FAKULTÄT FÜR !NFORMATIK

Faculty of Informatics

Design and Implementation of an Agent Architecture combining Emotions and Reasoning

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Computational Intelligence

eingereicht von

Janos Tapolczai, BSc.
Matrikelnummer 0825077

an der Fakultät für Informatik

der Technischen Universität Wien

Betreuung: a.o. Univ.-Prof. Dr. Hans Tompits

Wien, 9. Mai 2016

Janos Tapolczai	Hans Tompits

Erklärung zur Verfassung der Arbeit

Janos Tapolczai, BSc.
Address

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 9. Mai 2016

Janos Tapolczai

Acknowledgements

This thesis is represents my first, but, I hope, not last, attempt at an artificial intelligence and I would like to use these lines to express my gratitude to some of the people who have helped me along the way.

I would, first of all, like to thank my mother for tirelessly pursuing my education from my early years on, instilling in me the value of bettering oneself through education. She went well above and beyond her duty in her efforts and without her, I would not be where I am today.

Another person I would like to mention is Matthias Österreicher, who always was a good friend and an attentive listener, as well as an inspiration through his perseverance and his strength of character.

Then there are Friedrich Nietzsche, the eternal optimist, and Douglas Hofstadter who, with their books *Also Sprach Zarathustra* and *Gödel, Escher, Bach*, opened my eyes to a world I had never even suspected to exist.

Thanks also go out to my friends Michael Abseher, Jürgen Cito, Manuel Mertl, Bernd Rathmanner, Gerald Schermann, Manuel Wiesinger, and Simon Wolfsteiner for their support and friendship through these years, for the interest they showed, and the feedback they provided.

Furthermore, I would like to sincerely thank my professors Hans Tompits and Rudolf Freund for their advice and their forbearance in discussing my admittedly idiosyncratic ideas at length.

Abstract

We design and implement a hybrid agent architecture that combines emotional reactions to stimuli with reasoning about the consequences of actions as a means of developing effective behaviour in a toy world. The fundamental task of our agents is to survive by periodically finding and eating food and by dealing with hostile predators, either through flight or through flight. Agents are free to cooperate, antagonize, or ignore each other and agents with different emotional profiles will pursue different strategies.

The thesis begins with theoretical investigations of different models of computations and their relation to the biological brain. Our assumption is that the brain's function is, to some degree, analogous to a collection of white boxes, observable to each other. Accordingly, our agents are modelled as a collection of loosely coupled components which communicate with each other through messages. Any component is free to read any message and components have no information about which other components read the messages which they insert into the agent's message space.

The affective evaluation of its environment forms the basis of each agent's decision-making, though it is complemented by a belief generator which makes inferences about future world-states resulting from certain choices. Though evaluation of these future states as well, agents can optimize their behaviour, as they can foresee likely positive or negative consequences of their actions.

The thesis ends with an evaluation of individual behaviour as well as a population-based evaluation: we evaluate our agents qualitatively by placing agents in a number of simple scenarios and observing whether they perform tasks like collecting food or avoiding predators. After that, we evaluate them in a population-based manner by placing various populations into larger scenarios and recording the survival of different personality types over time.

Kurzfassung

Wir entwerfen und implementieren eine hybride Agentenarchitektur, welche emotionale Reaktionen auf Stimuli mit dem Schließen über den Folgen von Aktionen zum Zwecke der Ausarbeitung effektiven Verhaltens in einer einfachen Welt kombiniert. Die fundamentale Aufgabe unserer Agenten ist es, durch das Finden und Essen von Nahrung, sowie der Handhabung von Feinden durch Kampf oder Flucht, zu überleben.

Die Arbeit beginnt mit einer theoretischen Untersuchung mehrerer computationaler Modelle und ihrem Verhältnis zum biologischen Gehirn. Unser Ansatz ist, dass das Gehirn zu einem gewissen Grad analog zu einer Ansammlung von *White Boxes* ist, die die Arbeit der jeweils anderen beobachten können. Demgemäß sind unsere Agenten als lose gekoppelte Komponenten modelliert, die über Nachrichten miteinander kommunizieren. Jede Komponente kann jede Nachricht auslesen und keine Komponente weiß, welche anderen die Nachrichten lesen werden, welche sie in den zentralen *Message Space* des Agenten einfügt.

Die affektive Evaluierung ihrer Umgebung ist die Grundlage der Entscheidungsfindung der Agenten, sie wird aber ergänzt durch einen *Belief Generator*, der zukünftige Zustände der Welt als Folge bestimmter Aktionen simuliert. Durch die Bewertung dieser zukünftigen Zustände können Agenten ihr Verhalten anpassen und sowohl die negativen als auch die positiven Folgen ihrer Aktionen absehen.

Die Arbeit endet mit einer Evaluierung des individuellen Verhaltens sowie einer populationsbasierten Bewertung: Wir evaluieren die Agenten zuerst individuell, indem wir sie in simple Beispielszenarien situieren und testen, ob sie in darin einfache Aufgaben, wie das Sammeln von Essen oder die Flucht vor Feinden, ausführen. Danach bewerten wir sie auch auf Populationsbasis, indem wir verschieden zusammengesetzte Gruppen von Agenten in einer größeren Welt platzieren und die Überlebensrate verschiedener Persönlichkeitstypen beobachten.

Contents

CHAPTER 1

Introduction

The history of AI is marked by vacillations between two paradigms: the biological and the ideal one. The biological school of thought subsumes ideas like connectionism, which envisions the mind as an interconnected system of simple components, generally single neurons, and tries to build AIs via neural networks. Opposed to this view stand those schools that view the mind as an abstract machine: computationalism holds it to be an information processing system that deals in the manipulation of symbols, and which possesses structures like subsystems, rules, and syntax.

In this work, we shall build upon this latter, computationalist approach and, more specifically, upon the work of Marvin Minsky and Aaron Sloman, who have very much advocated the idea of the mind as a control system with an intelligible structure in books like *The Emotion Machine* [Min06], *Society of Mind* [Min88], *The Mind as a Control System* [Slo93], and *What Sort of Control System Is Able to Have a Personality?* [Slo97]. One of the running themes in Minsky's and Sloman's work is the criterion of *evolvability*: it is not sufficient, they argue, to merely propose some ideal reasoning apparatus; if artificial intelligences faithful to their biological inspirations are to be constructed, we must structure them similar to the structure of biological minds — and that is best accomplished by thinking about what sorts of subsystems might have evolved in what order, in what way, and for what task. The human-level AI must therefore replicate the brain's functions, warts included.

In the rest of this thesis, we shall pursue this idea, with special attention given to the interaction between emotions and reasoning in the sense of logical deduction. The result will be a small cognitive architecture that combines both, but privileging neither. Both Sloman and Minsky have sketched such architectures in the past, and ours will be similar to these in its broad outlines; however, in our preliminary considerations, we will discuss two specific issues:

1. *How* might certain subsystems have evolved? In particular: In what order capabilities did like pain, anger, deduction, or introspection come about and were they simply re-purposed from other, existing components, or were they created, so to speak, from scratch?

2. We look at the interactions between subsystems. How could, in evolutionary terms, something like a discrete subsystem, develop? In what manner do different subsystems in the brain communicate? Is there some universal, perhaps symbolic, communication protocol — or a suite of protocols?

These are deep and open questions and our goal will not be to answer them here, but we will make certain assumptions based on knowledge from neurology and evolutionary biology which will, in turn, motivate the design of the architecture of our toy artificial intelligence.

White-box model. The first issue will be discussed in the next chapter in which we give a hypothetical description of the brain as a collection of white boxes. The brain evidently possesses large-scale structure, but its computational model of massively connected neurons is very different than the ones present in man-made programming languages [Arb02, Section "Introducing the Neuron"] and our white-box model, although supported by circumstantial evidence from neurology, is primarily a working hypothesis. Conventional structural programming models programs via functions and procedures that work as black boxes, which are called and return an answer without their caller being aware of their internal workings; we propose that one might gain some useful insights by conceiving of the brain as a collection of white boxes, wherein components can interact with and observe each other's functioning. The reasoning behind this notion is that, in a massively interconnected system such as the brain, there are no strict boundaries between parts that would be analogous to the narrow interaction between a function and its caller in a computer program; rather, typical neurons have one axon that branches into thousands of synapses that connect to other neurons [Arb02, p. 4] and can, moreover, make new connections to other neurons at any time.

We will transport this conception of a white-box model into software by modelling our artificial intelligence as a collection of loosely coupled components that do not directly call each other but rather communicate by putting messages into and reading from a message space.

Reasoning and emotions. After this groundwork, we will come back to the large-scale systems, specifically imagination, its relationship to affect, and reasoning. We contend that imagination — perceiving events that are not happening — is the antecedent of abstract reasoning, and that they both were gradually evolved from older functionality, rather than either being *sui generis*. With support from fMRI studies, we conceive of imagination as a re-purposing of sensory perception; the same neural circuitry that had been used for the processing of physical stimuli like sights and sounds came to be used for

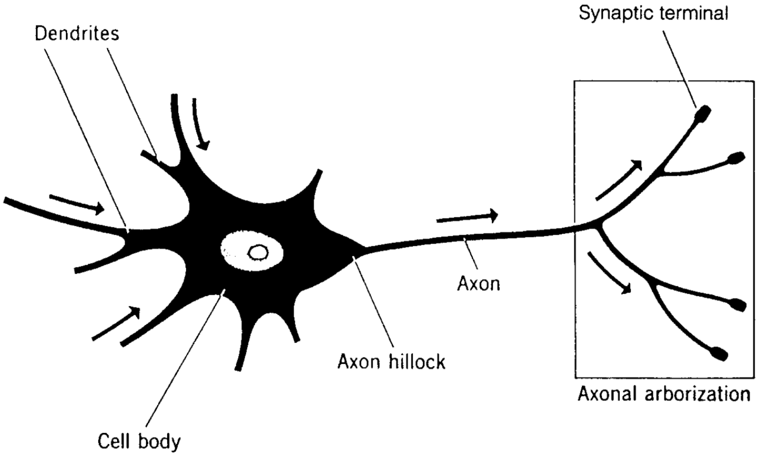


Figure 1.1: Schematic of a neuron. From the caption of the source [Arb89, p. 52]: *A "basic neuron" abstracted from a motor neuron of* [*the*] *mammalian spinal cord. The dendrites and soma* (*cell body*) *constitute the major part of the input surface of the neuron. The axon is the "output line". The tips of the branches of the axon form synapses upon other neurons or upon effectors* (*although synapses may occur along the branches of an axon as well as at the ends*).

the processing of those which were internally generated, with specific inhibitory signals prevent the imagined interfering with the real.

The concept of imagination being neurologically related to actual perception implies that emotions play the same role in both — real as well as imagined situations evoke emotional responses, but whereas real ones influence immediate behaviour, imagined ones chiefly influence planning. Being able to think ahead, to play out scenarios in one's head allows one to effectively ask questions like "How would I like this?" and "Would this be harmful?". Such questions confer upon organisms the ability to plan and to select beneficial courses of action, but without the need for explicit utility functions; their brains merely re-use their old circuitry in a novel way.

Abstract reasoning, too, is assumed to be an incremental development of pre-existing capabilities: with the ability to simulate physical worlds in place, brains were able to develop the ability to simulate *symbolic* ones. The same mechanism that had dealt with the processing of internally generated physical stimuli was employed in the simulation and mental manipulation of things like numbers, glyphs, propositions, or the minds of other individuals.

Toy AI. The second part of the thesis puts the above considerations into practice. We derive an architecture that is influenced by the work of Minsky [Min88] and Sloman [Slo93, Slo97], as well as of Sander et al. [SGS05] and Gadanho and Hallam [GH01]. This AI will consist of subsystems for affect, belief generation, decision-making, and perception.

For expediency's sake, we will not implement a true white-box model; rather, we will loosely couple the components so that each inserts messages into a common message space, from which other components may take what they desire. While implementing a fundamentally new computational model would be interesting, it would also be beyond the scope of the thesis. Instead, we loosely couple the components: each one will insert messages into a common container from which others may take whatever messages they can interpret.

The affective system will read messages coming from external perception and belief generation, and create emotional responses to them. These, in turn, will be read by the decision-maker and guide its actions. Actions can be external, in which case they cause the agent to act, or internal, in which case they tell the belief generator which possible world to simulate.

In terms of software engineering, our model has similarities, both to the Actor model developed by Carl Hewitt et al. [HBS73], and to publish/subscribe architectures [BJ87] — although more as a concession to practicality and less because of a similarity to their theories. The theoretical basis of our implementation is the postulate that the components of the brain function as white boxes and that other components may listen in on their activity, so to speak. Since this is diametrically opposed to the traditional idea of the procedure/function as a black box, which nigh every programming language follows, we compromise and model the cognitive structure as a mesh of loosely coupled components communicating via passing.

World and evaluation. The so constructed toy AIs will be placed in a simple grid environment that was inspired by the Wumpus world [RN10]. This world is populated by a number of agents, Wumpuses who function as predators, plants which provide food that the agents periodically have to consume, gold, which they can pick up and trade with each other, and dangerous chasms that kill any agent that steps into them. In this world, each agent can perceive and do a variety of things: it can see the cells ahead of it to a certain distance, smell the stench of Wumpuses which function as predators, and feel the gust around fatal chasms. It can move around, turn to rotate its sight cone, take the food from plants, attack other individuals, trade items like food or gold, and communicate with fellow agents by sending gestures in the form of arbitrary strings. The environment is so designed that the information available to the agent is minimal: it does not know about the disposition of other agents, the way in which they will interpret its gestures, the global topology of the world, what items other agents have, or any other information to which a real animal in a real environment would not have access.

The aim of this scenario is to test whether the agent architecture is viable at all and, if so, which affective profiles are more successful than others. While each agent works in the same way, they can be parametrised in their emotional reactions to stimuli and thereby exhibit different personalities, in a manner of speaking.

Structure of the thesis. The thesis consists of two large segments. The first one is a theoretical argument and empirical data supporting it in Chapter 2. It deals with the origin and purpose of neural systems, their evolution (insofar as is known), the components of which they are likely comprised, and how these could have come about. The constituent sections are

- Section 2.1, describing relevant previous work in the area of artificial intelligence,
- Section 2.2, containing the general evolutionary story,
- and Section 2.3, which describes our proposed white-box computation model of the cognition of humans.

The second segment begins with Chapter 3, wherein we discuss our model in greater detail. Section 3.1 explains how to represent white boxes as loosely coupled, interacting components and Section 3.2 introduces mathematical language to talk about such components in a formal way.

Chapter 4 then deals with subsystems as well as architectural patterns of which we will make use in the implementation. Chapter 5 then specifies the world in which our agents will have to survive, as well as the components and the architecture of our agents itself.

Thereafter, we present the results of our experimental evaluation in Chapter 6. Finally, the thesis closes with Chapter 7, which contains the conclusion, as well as possible future work.

CHAPTER 2

Preliminary Considerations

We begin this chapter with a look at related work in the field of artificial intelligence: approaches, architectures, implementations, and philosophies on which later parts of the thesis draw. After that, in Section 2.2, we will go through biological and neurological considerations that shall ground the models and architectures proposed herein. Lastly, Section 2.3 will outline a white-box model of cognition which assumes various parts of a central nervous system observing and influencing each other relatively freely. We specifically contrast this with the black-box model of structured programming, where the inner workings of functions and procedures are oblique to the caller.

2.1 Related Work

A great deal of work has already been done with the aim of creating general artificial intelligence approaches. Of special interest to us are *cognitive architectures*, the work of Alan Sloman, and the *nouvelle AI*.

Cognitive architectures. This thesis falls into the category of cognitive architectures and the integrated approach to AI, pioneered by people like Rodney Brooks and his subsumption architecture, and [Bro86], Douglas Hofstadter, who famously wrote about many aspects of AI in *Gödel, Escher, Bach* [Hof79], and who created the *Copycat* analogy-making program [Hof96]. Another important work is the *Hierarchical Control System* of James Albus [Alb96], in which cognitive tasks are organised hierarchically and delegated by nodes on higher levels to those on lower ones (this is similar to the mesh-like organisation of components described in Section 3.2, and to the layered structure of Minsky's *The Emotion Machine* [Min06]). The organisation described by Albus [Alb93] is, moreover, very similar to the one in Section 5, with world simulator, belief generator, sensory perception, and knowledge base (herein called "memory") modules being mostly analogous. Another large and conceptually similar system is Carnegie-Mellon's 4CAPS

[Uni], which posits small, relatively simple components, individually doing simple tasks, and having only limited computational resources. Most of 4CAPS's stated principles can be recognised in the coming sections [Uni, Operating Principles of 4CAPS]:

> *0. Thinking is the product of the concurrent activity of multiple brain areas that collaborate in a large-scale cortical network.* [. . .]
>
> *1. Each cortical area can perform multiple cognitive functions, and conversely, many cognitive functions can be performed by more than one area.*
>
> *2. Each cortical area has a limited capacity of computational resources, constraining its activity.*
>
> *3. The topology of a large-scale cortical network changes dynamically during cognition, adapting itself to the resource limitations of different cortical areas and to the functional demands of the task at hand.*
>
> *4. The communications infrastructure that supports collaborative processing is also subject to resource constraints, construed here as bandwidth limitations.*
>
> [. . .]

The probably earliest example of a cognitive architecture was Allen Newell's and Herbert A. Simon's *Logic Theorist*, created in 1955 [Cre93, p. 44]. Simon's theory of bounded rationality [GS01] — the idea of finding a merely satisfactory solution instead of a (provably) optimal one — is very similar to the loop between belief generation and evaluation described in Section 5. In both cases, agents with limited information search heuristically for the first solution that they find acceptable. Unlike exhaustive search methods (e.g. A*), this does not guarantee the best possible results, but it is much more cost-effective and closer to the way real humans solve problems. In spirit, this is also similar to the *Procedural Reasoning System* of Michael Georgeff et al. [IGR92], which is based on the belief-desire-intention (BDI) model[RMPG95, Bra87]. Much theoretical work has been done on BDI, but it is only tangentially related to this thesis.

