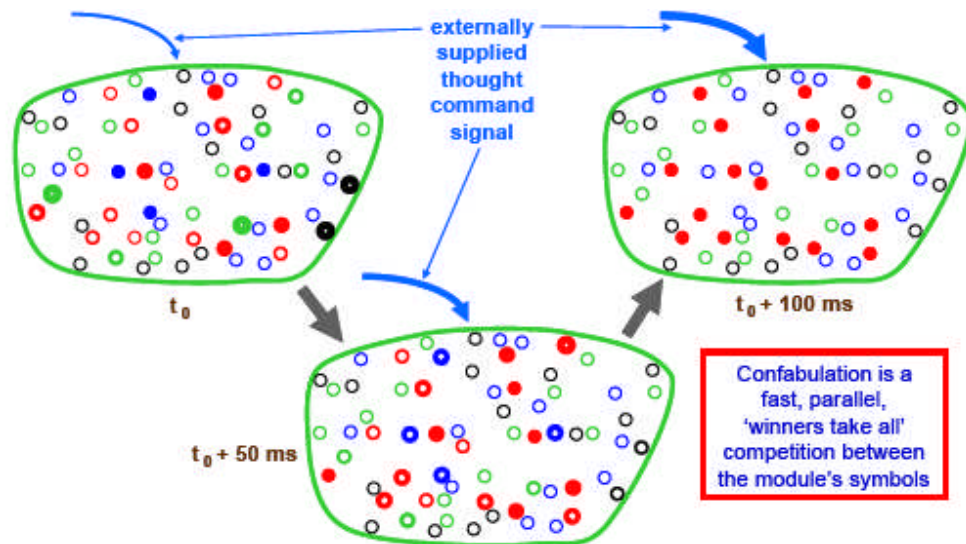


Figure 4.8 Snapshot of a single module at different time periods showing a winner-takes-all symbol selection process. Neurons of the same color are representing a single symbol, hollow circles represent inactive neurons, filled circles represent active neurons. Initially, all neurons in the module have varying activation levels. At the second time frame the red symbol is the winner of the externally triggered activation (has the most active subset of neurons, represented by the filled circles). Therefore in the last time frame, all neurons representing the other symbols are made inactive, and all neurons representing the red symbol are activated. Adopted from [32]

The theory hypothesizes that all knowledge is represented by connection between one symbol (a collection of neurons) in one module to another symbol in another module; such links are referred to as *items of knowledge* (figuring in the billions in humans, according to Hecht-Nielson). Items of knowledge in the theory are immediately formed on a temporary basis when *co-occurrence of symbol pair activity* occurs. Rehearsal of these links during a period of rest (sleep), determines where such links remain in short, intermediate, or long term memory.

When confabulation in the module results in a single active symbol, an *action command* that is associated with the symbol is immediately executed. This is the *reaction-action principle* of the theory. These mechanics aim to represent the situation in which a thought process reaches a single conclusion and launches a new thought or movement process as a result.

One can easily observe, however, that action commands of one symbol can be seen to conflict with action commands of another symbol (laughing and crying, throwing a ball and picking up the phone). Therefore such symbols in the theory are grouped in the same module or *lexicon*. As a result of confabulation in the module then, the active state of one symbol results in de-conflicting such actions, since all other neurons are made inactive. Hecht-Nielsen claims this is why animal behavior appears to be very decisive. The set of symbols in a feature attractor module in the model represents one *attribute* of objects in the mental realm. Symbols are created in the module as soon as external signals to the feature attractor module are delivered.

Throughout the creation, different levels of neurons emerge. In the beginning, feature attractor neurons receive their inputs from extra-modular neurons and begin to respond strongly to the input collection they repeatedly receive (initial connectivity of neurons is assumed to be mostly random). However, because of the competition for the incoming input the neurons in the module have with each other, each neuron is driven away from tuning in to the same collection of inputs as other neurons in the module. Hence, in this first stage of development these *simple* feature detector neurons have a high input range coverage and low information loss.

The secondary level of neurons, are composed of the neurons that happen to be mostly connected to the simple feature detector neurons instead of extra-modular neurons. The consequence of being connected to the simple inter-modular neurons, is that these secondary neurons end up learning statistically common *combinations of multiple co-excited* simple feature neurons. These secondary neurons also form inhibitory connections to the simple feature detector neurons that feed them their inputs. Thus when these secondary neurons are highly excited they shut down the simple neurons that they are connected to; this is called the *precedence principle*. An example within a visual detector module would be simple feature detector neurons representing a segment which is re-represented by a set of secondary neurons representing a longer segment which is made in part by the initial segment. The secondary neurons would shut off the simple neurons, thus having the module represent a whole object and not compete with a set of unassembled pieces. Each following level of neurons forms in a similar manner. At each

level, once the connections are fully formed (due to repetition) the responses of the neurons are frozen.

In Hecht-Nielsen's model there are five levels of feature detector neurons. When an external control input is sent to the module, the neurons that are at the highest levels of excitation will fire. If the control input to the module continues, then more feature detector neurons will fire before they are inhibited. According to the theory, this will result in fixed sets of activated neurons of approximately fixed size which will stabilize as the module development process continues. Each such stabilized set of neurons will then become a *symbol* of the module (*lexicon*). Because of the random and sparse nature of activation, a symbol in the module will look more like a cloud of neurons that can overlap other clouds representing other symbols. Although symbols can be added to the module or lexicon, once a symbol cloud stabilizes it becomes frozen in the module, and thus becomes available for the development of *knowledge links*.

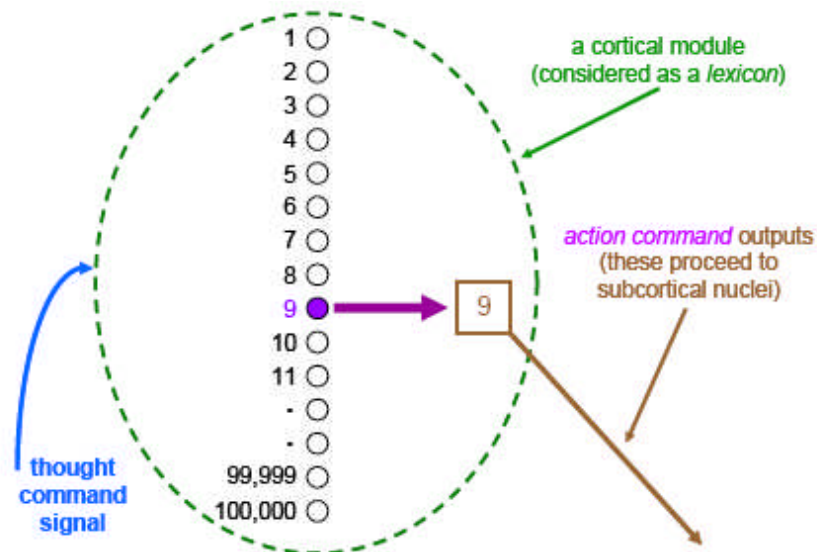


Figure 4.9 The symbol “9” inside the <number> lexicon module, adopted from [32]

A single knowledge link is a connection between a *source* symbol in one module or lexicon to the *target* symbol in another module. The set of all knowledge links is called the knowledge base. When a symbol (source) in one module is activated, its neurons send out their signals to millions of neurons that are residing in that region. Out of those many neurons that receive such signals, a few thousand of them do not receive

just a single signal but many signals originating from the activated symbol. These neurons are called *transponder* neurons. Their highly excited state results in them sending signals in turn to all neurons that they are connected to. These transponder neurons themselves also uniquely represent the source symbol, so in effect the symbol can seem to have been *transported* in some ways. According to the theory, the set of excited transponder neurons is larger than the set of active neurons of the source symbol (because the source symbol neurons have numerous outgoing connections)—this gives the effect of *amplifying* the source symbol. The activation does not continue beyond the transponder neurons because even though there are thousands of them that are excited, they are not excited enough to be *active*. However, *excited neurons can excite other neurons if those neurons are in a module or lexicon that is receiving an external control signal*.

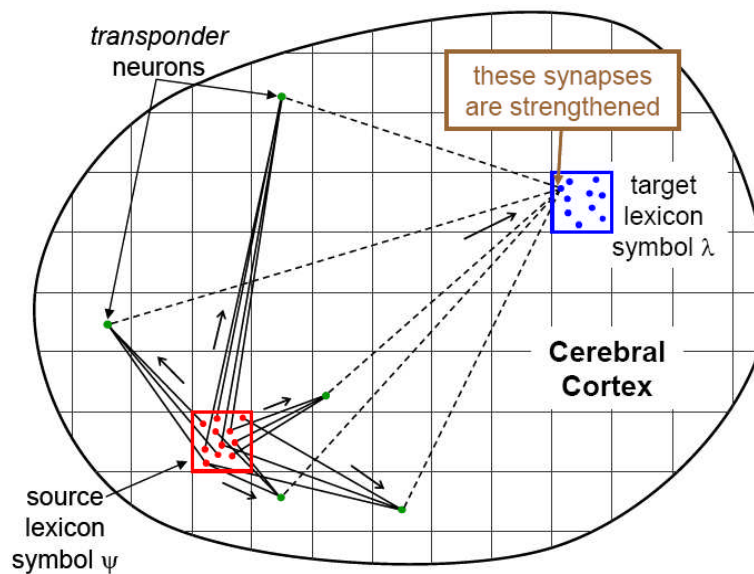


Figure 4.10 A Knowledge Link, adopted from [32]

A small portion of neurons belonging to the target symbol will happen to receive many signals from a subset of transponder neurons, thus completing the knowledge link. If most of the other target symbol neurons are active as this happens, then the transponder neuron connections to that certain subset of target symbol neurons will be strengthened. These synapses strengthening results in what the theory defines as being short-term memory. If the causal pairing of the source and target symbols are rehearsed in a period

of rest the resulting link strengthening induces intermediate or long-term memory. However, the connections from the source symbol to the transponder neurons are not strengthened because, the theory posits, the link has to be used intentionally with immediately preceding co-occurrence of source and target symbols neuron activation in order to induce learning. The transponder neurons of the source symbol will always be the same, so increasing the connection might result in premature activation of the target symbol.