Sloman. Many of the fundamental ideas in this thesis can be found in Alan Sloman's works [Slo93, Slo97, Slo99, Slo01, Slo], especially in *Beyond shallow models of emotion* [Slo01]. Therein, he formulated the criterion of evolvability in the context of cognitive architectures and postulated the possibility that nervous systems may be chaotic (but not unorganised). The agent architecture in Section 5 substantially resembles his, though it was not taken from there. The similarity is, however, indicative of a great deal of shared thought.

Implementation. In terms of software engineering, our model has similarities, both to the Actor model developed by Carl Hewitt et al. [HBS73], and to publish/subscribe architectures [BJ87] — although more as a concession to practicality and less because of a similarity to their theories. The theoretical basis of our implementation is the postulate that the components of the brain function as white boxes and that other components may listen in on their activity, so to speak. Since this is diametrically opposed to the traditional idea of the procedure/function as a black box, which nigh every programming language follows, we compromise and model the cognitive structure as a mesh of loosely coupled components communicating via passing. This description is reminiscent to the Actor model, although there are differences[1]: in the Actor model, the topology of the network may change through the creation of new actors, and messages are always passed from one source to known targets (via addresses). In our model, on the other hand, there is no topology in a strict sense; messages are put into a global message storage and every component is free to consume any message it deems relevant. Senders do not know who will read their output, and consumers do not know the sources. This arrangement can be seen as a particularly loose variant of a publish/subscribe architecture, in which the source and the target of a message are completely unaware of each other, and in which there are no specific channels to which one may subscribe. The only criterion by which messages may be accepted or rejected is their content.

Nouvelle AI. Lastly, the overall goal, if not the method, of this thesis echoes that of the *nouvelle AI* of, again, Brooks [Bro91], who claims that

> *the Von Neumann model of computation has lead Artificial Intelligence in particular directions. Intelligence in biological systems is completely different.*

The nouvelle AI approach stands in contrast to traditional AI in that it does not aim for human-level performance at specific tasks, but rather for the faithful reproduction of the behaviour of lower animals like dogs [Cop]. Brooks might be closer to the biological realities in his desire to abandon the von Neumann model in favour of biologically modelled computation, though we will take only general inspiration from his approach, not follow it closely. As our goal is merely a proof-of-concept implementation, and since the realization of truly novel programming, biologically oriented, paradigms is quite laborious, we opted for a compromise position and only tried to imitate biological computation in general spirit rather than in every detail.

2.2 Biological Foundations

In this section, we will go over the foundational ideas that, while serviceable on their own, will underlie the work in the second part of this work. The information will primarily concern biology, computational models, and the brain as a product of evolution.

[1] Note that we do not describe the implementation in the language of the Actor model, but that a translation into it would be quite easy. Such a translation would require using only very rudimentary features of the model, however, and, as that is not the focus, we forego the task.

Biologists will find all of it terribly basic, but this document is not intended for them; it is intended for computer scientists — who, we feel, have not truly taken to heart the consequences of the routes our nervous systems have taken through history for their present state. Sure enough, we have things *called* "evolutionary algorithms" and "machine learning", but names such as these invite us to a perilous confusion of labels with the real things. Although such mathematical abstractions may be inspired by biological processes, they are not the equivalents of these processes. Recreating the end-products of biology demands an understanding of biology on its own terms, not through the lens of misguidingly named mathematical abstractions. Providing the basis of such an understanding will be our aim for the next couple of pages.

2.2.1 Historical and Designed Artefacts

In order to understand how our brain works or could work, we must possess conceptual clarity — we must conceive of it, not as a product of one-time engineering, but as a historical artefact. Unlike "perfect" systems, like Peano arithmetic and the λ-calculus, those which grew historically does not make sense if one only looks at their current snapshot. One will find nonsensical solutions, and attempts to mitigate the consequences of earlier designs that have now become disadvantageous. The system as a whole might, at first glance, appear incomprehensibly and needlessly complicated. Of all such systems, the human brain might well be the most complex one; the task of understanding it correspondingly harrowing. Sloman asked whether the brain might have no architecture at all [Slo97, p. 5]:

> *Another question on which there is disagreement is whether the provision of a large set of capabilities, such as those listed above, necessarily involves the creation of an* intelligible *design, with identifiable components performing separate tasks, or whether the functionality could sometimes (or always?) emerge only in a very complex and incomprehensible fashion from myriad interacting components.*
>
> *For example, experimenters using genetic algorithms to evolve neural sets to control a robot sometimes create networks that work, but which seem to be impossible to understand (not unlike some legacy software which has grown over many years of undisciplined developments).*

It the classical sense, it probably does not, but we ought to be cognisant that "the classical sense" was induced by tradition and the limits of human cognitive ability. We might dismissively describe the brain as a jumbled, tangled chaos of neurons, but the fact that we do not recognise a structure by no means implies that one does not exist. Things, contrary to what is often espoused, do not "just work"; if they reliably produce complex results, they must have an architecture inside them, independent of our ability to recognise or understand it as such. We merely need to relax the notion of "architecture" to include structures that result from incremental change and the creative combination

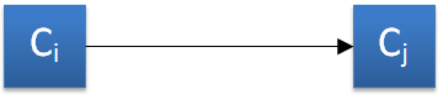


Figure 2.1: Relationship between the components of an organism without a nervous system.

of pre-existing parts. While the results of such processes are often extremely unintuitive and often even incomprehensible to us, we at least have a way of understanding them by re-tracing their evolution. Doing so is laborious and requires a huge amount of data (which we currently do not have), but this approach of regarding brain functions as through-and-through Darwinian (as opposed to having been pieced together) might bear results that have, so far, eluded the other schools of thought in the field.

What, one might now ask, is the consequence of such a view? The first is that each new feature in the developmental history had to have been useful on its own. The second is that it allows the distinction between what we will herein call *efficient* systems and *clean* systems. Since, at each stage of its evolution, the organism that carried the brain had to be viable, the end product is by definition guaranteed to be "efficient". Because of that same fact, however, it is all but guaranteed not to be "clean": for one, it was not possible to snap whole new components into the system; it would have also been impossible to combine old components in the elaborate and precise ways in which a human engineer might use parts. Worse, old components were almost certainly not discarded when new and better ones came into being. A good exposition of this process in humans can be found in Paul MacLean's seminal work *The Triune Brain in Evolution* [Mac90].

2.2.2 Origin of Nervous Systems

The evolution of nervous systems dates back to the development of primitive electrical signalling in eukaryotes, using calcium action potentials[2] and sodium channels [LHZ11]:

> *Voltage-dependent sodium channels are believed to have evolved from calcium channels at the origin of the nervous system.*

These sodium channels predated modern-day neurons, but served the same fundamental purpose of acting as control systems. We can readily conceive the benefits of imparting a control system onto an organism with the following thought experiment: let us imagine a microscopic organism without any sort of nervous system — all of its behaviour is hard-coded and mechanical. It can take in nutrients through its cell walls or through an opening; parts of it can contract or expand in response to stimuli like light or pressure; homeostatic conditions can influence its chemistry. Figure 2.1 shows this schema: if we enumerate the constituent parts or *components* of an organism as $\{C_1, \ldots, C_n\}$, the

[2] See any textbook on evolutionary biology.

organism's behaviour is caused by signals being sent between C_i and C_j (the case $i = j$ is possible). Such an organism suffers from three disadvantages: (a) reactions are localised, as two of its components might be too far apart to communicate in a timely manner or at all; (b) its repertoire of behaviours is necessarily simple and (c) it is not very adaptable.

Precursors to nervous systems communicated (a) first via action potentials, which were intracellular electrical signals [LHZ11] (emphasis ours):

> *Another key animal innovation was the nervous system, which is present in all but a few animals (i.e., sponges and placozoans).* Rapid, specific, long-distance communication among excitable cells *is achieved in bilaterian animals and a few jellyfish (cnidarians) through the use of action potentials (APs) in neurons generated by voltage-dependent sodium (Na_v) channels. Voltage dependent calcium (Ca_v) channels evolved in single-celled eukaryotes and were used for intracellular signalling.* It has been hypothesised that Na_v channels were derived from Ca_v channels at the origin of the nervous system [*the results in the paper support the hypothesis*]*, thereby conferring the ability to conduct action potentials without interfering with intracellular calcium. This view was reinforced by the apparent lack of sodium currents in sponges.*

The introduction of dedicated, long-distance[3] signalling cells between parts of an organism created the possibility of not only transmitting, but also modifying information. The moment an organism's parts do not communicate directly biochemically/mechanically, but over transmissions lines, evolutionary processes acting upon these lines are able to mutate them so that they change the signals. The first changes might consist of amplifying, diminishing, or distributing signals. Over time, the nerves may come to act as transducers on the stream of signals; in some rudimentary sense, they may begin to compute functions. Schematically, we see this in Figure 2.2, where a function F is interposed between two components. Not all components of an organism are created equal, of course. The first and most important use of nerve cells was the communication between sensory organs and the movement apparatus of the organism, and the bulk of nerve cells were located close to the sensory organs, where they processed information. A mere handful of neurons are not able to compute much, but they must have conferred considerable advantage to their owners.

The history of these developments is not entirely clear, but action potentials are present in all animals (with the exception of sponges) and in plants [LMM99, FL07]. A step up from mere stream transducers are the nerve nets that permeate the entire bodies of cnedaria (jellyfish) and the nerve cords that run along the bodies of bilateria (animals with left and right sides). In Figures 2.3b and 2.4 we see them in the phylogeny of

[3] The term "long-distance" may very well mean "long-distance within a single cell". Baluška and Mancuso argue in *Deep evolutionary origins of neurobiology: Turning the essence of "neural" upside-down* [BM09] that neural analogues already existed in prokaryotes (bacteria and archaea; organisms without cell walls and nuclei) and unicellular eukaryotes.

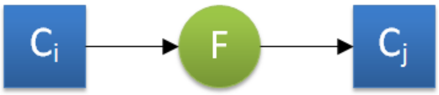
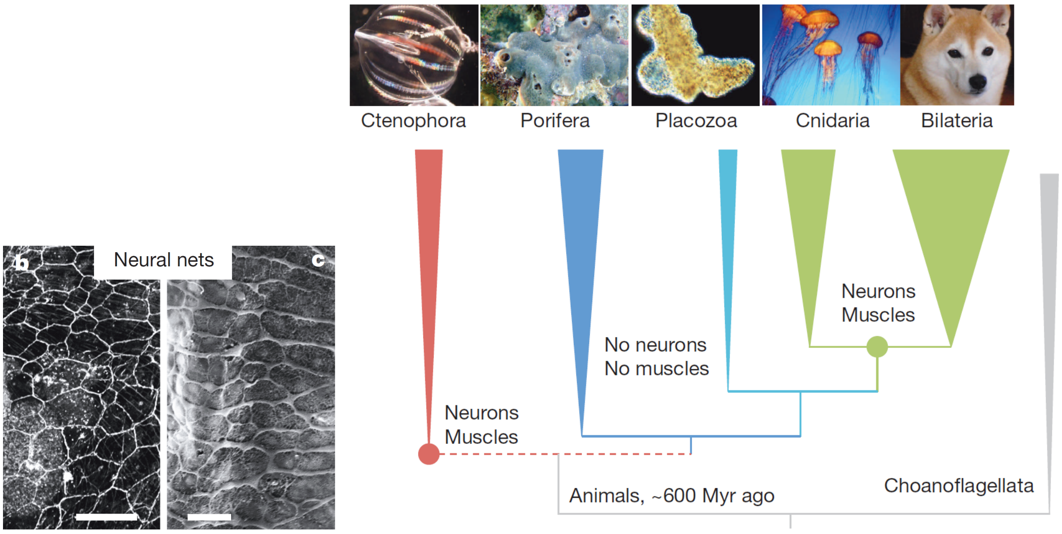


Figure 2.2: Relationship between the components of an organism possessing a nervous system. F can be understood as a simple signal transformer or a central coordinating mechanism.



(a) Nerve nets in ctenaphora.

(b) Phylogeny of the animalia. Even Ctenophores, macroscopic marine invertebrates which predate both jellyfish and bilateria, have nervous systems in the form of distributed nerve nets.

Figure 2.3: Nerve nets and phylogeny of animalia. From *The ctenophore genome and the evolutionary origins of neural systems* [MKC+14, p. 100].

the kingdom animalia. Both can process signals in a sophisticated way, and enable the performing of varieties of complex tasks, although the sets vary widely from species to species.

Central nerve cord and cephalisation. Nerve nets, while interesting, are not our aim. Unlike jellyfish, bilateria have a central nerve cord which runs from their front to their back. At various points alongside the cord, we find ganglions — thickenings containing larger amounts of nerve bundles. In all animals but worms, the frontal ganglion further thickened until it came to contain the overwhelming majority of the organism's neurons — forming the head. While substantial neural activity was occurring before this time, it is only here that it becomes to proper to speak of brains, and where we can begin to analyse macroscopic structures like lobes.

Vertebrate brains are subdivided into hinbrain, midbrain, and forebrain, having evolved

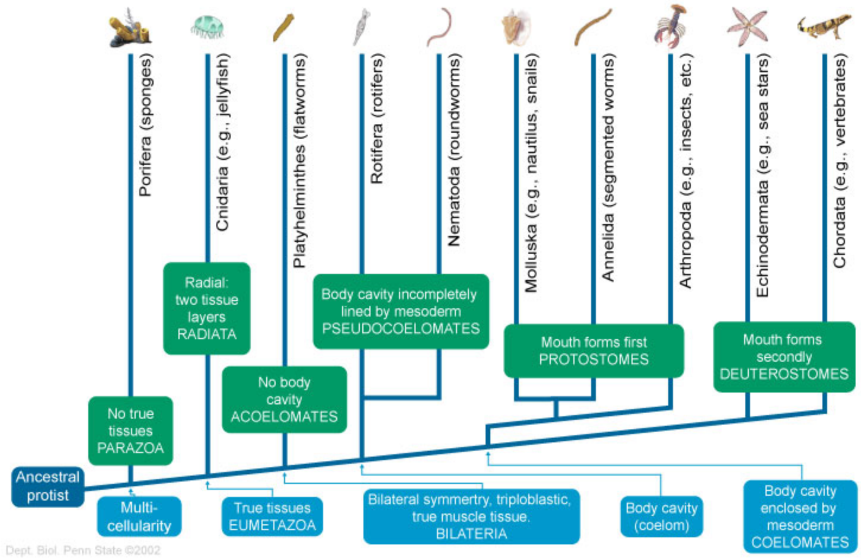

Porifera (sponges)
Cnidaria (e.g., jellyfish)
Platyhelminthes (flatworms)
Rotifera (rotifers)
Nematoda (roundworms)
Molluska (e.g., nautilus, snails)
Annelida (segmented worms)
Arthropoda (e.g., insects, etc.)
Echinodermata (e.g., sea stars)
Chordata (e.g., vertebrates)
Radial: two tissue layers RADIATA
Body cavity incompletely lined by mesoderm PSEUDOCOELOMATES
Mouth forms first PROTOSTOMES
Mouth forms secondly DEUTEROSTOMES
No body cavity ACOELOMATES
No true tissues PARAZOA
Ancestral protist
Multi-cellularity
True tissues EUMETAZOA
Bilateral symmetry, triploblastic, true muscle tissue, BILATERIA
Body cavity (coelom)
Body cavity enclosed by mesoderm COELOMATES
Dept. Biol. Penn State ©2002

Figure 2.4: Phylogeny of the animalia. Note the cnidaria and bilateria; both of these have types of nervous systems. From *Animals I – An Overview of Phylogeny and Diversity* [Woo].

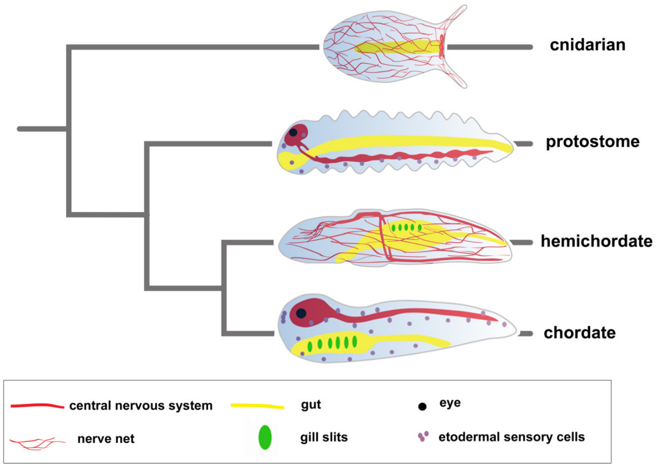


Figure 2.5: Body plans for metazoans. The bottom three items are all bilateria and all have nerve cords of some kinds, but only the bottommost (chordates) have a dorsal (upper) nerve cord. Vertebrates are a subphylum of the chordata. From *Evolution of bilaterian central nervous systems: a single origin?* [HCE+13, p. 3].

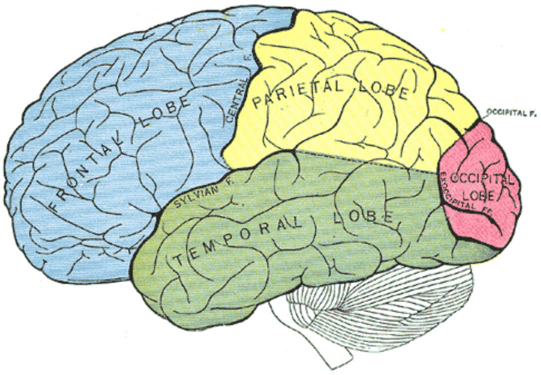


Figure 2.6: Illustration of the cerebrum with lobes shown. Hidden: limbic lobe, insular cortex. From *Anatomy of the human body* [Gra18, Fig. 728].

in this order. The forebrain (cerebrum) is responsible for all higher functions and is again divided into six lobes, which we see in Figure 2.6. The functions performed by these lobes are not precisely understood, but a number can be clearly associated to one lobe. The frontal lobe, for instance, is responsible of conscious thought; the temporal lobe processes auditory and olfactory signals; the occipital lobe deals with sights.[4] While some functions, like memory, are not neatly localisable, we can nonetheless see in the anatomy of vertebrate and mammalian brains the accruing of large groups of functions: motor control, smell, hearing, sight, reason, emotions. The question of organisation remains, however: it is one thing to say that we have hearing and smell, but what, if anything, ties these experiences together? We, after all, perceive the data from all our senses as one integrated experience. Here, views diverge. The common-sense belief is that we simply have one, indivisible consciousness. Such a view would implicate the frontal lobe as an central organising unit, without which an organism, even if it could smell or see, would not consciously do so. Minsky, Sloman, and Dennett argue persuasively, but speculatively, against this view in their works [Min06, Min88, Slo91, Den91]. They differ on the details, but all agree that the unified consciousness is an illusion; that it is not a single "I am"-thing gathering raw data, but a dispersed locus of experiences that we merely perceive as immediate.[5] In this view, the frontal lobe, while still instrumental, would not be the only contributor. All other regions of at least the cerebrum would contribute in some way to the organism's conscious experience. An animal without a frontal lobe would not be conscious in the same way as we are, but it would not be

[4] Interestingly, the occipital lobe is at the *back* of the head.

[5] Dennett criticises the idea of a *consciousness-thing* with the concept of the "Cartesian theater" [Den91]. According to him, positing that there is such a thing in our brains, and that it observes all other brain functions, is fundamentally problematic: if there is such a sort of homunculus in our heads that, say, sees the result the output of visual processing in the manner in which one would see a film, then how its visual perception work? Is there yet another a homunculus inside the homunculus that interprets visual information? Such a view implies either an infinite regress, or the algorithmic inexplicability of some part of the brain.

utterly blind either — it would already have some dim awareness of its existence; some rudimentary “I am” that we can hardly imagine would already be present in it. To quote Sloman (emphasis ours):[6]

> *It is not worth asking how to define consciousness, how to explain it, how it evolved, what its function is, etc.,* because there’s no one thing for which all the answers would be the same. Instead, we have many sub-capabilities, for which the answers are different*: e.g., different kinds of perception, learning, knowledge, attention control, self-monitoring, self-control, etc.*

Implications. The point in all this is not to give an detailed summary of evolutionary neurobiology; it is to show that nervous systems are ancient, gradually developed things. They have been shaped by the vicissitudes of hundreds of millions of years, and they could have developed in other ways. They were not planned, as a human would understand the word. If we are to gain headway in piecing together the “big picture”, we must take these facts to heart, and choose our modelling methods accordingly.

In the abstract of this work, we described the biological and the idealistic approaches as being polar opposites, and this is true as far as engineering is concerned, but in terms of their assumptions, false. They are both idealistic. Neural networks, insofar as their users want to re-create human behaviour, implicitly presuppose an intelligence in neurons that is not there. The comparatively small network is taught to compute some desired function, the hope being that it might thereby come to perform some complex, real-world function like common-sense reasoning. In principle, this strategy could work, but in practice, it is unrealistic — the environment in which real organisms had to succeed was the planet’s ecosphere; billions upon billions of nervous systems of all complexities were run over millions of years; nervous systems died off and were re-created from scratch by genes. It is therefore entirely unreasonable to assume that neural networks, trained against an objective function over a period of hours or days could re-create the function a biological organism, unless one were to suppose that there is some inherent quality in neurons that strives for such; that groups of cells somehow *wish* to organise themselves into specific configurations in which they are able to perform activities we would call “cognition”.

All this being said, we should not confuse criticism of the suitability of a method for a specific purpose with criticism of its suitability for any purpose. Neural networks have proven useful in understanding mental activity at small scales; both they and the symbolic/logic-based approaches have had a myriad of industrial applications. From this, however, it does not follow that we can build genuinely intelligent agents with them. Our only means of doing that (the only means that remain) is to laboriously unravel the developmental history of animal brains, step by step, making sense of each development

[6] The quotation appears in *The Emotion Machine* [Min06, p. 97] and Minsky attributes it to a post made by Aaron Sloman in the `comp.ai.philosophy` newsgroup, but we have been unable to find the original.

in context. Where empirical data are not available, we at least have to hypothesise how things could plausibly have happened. To this day, structural and genetic analyses have been done (via genetic sequencing and MRI), but they do not deliver sufficiently detailed data. Such methods are rather akin to measuring voltages and task time in a PC — they do tell us something, but an observer would never infer the existence of, e.g., compilers, call stacks, or type systems from such observations. For an understanding of the brain so specific that we can re-implement it in a computer, we will need currently non-existent and not-conceived-of technology. Until that day, guesswork, like the assumptions made in this thesis, will have to suffice.

2.2.3 Ways of Adaptation

After the philosophical groundwork and biological basics, let us describe possible means by which nervous systems can change and acquire new features. We begin with the observation that the existence of neuron bundles between parts of an organism is analogous to a loose coupling of components in a software systems. By having intermediaries that take over the task of communication, selection pressure can produce more and more complex functions, since it no longer has to act upon the body parts that send various signals, but change the nervous system that processes these signals instead. As example: pain receptors, muscles fibres, and the optical nerve have been unchanged for quite some time, long pre-dating the human species, but more recent brain developments have given us the ability to utilise them in novel ways — by providing a rich mental experience of suffering, playing instruments, and mentally rotating objects, respectively. Manipulating the software is far easier and more quickly done than doing so with the hardware, so to speak.

Having said that, the changes still have to have occurred incrementally. Even if a nervous system can change quickly (for evolutionary timescales), it still has change in tiny steps. We shall leave the matter that for the time being, but, as we will discuss later, this simple fact has profound computational consequences that are seldom thematised in discourse on this matter.

Let us return to the consideration of primitive life forms. We can imagine the malleable neuron bundles of such ancient organisms changing in a variety of ways in the face of selection pressure: when the environment required it, they could, after several generations, start to compute different or more elaborate functions. An organism which had had developed in an environment where food was abundant in bright places and which had now found itself in darkness would have benefited from a variety of plausible changes, such as

- an inversion of its light-seeking behaviour,
- switching off its metabolism in light places to conserve energy,
- accelerating its metabolism in dark places to make better use of the food there.

Of course, other changes would have also been possible, such as the metabolisation of different food sources,[7] but we can see how the aforementioned three could have been effected through mutations in a simple nervous system. For a system to permit such mutations, it must be far more robust than most products of human engineering, however. If one were to take out a piston in a car or replace a cogwheel in a mechanical clock with a differently sized one, the machine would, in most cases, simply break. In all others, it would catastrophically malfunction. Machines are designed to fit together perfectly and their complexity tends to be irreducible. Even software, which is more readily changed, is easily broken by small-scale tinkering.

When discussing how they can evolve and, in particular, *evolve to perform new tasks* and not just variations on old ones, explanations are again constrained by two criteria: (a) the change has to be small, or at least have a small cause[8] and (b) each change must be beneficial in the short term.[9] Something that we would conventionally recognise as a program, something which has precise notions like "instruction" and "call structure" is probably not suited to this pattern of changes.[10] Instead, we ought to imagine the brain as a mesh of computation in which functions are computed cumulatively, so that small changes in neural structure only lead to small changes in output.

To illustrate this, we can look at a simple neural network in Figure 2.7a, with a marked node N_x. Figure 2.7b shows an unlikely change scenario in which some new component/-function is cleanly grafted onto the system. Figures 2.7c and 2.7d then show two more likely scenarios: in the first a mutation causes N_x to be split and the new nodes take over some of its connections. In the second, a larger component is accidentally copied as-is and, over time, is moulded to do something useful.[11] In time, new functions can thus grow into the system, but never in the manner in which, say, an engineer would implement a new feature.

Sloman's brain. One might ask what the relationship between the gradual growth of neural bundles and the observed, large-scale functions in the brain is. We have now supposed at some length that the organisation is not neat, but the question remains whether we can speak of an organisation at all (even a messy one). In *Beyond shallow models of emotion* [Slo01, p. 8], Sloman illustrates the possible chaotic organisation of the

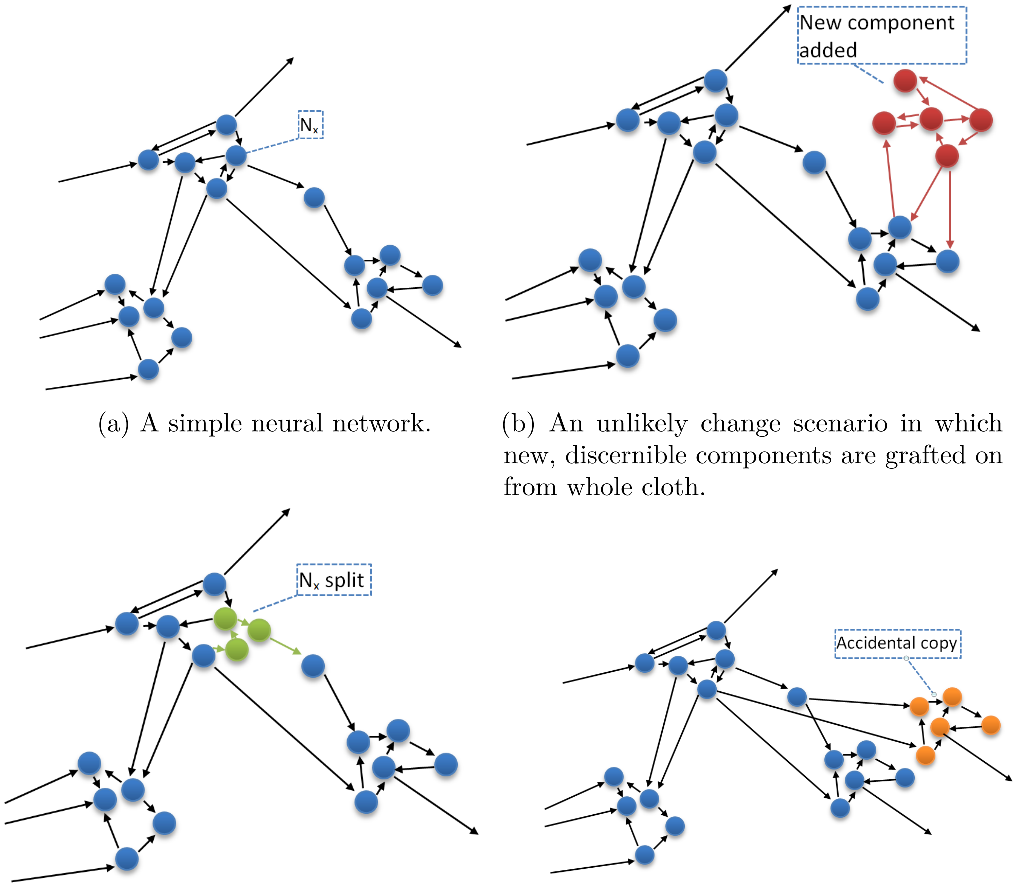
[7] A current-day example is given by nylon-eating bacteria, which have developed in the last century and which now have an abundant food source and no competition.

[8] The effect does not have to be small — changes in single genes can switch entire components on or off. The MYH16 gene, which is present in non-human primates but has been switched off in humans, is an example. In us, its disabling lead to a drastic reduction in the size of jaw muscles and a corresponding increase in brain size [Car05]. Nonetheless, such events are rare and not the main drivers of evolution.

[9] Caveats apply: if the selection pressure on a group of organisms is not too strong, changes which may be suboptimal but perhaps beneficial at some later point may spread, and non-selective processes like genetic drift can also play a role.

[10] Cf. evolutionary program generation, in which expression trees mutated. We charge that such algorithms are not adequate models of what happened in the evolution of our brains.

[11] Such copies can be caused by mutations and are known to happen with some frequency in nature.



(a) A simple neural network.

(b) An unlikely change scenario in which new, discernible components are grafted on from whole cloth.

(c) A more likely change scenario in which one part is split into three but where the overall shape of the network is not appreciably altered.

(d) A second change scenario in which an entire component is accidentally copied. While the resultant change is large, a small genetic mutation can cause it.

Figure 2.7: Means of change in neural networks.

brain with Figure 2.8, conjecturing that it might be a jumble of parts that just happen to work together:

> *Any observed behaviour might be produced by an unintelligibly tangled and non-modular architecture.* (*Rectangles represent information stores and buffers, ovals represent processing units, and arrows represent flow of information, including control signals.*)

It sounds somewhat like cheery optimism to presuppose that there even are information stores and control signals. The actual situation is likely a far worse one: it is not just different programs that are run in the brain, but entire different models of computation,

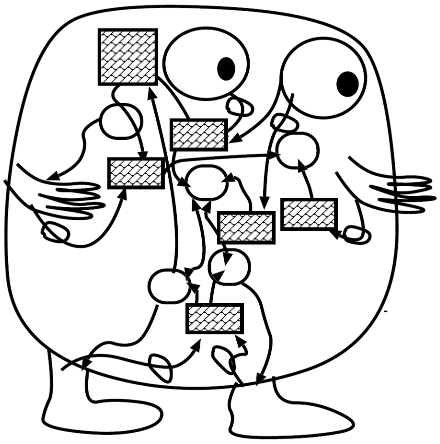


Figure 2.8: Sloman's illustration of the brain as an "unstructured mess".

with the same pattern of activity being interpreted simultaneously in more than one way. Today, we can scarcely imagine how such an "architecture" would work, let alone how one would program it — but if we really want to create genuinely animal intelligences, we will have to find out. Sloman himself admits to this difficulty in *What Sort of Control System Is Able to Have a Personality?* [Slo97, p. 6, Section 9 "Is the task too hard?"]:

> *Given the enormous diversity in both design space and niche space and our limited understanding of both, one reaction is extreme pessimism regarding our ability to gain significant insights.*

The following remedy is offered:

> *My own attitude is cautious optimism: let us approach the study from many different directions and with many different methodologies and see what we can learn.* [. . .]
> *In particular, the Cognition and Affect group at Birmingham has been trying to use a combination of philosophical analysis, critical reflection on shared common sense knowledge about human capabilities, analysis of strength and especially weaknesses in current AI systems, and where appropriate hints from biology, psychology, psychiatry and brain science, to guide a combination of speculation and exploratory implementation.* [. . .]

The methods listed all have their applications, but computational analysis is missing among them. When we talk of psychology, philosophy, and critical reflection, we have already supposed too much; we want to replicate the high-level output of the brain without having explored the mechanism that produces it. In a manner of speaking, we have seen the forest, but do not understand what trees are. If we are to gain the sort of knowledge of brains that we can implement into an AI, we must, empirically, find out about their method of computation. Barring that, we must at least approximate it as far as is practical, and accept that the result will necessarily be an inferior simulacrum.

What, then, is the computational model used in the brain? So far, nobody knows, and, presumably, that will stay that way for the foreseeable future. It is very much a guess, but from the concept of slowly growing neural networks (seen in Figures 2.7a-2.7d), one might infer something like the “active symbol” hypothesis in Douglas Hofstadter’s *Gödel Escher Bach*: that patterns of activity form little programs and pieces of data at once; that manipulate other patterns of activation and are manipulated yet other patterns during their lifetime. These are only imperfect analogies, of course. On the coming pages, we shall outline a conceptually compatible white-box model of computation as another, imperfect analogy that will serve as the basis for the model in Chapter 3, and for the implementation of the toy agents in Chapter 5.

2.3 The Brain as a Collection of White Boxes

We now leave the realm of established fact and venture into conjecture. What has been said up to this point has been good, general fact, but it does not suffice for building actual programs. Data on the computational structure of the brain is scarce, thus we will limit myself to positing general, plausible hypotheses about what sorts of structures and loci of computation could have plausibly arisen in it over the course of its evolution.

A plausible case has been made by Minsky, Sloman, and others (especially in *The Emotion Machine* [Min06]) that the brain must possess components in some form. Were it not so, the organ would have long ago succumbed to the inefficiencies of its design. As more and more functions are grafted onto a system, the number of interactions between its parts or regions, and therefore the bugs in it, increases. Worse yet, the system becomes brittle: even if, like in a neural network, some accidentally working configuration would have been able to be reached, small changes would surely have upset it again. The part-less system is an evolutionary dead-end from which no improvement is possible, and given how far along our cognition is, it is quite clear that we are not dealing such when we look at our brains.