In the model, the strength of the knowledge link is directly proportional to the probability of an assumed fact a given an answer e , $p(a|e)$ in the confabulation calculation discussed above. The source symbol in the model is in the role of the assumed fact, whereas the target symbol takes the place of the answer. In fact the probability that both source and target symbol would be coactive, $p(ae)$, could be represented by the strength of the connection between the transponder neurons and the target symbol neurons. The probability of $p(e)$ is approximated by using the target symbol neurons' average firing rate. From this one can calculate $p(ae)/p(e)$ which is equivalent to $p(a|e)$ (Bayes law). Hecht-Nielsen states that the difference between the weakest connection and the strongest connection between neurons is a factor of six and therefore the probabilities used for confabulation in the model are distributed accordingly.

The theory contends that given the entire neuronal space (cortex), the relative scarcity of knowledge links is necessary for cognition, for if the lexicon modules were fully connected with knowledge links cognitive functionality would be in peril. Coming back to the earlier language processing example *the train was going*, suppose one wanted to check the following set of candidates $\{south, down, sad\}$. In the model, only the first two candidates would presumably have strengthened connections to *all* of the four assumed facts' transponder neurons. Therefore *south* and *down* would win out in the confabulation process and be two answers in the expectation set.

As stated earlier, when a single symbol wins out and becomes active at the end of the confabulation process an action command is launched. The activation of the symbol results in the highest level neurons becoming highly excited. The outgoing signals of these high-level neurons leave the module and reach an action nuclei, one of many (each

having different functions)—thus enacting the *reaction-action* principle. Again the executed actions can be either movement or thought processes.

The question then becomes: how do these high-level neurons of the symbol become wired to a specific set of actions in the first place? Such connections, for the model to be plausible, need to be learned and perfect since executing a wrong action might have serious consequences. The symbol-to-action connections are hypothesized to be developed in childhood, which allows them to be developed through experimentation and trial and error. At first symbol activations trigger random action commands. A nuclei called the basal ganglia determines whether a reduction in goal level was observed (which is taken to be “good”). This nuclei’s functionality arises from its concentrated and dense input from the limbic system. However, reductions in goal states do not occur immediately, and may occur after a long period of time. It is hypothesized by some neuroscientists that the basal ganglia develops a number predictive elements called critics (recall the Emotion Machine) that learn goal state reduction patterns so as to determine such a value at the time of an action execution. Once an action is determined to be of value, the basal ganglia sends a signal to the lowest level of neurons of the active symbol. This in turn causes the connection between the high-level neurons and the action command neurons to be strengthened.

The theory hypothesizes that cognition is built from the following entities: lexicon modules, a knowledge base composed of numerous knowledge links, and the action commands that are associated with each symbol in the lexicon modules.

4.7 The SINBAD System

In the classic associationism put forth by David Hume, if two events occur at the same time or one after the other, these events become associated so that when one thinks of A this causes one to also think about B. However, pair-wise association is not powerful enough to account for the mechanistic process of thought that David Hume wishes it to. Hume’s account of abstraction states that when two similarities between entities occur, there is an abstraction of the feature that those two entities have in common. The problem, of course, is that one can infinitely enumerate the number of similarities two entities share. If associationism is to work, there has to be some

mechanism that explains how the mind only abstracts the useful similarities. The well-known attack on associationism by Fodor states that in order for associationism to account for cognitive function, it has to be endowed with logical functions such as OR and XOR or even more complex functions for it to stand a chance. However, introducing such complexity into the “fundamental” features of associationism creates a problem, in that it fails to explain how the complexity of representations in the mind are learned by the processing of raw sensory data as was originally hypothesized since the complexity in the system is already deemed to be fundamental. However, Ryder and Favorov aim to create a mechanistic associative system that learns through the association of raw environmental input and is also capable of cognitive reasoning.

Like Hecht-Nielsen, Ryder and Favorov [33,34] believe that there is a fundamental cognitive operation that is responsible for almost all of human cognition. However, they believe that such an operation is in essence a predictive mechanism. In their neuroscientific approach to cognition (SINBAD), they believe that the task of learning function belongs to the principal dendrites (the part of the cell that receives the input) of a neuron. The dendrites themselves send their output to the center of the neuron (soma) which sums up all the signals coming in from the dendrites.

The main crux of Ryder and Favorov’s theory is that even though the dendrites of the neuron don’t share the same input channels, they learn to output the same signal as each other. In order to do so, the dendrites modify their own behavior according to the other’s output. So they somehow learn to coordinate their outputs despite the fact that they could be receiving mutually-exclusive environmental information. The authors conclude that in such a case, the differing environmental input the dendrites receive are in fact “mutually predictive” with respect to one another.

The pyramidal cells on which this model is based usually have 5-8 principal dendrites. However, in order to illustrate that these dendrites in their model can learn what they claim they do, Ryder and Favorov only use a pyramidal cell with two dendrites in their example. Each dendrite in the model is made up of ten hidden units, which are themselves connected to two input channels. The input channels of the two dendrites, as stated, are mutually exclusive. The output unit of the dendrite is connected to the soma, which sums up the output coming from both dendrites. The dendrites are

backpropagation networks and their output is a sigmoid function (used for its adeptness for calculating non-linear functions) with the hidden unit values as parameters. The idea is that under each others' guidance the dendrites will be able to learn complex nonlinear functions.

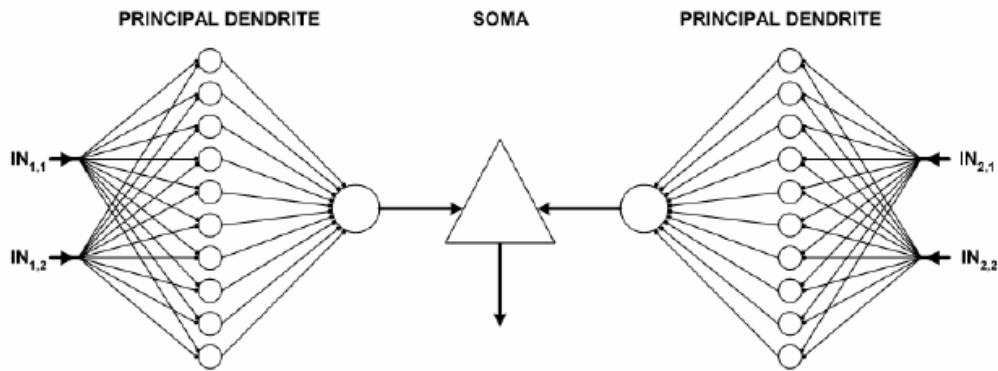


Figure 4.11 The main building block mechanism in SINBAD is the pyramidal cell, with two dendrites connected to the soma (triangle). Each dendrite represents an error propagation network with one output connection, 10 hidden units, and two input connections. Adopted from [33]

In a simple scenario it is assumed that the incoming sensory data involves a single object X . The object X can be either present or absent in the environment ($X=1$, $X=0$), and it is represented in the network with the following functions:

$$X = IN_{11} \text{ XOR } IN_{12}$$

$$X = IN_{21} \text{ XOR } IN_{22}$$

This means that the object X is observed to be present by the network if either IN_{11} or IN_{12} are active, but not both, *and* if either IN_{21} or IN_{22} is active, but not both.

Therefore that allows there to be eight different environmental situations that can be represented by the input channels. The question becomes whether the dendrites can learn whether the two functions are predictive of each other and also whether they can learn that the cause of the matching outputs is the presence of object X in the

environment. In the initial state, the presence of object X is not made known to the cell in any singular way, but its presence lies under the regular patterns in the input combinations of the network. What happens in the training process is as follows: because the dendrites are a backpropagation network, the connection weights of the hidden units are adjusted after each cycle. However, the supervising teacher in this case happens to be the other dendrite, and what results is a situation where one dendrite is guiding the other while modifying its own behavior. The adjustments in the network continue until both dendrites have near identical outputs while receiving signals from different input channels. What has been accomplished as a result, the authors point out, is that each dendrite uses its own input *in order to predict how the other dendrite is going to respond to its own input*.

In such a model, one issue that has to be resolved is how to ensure that the input channels are distributed in such a way so as to ensure that the manner in which the input channels are connected to the dendrites allows for the learning of nonlinear functions to continue unhindered. The previous example was set up specifically for the encoding of one object in the environment, but when the environment and the objects within it are not known in advance one needs to find a general way to connect the input channels to the dendrites.