If we concede that we are dealing with identifiable parts, a second question arises: how do these parts communicate? We would like to deal with this question in some detail. In the literature, this issue is often glossed over — in diagrams, one frequently finds unannotated arrows going between functions; the accompanying texts mention concepts like “selection”, ”message”, and “sending” under the implicit assumption that these are merely primitives in no need of further explanation. When we consider the workings of neurons, however, it is not at all clear how groups of them could put together any sort of complex message, and, once put together, how it would travel, and how another group of neurons could receive and interpret it. Are there dedicated interpreters, akin to compilers and runtimes in computer systems? This is not known. We cautiously propose that it is not so, but we can present plausibly-sounding scenarios for both outcomes:

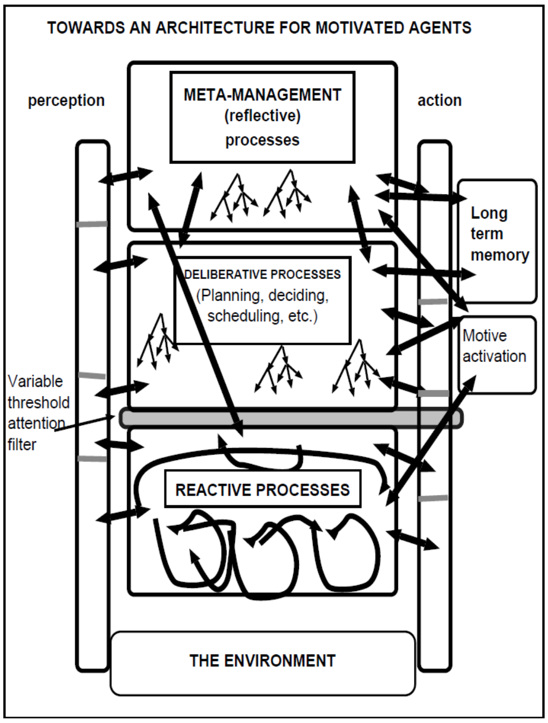

TOWARDS AN ARCHITECTURE FOR MOTIVATED AGENTS
perception
action
META-MANAGEMENT (reflective) processes
Long term memory
DELIBERATIVE PROCESSES (Planning, deciding, scheduling, etc.)
Motive activation
Variable threshold attention filter
REACTIVE PROCESSES
THE ENVIRONMENT

Figure 2.9: Architecture for motivated agents. From [Slo97, p. 10]

- On the one hand, we may image that, early on, some simple message format developed, allowing more efficient communication between not quite differentiated regions of a nervous system. Over time, this was extended as more components came into play; these new components would have found it easier to make use of the pre-existing protocol. Larger clusters of parts might have even repeated the process and developed simple, internal message formats for communication among themselves. As an orthogonal development, newly developed components might have performed more abstract duties, using older ones as subsidiaries. To solve conflicts whenever these new and old parts proposed different solutions to whatever issue the organism faced, some other component could have received inputs from both, and adjudicated. In such way, a hierarchical and layered structure could have come into being — different layers working at different levels of abstraction, and each component only communicating on an on-basis with others. All in all, the whole system would come to resemble a human-developed program.

- On the other hand, we could imagine quite a different scenario: suppose that the basic scheme of neurons sitting as growths on the communication lines between components never changed. Their basic task was the modulation of signals, and if some new function was to be grafted into the system, then this would have been achieved by growing more neurons that modulated the signals of their fellow

> neurons. They would not have opened a communication channel with the existing components, but would have listened in on their activity in the manner of interlopers surreptitiously modifying messages. Since neurons would have had no reason to hide their activity (as a black box does), this would have been quite easy and straightforward to do. New bundles of functionality could have inserted themselves into the middle of the information flow (enhancing existing functionality, or adding administrative features), before it (providing pre-processing), or after it (providing post-processing, or usage of the output for higher-level tasks). In contrast to above, we concentrate on the *process* of software development instead of its product: a human programmer adds functions one at a time, here and there, extending and refining functionality where fancy strikes or necessity requires.

One could thus call the first scenario the *product-oriented view* and the second the *process-oriented view*: the first looks at the end product of a development, the second posits that the very process of that development, fossilised, is present in the end product.

Evidence is scarce and equivocal for both. In fact, it need not even be the case that they form a dichotomy: we might just as well speculate, for instance, that the second is the low-level reality, but that the first emerged from over time due to the efficiency of its design. The components could be fuzzy, to some degree. We could also posit that the first one "degenerated" into the second one; that a formal system is emulating an informal one because of the latter's greater versatility and dynamism.

For the rest of the thesis, we will explore the second of the above two hypotheses, not necessarily because we firmly believe it to be true, but rather because the first one has been tried for some time, and has so far not produced a general AI.

Practical abstraction. While such a white-box model, and the hypothesising that preceded it, are conceptually useful, a mesh of gradually grown patterns does not lend itself to implementation in a program. We do not have the capability of faithfully pouring the structure of the human brain into a computerised mould just yet, but, for the time being, we may opt for the next best thing and take cues from it in the hopes of improving our imitations.

Therefore, we will present a simplified model which, while attempting to remain true to the conceptual view, will, pragmatically, contain discrete functions and components. The white-box nature of brain activity will be emulated by a message-passing scheme in which messages model the internal activity of components. Instead of each component blindly acting in some fashion on the activity of another, components will have explicit parsers and interpreters and later, these will be further simplified into localised message formats and tagging, for the sake of easy implementation. This effort is guided by the same thought as Sloman's cognitive architecture depicted in Figure 2.9. It is not a truly accurate representation of the brain and it does not claim to be, but it is *something like it*; something that is close, and good, enough. We will meet this cognitive architecture again later, but for now, we move on to the description of neural components.

CHAPTER 3

Overall Component Model

Having gone through a number of biological considerations, we now translate these into a mathematical model of a component system. These components can send and receive messages, filter them out, and interpret them in various ways. This model does not yet specify our architecture, but our artificial intelligence, as described in Chapter 5, will be comprised of such components.

3.1 Components as White Boxes

We can imagine the components of the mind as white boxes which inform other components by their very functioning — however, this does not lend itself to easy implementation. Instead, we can emulate this behaviour via a *message space*, from which individual components take their input and into which they put their output. A *component* is then a local processing unit which continuously scans the message space, running messages through its *filter*. If the filter detects a relevant message, it is then passed to the *interpreter*, which parses the message into the needed format and hands it over to the *processor*. The processor, after having finished, puts its output back into the message space for other components to read. Figure 3.2 illustrates this scheme. Note the lack of explicit hierarchical structure and central organising units.

However, as we will show in the next section, this model is generic enough to accommodate special-purpose structures like a message space. Figure 3.2 shows the message-passing scheme, but it also specifies a graph in which the nodes are the components and fixed, while the edges are the accepted messages and are determined by the nodes; through their filters, components control the shape of the graph. By imposing invariants on these filters, we can have the graph take any shape we desire. In particular, we can model the kinds of structures that occur in many other cognitive models and in empirical research: central organisers, sequences of components (“pipelines”), localized messages affecting

Symbol	Description
C P	Processing component
Ft Ch	Choice
Messages	Data container (Queue, List, etc.)
Head Body	Data
○	Stream generator
Head Body	Counterfactual (imaginary) data

Figure 3.1: Notation for the diagrams in this and the following sections.

only a small part of the mind, a component reading its own messages, loops and iterative messages between two or more components et cetera.

Messages. We may now ask how such messages between components are structured. Here, we make two empirical claims:

1. messages have a priority and
2. they are effectively unstructured.

To the best of our knowledge, the veracity of either has thus far not been determined by neuroscience. For the first, Marvin Minsky's "The Emotion Machine" provides some circumstantial evidence [Min06, p. 222]:

> *Of course, when one activates two or more Critics or Selectors, this is likely to cause some conflicts, because two different resources might try to turn on a third resource both* on *and* off*. To deal with this, we could design the system to use various policies like these:*
>
> 1. *Choose the resource with the highest priority.*
> 2. *Choose the one that is most strongly aroused.*
> 3. *Choose the one that gives the most specific advice.*
> 4. *Have them all compete in some "marketplace".*

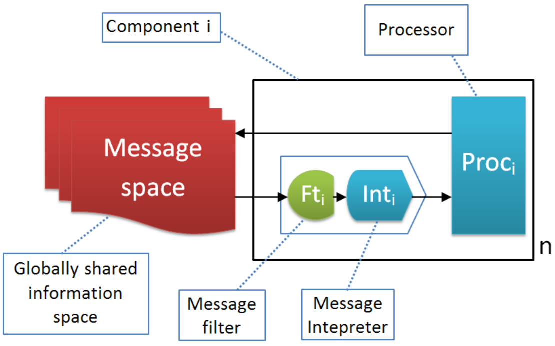


Figure 3.2: Global neural architecture.

The selection strategies Minsky lists imply that there is some mechanism in the brain to determine the urgency of a signal. While it is possible that higher brain functions like reasoning or affect make an additional, rational evaluation, sensations like intense pain, bright lights, or great sadness can likely be communicated most easily by the appropriate components causing a flood of activity which, by its very intensity, informs other components of the urgency of their messages.

The second claim — that messages are essentially unstructured — means that there is no common, agreed-upon format in which they are stored. In addition to the evolutionary implausibility of such a format being created, an unstructured message format is in line with the white-box nature of components: since components merely “listen in” on others, and since each components will have its own pattern of activity, a listener would simply have to try and make sense of this activity as best it could. The proposed structure of messages is thus shown in Figure 3.3: every message comprises a priority header, together with an unstructured body which, for our purposes, is simply a string of bits.

Filters. Before a component can respond to a message by another, such a message must be assessed for the presence of relevant information. Conceptually, this happens via a *filter* in each component, which pattern-matches incoming messages and, if a certain threshold is reached, signals relevance and hands the message over the *interpreter* for parsing. Figure 3.4 shows such a filter: it is composed of a directed graph of nodes, and a node is activated if it detects some specific content in the message. Nodes, in turn, are

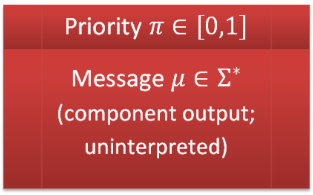


Figure 3.3: Structure of a neural message.

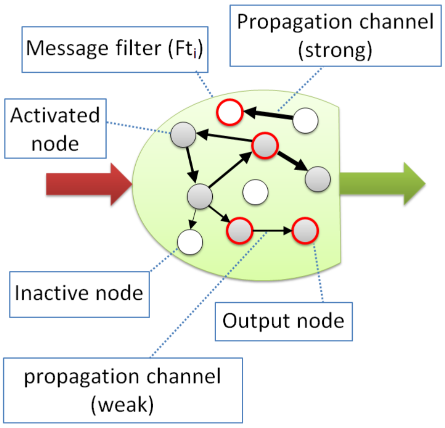


Figure 3.4: A pattern-matching filter for a component C_i.

connected via edges of strength within the interval $[0, 1]$. When a node is activated, it sends a charge proportional to the strength of its link to its neighbours, contributing to their activation as well. Some nodes are marked as *output nodes*; if enough such output nodes become activated, the message is deemed to be sufficiently relevant. This model of filters is inspired by the *spiking neural P Systems* of Georghe Paŭn et al. ([PRS10, p. 337] and [IPY06]), in which charges sent along directed graphs of neurons are used to compute functions.

3.2 Mathematical Model

We now create a mathematical model for the description of the architecture. This model will be split into two parts: the structural and the operational semantics. The structural semantics encode the static properties of neural systems, whereas the operational semantics describe the behaviour of such a system at runtime.

3.2.1 Preliminaries

Since the mathematical model is built with implementation in mind, we will use some basic type theory in the coming sections. The following notions are from the λ-calculus and its attendant type systems; anyone familiar with such can therefore freely skip this section. We will introduce types, type constructors, and their relation to functions, together with a few example types which will come in handy later on. The following can be found in any introduction to type theory and was taken (with simplification) from [Men88], [CD94], and [JR97].

Definition 1 (Syntax: Type). *For our purposes, types are defined inductively thus:*

1. Basic type: $\mathbb{R}$ *and* $\emptyset$ *are types.*

2. Sum type: *If* $\mathtt{T}_1, \mathtt{T}_2$ *are types, the* sum type $\mathtt{T}_1 + \mathtt{T}_2$ *is a type.*

3. Product type: *If* $\mathtt{s}$ *is a string and* $\mathtt{T_1}, \ldots, \mathtt{T_n}$ *are types, the* product type $\mathtt{s}\ \mathtt{T_1} \ldots \mathtt{T_n}$ *is a type. A special case is the* anonymous product type *(tuple), where* $s = \text{“}\langle\rangle\text{”}$*. There, we just write* $\langle \mathtt{T_1}, \ldots, T_n \rangle$*.*

4. Full application: *If* $\mathtt{T_1}, \ldots, \mathtt{T_n}$ *are types and* $[\forall x_1, \ldots, x_n]\, \mathtt{C}$ *is a type constructor (see next definition), then* $\mathtt{C}\ \mathtt{T_1} \ldots \mathtt{T_n}$ *is a type.*

5. μ-abstraction: *If* $\mathtt{T}, \mathtt{S}$ *are types and* $\mathtt{S}$ *occurs in* $\mathtt{T}$*, then* $[\mu\alpha]\, \mathtt{T}[\mathtt{S}\backslash\alpha]$ *(for a fresh variable name* α*) is a type.*

Definition 2 (Syntax: Type constructor). *Type constructors are the defined thus:*

1. Base case: *Every type* $\mathtt{T}$ *is a type constructor.*

2. Abstraction: *If* $\mathtt{C}$ *is a type constructor and* $\mathtt{T}$ *is a type,* $[\forall x]\, \mathtt{C}[\mathtt{T}\backslash x]$ *is a type constructor.*

3. Sum types: *If* $\mathtt{C_1} \ldots, \mathtt{C_n}$ *are type constructors with* $\mathtt{C_i} = \left[\forall \vec{X}_i\right] \mathtt{T_i}$ $(1 \leq i \leq n)$*, then* $\left[\forall \vec{X}_1; \ldots; \vec{X}_n\right] (\mathtt{C_1} + \cdots + \mathtt{C_n})$ *is a type constructor.*

4. Partial application: *If* $\mathtt{T_1}, \ldots, \mathtt{T_i}$ $(i < n)$ *are types and* $[\forall x_1, \ldots, x_n]\, \mathtt{T}$ *is a type constructor, then* $[\forall x_{i+1}, \ldots, x_n]\, \mathtt{T}[x_1\backslash\mathtt{T_1}, \ldots, x_i\backslash\mathtt{T_i}]$ *is a type constructor.*

Every type is interpreted as a set of values which are of that type; type constructors are interpreted as universally quantified templates for actual types. Their formal semantics are as follows:

Definition 3 (Semantics: Type). *Let* $\mathtt{T}$ *be a type. Its interpretation function* $\mathrm{int}(\mathtt{T})$ *is defined thus:.*

1. Basic type: $\mathbb{R}$ *is interpreted as the set of real numbers.* $\mathrm{int}(\emptyset) = \{\}$*.*

2. Sum type: *If* $\mathtt{T_1}, \mathtt{T_2}$ *are types, then* $\mathrm{int}(\mathtt{T_1} + \mathtt{T_2}) = \mathrm{int}(\mathtt{T_1}) \cup \mathrm{int}(\mathtt{T_2})$*.*

3. Product type: *If* $\mathtt{T_1}, \ldots, \mathtt{T_n}$ *are types and* $\mathtt{s}$ *is a string, then*

$$\mathrm{int}(\mathtt{s}\ \mathtt{T_1} \ldots \mathtt{T_n}) = \begin{cases} \{\mathtt{s}\} & \text{if } n = 0 \\ \{\mathtt{s}\} \times \mathrm{int}(\mathtt{T_1}) \times \cdots \times \mathrm{int}(\mathtt{T_n}) & \text{if } n \geq 1. \end{cases}$$

4. Full application: *If* $\mathtt{T_1}, \ldots, \mathtt{T_n}$ *are types and* $[\forall x_1, \ldots, x_n]\, \mathtt{C}$ *is a type constructor, then*

$$\mathrm{int}(\mathtt{C}\ \mathtt{T_1} \ldots \mathtt{T_n}) = \bigcup_{v_1 \in\ \mathrm{int}(\mathtt{T_1})} \cdots \bigcup_{v_n \in\ \mathrm{int}(\mathtt{T_n})} \left(\bigcup_{C' \in\ \mathrm{cint}(C)} C'[x_1\backslash v_1, \ldots, x_n\backslash v_n] \right).$$

5. μ-abstraction: *If* $[\mu\alpha]\,\mathtt{T}$ *is a type, then*

$$\text{int}([\mu\alpha]\,\mathtt{T}) = \text{int}(\mathtt{T}) \cup \text{int}(\mathtt{T}[\alpha\backslash\mathtt{T}]) \cup \text{int}(\mathtt{T}[\alpha\backslash\mathtt{T}][\alpha\backslash\mathtt{T}]) \cup \ldots$$

with $\text{int}(\alpha) = \{\}$*.*

Definition 4 (Semantics: Type constructor). *The partial interpretation function* cint *for type constructors is defined as follows: if* $\mathtt{C}$ *is a type constructor containing exactly the types* $\mathtt{T_1}, \ldots, \mathtt{T_n}$*, then*

$$\text{cint}(\mathtt{C}) = \bigcup_{v_1 \in\ \text{int}(\mathtt{T_1})} \cdots \bigcup_{v_n \in\ \text{int}(\mathtt{T_n})} \mathtt{C}[\mathtt{T_1}\backslash v_1, \ldots, \mathtt{T_n}\backslash v_n].$$

Intuitively, sum types are simply unions, product types are named Cartesian products, and full applications are instantiations of type constructors with all possible values. μ-abstraction represents recursive data types such as lists or trees. Type constructors themselves are just generic types.