In order to accomplish this, the authors consider a richer and more complex environmental scenario. The raw sensory information about the environment in this example is represented by 32 binary parameters, what the authors refer to as the “elementary conditions.” The environment happens to contain to complex objects A and B, which are both dependent on the presence of less complex properties. The presence of object A is determined by the functions, $A = (a \text{ XOR } b)$ and $A = (c \text{ XOR } d)$. If object A is a physical object, for example, these two functions may represent its different sides. Likewise the presence of B in the environment is determined by $B = (e \text{ XOR } f)$ and $B = (g \text{ XOR } h)$. The intermediate properties a, b, c, d, e, f, g, h are all in turn determined by functions of the lowest level 32 binary elements or elementary conditions. For instance:

$$a = (1 \text{ XOR } 2), a = 3 \text{ AND } 4$$

$$b = (5 \text{ XOR } 6), b = 7 \text{ AND } 8$$

....

$h = (29 \text{ XOR } 30), h = 31 \text{ AND } 32.$

Therefore in the first level there are 32 elements, in the second there are eight, and in the most complex level there are two elements. Each of the 32 input channels are then randomly assigned to one of the dendrites, but not to both of them. After the network is run on 100 input combinations, the dendrites converge towards near identical outputs. Unlike the first example, since the input channels were distributed randomly there is no guarantee that the binary parameters that determine an intermediate property (a...g) are connected to the same dendrite. This would mean the predictive set of channels may not have been placed on opposing dendrites, possibly making it impossible to determine a specific environmental property.

One way the authors suggest to solve this problem is by having more than one pyramidal cell in the network. The suggested model is a network with 32 pyramidal cells with the input channels randomly assigned to their dendrites. Thus, each cell has a different set of input channels associated with it. The idea is to have a test dendrite that does learn a specific prediction by using the raw 32 input channels (the control case) or by using the information received from the 32 pyramidal cells themselves. The regular cells in all likelihood will be able to pass on information that goes in some ways to represent the intermediate elements a...g. The test dendrite is still modeled as a backpropagation network where it has a set of hidden unites and has the 32 pyramidal cells act as the input channels. The test dendrite was modeled separated for three cases, one to be trained for object A, one to be trained for object B, and one to be trained for object A XOR B.

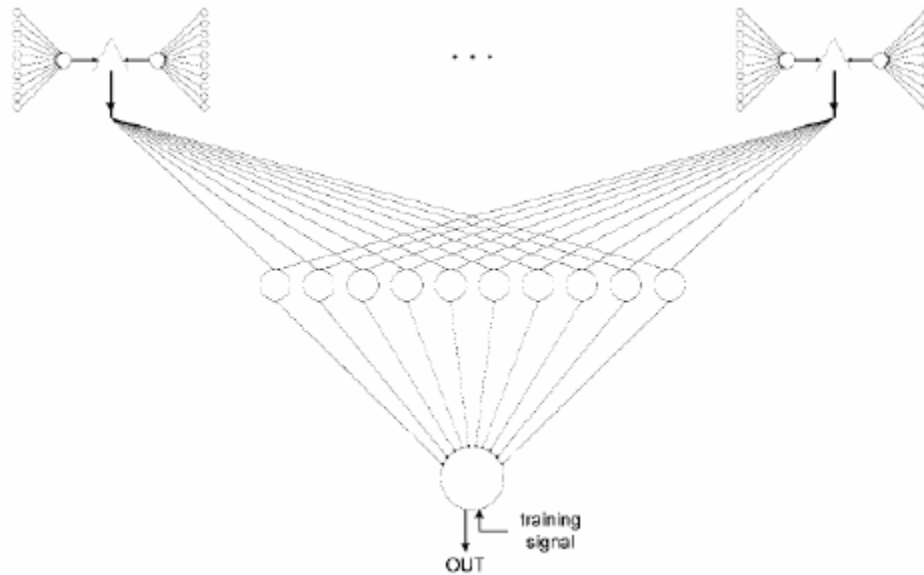


Figure 4.12 A layer of 32 SINBAD cells that have their outputs connected to a test dendrite, which is modeled as a backpropagation network with one output connection and a single layer of hidden units. Adopted from [33]

The control test dendrite that used the raw 32 input channels eventually converged to A but failed to converge to A XOR B. The authors attribute this failure to the difficulty and complexity calculating a third level nonlinear (XOR) function using backpropagation networks. However, when the test dendrite used the 32 pyramidal cells as input channels, the network was not only able to converge to A and the difficult third degree A XOR B, but did so with exposure only to 0.5 % of all possible input combinations. The reason for the improvement between using the raw 32 input channels and the 32 pyramidal cells is most likely due to the ability of the individual cells to be able to relay information to the test dendrite about the second-tier environmental properties a...g. Hence, when such information arrives at the test dendrite the complexity of the function has decreased by one level, so that calculating third level function like A XOR B becomes equivalent in difficulty to calculating second level intermediate function like (a XOR b). Therefore, the authors claim that individual pyramidal cells can function as building blocks for creating prediction networks that have incrementally complex nonlinear functions, and can realize higher-level of environmental features, even when input channels are distributed randomly among dendrites.

Ryder and Favorov believe that these networks will be able to discover various

important high-level cognitive relations. They believe, for example, such structures could be useful in reasoning about the existence of something not directly observable, by analyzing its effect on other entities. They mention how Mendal discovered the existence of genes by studying the observable traits of parents and their offspring, and see no reason as to why SINBAD shouldn't appear to derive the pattern from its layered and hidden units. Other relations that would be learned by the system would be cause and effect properties, and also the correlation between interconnected parts, such as the presence of a head predicting the presence of a body, another cognitive task that the system is well equipped to handle. In essence, Ryder and Favorov believe that what they have proposed is a system that would excel at abduction, inference-to-the-best-explanation.

Chapter 5

Analysis of Cognitive Architectures

CT and SINBAD are models where the smallest building blocks are nodes that are not semantically endowed. SHRUTI, Novamente, and NARS, on the other hand, depend mostly on units that already come with semantic baggage attached. In the latter group of models, the main aim is to obtain high-level human cognitive functioning. Therefore some of the basic building units may themselves contain semantic elements that are beyond the abilities of animal or infant cognition.

The working premise for the former group of models is the idea that non-semantic sensory data can be semantically interpreted by the system through the layered interactions of non-semantic units. The formation of semantic clusters through this interaction will then help guide higher cognitive function. This approach aims to capture the developmental process of cognition more accurately than the approach of the latter group. There is also a belief couched in many of these models that high-level cognitive function results from a more *complex organization* of the same units available to animal cognition, as opposed to the idea that high-level cognitive function can be obtained from the organization of more *complex atomic units*.

The developmental approach is fueled by evidence such as that evidence cited in [35] that show that animals such as rats are able to distinguish features based on their approximate numeric value even though they are clearly incapable of exercising the required symbol manipulation that is needed for exact counting. The results are similar to those obtained when humans are forced to approximate numeric values without having time to count objects exactly. Therefore, models that have atomic units that already come with symbol manipulation abilities in order to account for mental arithmetic won't be

able to duplicate the numeric approximation ability that occurs without symbol manipulation.

5.1 Non-Axiomatic Reasoning System (NARS)

One of the issues with NARS, is to determine how well it handles notions like attention and interruption. Regarding NARS, Wang states:

What should a system do if it is occupied by one task when another one shows up? Since for NARS new tasks do not come from a predetermined set, usually the system cannot tell at the beginning what kind of solution can be found, or how much resources it will cost. In such a situation, it is usually undesired either to let the new task wait for an unlimited period, or to let the new task interrupt the processing of the current task for a unlimited period.

In the model, attention to tasks seems to be handled in a zero sum fashion. For example, increasing one's concentration for a given task is a common ability. However, it requires that one ignores other things that would normally grab one's attention. Not only that, but the things one ignores are not arbitrary. If one is listening to music while doing work, he can ignore the music but can nevertheless interrupt his work if the telephone rings. He might also notice the silence when the music stops. This allows one to conjecture that it isn't the activation of information that's important, but the expectancy of activation. Therefore in terms of the our own model (the Inscriptor Model, later discussed), we may want to have a node that represents what we commonly refer to as "silence." Silence is not "nothing," rather, it is the low frequency threshold of sound that we expect to hear most. Since we expect it often, it is a node whose activation we routinely ignore. Only when we are not expecting it does our attention turn to it when it's activated. These seem to be essential abilities for adequate task or attention management.

With regard to NARS, these activations can be taken as tasks that the system has to address at some point. The key issue in this example is that only tasks that are expected can be ignored. At some point the mind addresses the task "music is playing," and

handles it by decreasing its priority. When new input arrives in the next time step, the system should not view “music is playing” as a new task and address it. However, it must view “music is not playing” as a new task and turn its attention toward it if it has not predicted it. This creates a problem in the following way: Does NARS have to receive input for negative data (“music is not playing etc”), and if it does how does one keep the enumeration of negative data finite? It seems intuitive to assume that the system may only need negative data regarding activations or tasks that it has been predicting. The task’s expectancy should dictate its priority. In the same example, if one has the window open he will ignore a slight breeze because he is expecting it but he may not be expecting the music to stop, which would grab his attention immediately.

Though Wang may not have taken note of these issues regarding task management, there seems to be way that NARS can accomplish the predictions. When a new task “feel slight breeze” comes through, NARS may check its belief network to see to the extent at which its beliefs predict the task. Since the belief “window is open” has been stored in its memory recently, through the system’s inference rules it can come to the realization that the new information is not that unusual, hence it can give the thought of “feel slight breeze” little attention. However, this action in and of itself is a task (or meta-task), because the system engages in the same process as it would when it tries to find an answer to a question or when it tries to reach a goal.