Whenever we want to assert that an expression has a specific type, we write:

Notation 5 (Typed expressions). *Let* x *be an expression and* $\mathtt{T}$ *a type.* $x :: \mathtt{T}$ *asserts that* x *has type* $\mathtt{T}$*.*

Henceforth, by convention, we will write type variables in lower-case and concrete types in upper-case, omitting the explicit $\forall$-blocks. That is, a type like $[\forall x, y, z]\,\mathtt{C\ x}\ (\mathbb{N} + \mathtt{T_1})\ \mathtt{y\ z}$ will simply be written as $\mathtt{C\ x}\ (\mathbb{N} + \mathtt{T_1})\ \mathtt{y\ z}$ and it will be clear that $\mathtt{x}, \mathtt{y}, \mathtt{z}$ are type variables, while $\mathbb{N}, \mathtt{T_1}$ are concrete types. A special kind of type constructor is the function arrow $(\rightarrow)$ which induces the function type:

Example 6 (Function arrow). *If we take, say, the type* $\rightarrow \mathtt{S1\ S2}$ (*a product type with the product types* $\mathtt{S1}$ *and* $\mathtt{S2}$ *as arguments*) *and abstract twice, we get* $[\forall s, t] \rightarrow \mathtt{s\ t}$*. Here,* $\rightarrow \mathtt{s\ t}$ *is the type constructor for unary functions from* $\mathtt{s}$ *to* $\mathtt{t}$*, also written infix as* $\mathtt{s} \rightarrow \mathtt{t}$*. Functions with multiple arguments, mapping* $\mathtt{t_1}, \ldots, \mathtt{t_{n-1}}$ *to* $\mathtt{t_n}$*, can be modelled in two ways: either through n-tuples, or through nested function arrows:*

$$\begin{array}{l} \langle \mathtt{t_1}, \mathtt{t_2}, \ldots \mathtt{t_{n-1}} \rangle \rightarrow \mathtt{t_n}; \\ \mathtt{t_1} \rightarrow (\mathtt{t_2} \rightarrow \cdots \rightarrow (\mathtt{t_{n-1}} \rightarrow \mathtt{t_n}) \cdots). \end{array}$$

The first method necessitates that we supply all arguments at once, whereas the second allows them to be given one after another.

Function arrows allow the execution of functions in the obvious way:

Definition 7 (Function application). *Let* $f :: \mathtt{S} \rightarrow \mathtt{T}$ *and* x *be an expression of type* $\mathtt{S}$*. Then* $f\ x$ *is an expression of type* $\mathtt{T}$*. Function application associates to the left, that is:* $f\ x_1 \ldots x_n = (\cdots((f\ x_1)\ x_2) \ldots x_n)$*.*

We can combine type constructors, sum types, and product types into *algebraic data types* (ADTs).

Definition 8 (Algebraic data type (ADT)). *Let* $\mathtt{s}$ *be a string and* $\mathtt{C_1}, \ldots, \mathtt{C_n}$ *be type constructors such that* $\mathtt{C_i} = [\forall x_1, \ldots, x_n]\,\mathtt{T_i}$ *and* $\mathtt{T_i}$ *is a named product type with type variables* $(1 \leq i \leq n)$*. Then* $[\forall x_1, \ldots, x_n]\,(\mathtt{T_i} + \cdots + \mathtt{T_n})$ *is an ADT. If we want to give a name to an ADT, we write it as* $\mathtt{s}\ \mathtt{x_1} \ldots \mathtt{x_n} = \mathtt{T_i} + \cdots + \mathtt{T_n}$.

Since an ADT is merely the sum of product types, it is itself a type constructor. If it has no type variables, it is also a type. Next, we define a couple of example ADTs, some of which we will use in the next section.

Example 9 (ℕ, 𝔹, ℚ, ℂ, Maybe, Either, List).

$$\begin{aligned}
\mathbb{N} &= [\mu\alpha]\,\mathtt{Z} + \mathtt{S}\ \alpha;\\
\mathbb{B} &= \mathtt{False} + \mathtt{True};\\
\mathbb{Q} &= \mathtt{Rat}\ \mathbb{N}\ \mathbb{N};\\
\mathbb{C} &= \mathtt{Complex}\ \mathbb{R}\ \mathbb{R};\\
\mathtt{Maybe\ t} &= \mathtt{Nothing} + \mathtt{Just\ t};\\
\mathtt{Either\ l\ r} &= \mathtt{Left\ l} + \mathtt{Right\ r};\\
\mathtt{List\ a} &= [\mu\alpha]\ \mathtt{Nil} + (\mathsf{a} : \alpha).
\end{aligned}$$

ℕ *is the usual Peano definition of natural numbers, with a nullary product type* `Z` *representing zero, and a unary product type* `S`*, which allows recursion.* 𝔹*,* ℚ*,* ℂ *are the sets of Boolean number and rational/complex numbers, respectively, with* `False` *and* `True` *being nullary product types, and with* `Rat` ℕ ℕ *and* `Complex` ℝ ℝ *being binary ones.* `Maybe` *represents an optional value, which may or may not be present.* `Either` *represents a choice between two values, of which either the left or the right one is present, but not both.* `List a` *(or just* [a] *as a shorthand) denotes a list of values of type* `a`*. There,* `Nil` *is the nullary type constructor for an empty list and* : *is an infix binary type constructor that stores the head and tail of a list.*

We also define the usual convenience functions for these types:

$$\begin{aligned}
\mathtt{isJust} &:: \mathtt{Maybe\ a} \to \mathtt{Bool};\\
\mathtt{isJust}\ m &= \begin{cases} \mathtt{True} & \text{if } m = \mathtt{Just}\ x,\\ \mathtt{False} & \text{otherwise;} \end{cases}
\end{aligned}
\qquad
\begin{aligned}
\mathtt{head} &:: [\mathsf{a}] \to \mathtt{a};\\
\mathtt{head}\ l &= \begin{cases} x & \text{if } l = x : xs,\\ \bot & \text{otherwise;} \end{cases}
\end{aligned}$$

$$\begin{aligned}
\mathtt{fromJust} &:: \mathtt{Maybe\ a} \to \mathtt{a};\\
\mathtt{fromJust}\ m &= \begin{cases} x & \text{if } m = \mathtt{Just}\ x,\\ \bot & \text{otherwise;} \end{cases}
\end{aligned}
\qquad
\begin{aligned}
\mathtt{tail} &:: [\mathsf{a}] \to [\mathsf{a}];\\
\mathtt{tail}\ l &= \begin{cases} xs & \text{if } l = x : xs,\\ \bot & \text{otherwise.} \end{cases}
\end{aligned}$$

Definitions 1–8 specify a fragment of System F_ω,[1] which is used to type expressions in the lambda calculus. Although System F_ω is strictly more powerful, our definitions are enough to provide a description language for the data types and functions in the rest of this work.

3.2.2 Neural Systems

Definition 10 (Neural component). *Let I be an index set and let* `T` *be any type. Then, a neural component C with a name from I and message type* `T` *is a four-tuple*

$$\langle \mathtt{name}, \mathtt{ft}, \mathtt{int}, \mathtt{proc} \rangle$$

where

1. $\mathtt{name} :: I$ *is the* name *of C,*
2. $\mathtt{ft} :: \mathtt{T} \to \mathbb{B}$ *is called the* filter *of C,*
3. $\mathtt{int} :: \mathtt{T} \to \mathtt{Maybe\ T}$ *is called the* interpreter *of C, and*
4. $\mathtt{proc} :: \mathtt{T} \to \mathtt{T}$ *is called the* processor *of C.*

Formally, the type of C is $\mathtt{Comp}_{\mathtt{T},I}$. As a shorthand, we denote the name, filter, interpreter and processor of a given component C as $\mathtt{name}_C$, $\mathtt{ft}_C$, $\mathtt{int}_C$, $\mathtt{proc}_C$, respectively.

A set of neural components, together with a set of messages, induces a *neural system*:

Definition 11 (Neural system). *Let T be any type and let I be an index set. Then, a* neural system *with message type T and component names from I is a tuple*

$$\langle \boldsymbol{Co}, \boldsymbol{Me} \rangle$$

where

- $\boldsymbol{Co}$ *is a set of neural components (with message type T and names from I) and*
- $\boldsymbol{Me}$ *is a set of elements of type T, called the* set of messages.

3.2.3 Sending and Receiving Messages

We now give a notation for the sending and receiving of messages in a system. Here, we distinguish two aspects: first, the structural, which describes how messages *can* travel in a system and the operational, which describes how they *do* travel in some given scenario.

[1] Specifically, the decidable fragment of System F_ω without higher kinds and only prenex-polymorphism. That is, type constructors can only take types as arguments and are of the form $[\forall x_1, \ldots, x_n]\,\mathtt{C}$ for quantifier-free `C`. This is also called the Hindley-Milner type system. For details, see [Bar91].

Structural Notation

The elements of a component statically determine which messages it can receive and send. Based on the behaviour of the filter, interpreter and processor of a component, we can express a number of properties.

Definition 12 (Message reception). *Let C be a component and m a message. C can receive m if and only if $\texttt{ft}_C\ m = \texttt{True}$ and $\texttt{int}_C\ m = \texttt{Just}\ m'$ for some m'. When C can receive all messages in $\{m_1, \ldots, m_n\}$, we write:*

$$\{m_1, \ldots, m_n\} \rightarrowtail C.$$

We denote the opposite statement, that C cannot receive any message in $\{m_1, \ldots, m_n\}$, by:

$$\{m_1, \ldots, m_n\} \multimap C.$$

Definition 13 (Message sending). *Let C be a component and $m, m_1, \ldots, m_n$ messages. C can send out a message m if and only if there exists a message m_{in} such that $\texttt{proc}_C\ m_{\text{in}} = m$. When C can send all messages in $\{m_1, \ldots, m_n\}$, we write:*

$$C \rightarrowtail \{m_1, \ldots, m_n\}.$$

The opposite statement, that C cannot send any message in $\{m_1, \ldots, m_n\}$, is denoted by:

$$m_1, \ldots, m_n \multimap \{C\}.$$

Definition 14 (Receiving set). *The set of components which can receive a message m is denoted by*

$$\texttt{rec}(m) \equiv \{C \in \mathbf{Co} \mid \{m\} \rightarrowtail C\}.$$

$\texttt{rec}$ *can also be overloaded to refer to the set of components which can receive and interpret at least some message of a component C:*

$$\texttt{rec}(C) \equiv \{C_i \in \mathbf{Co} \mid \exists m :\ C \rightarrowtail \{m\} \wedge \{m\} \rightarrowtail C_i\}.$$

Operational Notation

Whereas the structural notation pertained to the static properties of a neural system, the operational notation describes *traces*: lists of sent and received messages, and the changes they induced in the system.

Definition 15 (Message action). *When a component C_i outputs a message m_{out} that another component C_j receives and interprets as message m_{in}, we write*

$$C_i \to [m_{out}, m_{in}] \to C_j.$$

We refer to this as message action. *If it is clear that the message m does not change, we just write*

$$C_i \to [m] \to C_j.$$

Definition 16 (Trace). Traces *are defined inductively thus:*

1. *Every message action is a trace.*
2. *If T_1 and T_2 are traces, $T_1; T_2$ is a trace.*

"; " denotes sequential execution and is associative. Thus, the semantics of a trace $T_1; T_2; \dots ; T_n$ are that T_1 is executed first, followed by T_2, and so forth, until T_n is reached and the execution ends. For readability, $T_1; \dots ; T_n$ will sometimes be written line-by-line as

$$\begin{array}{c} T_1 \\ \vdots \\ T_n \end{array}$$

Definition 17 (Component mutation). *Let $f_1, f_2, \dots$ be functions $\mathtt{Comp}_{T,I} \to \mathtt{Comp}_{T,I}$ which preserve the names of components, m, m' messages of type T and C a component of type $\mathtt{Comp}_{T,I}$. When C is changed into $(f_n \circ \dots \circ f_1)\ C$ by a message m it receives, or changed into $(f_n \circ \dots \circ f_1)\ C$ by a message m' it sends, we write, respectively:*

$$\begin{array}{c} \dots \to [m] \to \langle f_1, \dots, f_n \rangle C; \\ C\langle f_1, \dots, f_n \rangle \to [m'] \to \dots . \end{array}$$

If no change occurs, that is, if

$$\begin{array}{l} C\langle\rangle \to [m] \to \dots \quad \text{or} \\ \dots \to [m] \to \langle\rangle C \end{array}$$

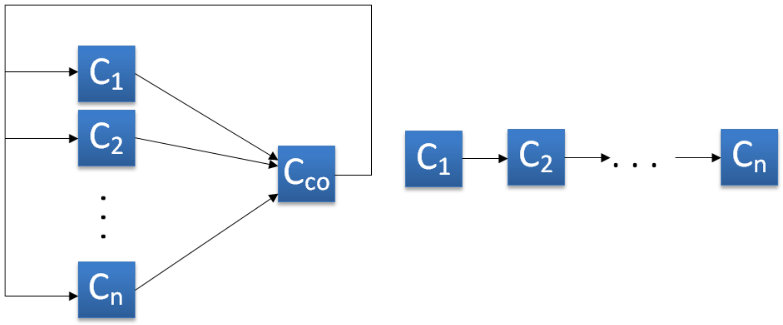
we omit the angle brackets. The semantics are as follows: after by sending or receiving a message, **Co** *is replaced by* $(\mathbf{Co} - \{C\}) \cup \{(f_n \circ \dots \circ f_1)\ C\}$.

Definition 18 (Plastic and non-plastic neural systems). *If, for all messages m and components C, C' in a neural system:*

$$C\langle\rangle \to [m] \to \langle\rangle C'$$

holds, we call the system non-plastic*. Otherwise, we call it* plastic*.*

This definition intends to roughly convey the notion of neuroplasticity, as used in neuroscience: areas in the brain are changed over time through specific patterns of activity. Here, such change is modelled by the execution of functions and the replacement of C in the system by $f_n \circ \dots \circ f_1(C)$.



(a) Components communicating via a central organising mechanism.

(b) A sequence of components.

Figure 3.5: Two special-purpose component arrangements.

3.2.4 Invariants

Such a model does not necessitate the existence of special structures, such as central organizers or sequences of components, one activated after another,[2] but it does not preclude them either. In fact, we can enforce certain features via first-order invariants. For example, a central organizing units for the components $C_1, \dots, C_n$ can be emulated by a component C_{co} which accepts messages and transforms them into an appropriate format for the some other components. Figure 3.5a depicts such an organiser, and Invariant 19 gives a symbolic definition.

Invariant 19 (Central organiser). *Let $C_1, \dots, C_n, C_{co}$ be components, with $C_1, \dots, C_n$ being* peripheral components *and C_{co} being the the* central organiser. *Then, the invariant encoding that $C_1, \dots, C_n$ communicate with each other by sending messages via the central organiser C_{co} is*

$$[\forall i \in \{1 \dots, n\}][\forall m] : \\ (C_i \rightarrowtail \{m\} \Rightarrow \texttt{rec}(m) = \{C_{co}\}) \wedge \left((\mathit{proc}_{C_{co}} \circ \mathit{int}_{C_{co}}(m)) \in \bigcup_{1 \leq j \leq n} \texttt{rec}(C_j) \right).$$

Similarly, sequences can be created by components $C_1, \dots, C_n$, where each components reads the message of the last one. We see such an invariant depicted in Figure 3.5b and formally described in Invariant 20.

Invariant 20 (Sequence). *Let $C_1, \dots, C_n$ be components. Then, the invariant encoding that, for $1 \leq i < n$, a component C_i may only send messages to the next component C_{i+1} is*

$$[\forall i \in \{2 \dots, n\}] : \texttt{rec}(C_{i-1}) = \{C_i\}.$$

[2] An example of such a sequence is discussed by Sander et al. [SGS05] where the authors model the emotion process as a four-step pipeline of relevance, implication, coping and normative significance.

CHAPTER 4

Affective Architecture

Having laid out a general component model, we now describe the components our artificial intelligence will actually use. In this, we make use of the works of Minsky and others to design components that are likely analogous to those that exist in the brains of biological organisms. Of special interest to us are perception, emotional systems, and planning.

4.1 Important Subsystems

If an agent is to efficiently navigate a complex environment, it has to possess certain cognitive capabilities. First, it needs a means of sensory perception — that is, the processing of raw sensory input into an format intelligible to its other cognitive components. Second, if it wants to plan, it also requires the ability to reason about its actions — we might call this belief generation or imagination. Third, it needs the ability to prefer some courses of action over others. In biological organisms, we might think of one's affect, that is, emotions, fulfilling this role.

4.1.1 Sensory Perception

The model presented herein is inspired by Marvin Minsky's *The Emotion Machine* [Min06]. Therein, Minsky proposes a layered mental structure where each successive layer operates on more and more abstract representations of the world, starting with primitive sensations and proceeding all the way to self-conscious reflection and rational planning. Figure 4.1 shows such a layered structure.

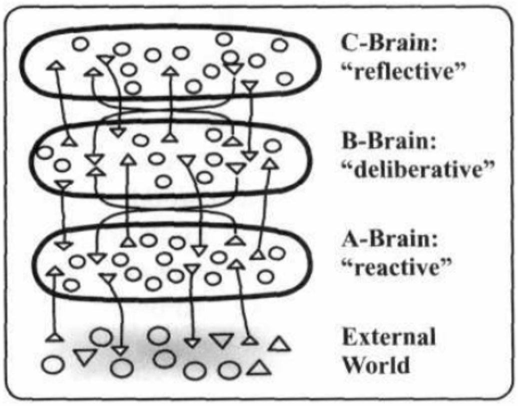


Figure 4.1: Layered perception of the world, from [Min06, p. 100].

The diagram is explained thus [Min06, p. 100]:

> *Now suppose that your A-Brain gets some signals from the external world (via such organs as eyes, ears, nose, and skin) — and that it also can react to these by sending signals that make your muscles move. By itself, the A-Brain is a separate animal that only reacts to external events but has no sense of what they might mean. For example, when the fingertips of two lovers come into intimate physical contact,* the resulting sensations, by themselves, have no particular implications. *For there is no significance in those signals themselves: their meanings to those lovers* lie in how they represent and process them in the higher levels of their minds.

If we apply this to the architecture of Section 3.2, we can devise a system in which each sense S has an associated component C_S which does two things:

1. It consumes the raw sensory information delivered by various organs and output processed input for higher brain functions;
2. as a side effect of this processing, it causes instinctive, low-level reactions in the body, such as pulling away from pain or jumping at a sudden fright.