The structure of memory storage is also significant for cognitive models. In NARS, memory is composed of a bag of concepts that in turn have a bag of tasks and a bag of beliefs in them. Wang describes the storage as follows:

For instance, if the statement of a new task is “raven $\rightarrow$ bird,” then the task will be put into the task bag of the concept “ C_{raven} ” and that of the concept “ C_{bird} .” If the task is a judgment, it will add a corresponding belief into the belief bag of the two concepts, too. If there is no such concept (i.e., the term is novel to the system), it will be created as a result of the acceptance of the task.

Here, it’s evident that there’s a redundancy in storage as the new tasks have to be stored in two concepts instead of just one. If one’s aim is to create a cognitive model that

operates under limited time and storage, this is not very ideal. The redundancy in storage may perhaps improve the speed of the computation under this architecture. The main problem, however, seems to be that although C_{raven} and C_{bird} are related in the model abstractly, their relation is not echoed in the storage architecture. Why is this a problem? Well, it may be the case that our cognitive capabilities arise from our interaction with a rather rich sensory environment. This means that the NARS system must be able to handle receiving dense amounts of inputs at a single time step. Although all such sensory information will be in the form of term logic, one will have to sift through all that information in the belief system. If related concepts are not connected in storage, then searching through concepts becomes the only option. However, if one takes into account the speed at which the mind is able to recognize a single concept out of all the concepts in memory, the idea that the mind does a search through its entire memory in order to find the initial concept seems implausible. When one hears the word “dog,” does the mind search for the concept of sound clip and see what mental representation it’s connected to?

A large part of our intelligence seems to be dependent on our ability to determine the relevancy and importance of information. Therefore, the success of NARS is dependent on whether it can truly retain the right kind of information and dismiss what is not relevant to its state of being. What Wang calls “the subtlest part of the control mechanism” may have significant repercussions regarding the limits and abilities of the kind of cognition that can be achieved through this system.

How are the priority and durability of tasks determined? Wang states that the priority and durability of a task depends on the *origin, quality, usefulness, and relevance* of the task. This is all well and good, Wang does not elaborate on how such factors are determined (especially with regard to the last two). The priority values of beliefs are increased when they help in solving tasks, but this will work only if the tasks in question are truly important tasks. Furthermore, Wang explains that the priority and durability of input tasks are determined by the environment, which is troubling as far as cognitive plausibility is concerned. The priority a cognitive agent gives to a sensory input must be determined by a criteria completely internal to the agent. When we observe events in the environment, they don’t enter our mind with tags already attached proclaiming, “We are important, process us first!” Our internal processes make that determination.

5.2 SHRUTI

In SHRUTI, the most basic facts inputted into the system and the queries made to it involve semantic relations. Thus, it would be difficult to argue that the semantic data that is allowed into the system is nearly as rich as the sensory data available to human and animal cognition. Furthermore, this is a self-imposed limitation of design rather than a limitation of the current implementation. That is to say, it is not a matter of *getting to that part*, the system simply wasn't built with the intention of incorporating non-semantic sensory input to begin with. Nevertheless, though the nodes in SHRUTI are intrinsically semantic, the mechanisms that have been proposed to handle cognitive control are primitive, and in fact, are rather compatible with the building block structures of CT, SINBAD, and RA. These units are presented as simple information-processing nodes. As one will recall from an earlier section, there are maintainer nodes that "lock" certain activated nodes so that they remain activated even if they no longer receive signals from their input; and also filter nodes that increase the activation of some nodes while dimming others. SHRUTI, like CT, also employs a winner-takes-all (WTA) mechanism to regulate the spreading activation of nodes. In SHRUTI, all these nodes are activated by "higher-level control nodes," but like in many models based on spreading activation—and because of a lack of a centralized decision-making module—the rules or protocols that govern these high-level control mechanisms remain elusive. In order to explain what is going on, or what *should* be going on, it is necessary to meaningfully interpret and explain the nature of the activations that are taking place. The relation between the semantic nodes that compose the knowledge base and the control nodes that occasionally manipulate them has yet to be adequately explained in SHRUTI. In what has been initially presented, the control nodes are the ones influencing the activation of semantic nodes. However, how realistic is it to claim that the activation of the control nodes themselves are not dependant on the system's knowledge base? Therefore, in order to establish the feedback that would be needed for cognitive control one has to sketch out how the activation of the semantic network comes to affect signals the control nodes receive that in turn manipulate the activation of the semantic network. SHRUTI does not

satisfy this requirement.. On this point, it may have to pay a price for having the nodes of its semantic network be of a completely different type than the nodes it employs for cognitive control. That is to say, it needs to reformulate its complex semantic nodes in terms of its primitive control nodes, thereby integrating both hierarchical structures.

5.3 Novamente Engine

The Novamente Engine and the Emotion Machine share a general similarity that they, for the most part, don't share with others. They both view cognition as something that depends on the integration of functionally distinct components, rather than something that depends on mostly functionally identical units that create functional differentiation at higher levels through interaction with the environment. Both models propose having complex procedural nodes present from the initial point of learning. However, in the Novamente Engine, the narrow AI processes themselves such as reinforcement learning, logic-planning algorithms, and evolutionary programming are also part of the integration. In this way, the Novamente Engine, is the only model discussed here that specifically integrates well-known narrow AI methods in order to accomplish general cognition. One should however remember that there is a difference in integrating narrow AI approaches with the intent of general cognition modeling and developing narrow AI approaches without such an explicit intent, and then expecting them to converge at some point in the future.

On the other hand, when it comes to assigning an "Attention Value" to Atoms in the systems, Novamente has a criteria similar to the one NARS employs in order to weight the priority of a certain concept. In order to assess how much attention a given Atom should receive, Novamente employs a utility function that determines the effect the atom had on recent processing. However, since Novamente functions by executing a circular queue of highly distributed and functionally differentiated Mind Agents, it is unclear whether the Attention Value attributed to the Atom is dependent more on local processing factors, or global processing factors. For example, an Atom might be very useful to a specific Mind Agent but if the Mind Agent's own heavy independent processing isn't queried often by the general system, the Attention Value given to the

Atom may be overvalued. Furthermore, Attention Values at present are only assigned to Atoms and not Mind Agents, which means that one cannot adjust an Atom's global Attention Value based on the importance of the Mind Agents that it serves. This would result in Atoms that are very useful in highly domain-specific instances being promoted globally, thus forcing Mind Agents from other domains to access those Atoms before other candidates.

Furthermore, Novamente has risen out of a previous general cognition architecture also proposed by Goertzel *et al*: Webmind. In Webmind, each node or Atom was built as an autonomous software agent. As a result, the authors note, the computational complexity of such a massively multi-agent system was tremendous. In Novamente, therefore, autonomous software agents are composed of a cluster of nodes and not single nodes. Hence, whereas Webmind had millions of autonomous agents, Novamente only has a few hundred. However, the fact that the structure and functionality of components were altered due to computing costs and not directly because of its inability to handle certain cognitive functions (albeit a consequence of huge computing costs) does make one slightly uneasy. A large overhead as a result of programming in a high-level programming language can conceivably be reduced if one adopts something closer to machine-language. Given the complexity of the existent Mind Agents in Novamente, however, the problem may not have been there being millions of autonomous software agent, rather, the problem may have been that the individual software agents were *too complex*.

5.4 Confabulation Theory

The main aspect that is missing from Hecht-Nielsen's Confabulation Theory is the cognitive function that neuroscientists currently attribute to the basal ganglia, in other words, the cognitive function that determines which actions or thought processes should be preferred above others based on the interaction with the environment. One alternative way such a process may work given the current underlying nature of the model is perhaps if there are intrinsically "positive" and "negative" neuron clouds that can account for the goal function. The activity of "positive" neuron clouds will be maximized by the model,

whereas activity of “negative” neuron clouds will be minimized. The positive and negative neuron clouds in the mind may be taken to represent such basic intrinsic features such as the pleasure and pain states in the human brain. If the activation of positive neuron clouds co-occur with the activation of the action or thought processes then the connection between them is strengthened, and if the activation of negative neuron clouds co-occur with the activation of action and thought processes then the connection between them will be strengthened. The reason for introducing neuron clouds that represent the concept of “positive” and “negative” into the model is to account for the behavior that human cognition engages in when it activates actions and thought processes that increase reward and avoid hardwired negative attributes such as pain. With respect to an action-goal function, there should be an activation bias towards those thought processes and actions that are strongly connected to the positive neuron cloud, and a bias against those that are connected to hardwired negative attributes.

In order to determine whether a reduction in the goal level (interpreted as increased reward) has taken place, the satisfaction conditions of the goal have to be known as well. In fact, a satisfied goal can be defined as a set of conditions that have to be simultaneously active. Even though one may propose the idea of critics that learn patterns of goal state reductions, one still needs to frame such concepts within the neural network that has been proposed by CT, otherwise there is a risk of losing the benefits that are a result of such a model. However, the basal ganglia that is hypothesized to handle the goal function is a black box in the current model. As for the bottom-up and top-down supervision that is needed for AGC, CT thus far has established its foundations for only bottom-up supervision of knowledge representation.

Alternatively, one may have implemented confabulation in a symbolic rather than connectionist framework. Such a model would perform satisfactorily in the kind of natural language processing examples as outlined above. However, the development of knowledge links in CT present a process that is much more adept in handling continuous environmental data while still establishing contextual information as dictated by confabulation. In a symbolic model, it may be easier to confuse random co-occurrence of symbol activation with non-random co-occurrence since, unlike neuron clouds in the connectionist framework, its symbols are not composed of smaller components that are

also activated when other symbols are being represented. This characteristic allows a connectionist model based on the same concept to reduce the consequences of a random activation of a few neurons (possibly reflecting a random event in the environment) on any inferences or associations that are to be made.

5.5 SINBAD System

SINBAD is similar to CT primarily because the emphasis is on the simplicity of the building blocks of the system, and the idea that most of general cognition arises from the successive layering of a primitive mechanism. One of the interesting things about SINBAD is that it gives one an idea of how to integrate different complex modules in the system, even though it is primarily concerned with low-level mechanisms. The basic idea in SINBAD is that two nodes with different inputs can use the other's output as feedback supervision for predicting each other's output. What's more is that the process of prediction between the two nodes results in their ability as a unit to represent an existing sensory phenomenon.

Like Confabulation Theory, the nodes in the SINBAD model do meet the criteria of "spontaneous self-organization" that is necessary for models aimed at general cognition, but they do not yet meet the criteria of goal-oriented behavior. Of course, what is to be commended is the fact that SINBAD goes some ways in achieving complex semantic representations with non-semantic mechanical primitives. This is why the authors view SINBAD as a theory of neurosemantics and mental representation. One may question whether the concept categorization that is achieved can be termed "semantic" in the way humans think their conceptual understanding is semantic, but the important point here is not what we call a certain object, rather the point is that the system is able to categorize and identify the certain object the same way we do. Therefore, the model doesn't represent an object simply because its designers wish it to represent that object. The ability of SINBAD to achieve concept recognition through primitive prediction mechanisms is definitely interesting in this respect. However, one needs to be able to harness the ability of the model in using prediction as a means of concept recognition so that it can work towards satisfying self-defined tasks and exhibit goal-directed behavior.

One of the major issues with models of cognition that depend heavily on co-occurring stimulus is that one has to ensure that co-occurring activations are a result of mutually-exclusive inputs. Otherwise, the co-occurring activation is trivial. Although neuronal components such as dendrites in the brain do not share inputs, when it comes to ensuring the same mutual exclusivity in computational models the task is a difficult one. Establishing a high-level of connectivity and randomly assigning first-level nodes to sensory input will not guarantee mutual-exclusivity. On the other hand, deterministically assigning connections may be an expensive procedure and also inhibit development. Thus, without presenting a clear method with which to limit, if not prevent, mutually-inclusive inputs of sibling dendrites, the SINBAD's power of representation will be restricted.

5.6 Summary and Implementation Pointers

	General Type	Integrated Knowledge Base	Primitive Nodes	Supervision	Modular	Low-level Cognition	Limited resources Restraint	Behavioral Feedback
EM	Distributed High-level	Yes	No	Top & Bottom	Yes	No	No	Yes
NARS	Knowledge Representation	Yes	No	Bottom-Up	No	No	Yes	No
SHRUTI	Knowledge Representation	Yes	Mixed	Top & Bottom	No	Semi	No	No
NE	Distributed High-level	Yes	No	Top & Bottom	Yes	No	Yes	Yes
CT	Knowledge Representation	Yes	Yes	Bottom-Up	No	Yes	No	Yes
SINBAD	Prediction	Yes	Yes	Bottom-Up	No	Yes	No	No

Table 5.1 Comparison of Models

In Table 1, one can see that many of the models have a primary cognitive ability around which to build a general cognitive agent. Only NARS and NE have been developed specifically with working under limited time and memory resources in mind. Among the models, EM, NE, and CT have outlined mechanisms for behavioral feedback. The exceptions are the Emotion Machine and the Novamente Engine, which are built by integrating functionally distinct modules and nodes. SINBAD and CT are the only models composed of solely primitive nodes, whereas SHRUTI contains primitive nodes for cognitive control only. In NARS nodes represent concepts and thus are comparatively high-level. The nodes that form the semantic network in SHRUTI represent complex semantic relations. In EM and NE, the nodes themselves may perform complex algorithmic functions.

With respect to building a connectionist architecture for general cognition we believe that such an architecture should consist of nodes similar to the primitive nodes

seen in SINBAD and CT rather than the high-level concept nodes in NARS, and the procedurally complex nodes in NE and EM. Furthermore, we believe cognitive control nodes like the ones featured in SHRUTI should also be implemented, but only in conjunction with a network composed of similarly primitive nodes and not the high-level conceptual nodes featured in SHRUTI's semantic network. In CT, for example, the co-occurrence of activations of nodes represents a syntactic and statistical co-occurrence. As these co-occurrences increase and proceed up the hierarchy they get closer to producing results that are meaningful to an observer. The point here is that the task is accomplished without the help of atomic nodes that explicitly try to establish semantic relations between different data. This approach more accurately reflects the development of human cognition and should in turn result in an agent with comparatively better cognitive function.

EM's use of narratives to allow the cognitive agent to grasp different contexts may be one of the better approaches to achieving high-level cognitive understanding. However, since we hold that low-level cognitive processing is essential, an ideal implementation should focus on how to construct narrative-like data structures for non-conceptual sensory data.

As previously mentioned, the term-logic statements in NARS that relate one concept to the other are stored in the data structures for both concepts. Instead, connections should be created to the original source. This would ease navigation to associated nodes and conserve memory. Furthermore, the relevance and quality factors in NARS that are used to determine the priority of a concept have not been focused on in the initial development of the system, despite the fact that a proper assessment of a concept's priority is necessary for the system to function. Hence, in the implementation of a connectionist framework, methods that determine the relevance and priority of nodes should be established from the very beginning.

In CT, behavioral feedback between the action nuclei and the symbol-inducing neurons allows behavioral action to alter the symbols that are represented from the initial stages of symbol representation. This is in contrast to models such as the recently developed Recommendation Architecture (RA) [36] that allow behavioral feedback only by adjusting the connection weights between the symbol-clustering modules and the

action modules. It does not allow the action module to bias the specific symbols that are being clustered. Both approaches have their advantages at different stages in cognitive development. The approach of RA has its advantages in the initial stage of learning, whereas the CT approach is beneficial for a later stage of learning. At the early stage, the agent's actions will be mostly random, therefore altering the way symbols are represented through behavioral feedback at this stage will have negative consequences for later learning. However, once the actions establish meaningful connections to the existing symbols, it would be beneficial for the system to shift its symbol representation resources to differentiations that have significant impact on the agent's interaction with the environment. Thus, unlike the current CT and RA models, the implementation of the cognitive agent should have behavioral feedback mechanism that shifts from the RA approach to the CT approach as learning proceeds.

Finally, an implementation should have a foundation that aims to accomplish an extended version of what the current SINBAD system provides. In SINBAD, nodes that were built for prediction are joined together in order achieve a different cognitive ability: symbol representation. Therefore, in essence, the structure of the model should allow for atomic nodes to be reformulated into building blocks for various cognitive abilities without have to alter their original functions.

Chapter 6

A Look at the Inscriptor Model

As initially proposed by Davenport [37], our connectionist Inscriptor Model (IM) centers on two main ideas. First, a parent node and its children in the system function as if-then statements, where the parent is the antecedent term and its children are the consequent terms. Second, an activation principle that allows the spreading of activated nodes functions as a low-level reasoning mechanism. As with all other connectionist models and neural networks, the Inscriptor Model is composed of a hierarchy of nodes.

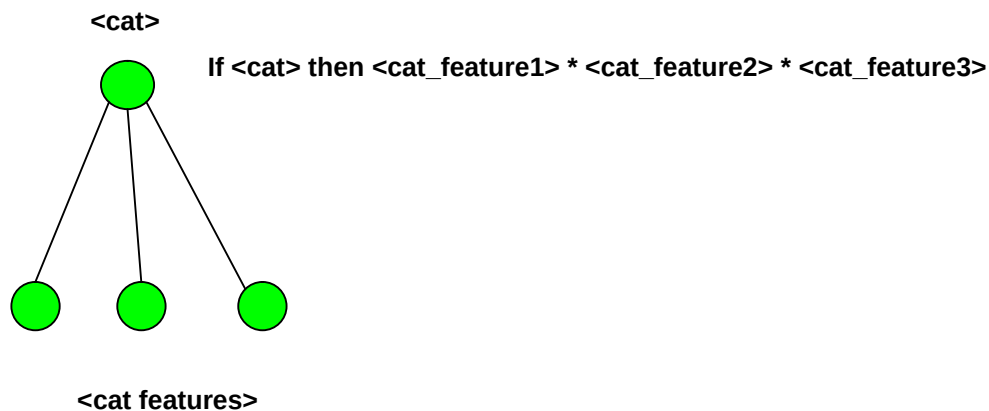


Figure 6.1 An inscriptor parent node and child nodes. Nodes are representing sensory data, not abstract symbols

The lowest level in IM is for sensory interaction with the environment. The nodes at this level are atomic in the sense that they are activated by sensory data that will not be perceived by the model if they are broken down into smaller components (they can be

understood as the smallest perceptible pixel, to give a visual example). Thus at the lowest level of the model every node represents a unique sensory item. Intuitively, the level above consists of nodes that are connected to nodes in the atomic level. A node at this level may be connected to many atomic nodes. If the activation of a subset of its input nodes co-occur frequently, the node “freezes” those connections. Once those connections are frozen, the node will only fire when the activation of the subset of atomic nodes it has locked co-occur again. That is, their collective activation will push it over its own activation threshold. The node is thus trained to represent a co-occurring subset of atomic nodes. If the input nodes are taken to be separate features, the node itself represents the conjunction of those features.