In Figure 4.2, a slice of just such a system is shown for visual, auditory, olfactory/gustatory, and tactile sensation. The produced data can be of two kinds: one is more abstract than the input and facilitates deliberative action, and the other contains instructions for instinctive behaviour for the body.

4.1.2 Belief Generation and Planning

Broadly speaking, belief generation can be described as “imagination”, and is closely related to sensory perception and world simulation. In examining the system, we might

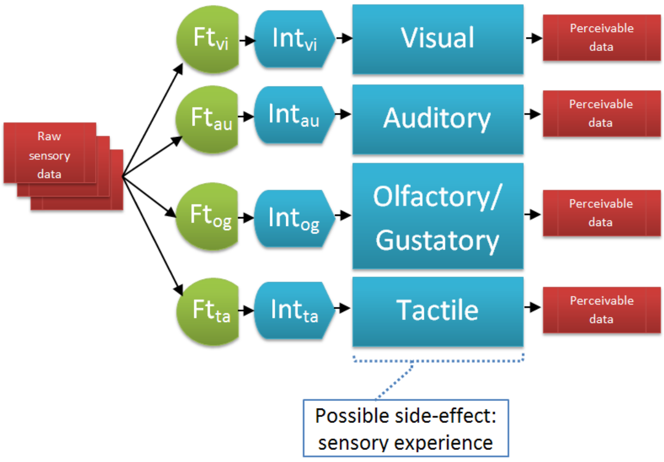


Figure 4.2: Partial structure of sensory perception - raw sensory data is processed and made available to higher functions such as the affective subsystem. The comment "Possible side-effect: sensory experience" signifies the fact that conscious and subconscious sensory experiences might occur as a side-effect of this processing. However, it is currently unknown to neuroscience whether this is indeed the case.

broadly classify its processes into three categories:

1. Belief generation — imagining sights, sounds, etc. Such experiences have much in common with those caused by our sensory organs, yet are marked not as real. In particular, imagined experiences evoke only parts of the conscious experience that accompanies real perceptions. Research by Berthoz [Ber96] and Lotze et al. [LME$^+$99] suggests that (a) the brain indeed uses similar circuitry for real and imagined experiences and that (b) imagined experiences are prevented from being confused with real ones via inhibitory signals. Lotze et al. write:

   > *The results of cortical activity support the hypothesis that motor imagery and motor performance possess similar neural substrates. The differential activation in the cerebellum during EM and IM is in accordance with the assumption that the posterior cerebellum is involved in the inhibition of movement execution during imagination.*

   From the abstract of Berthoz's paper:

   > [...] *experimental evidence suggesting that the brain can use the same mechanisms for the imagination and the execution of movement. In particular the fact that adaptation of the vestibulo-ocular reflex can be obtained by pure mental effort and not solely by conflicting visual and vestibular cues has been suggestive of the fact that the brain could internally simulate conflicts and use the same adaptive mechanisms used when actual sensory cues were in conflict.*

2. World simulation — the imagination of future states. Simulating worlds goes beyond the imagination of sensory experiences; it involves constructing models of worlds and simulating their behaviour. The details of this process are unknown, but we can assert that it is capable of a number of things:

   (a) construction of non-physical worlds, such as mathematical models,

   (b) extrapolation into the future and the past, and

   (c) simulation of the minds itself and other agents.

3. Executive planning — humans can plan both both in immediate and concrete terms (such as body movement) and in the abstract. It is likely that different circuitry is used for movement planning and for planning involving abstract reasoning, in both cases it is necessary that the brain simulate the world in some way. The simulation of the consequences of body movement is likely older than humanity and distinct from the kind of world simulation described above, but both share their function: the agent proposes as series of actions to take, inserts them into some mental world and judges the utility of those actions based on the predicted consequences.

Needless to say, that this process in all its subtleties is immensely complex and thus we simply endeavour to sketch its possible structure only in extremely rough outlines. This sketch is shown in Figures 4.3, 4.4, and 4.5: the world simulation is an ordinary component with a filter and interpreter which outputs, for simplicity's sake, messages marked as imaginary. We can imagine such messages to be very much like ordinary sensory ones, with the exceptions that they have no accompanying sensation and, more importantly, that we are aware of their non-reality. The planning component receives instructions about desirable states and outputs hypothetical actions which the world simulator incorporates. The world simulator's output is in turn read by the planner, which then abandons the plan or decides to pursue it further.

The planner, minimally, has to perform two functions — first, it has to judge the desirability of various world states and second, it has to be able to devise possible steps for the agent based on some strategy. If these two functions and some desired goal(s) are given, the planner can do its work by issuing the following commands, as shown in Figure 4.4:

1. If some goals are not yet reached but appear possible, devise possible steps to take and have the world simulator predict their outcomes.

2. If the goals appear impossible the necessary steps prohibitively undesirable, command the world simulator to cease its activity.

3. If earlier proposed steps turn out to fulfil some goal, contact the agent's executive component.

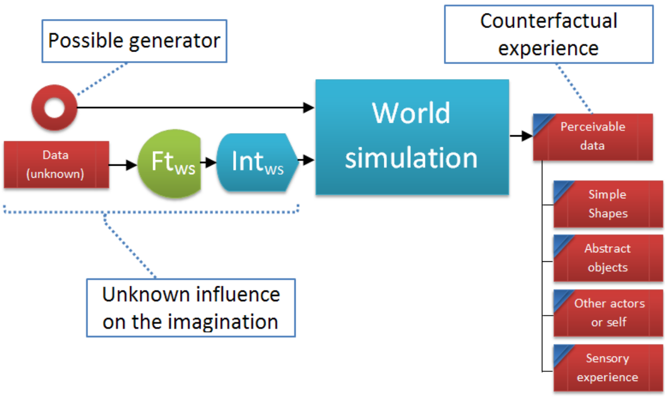


Figure 4.3: Structure of of belief generation & world simulation: messages emulating the output of sensory perception are generated, but are marked as imaginary by unknown means.

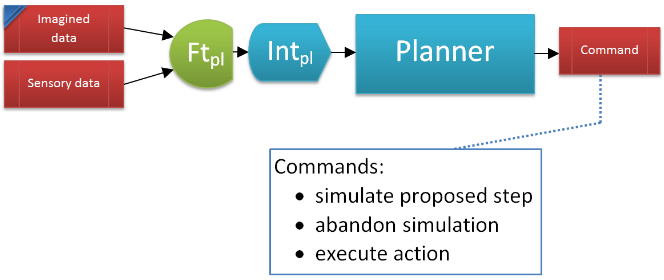


Figure 4.4: Planner with two kinds of inputs: (1) real sensory data and (2) imaginary data which comes from world simulation. On the basis of these inputs, possible steps are developed and sent out as commands.

World Simulation as Rationality

The way in which we just described the interaction between the world simulator and the planner suggests that they function as a pair of guesser and checker: the planner generates ideas on what to do and the world simulation tests their viability in some setting. Indeed, we can model rational thinking as embedded in the world simulator, especially if we make use of a plastic neural system. The proposed steps of the planner might be quite chaotic and irrational, but when given to the world simulator, it recognises them as such and returns a failure signal to the planner, causing it to abandon "bad" paths of cognition. A plastic planner can learn from the consistent failure of certain kinds of steps and, in time, propose them less and less often. Observed as a whole, this system of planner and simulator appears to simply deliver good plans by intuition, even though, in isolation, neither part is very clever.[1]

[1] We do not wish to idealise rationality too much; world simulation is only partly rational and, given faulty information about the world, will err considerably and in documented ways. Similarly, it

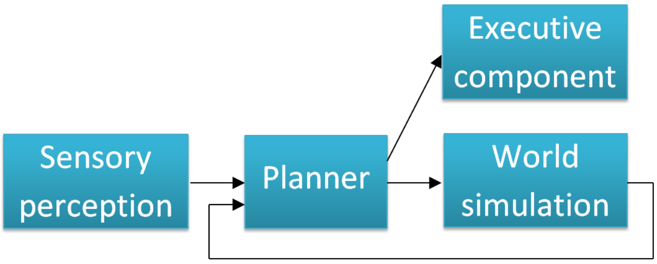


Figure 4.5: Interaction between world simulator and planner: the planner devises possible steps and feeds them into the world simulator, which, in turn, tries to calculate their effects. The results are fed back to the planner.

Model. In a simplified way, we can model the process of logical deduction in a formal system $F = (A, R)$, where A is a recursive set of axioms and R is a recursive set of production rules of the form $(r_{\text{from}}, r_{\text{to}})$ s.t. $r_{\text{from}} \rightarrow r_{\text{to}}$ is a valid production in the system. Let

1. W be a world simulator for the world of propositions $\mathcal{P}$ in (A, R),
2. P a planner,
3. $\text{St} = \{s_1, \ldots, s_p\}$ a set of messages about steps to take,
4. $\text{Cat} = \{K_1, \ldots, K_q\}$ a list of message categories,
5. $\texttt{cur} : W_S$ the current state of the world simulator,
6. $\texttt{ins} :: W_S \rightarrow \text{St} \rightarrow W \rightarrow W$, $\texttt{del} :: \text{St} \rightarrow W \rightarrow W$ functions for inserting or deleting a state change into the world simulator or the planner,
7. $t(i)$ and $b(i)$ functions which increase or decrease the likelihood of sending a message belonging to category K_i and
8. $\bot_i, \top_i$ the failure and success signals of a message belonging to the category K_i.

One step of the interaction between W and P, in a scenario where P proposes steps $s_{i_1}, \ldots, s_{i_n}$, can then be modelled with two traces T_{guess} and T_{check}:

$$\begin{aligned}
T_{\text{guess}}(\texttt{step}) \quad &\equiv \quad P\langle \texttt{ins cur step})\rangle \rightarrow [\texttt{step}, \texttt{step}] \rightarrow \langle \texttt{ins cur step})\rangle W; \\
T_{\text{check}}(\texttt{step}) \quad &\equiv \quad \forall K_i \in \mathit{Cat} : K_i(\texttt{step}) \Rightarrow \\
&\qquad\qquad \text{if } [\exists s_j]\,(\texttt{cur}, s_j) \in R \;\text{ then }\; W\langle\rangle \rightarrow [\top_i, \top_i] \rightarrow \langle t\ i\rangle P \\
&\qquad\qquad \text{else }\; W\langle \texttt{del step})\rangle \rightarrow [\bot_i, \bot_i] \rightarrow \langle \texttt{del step}, b\ i\rangle P.
\end{aligned}$$

is certainly possible for the planner to derange the world simulator by evaluating certain states as so desirable/undesirable that it will pursue even scenarios which the world simulator reports as highly unlikely.

Axioms can be selected by executing $T_{\text{guess}}(\texttt{ax})$ for all $\texttt{ax} \in A$. We can then perform deduction via $T_{\text{guess}}; T_{\text{check}}$, for a probabilistically selected $\texttt{step} \in St$.

Intuitively, T_{guess} guesses a step to take. It does so but inserting it into the planner's world-state via `ins` and then sending a message to the world simulator, which also inserts it into its world state. T_{check} then checks whether the change from `cur` to `step` was legitimate. If so, it determines to which category `step` belongs and sends the $\top$-signal for that category back to the planner. Otherwise, it sends the corresponding $\perp$-signal. The purpose of this is to make it more or less likely, respectively, that the planner should choose the same category of step in the future. The categories, we can imagine, could be things like "modus ponens", "associative reasoning", "appeal to consequences" and so forth.

If we repeat this interaction (with different proposed steps $s_1, \ldots, s_p$ in each iteration), we get an algorithm for logical deduction — that is, since A and R are recursive, the system will recursively enumerate all valid logical formulas, provided that we pursue each path and that the probability of selecting any valid step is > 0. In addition, we could add a goal function g to P s.t. it would accept certain states and stop. Thereby, P and W could be used to prove logical propositions.

4.1.3 Affect

When discussing human affect, one can mean various things: the causation of emotion, its internal mechanisms, the expression of emotion, social communication of emotions, etc. In this document, we restrict our attention just to the internal mechanisms — that is, to the means by which emotions are evoked in an agent and how they shape its thinking.

Furthermore, the issue will only be the causative mechanism itself; taxonomy and hierarchy of emotions are deferred to future versions of this document.

The model presented herein is adapted from Gadanho and Hallam [GH01], who employed it in the context of robot learning. They constructed a system of *feelings* and *sensations* $\mathcal{F}$, *emotions* $\mathcal{E}$, and a hormone storage H.

Figure 4.6 shows this model: *sensations* enter the system and are connected to the *feelings*. They, in turn, determine the agent's *emotions*. The emotions then feed into a *hormone storage*, the contents of which influence, together with the *sensations*, the agent's *feelings*. In the context of their paper, this model had a very restricted application. Its purpose was to merely help guide a robot through a world, and accordingly, $\mathcal{F}$ and $\mathcal{E}$ were only defined as [GH01, p. 47]:

$$\mathcal{F} = \{\text{Hunger}, \text{Pain}, \text{Restlessness}, \text{Temperature}, \text{Eating}, \text{Smell}, \text{Eating}, \text{Proximity}\}$$
$$\mathcal{E} = \{\text{Happiness}, \text{Sadness}, \text{Fear}, \text{Anger}\}$$

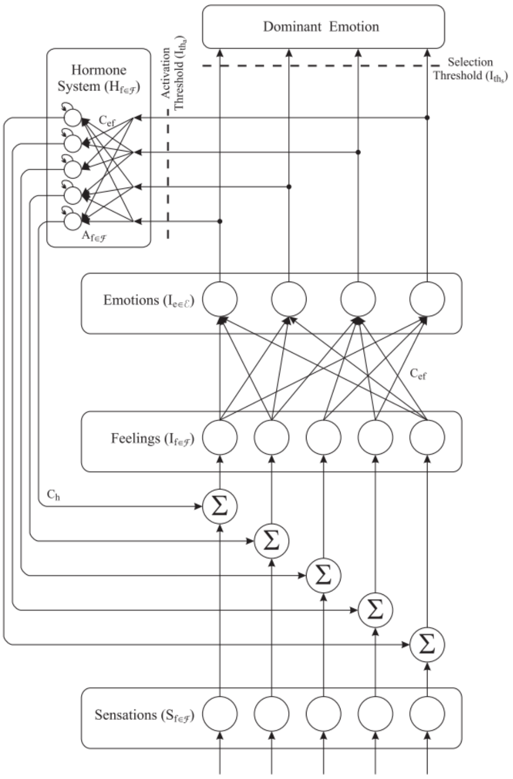


Figure 4.6: Emotional model of Gadanho and Hallam [GH01, p. 46].

The main advantage of Gadanho's and Hallam's model is that (a) it is sufficiently generic to accommodate various schemas and (b) posits an internal state (the hormone storage), giving agents a certain inertia. For example, one can imagine integrating a many-dimensional model like Brazeal's [Bre03] detailed taxonomy of emotion like Ortony's OCC model [OLCC88]. The existence of an internal state is necessitated by the simple observation that our internal world is not solely dependent on momentary stimuli, but merely influenced by them. The idea of a hormone storage might be a simplistic approximation but it, too, can be refined as needed.[2] Figure 4.2 shows the adapted model. The general structure was retained, but the set of sensations was replaced by the sensory processor described in Section 4.1.1 and, instead of a single dominant emotion, competing emotions simply emit messages which are used by execute components and the world simulation.

[2] It might be tempting to simply replace the hormone storage with the message space, but doing so would ignore the role that neurotransmitters like dopamine and serotonin play in cognition, irrespective of the purely computational activity of brain components.

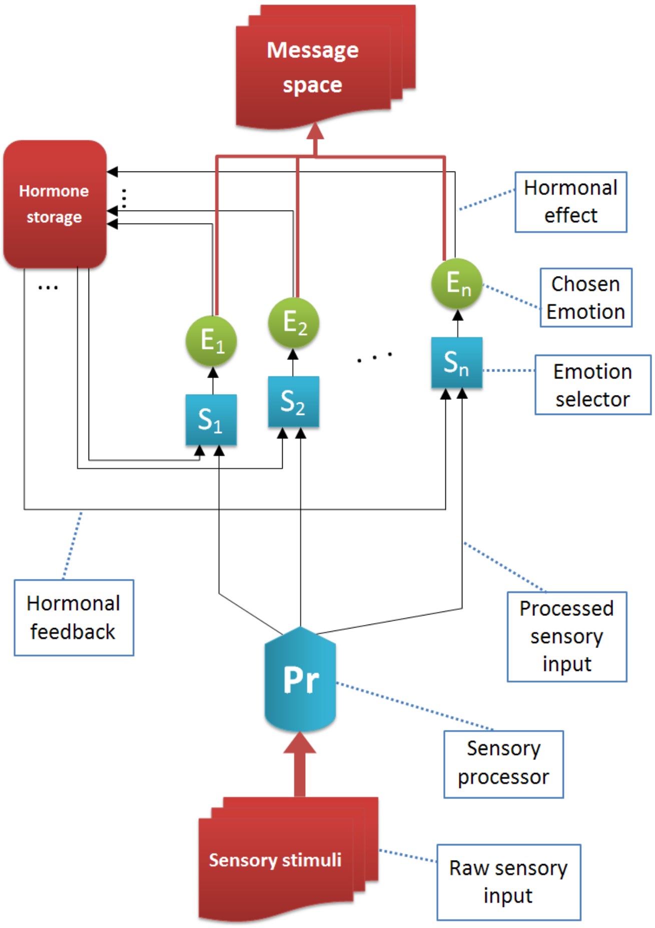


Figure 4.7: Affective subsystem; specialisation of the global neural architecture. In plastic neural systems, selections may change over time.