Furthermore, since IM is a model based on “spreading activation,” an active parent node will have a higher activation level than any of its input since it will be receiving activation signals from several nodes. The more active a particular node is, the more “conscious” the IM is of the node and the more it influences IM’s behaviour.³ Hence, the fact that one looks at an apple and sees it whole rather than in visually dissected parts is explained in IM by the higher activation level of the parent node with respect to its children. A decaying function must also be implemented in the model in order for priority to be given to the consistently activated nodes.

In its initial conception, the network of nodes or “inscriptors” needs to be acyclic. This is because deduction is represented by the activation of a high-level node spreading down to its descendents. If at any point the activation proceeds back up, then the process can no longer be called deduction. To give a simple short example: if $A \rightarrow B$, $C \rightarrow B$, one cannot deduce C if A is true. In the graphical representation, both A and C are parents of B , therefore it would conclude C from A if the activation were allowed to spread back up to a parent, thereby violating deduction.

To see why one can depend on deduction process in IM when unidirectional activation is ensured, one simply needs to recall how the initial parent-to-child connections are created. The connections are frozen in way that guarantees that *all* the instances of the parent node that are experienced have all the attributes the child nodes

³ A temporal chain of highly activated nodes is analogous to a thought process in IM.

represent. This is trivial to confirm since the connections are frozen on co-occurring child nodes only.

The problem with such a set-up is that one of the key elements of human cognition—and what a complete IM aims to perform—is abduction. Graphically this means inferring the presence of the parent node by the presence of the child node. In other words, the activation has to spread bottom-up. In order to accomplish this, the spreading of activation must be able to move in both directions. The goal, in effect, is to establish mechanisms that “mix abduction and deduction appropriately.” Within this initial framework, IM has a lot in common with NARS. However, in NARS, as previously noted, there was a redundancy in the way knowledge items were stored, and a system-wide search had to take place in order to find prioritized concepts. In IM, the prioritization is governed by activation levels and unique knowledge items are accessed through connections and are not duplicated.

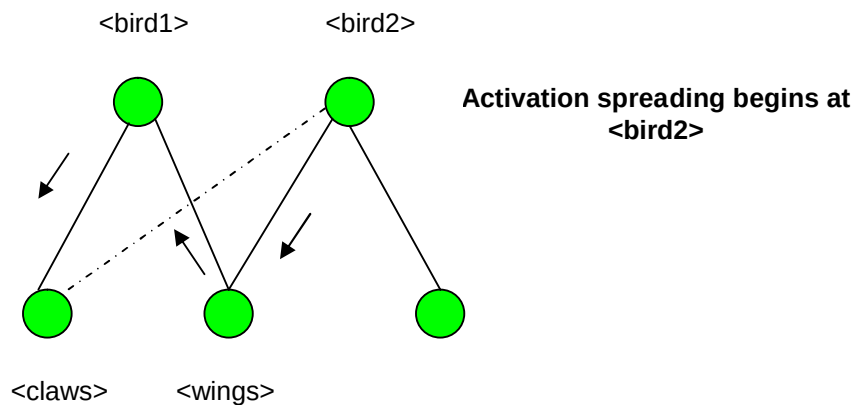


Figure 6.2 Mixing abduction and deduction: Without having $\langle \text{bird2} \rightarrow \text{claws} \rangle$ as an assumed fact, the system infers that $\langle \text{bird2} \rangle$ has claws based on bi-directional activation spreading via a shared node.

Furthermore, the system must know that it has to respond to the conclusions of abduction and deduction differently. Simply put, it has to know that the former should be treated as a guess needing verification, whereas the latter should be treated as a conclusive statement. For example, when someone talks to me about an eagle I can

conclude that it has wings without having to go and verify that conclusion because it is based on assumed facts. However, when somebody talks to me about something that has wings, and I use abduction and “conclude” that he’s talking about an eagle, I will feel obliged to continue to look for more evidence supporting that guess. Only after receiving an external affirmation (being told that it is in fact an eagle) will my conclusion be as reliable as the previous deduction.

Another feature that must be implemented is top-down supervision. Since most of cognition involves memory recall, nodes that were created as a result of sensory input should be able to be activated by higher-level nodes when sensory input is no longer present. For example, the system should be able to activate the node that represents a strawberry while all its child nodes that represent its separate visual features remain inactive. A situation where a parent node is active and its child nodes that represent primitive sensory input are inactive may be taken as being analogous to *imagining* a strawberry, whereas a situation with the parent node and child nodes being all active may be taken as being analogous to *seeing* a strawberry. When top-down supervision is introduced into IM, however, it has to be implemented through activation mechanisms. This brings up the issue of cyclic activation again and the importance of controlling its spread.

There is also a difference between nodes in the higher levels and nodes in the atomic level that affects the functionality of the network and needs to be addressed. As mentioned, each atomic node represents a unique attribute. This is not necessarily the case with all nodes in a higher level. In the second level, for example, it might be the case that there may be several nodes representing the same set of co-occurring features since they may all be connected to the same active subset of nodes. If the input nodes that they don’t share are inactive and the input nodes that they do share are active then this particular set of higher level nodes will be locked onto the same active input nodes. If two nodes represent the same collection of sensory items, then when one of the nodes is active it will always be the case that the other node is active as well. This creates a problem for representation in the next level. If a node in the third level has connections to second-level nodes that represent the same sensory information as each other, there is a high likelihood that it will lock onto those nodes since their activations will always co-

occur. Hence what would superficially appear to be the conjunction of two separate features would actually merely be the repetition of a single feature. This would collapse the representational capacity of the model. This is also one of the issues that have to be addressed in SINBAD as well as CT.

One “quick fix” would be to have a procedure that checks that nodes on the same level are not locked onto the same input, eliminating the repetitive nodes in the process. This is the sort of thing we would want to avoid. The more global procedures are introduced in order to compensate for unwanted local interaction, the more pointless it is to base the model on such local interaction. One possible local solution would be to have connections between nodes on the same level in order to inhibit each other in the event that they lock onto the same input.

This is how it could work: Before a node locks onto the input nodes that have been repeatedly activated together, it sends signals to all its child nodes that are to be frozen, activating them in the process. Then it temporarily blocks its incoming connections from its child nodes. Any other node at the same level that is exclusively locked on the same set of inputs will then become activated, and the node in question will be alerted to the fact through the lateral connection it shares with its sibling nodes. Therefore, when the activation of a sibling node coincides with its testing of the child input nodes the node in question aborts the freezing process.

Lateral Repetition Testing

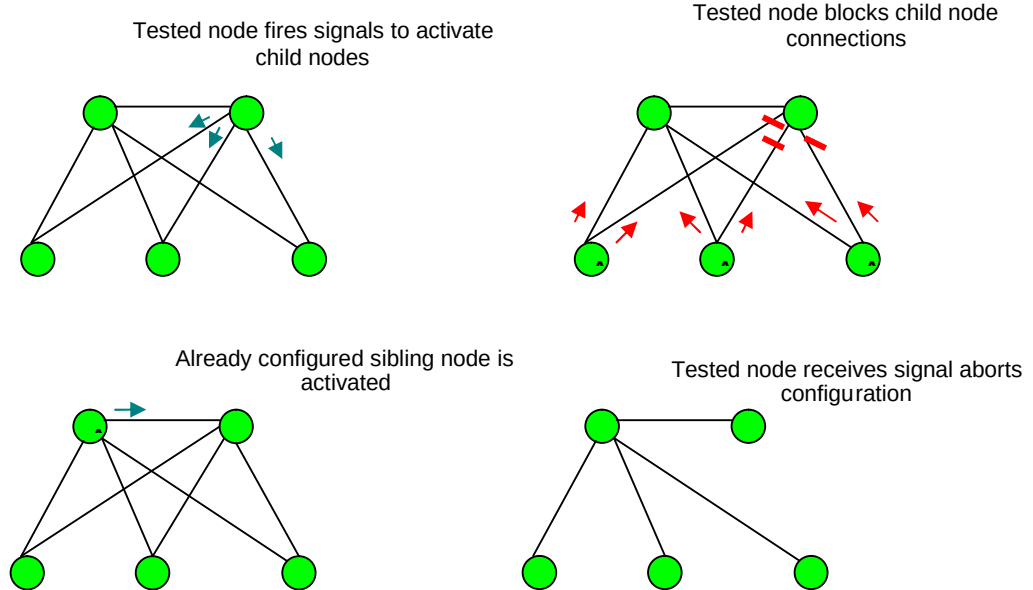


Figure 6.3

The problem with this approach is that the testing may coincide with the activation of a sibling node that has already locked on a completely independent set of input nodes and is responding to their current activation. Since a lateral open connection exists, the node that is about to freeze its connection would receive a signal from the sibling node. If this were the case, the node in question would wrongly conclude that the sibling node was representing the same set of input nodes and would needlessly abort the freezing process. This is why the freezing of connections should take place when there is a low density of primitive sensory input (similar echoes in Hecht-Nielsen's CT, where connections are frozen during sleep). One other—and more important—problem is that testing and freezing phases occurring in parallel among sibling nodes may need to be sequential. For example, if two nodes that share the same co-occurring input activations run the testing phase in parallel before freezing their connections, they will both be blocking their own input connections, and hence both will not receive the other's signal because neither will be activated.

Simultaneous Lateral Repetition Testing

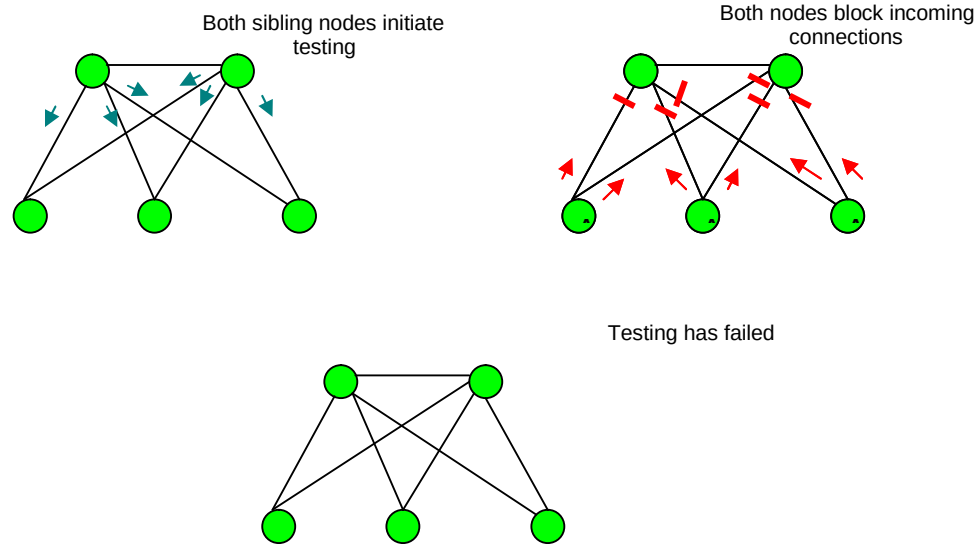


Figure 6.4

Alternatively, one may endow the parent node of the two sibling nodes the responsibility of inhibiting representation redundancies. Suppose two second-level nodes are already locked onto the same atomic nodes. Further suppose that a third-level node wants to freeze its connections to these nodes because of their co-occurring activations, but before it locks onto the two nodes it wants to make sure they don't represent the same subset of input nodes.

Parent Node Repetition Testing

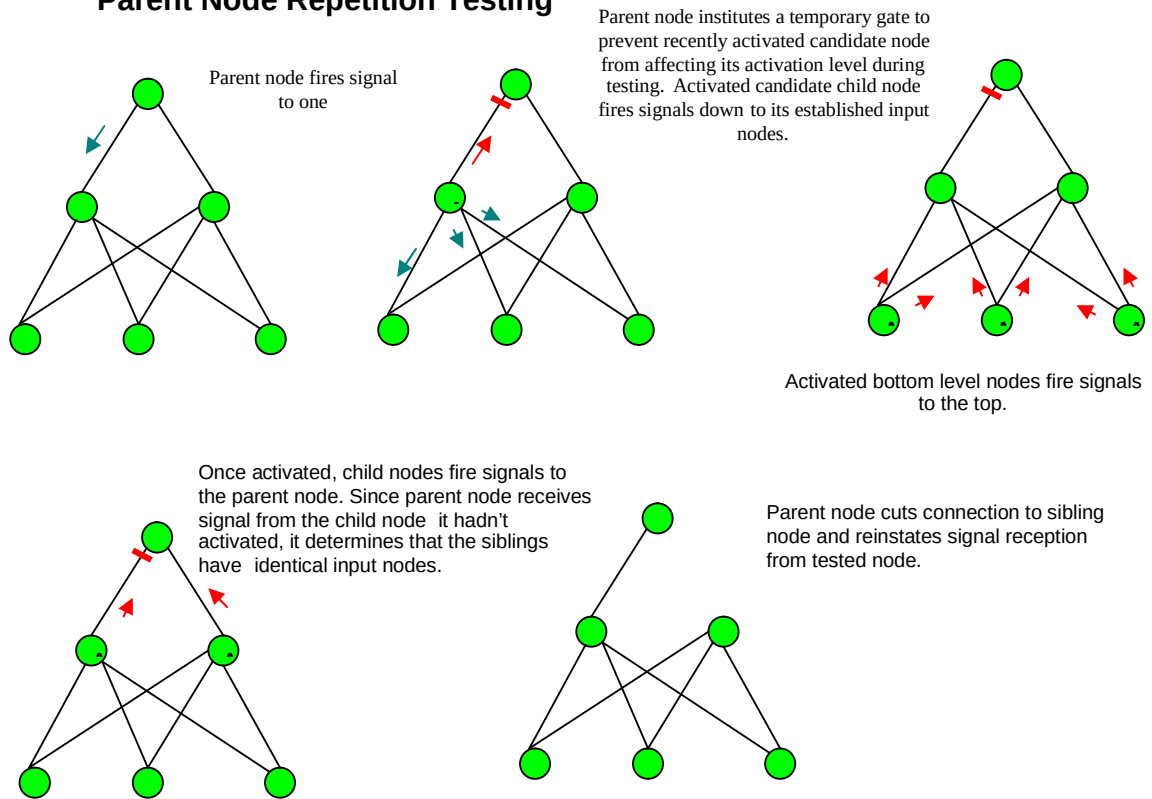


Figure 6.5

One downward activation protocol the parent node can establish is as follows: It can fire a signal to one of its input nodes, which in turn would activate its own input nodes that it has frozen connections to. After the parent node (third level) fires a signal down to one of its input nodes, it closes down the incoming connection from that node so as to temporarily prevent the activated input node from affecting its own activation level. The input node (second level) in its activated state fires signals through its own frozen connections, activating its own child nodes (atomic level). Now if another second level node connected to the third level parent is locked onto the same atomic nodes as the node the parent itself activated, the sibling second level node in question will be activated and fire a signal to the parent node. Thus, if the other co-active second level nodes do not represent the same set of nodes as the node the parent has just activated, then the parent will not be expecting any incoming signals. However, if it receives any signals from its other nodes in this testing phase it will know that such a node shares the same set of inputs as the node that itself has activated, and thus sever the connection with that node.

Thus, three levels of nodes are involved in the testing: the parent node that has yet to freeze its connections, the child nodes that already have frozen connections and the grandchild nodes at the other end of those connections (atomic level nodes in this case). Again, such testing would need to occur when the level of incoming sensory input is relatively low (when the cognitive agent is in a state of “sleep”), otherwise the input nodes being tested may be activated by the sensory input instead of the activations the parent node has triggered.

Once repetitive symbols have been eliminated through local processes, the co-occurrence of activations can establish an increasing set of complex of symbols, yielding, at the top of the hierarchy, representations of specific situations. As the agent experiences these specific situations through experience they will decay with each time step, but features that are common to many will remain vibrant in the inscriptor knowledge representation structure because of their repeated activation across many observed situations. This process in IM emulates mental abstraction.

Chapter 7

Conclusion

It is very unhappy but too late to be helped, the discovery we have made, that we exist. That discovery is called the Fall of Man. Ever afterwards we suspect our instruments. We have learned that we do not see directly, but mediately, and that we have no means of correcting these colored and distorting lenses, or of computing the amount of their errors. Perhaps these subject-lenses have a creative power...Once we lived in what we saw, now, the rapaciousness of the new power, which threatens to absorb all things, engages us. – Ralph Waldo Emerson

The construction of an artificial general cognitive system (AGC) is a long and arduous process which requires insight to predict the limitations of the mechanisms one designs. The significant increase in computational power and the use of this power to map the brain has given some the illusion that it is a matter of waiting for a “computational singularity” that will propel otherwise poorly performing systems into adept AGC machines. In this view, when exponential algorithms are processed in near-constant time the ability to function in a continuous sensory environment will be greatly improved, and since much of cognition relies on learning from the interaction with the environment, this will make the ultimate goal much more attainable. However, the construction of a system for general human cognition is a design problem, and the reliance on computational power may blind one to the difficulties that exist in this regard. As Ellen Ullman put it, “There’s no Moore’s Law for software. Quite the contrary: as systems increase in complexity, it generally becomes harder to write reliable code.”

Our motivation for looking into philosophical and neuroscientific research has to do with their practical relevance in constructing a comprehensive cognitive agent. A cognitive system having symbols semantically grounded is not merely of theoretical importance, but of practical necessity. Rather than simply be about what cognitive agents

label certain concepts as, it embodies the notion that a cognitive must categorize “water” *the way we do*, and engage in similar action when confronted with the same concepts. Similarly, modularity in human cognition has to be addressed. If a system is composed of many independent modules, then as Fodor pointed out there has to be an entirely different module to ensure communication between all of them. In this regard, the examination of dissociation and association of different cognitive functions in Chapter 3 indicates that often times what appears on the surface to be dissociative is actually quite associative. Thus the challenge then is not only to explain the association of two cognitive functions, but to explain why we also subjectively regard them as dissociative (e.g. short-term and long-term memory). Our view is that increased modularity occurs through development and should not be hardwired into the system. What needs to be proposed is an architecture that allows agents to begin from general processing and to develop modular states as learning progresses. Many of the cognitive models that exist today either are general frameworks that don’t modularize through development or they are models that are simply pre-programmed with the specific modules in place.