Affective Subsystems

We now develop the concept of “emotion” in greater detail. The process shown in Figure 4.7 might suggest we simply have a collection of emotions and that all emotions are essentially equal, but we submit that this is not so. Instead, we propose the existence of various subsystems, each responsible for a group of emotions, and each with its own history and distinctive tasks. Thus, we will make the following two assumptions:

1. *“Emotion” is not a singular phenomenon.* Specifically, we will assume that emotions are not simply vectors in a vector space of two, three, or four dimensions, with the only difference between, say, happiness and fear being a different value in the positivity-component. Rather, we will assume that emotions are fundamentally different from each other and that each emotion induces a distinct subjective experience.

2. *There exist emotions which are both different in kind and which pertain to different subsystems in the brain.* This implies that emotions cannot conceptually be seen as a homogeneous set $\{E_1, \dots, E_n\}$. Instead, a number of distinct subsystems are necessitated, each responsible for the causation and processing of a group of emotions. Given this, the only substantial aspect any two emotions might have in common would be our referring to both of them as “emotion”.

Both of these assumptions are rather concrete and thus deserve evidence. In 1999, Davidson and Irwin, using PET and fMRI scanning, found two different systems mediating approach- and avoidance related behaviours [DI99, p. 13]:

> *A large body of lesion, neuroimaging and electrophysiological data supports the view that the prefrontal cortex (PFC) is an important part of the circuitry that implements both positive and negative affect. (...) A number of early studies that evaluated mood subsequent to brain damage suggested that patients with damage to the left hemisphere, particularly in PFC, were more likely to develop depressive symptoms compared with patients having lesions in homologous regions of the right hemisphere. (...) The general finding of left dorso-lateral PFC damage increasing the likelihood of depressive symptoms has been interpreted to reflect the contribution of this cortical territory to certain features of positive affect, which, when disrupted, increases the probability of depressive symptomatology.*

In this, they echo earlies findings by Cacioppo et al. [CG99], Gray [Gra94] and Lang et al. [LBC90] that affect is lateralised, with different hemispheres being responsible for different categories of feeling. It therefore stands to reason that different emotions, being generated by different brain regions, should therefore also be different in their character.

Further, much research has been done in the area of so-called *basic emotions* — a small set of emotions are acknowledged as being both elementary and characteristically distinct from each other. The *Cambridge Handbook of Affective Neuroscience* provides a good overview of the basic emotion theory [AV13, pp. 9-10]. Matsumoto and Eckman [ME09], for instance, identified seven basic emotions: happiness, surprise, contempt, sadness, fear, disgust, and anger.

Damasio [Dam98], drawing upon neuroscientific findings, sketches a model of affect mainly involving the prefrontal cortex, but also the amygdala, the hypothalamus, and the anterior cingulate cortex, as seen in Figure 4.8.

In the same article, he describes how different brain regions are responsible for different kinds of emotion (emphasis ours):

> *Equally problematic is the widespread view that the limbic system is the neural basis for all emotions. A rich body of evidence tells us that this is just not the*

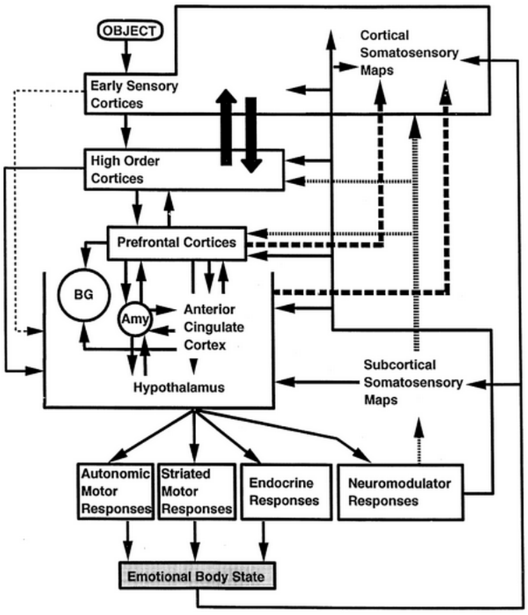


Figure 4.8: Neurological structure of affect, according to Damasio [Dam98].

> *case. Both within and around the limbic system, circuitry connection varied neural sites supports the operation of different emotion. For instance, work on aversive conditioning in rodents has shown that the amygdala is certainly involved in negative emotions such as fear [10,6].* Work in humans, on the other hand, has not only confirmed the amygdala's involvement in negative emotions such as fear and anger, but also shown that the amygdala is not involved in the processing of positive emotions such as happiness, or negative emotions such as disgust.

The last sentence of that quotation is especially revealing: it states that the neurological distinction is not simply one between positive and negative, or one between approach- or avoidance-related emotions, but that each emotion has its own profile of neurological activity and involves its own peculiar set of brain structures.

These facts make it quite clear that emotions are not simply homogeneous phenomena, being induced by a single system in the brain; rather, they are different in character and in the neural structures they involve.

Structure of affect. The system depicted in Figure 4.7 left several parts unspecified: the sensory processor Pr, the emotion selectors $S_1, \ldots, S_n$ and the messages sent by the chosen emotions into the message space. In the following paragraphs, we will flesh out that model in greater detail, building principally on the work of Sander, Grandjean and Scherer [SGS05]. Sander and colleagues partitioned the emotion process into four stages,

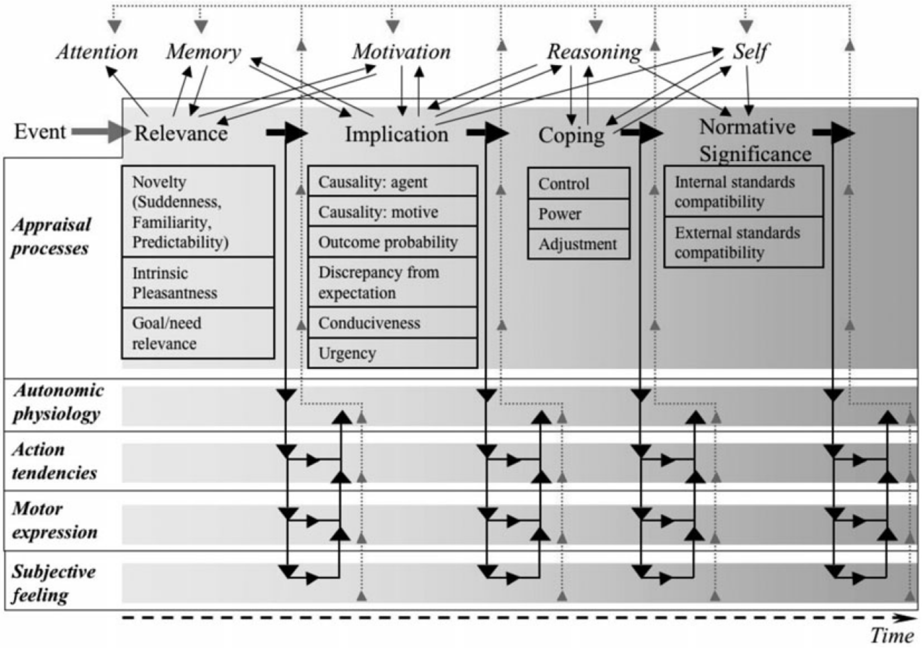


Figure 4.9: The four-stage emotion process according to Sander et al, consisting of relevance, implication, coping and normative significance.

as shown in Figure 4.9. The first is *relevance*, which functions as a filter and detects the intrinsic pleasantness and the level of (emotional) attention that a stimulus demands. The processes of this stage, roughly speaking, correspond to the work of the sensory processor Pr. The second stage is *implication*, where reasoning becomes engaged in order to determine the cause, likely outcome, and urgency of the perceived facts. At this stage, emotions like joy, anger, contentment, disgust, etc. are evoked, together with approach- and avoidance-related behaviours — this corresponds to the emotion selectors $S_1, \dots, S_n$. Deliberate strategies come only in the next stage: *coping*. In it, reasoning and planning become fully engaged. The fourth stage is *normative significance* and deals, in essence, with moral concerns, both internal and those of other agents.

Sander et al. give a good, detailed account of the interactions of affect with other systems, although we would argue that theirs is unduly suggestive of a simple *pipeline*, rather than a mesh of systems into which the affective ones are embedded. In addition, it does not address the interactions with perception, memory, and reasoning. Based on the evidence discussed above, we shall now present a more horizontal view and construct a model of the hypothesised emotional subsystems and their interactions with other parts of the brain. Since no established vocabulary seems to exist in this specific are we shall first introduce a number of terms.

Definition 21 (Evocative system). *An evocative system is a subsystem in the brain responsible for evoking consciously experienced affect within an agent based on internal* *or external stimuli.*

Various such evocative systems can be imagined. For the purposes of this thesis, we will work with the following rough categorization:

Pre-social emotions. Certain behavioural mechanisms can be observed in non-social as well as social animals. The fight-or-flight instinct, for example, is nearly universal, as is the inclination to seek out food, shelter, and other resources. “Instinct” is indeed a more appropriate term in the case of most species, rather than “emotion”, which connotes a certain richness of experience. Nonetheless, we can clearly see that, in more intelligent, social animals, emotions like anger, fear, and joy, have grown out of just these instincts. Hence the term “pre-social emotions”: while emotion itself is quite possibly inherently social, certain emotions are rooted in instincts which are not, and an emotional animal would feel them even if it were the only one of its kind in an environment.

Social emotions. A by far richer subset of emotions are the social ones. Indeed, social situations are the ones where affect can and must truly shine: the presence of other individuals, or of the entire tribe, demand a variety of affect relating to the appraisal of the agents, sympathy/antipathy, respect/contempt, the appraisal of oneself, showing dominance or submission, influencing other group members, taking action as a group, judging the behaviour of agents against norms, etc. It is also in social emotions in which it even makes sense to *show* emotion: facial expressions and gestures provide the signalling and mechanism needed for group coherence and coordinated action.

We can identify several subsystems in the category of social emotion:

1. Reflective judgement about oneself in relation to the group or to abstract norms, primarily pride and shame [TD08], but possibly also jealousy and humiliation (which, in contrast to shame, is attributed to external causes) [Fon09];
2. other-related judgement which determines whether to feel sympathy or antipathy, compassion, respect or contempt, trust or distrust for other individuals;
3. normative judgement, which determines whether others or oneself is acting in accordance with instinctive or cultural norms.

Other classifications are also possible. Haidt [Hai03], for example, identifies those that are other-condemning (disgust, contempt), self-conscious (shame, embarrassment), other-suffering (compassion), other-praising (gratitude, awe). The picture is immensely complex and the neurological structure is presently not known. For the purposes of this thesis, we will therefore content ourselves with only this roughest of outlines.

Aesthetic emotions. This type of emotion is perhaps the least studied in neuroscience and AI. It is certainly the most subtle and the least “utilitarian” type — as such,

> it is philosophers, rather than AI researchers, who study it. For instance, Jenefer Robinson, in *Deeper Than Reason: Emotion and Its Role in Literature, Music, and Art* [Rob05], writes about the affective appraisal of artwork as an unconscious process which partly reproduces the emotions of its creator. In this, she builds upon and modifies Collingwood's 1983 *The Principles of Art* [Col05, Kem12]. Since aesthetics are not the focus of this work, we shall leave it at this mention. A more thorough exploration would be interesting future work, however.

The emotions just listed can all be found in the more extensive taxonomies, chiefly among them in Ortony's OCC model [OLCC88]. The taxonomies, however, tend to neglect the underlying neurology and the chronology of the development of these systems. Ortony's classification specifically is persuasive up to a point, but, despite it being fine-grained, one is left wondering about the underlying structure: which emotions are caused by the same brain regions, what structure, if any, do two given emotions share, to what degree is the classification scheme isomorphic to the actual neurology? This is an active area of research and while these questions are interesting, we have to leave them largely open for now.

The evoked feelings tie into and directly influence the agent's actions. This includes conscious, deliberate ones, such as avoiding an unsympathetic person, but also subconscious ones and those that are purely internal, such as the focusing one's attention to an important topic. These actions all fall under the umbrella term of *executive system*:

Definition 22 (Executive system). *An* executive system *is a subsystem in the brain which makes decisions about the behaviour of an agent's mind or muscular system.*

This definition leaves open what exactly a decision is. In principle, any neural activity in a part of the brain could be seen as a decision of sorts, since it influences neural activity in other parts. While we do perceive certain processes as deliberate and others as automatic, this is simply what our introspection tells us and does not reflect the underlying reality; (conscious) decision-making is as mechanical as any other process in the brain, the chief difference being that we are aware of the workings of that process and perceive the control it exerts over cognition as coming from us.[3]

Nonetheless, there are properties by which we can identify executive systems in the brain: on a sufficiently high level of abstraction, we can see that certain components are receptive to control signals. Certain other components — these are the executive

[3] We should add that we are not even aware of the entirety of our decision-making. This is especially apparent when we are asked to make trivial or random choices. A person who is asked to press a left or a right button, for example, will choose one, seemingly at random, but will not be able to explain why one button was chosen over another. Moreover, there is evidence that the choice is made before the person *knows* that a choice was made: Soon et al. [SSBHH08] instructed subjects to press a button and to record when they thought they made the decision to do so. Brain scanning revealed spikes in the activity of the lateral and medial frontopolar cortices and the posterior cingulate cortex *before* the subjects claimed their decisions were made. In effect, they only became aware of their supposedly free decisions after they had already been made. From their conscious perspective, the decision simply "popped into their heads".

systems — have as their *chief purpose* the sending of such control signals. The former accomplish some conceptually small task and essentially serve as building blocks. The latter structure the work and assemble the small building blocks into compound actions. See Section 4.1.2, where planner and world simulator work in tandem, with the world simulator bearing the workload and the planner having control.

We can now distinguish certain kinds of action. While those performed with the "body" (i.e. the skeletomuscular system) are the most visible ones, we, as shown, also make decisions regarding the contents of our minds — we decide *what to think about.* We then add the distinction between consciously and subconsciously made actions and get the following four categories of executive system:

- *Subconscious motor control:* instinctive reaction, such as the jerking away from pain, jumping when startled, and turning towards interesting visual stimuli.

- *Conscious motor control:* deliberate, planned action which the agent experiences as a choice.

- *Subconscious mental control:* involuntary but consciously experienced changes to the mind-state of an agent which are perceived as activity rather than mere feeling. This includes like obsessing over an issue, manias, fantasies insofar as involuntary, etc.

- *Conscious mental control:* deliberate mental changes of an agent. This includes the making of decisions, the deliberate focusing of attention, deliberate planning, deliberate strategy selection, and so forth.

We stress that these are *categories* of systems, not systems themselves. We control our minds and our bodies in a variety of ways and there is no evidence that there is some sort of master control system anywhere in the brain responsible for these tasks. The planner from Section 4.1.2 only controls one other component — and it might very well be that it does not even exist in the brain as one compact component. It might be that a variety of smaller systems are tugging and vying for control and balanced against each other in such a way that the illusion of dedicated planning component is created.

4.2 Interaction Between Affect and World Simulation

Section 4.1.2 outlined what could be called *deliberate action* in the from of a planner-world-simulator loop. Section 4.1.3 described the structure and components of affect. These systems are of course not isolated from each other; emotional states influence both the planner's chosen heuristics and the world simulator's creation of worlds. In addition, attention, also influenced by affect, controls the allocation of cognitive resources. We now explore these relationships in further detail.

Planning as search. In the AI literature, search algorithms are of great importance. In this context, we can view the loop between planner and world simulator as a greedy search: the planner chooses the nodes which are to be expanded and sends them to the world simulator. It, in turn, performs the expansion by simulating the appropriate worlds. These simulated worlds are sent back to the planner for evaluation regarding desirability (i.e. cost). This presents an obvious problem: since greedy search is not complete, our planner-world-simulator loop can't be complete either. In fact, the situation is worse — greedy search computes the cost of all candidates for expansion and chooses the cheapest, whereas our planner, being heuristic, might not consider certain nodes at all.

This might seem damning, but we must also consider the interaction with attention and memory. First, planned steps are committed to memory and thus, we gain access to past costs. An agent does not plan blindly, but can recall how long its plans are and what costs past planned steps entail. Given this information, we can turn the greedy algorithm into an A* search, with the qualification that the planner might not consider certain nodes. The mechanism of attention can further be used to enhance the search: if planning along a certain path takes too long, the agent might decide to abandon it altogether and start afresh with a different strategy. This failure too is stored in memory and can influence the planner in the new planning process by making the proposing of steps of the previously pursued path unlikely.

CHAPTER 5

Implementation

Having laid the theoretical framework, we come to the practical part of this thesis — a proof-of-concept implementation of multiple affective agents interacting with each other. This chapter describes the world in which the agents and the Wumpuses will act, as well as the architecture of these agents.

The goal is the creation of a toy AI that semi-realistically mimics animal intelligence, the operative word being "mimic". As Sloman [Slo01] pointed out, naming a variable *anger* or *love* does not give a program some qualitative experience. Indeed, our much more modest goal is to *emulate* the behaviours that are associated with certain mental states — and to show how such emotional states, interacting with reasoning, can help an agent thrive in its environment. These programs will really only be soulless automata, employed to illustrate a point about living beings with brains, acting with incomplete information.

5.1 World

The choice of the world profoundly affects the implementation of the agent — its knowledge base, mechanism of perception and interaction, the required complexity of the implementation. On the one hand, the world should be simple enough to permit a reasonably small and effective agent which does not have to solve hard AI problems (like human-level sight) to deal with what we, in this context, might call details — but on the other hand, the world should be sufficiently complex to allow agents to distinguish themselves. This is especially true in the case of an affective agent whose actions should be visibly influenced in rich and subtle ways by its emotional state. We shall first lay out the design goals and then evaluate three possible worlds for agents.