Another issue is that our human preference to solve tasks through mental simulation instead of rule-based logic, as mentioned in Chapter 3, should alert those who are constructing cognitive agents that are designed to have a preference for solving tasks with “high-level” cognitive capabilities whenever possible. After all, it is easy to fall into that trap when it is often times hard to avoid regarding cognitive development as a transition from low-level functions to high-level functions. In this case, “high-level” logic formulations are inhibited in favor of “low-level”: visual simulations.

Furthermore, connectionist models are advantageous because they are able to account for the developmental stages of cognition. Thus, although other approaches may garner success with respect to what the mind *does*, this approach will also be able to address *how* the mind works. In conclusion, by taking into account the advantages of each of the cognitive models described in this thesis and the philosophical and neuroscientific considerations presented here, we hope to have prepared ourselves for further developing our own comprehensive cognitive framework that was introduced here in our future research.

References

- [1] Newell, A.; Shaw, J.C.; Simon, H.A. "Report on a general problem-solving program." *Proceedings of the International Conference on Information Processing*. pp. 256-264, 1959.
- [2] Smolensky, P. "On the proper treatment of connectionism." *In Brain and Behavioral Sciences*, 11, 1988.
- [3] Dietrich, E. "It Does So: Review of The Mind Doesn't Work That Way: The Scope and Limits of Computational Psychology." *AI Magazine*, Vol 22, No. 4, 2001, pp. 141-144.
- [4] Fodor, J. *In Critical Condition*, MIT Press, 1998.
- [5] Chalmers, D. "Why Fodor and Pylyshyn Were Wrong: The Simplest Refutation." *Proceedings of the Twelfth Annual Conference of the Cognitive Science Society*, 1990, pp.340-347.
- [6] Wallis, C. "Computation and Cognition." *Journal of Experimental Artificial Intelligence*. Vol 15, 2003, 177-193.
- [7] Harnad, S. "Symbol Grounding and the Origin of Language." *Computationalism: New Directions*. Cambridge, MA: Press: 2002, pp 143-158.
- [8] Klahr, D. "Commentary: new kids on the connectionist modeling block." *Developmental Science*, Vol. 7 No.2, 2004.
- [9] Hegarty, M. "Mechanical reasoning by mental simulation." *Trends in Cognitive Sciences*, Vol. 8, No. 6, June 2004.
- [10] Carpenter, M.; Call, J.; Tomasello, M. "Twelve- and 18-month-olds Copy Action in Terms of Goals." *Developmental Science* Vol. 8 No. 1, 2005.
- [11] Apperly, I. A.; Samson, D.; Humphreys G. W. "Domain Specificity and Theory of Mind: Evaluating Neuropsychological Evidence." *Trends in Cognitive Science*, Vol. 9, No.12, 2005.
- [12] Frith, U; Frith, C.D. "Development and neurophysiology of mentalizing." *Philos. Trans. R. Soc. Lond. B Biol. Sci.* 358, 2003, pp. 459–473.
- [13] Langdon, R *et al.*, "Disturbed communication in schizophrenia: the role of poor pragmatics and poor mind-reading." *Psychological Medicine*.. 32, 2002, pp. 1273–1284.

- [14] Maunsell, J. H.R. "Neuronal Representation of Cognitive State: Reward or Attention?" *Trends in Cognitive Sciences*, Vol.8, No. 6, 2004.
- [15] Schultz, W. "Multiple reward signals in the brain." *Nature Reviews Neuroscience*, 1, 2000, pp. 199–207.
- [16] Treue, S. "Neural correlates of attention in primate visual cortex." *Trends in Neuroscience*. 24, 2001, pp. 295–300.
- [17] Davelaar, E.J. *et al.* "The demise of short-term memory revisited: empirical and computational investigations of recency effects." *Psychological Review*. 112, 2005, pp. 3–42.
- [18] Speer *et al.*, "Strategy-dependent changes in memory: effects on behavior and brain activity." *Cognitive, Affective, and Behavioral Neuroscience*, 3, 2003, pp. 155–167.
- [19] Ranagath, C; Blumenfeld, R. S.."Doubts about Double Dissociation Between Short-term and Long-term Memory." *Trends in Cognitive Sciences*, Vol. 9, No. 8, 2005.
- [21] Maia, T; Cleermans, A. "Consciousness: converging insights from connectionist modeling and neuroscience." *Trends in Cognitive Sciences*, Vol. 9, No. 8, 2005.
- [20] Zelazo, P. D. "The Development of Conscious Control in Childhood." *Trends in Cognitive Sciences*, Vol. 8, No.1, 2004.
- [22] O'Reilly, R.C *et al.* "A Biologically-based computational model of working memory." In *Models of Working Memory: Mechanisms of Active Maintenance and Executive Control* (Miyake, A and Shah, P., eds), pp.375-411, Cambridge University Press.
- [23] Frank, M. J *et al.* "Interactions Between frontal cortex and basal ganglia in working memory: a computational model." *Cognitive, Affective, and Behavioral Neuroscience*, 1, 137-160.
- [24] Minsky, M. *The Emotion Machine (Draft)*. <http://web.media.mit.edu/~minsky/>, August 2006.
- [25] Singh, P. *EM-ONE: An Architecture for Reflective Commonsense Thinking*. Ph. D. Dissertation. MIT, 2005.
- [26] Singh, P.; Minsky, M. "An Architecture for Cognitive Diversity." *Visions of Mind*, Darryl Davis (ed.), London: Idea Group Inc, 2005.

- [27] Wang, P. *Rigid Flexibility: The Logic of Intelligence*. Springer, Forthcoming, 2006.
- [28] Wang, P. "Experience-Grounded Semantics: A theory for intelligent systems." *Cognitive Systems Research*, Vol.6 No.4, 2005.
- [29] Wendelken, C.; Shastri, L."Connectionist Mechanisms for Cognitive Control." *Neurocomputing* 65, 2005, pp. 663-672.
- [30] Goertzel, B; Pennachin, C. "The Novemente Artificial Intelligence Engine." In *Artificial General Intelligence* (Goertzel, B.; Pennachin, C. eds.). Springer, Forthcoming, 2006
- [31] Hecht-Nielsen, R. "Cogent Confabulation." *Neural Networks*,18, 2005, pp. 111-115.
- [32] Hecht-Nielsen, R. *A Theory of Cerebral Cortex*. UCSD Institute for Neural Computation Technical Report #0404. December, 2004.
- [33] Ryder, D; Favorov, O. "The New Associationism: A Neural Explanation for the Predictive Powers of Cerebral Cortex." *Brain and Mind* 2(2), 2001, pp.161-194.
- [34] Ryder, D. "SINBAD Neurosemantics: A Theory of Mental Representation." *Mind and Language*, Vol.9, No. 2, 2004.
- [35] Piazza, M; Dehaene, S. "From Number Neurons to Mental Arithmetic: The Cognitive Neuroscience of Number Sense." In *The Cognitive Neuroscience, Third Edition* (Gazzaniga ed). MIT Press, 2004.
- [36] Coward, L. A. "Accounting for episodic, semantic and procedural memory in the recommendation architecture cognitive model." *Proceedings of Ninth Neural Computation and Psychology Workshop*, 2005.
- [37] Davenport, D. "Inscriptors: Knowledge Representation for Cognition." In *Proceedings of ISCIS-8* (Gün, L, Önvural, R.,Genelbe, E. eds.), Istanbul. November, 1993.

Further Reading

Fodor, J. *The Mind Doesn't Work That Way. The Scope and Limits of Computational Psychology*, MIT Press, 2000.

Chalmers, D. *Philosophy of Mind: Classical and Contemporary Readings*. (Editor). Oxford University Press, 2002.

Dennett, D. *Consciousness Explained*. Penguin Books Ltd, 1993.

Dietrich, E; Markman, A.B. "Discrete Thoughts: Why Cognition Must Use Discrete Representations." *Mind and Language*. Vol 18, Issue 1, 2003.

Taddeo, M.; Floridi, L. "Solving the Symbol Grounding Problem: a critical review of fifteen years of research." *Journal of Experimental and Theoretical Artificial Intelligence*, Vol. 17, No.4, 2005.

Hebb, D.O. *The Organization of Behavior*. John Wiley, New York, 1949.

Mauk, M. D.; Bunoman, D. V. "The Neural Basis of Temporal Processing." *Annual Review of Neuroscience*, Vol 27, Issue 1, 2004.

Wang, H.; Fan, J.; Johnson, T. "A Symbolic Model of Human Attentional Networks" *Cognitive Systems Research*, 5, 2004.

Coward, L. A.; Gedeon, T. D.; Ratanayake, U. "Algorithms for Learning Complex Combinations of Operations." *Proceedings of the IEEE International Conference on Fuzzy Systems*, 2004

Coward, L. A. "Accounting for episodic, semantic and procedural memory in the recommendation architecture cognitive model." *Proceedings of Ninth Neural Computation and Psychology Workshop*, 2005.

Schmidhuber, J. "Optimal Ordered Problem Solver." *Machine Learning*, 54, 2004.

Schmidhuber J. "Gödel Machines: Self-Referential Universal Problem Solvers Making Provably Optimal Self-Improvements." <http://www.idsia.ch/~juergen/gmweb4/gmweb4.html>, December 2004.