Design Goals. The two most important criteria for prospective worlds are richness of interaction and world complexity, in that order. As said, an evaluation of affective agents is only possible if they can interact with their environment and other entities in a sufficiently complex way to allow agents with different emotional profiles to be distinguished from each other. Mechanisms of problem-solving like STRIPS [FN71], A* search [HNR68], answer-set programming [GL88], forward-/backward-planning, etc. have been explored in the context of structurally simple worlds, generally those representable through propositional logic, cost-functions, decision tress, and the like. While these are useful, they are less appropriate in an affective scenario for the following two reasons:

1. they are geared towards finding provably optimal solutions to computationally expensive but conceptually simple problems like planning or game-playing and

2. they rely heavily on hand-crafted ontologies and domain knowledge on the part of the human programmer.

For a world to be useful to us and to avoid these pitfalls, it should be in some sense realistic: it should permit a large number of different kinds of interactions, and it should not provide agents in it with perfect knowledge about its rules.

We admit that we, in this matter, stand in opposition with Marvin Minsky, who famously recommended the use of idealised micro-worlds to study artificial intelligence, in that same vein in which physics makes use of ideal, frictionless planes and perfect spheres. His argument certainly has merit, but we believe that emotion is too complex a phenomenon for such abstract scenarios. In too simple a setting, pure reasoning not only easily outperforms emotional behaviour, but avenues for exhibiting emotional behaviour are scarce to begin with. For this reason, we propose that, in this context, rich interactions should take precedence over idealization and simplicity.

It is of course still desirable to minimise complexity as far as possible. An overwhelmingly complex world has two obvious drawbacks: first, the required complexity of an agent scales with the complexity of the world; second, the more complex the world, the harder it is to reason about it. If there are a hundred ways to succeed, for instance, agent performance becomes quite difficult to measure.

5.1.1 Wumpus World

The traditional Wumpus world, as described in Russell and Norvig's *Artificial Intelligence: A Modern Approach* [RN10, p. 236], is a grid-based, 4x4 cave world with one agent, one monster — the Wumpus — and gold placed in random rooms. The agent starts at position $\langle 1, 1 \rangle$ and can move forward or turn 90° to the left or right. If it enters a room with a pit or a live Wumpus, it dies; its goal is to find and collect the gold and then move back to position $\langle 1, 1 \rangle$ to climb out of the cave. In addition, it has one arrow which he can fire straight ahead to defend against the Wumpus. The agent cannot see, that is, it only has access to information about its own cell and cannot directly observer other ones. To quote Russell and Norvig, the agent has only the following local information [RN10, p. 237]:

- *In the square containing the Wumpus and in the directly (not diagonally) adjacent squares, the agent will perceive a* Stench*.*
- *In the squares directly adjacent to a pit, the agent will perceive a* Breeze*.*
- *In the square where the gold is, the agent will perceive a* Glitter*.*
- *When an agent walks into a wall, it will perceive a* Bump*.*
- *When the Wumpus is killed, it emits a woeful* Scream *that can be perceived anywhere in the cave.*

This type of world is simple enough to be amenable to rule-based reasoning, although it can contain ambiguous situations where the agent does not have enough information to make the best choice. For example, if an agent moves to position $\langle p_x, p_y\rangle$ and experiences a breeze, 1, 2, or 3 adjacent rooms may contain pits, but it cannot be safely determined which ones these are. Thus, occasionally, the agent must choose between climbing out without the gold and risking death by pit or Wumpus.

For our purposes, this is a bit too simple, however. Caution/bravery is the only axis along which agents can be differentiated and although various complex behaviours — such as trying one dangerous cell, then going back and trying another one to explore the world — are possible, these do not have a clear relation to emotional states.

Let us, while staying true to the spirit of the original, now define a type of extended Wumpus world $\mathcal{W}_{\text{ext}}$ that allows more varied interaction between agent an environment.

Definition 23 ($\mathcal{W}_{\text{ext}}$-type world). *Let* $\texttt{T}_\texttt{v}$*,* $\texttt{T}_\texttt{e}$*,* $\texttt{T}_\texttt{g}$ *be arbitrary types. Further, let* G *be a directed graph with vertex labels of type* $\texttt{T}_\texttt{v}$ *and edge labels of type* $\texttt{T}_\texttt{e}$*, and let* gl *be an object of type* $\texttt{T}_\texttt{g}$*. Then, the tuple* $\langle G, \text{gl}\rangle$ *is a* $\mathcal{W}_{\text{ext}}$-type world *(with type parameters* $\texttt{T}_\texttt{v}$*,* $\texttt{T}_\texttt{e}$*, and* $\texttt{T}_\texttt{g}$*). We call* G *the* world frame *and* gl *the* world data*.*

We can interpret each vertex v in the graph as a room with attached data $l(v)$ of type $\texttt{T}_\texttt{v}$, and each edge e as an unidirectional connection between rooms with attached data (such as path costs) $l(e)$ of type $\texttt{T}_\texttt{e}$. The object gl is the global world data.

Next, we specify some properties of the world frame:

Definition 24 (World properties). *Let* $W = \langle G, \text{gl}\rangle$ *be a* $\mathcal{W}_{\text{ext}}$*-world. Then,* W *is*

1. reflexive*, if, for all* $v \in V(G)$*,* $(v, v) \in E(G)$*,*
2. non-Euclidean*, if for all pairwise distinct* $v_1, v_2, v_3 \in V(G)$*,* $\{(v_1, v_2), (v_1, v_3)\} \subseteq E(G)$ *implies* $(v_2, v_3) \notin E(G)$*,*
3. symmetrical*, if for all* $v_1, v_2 \in V(G)$*,* $(v_1, v_2) \in E(G)$ *implies* $(v_2, v_1) \in E(G)$*,*
4. connected*, if for all* $v_1, v_2 \in V(G)$*, there exists a path from* v_1 *to* v_2 *in* G*, and*
5. n-dimensionally embeddable*, if there exists an infinite,* n*-dimensional grid* S *such that* $G \subseteq S$*.*

The first four properties speak for themselves. As for the fifth — Figure 5.1 shows an example of a 2-dimensionally embeddable frame. A frame G is n-dimensionally embeddable if it is a fragment of an infinite, n-dimensional, square grid of nodes S, plus any loops G might have. When we embed this infinite grid S into $\mathbb{R}^n$ through an embedding, every edge corresponds to a vector of length 1 along exactly one dimension. If we additionally take G's loops to correspond to null-vectors, this induces an *edge direction function* and a *position function*:

Definition 25 (Edge direction and position). *Let $W = \langle G, \mathrm{gl}\rangle$ be an n-dimensionally embeddable world (for some n) and ϵ an embedding of W into $\mathbb{R}^n$. Then, the* edge direction function *is given by*

$$\Delta_n^\epsilon : E(G) \to \{0, x_1^+, x_1^-, x_2^+, x_2^-, \dots, x_n^+, x_n^-\}$$

with 0 corresponding to a loop and x_i^+/x_i^- corresponding to forward/backward movement in the ith dimension. Furthermore, a position function *is an injective mapping*

$$\pi^\epsilon :: V(G) \to \mathbb{R}^n,$$

with $pi^\epsilon(v) = r$ indicating that under ϵ, v was mapped to position r in $\mathbb{R}^n$. When the number of dimensions and the embedding are obvious, we omit n and ϵ. Finally, the indexing function *of W is given by:*

$$\begin{aligned} &[.] : n\text{-}\textit{dimensionally embeddable world} \to \mathbb{R}^n \to \texttt{Maybe}\ V(G),\\ &W[p] \equiv \begin{cases} \texttt{Just}((\pi^\epsilon)^{-1}\ p) & \textit{if } (\pi^\epsilon)^{-1}\ p \textit{ is defined},\\ \texttt{Nothing} & \textit{otherwise.}\end{cases}\end{aligned}$$

Note that, since π^ϵ is injective by definition, an inverse $(\pi^\epsilon)^{-1}$ also exists.

We will give agents access to Δ_n^ϵ and π^ϵ (or simply Δ and π) to allow them to determine their position and direction in the world. Providing such information might seem problematic, but we thereby free ourselves from having to insert things like landmarks, wind currents, stars, and other navigational aids into the world. Given that navigation is not the focus of this thesis, this seems an appropriate simplification. Using the above properties, we can specify a subtype of $\mathcal{W}_{\text{ext}}$-type worlds:

Definition 26 (2D grid world). *Let $W = \langle G, \mathrm{gl}\rangle$ be a $\mathcal{W}_{\text{ext}}$-type world (with type variables $\mathtt{T_v}, \mathtt{T_e}, \mathtt{T_g}$). If W is reflexive, connected, and 2-dimensionally embeddable W is a* 2D grid world. *Every 2D grid world has an associated function $\Delta_2 : E(G) \to \{0, x_1^+, x_1^-, x_2^+, x_2^-\}$ and a position function $\pi : V(G) \to \mathbb{R}^2$.*

Note that every n-dimensionally embeddable world is also symmetrical and non-Euclidean.

Grid worlds, as we have seen, are potentially infinite, n-dimensional grids, although their cells need not form a square or cube. Their shape can be irregular in that some rooms and connections may be missing, as long as the shape as a whole stays connected.

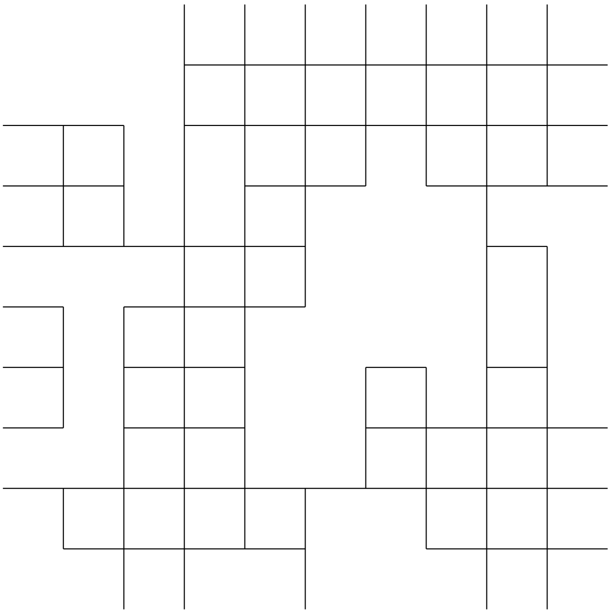


Figure 5.1: A segment of 2-dimensionally embeddable world. The vertices are its rooms, the edges are the connections between the rooms.

2D grid worlds are representationally the same as $\mathcal{W}_{\text{ext}}$-type worlds; they just have some structural invariants on their frames. If we additionally specialise the representation through the type parameters $\mathtt{T_v}$, $\mathtt{T_e}$, and $\mathtt{T_g}$, we arrive at the type of world which will serve as the environment for our agents: the "jungle world" $\mathcal{W}_{\text{jun}}$.

Definition 27 **($\mathcal{W}_{\text{jun}}$).** *Let* $\mathtt{T_v}$*,* $\mathtt{T_e}$*,* $\mathtt{T_g}$ *be the following tuples:*

$$\begin{aligned}
\mathtt{TV}_{\text{jun}} &= \langle \mathtt{entity} :: \mathtt{Entity}, \\
&\quad\ \mathtt{plant} :: \mathtt{Maybe}\ \mathbb{R}, \\
&\quad\ \mathtt{stench} :: \mathbb{R}, \\
&\quad\ \mathtt{breeze} :: \mathbb{R}, \\
&\quad\ \mathtt{pit} :: \mathbb{B}, \\
&\quad\ \mathtt{meat} :: \mathbb{N}, \\
&\quad\ \mathtt{fruit} :: \mathbb{N}, \\
&\quad\ \mathtt{gold} :: \mathbb{N}\rangle, \\
\\
\mathtt{TE}_{\text{jun}} &= \langle \mathtt{danger} :: \mathbb{R}, \\
&\quad\ \mathtt{fatigue} :: \mathbb{R}\rangle, \\
\\
\mathtt{Temp} &= \mathtt{Freezing} + \mathtt{Cold} + \mathtt{Temperate} + \mathtt{Warm} + \mathtt{Hot}, \\
\\
\mathtt{TG}_{\text{jun}} &= \langle \mathtt{time} :: \mathbb{N}, \\
&\quad\ \mathtt{temperature} :: \mathtt{Temp}\rangle.
\end{aligned}$$

$\texttt{Entity}$, $\texttt{Item}$, $\texttt{Agent}$ *and* $\texttt{Wumpus}$ *are the following records:*

$$\begin{aligned}
\texttt{Entity} &= \texttt{Ag Agent} + \texttt{Wu Wumpus} + \texttt{None}, \\
\texttt{Item} &= \texttt{Gold} + \texttt{Fruit} + \texttt{Meat}, \\
\texttt{Agent} &= \langle \texttt{name} :: \texttt{String}, \\
&\quad \texttt{direction} :: \texttt{X}_1^+ + \texttt{X}_1^- + \texttt{X}_2^+ + \texttt{X}_2^-, \\
&\quad \texttt{health} :: \mathbb{R}, \\
&\quad \texttt{fatigue} :: \mathbb{R}, \\
&\quad \texttt{inventory} :: [\langle \texttt{Item}, \mathbb{N} \rangle], \\
&\quad \texttt{state} :: \texttt{S} \rangle, \\
\texttt{Wumpus} &= \langle \texttt{health} :: \mathbb{R}, \\
&\quad \texttt{fatigue} :: \mathbb{R} \rangle.
\end{aligned}$$

The last component of $\texttt{Agent}$, $\texttt{state} :: \texttt{S}$, *is the internal state of agents which we will discuss later.*

Let gl *also be a value of type* $\texttt{TG}_{\text{jun}}$ *and let* G *be any 2D grid world with node labels of type* $\texttt{TV}_{\text{jun}}$ *and edge labels of type* $\texttt{TE}_{\text{jun}}$. *Then,* $\langle G, \text{gl} \rangle$ *is a* $\mathcal{W}_{\text{jun}}$*-type jungle world.*

The intuitive meaning of $\mathcal{W}_{\text{jun}}$is the following: the two-dimensional grid world is inhabited by multiple agents and wumpuses, where the former act according to their agent function and the latter act mechanically. In addition, each cell in the world may have a plant or a deadly pit on it, in addition to a certain amount of fruit, meat, and gold. Agents and wumpuses move in the world by traversing edges which have associated fatigue and danger levels, representing easy and difficult paths. Local information is available to expedite navigation: stench (emanating from wumpuses) and breeze (emanating from pits). Finally, the temperature and the time dictate global environmental conditions.

Although the field names are suggestive of the way in which a $\mathcal{W}_{\text{jun}}$-type world works, the type, strictly speaking, only specifies the data and frame properties. We can employ such worlds in any sort of scenario, with whatever semantics we wish. Notwithstanding, our implementation will use a straightforward *standard semantics*, that have the world work in the manner of a simple ecosystem in which predators hunt for prey and compete with each other. The wumpuses fulfil the role of carnivorous predators which roam the world, hunting and attempting to kill agents on sight. Agents, in turn, are hunter-gatherer omnivores who can sustain themselves either through eating plants, killing Wumpuses for their meat, or by acquiring resources from other agents. They may carry meat or fruits in their inventory, or gold, which has no intrinsic use, but which may be used as an exchange medium, provided that multiple agents have the mental ability to facilitate bartering. The term “jungle world” reflects the uncertainty under which its actors must act. They only have access to quite limited local environmental information, and they possess no communication protocol upon which they could base their cooperation. Analogously to real-world situations, agents must rely on simple gestures to infer the intentions of their peers, and they cannot know whether they are misunderstanding these, or whether they are being deceived. The aim of this mechanism is to allow the experimentations with things like social adaptation, prejudice, and trust. The goal of simulating affective agents in such a world is to see which behavioural profiles are successful, how they develop over multiple generations, and how they engage each other.

Definition 28 (Semantics and runs of $\mathcal{W}_{\mathbf{jun}}$-type worlds). *A function φ of type $\mathcal{W}_{\text{jun}} \to \mathcal{W}_{\text{jun}}$ is called a* semantics of $\mathcal{W}_{\text{jun}}$-type worlds*. For a $\mathcal{W}_{\text{jun}}$-type world, W, the iterated application of φ to W, given by the list $[W, \varphi\ W, \varphi^2\ W, \varphi^3\ W, \ldots]$, is called a* run of W (with semantics φ)*. Furthermore, $\varphi^n\ W$ is referred to as the* state of W's simulation at time n (with semantics φ)*.*

Definition 29 (Standard semantics of $\mathcal{W}_{\mathbf{jun}}$-type worlds). *The standard semantics for $\mathcal{W}_{\text{jun}}$-type worlds is given by the function* sem :: $\mathcal{W}_{\text{jun}} \to \mathcal{W}_{\text{jun}}$. sem*, defined as*

$$\text{sem}\ \langle G, \text{gl}\rangle = \langle G', \text{gl}'\rangle,$$

where $\langle G', \text{gl}'\rangle$ is identical to $\langle G, \text{gl}\rangle$, except for the following changes:

1. Environment: *For all $v \in V(G)$, the following is defined:*

   - Wumpus: *Set v's stench to*

     $$\max\left\{0, 1 - \frac{\max\{0, ||v, w|| - 1\}}{3}\right\}$$

     where w is the closest cell that has a Wumpus on it. If there are no Wumpuses, set w's stench to 0.

   - Plant: *If there is a plant on v and it has a growth value of < 1, increase its growth by $\frac{1}{10}$.*

   - Pit: *If there is a pit in a cell w at a distance ≤ 3 from v, set the breeze to*

     $$\max\left\{0, 1 - \frac{\max\{0, ||v, w|| - 1\}}{3}\right\}.$$

2. Global data: *The* daylight function *is defined as*

   $$\texttt{light}\ t = \begin{cases} 0 & \text{if} \quad 20 \leq |t - 25|, \\ 1 & \text{if} \quad 15 \leq |t - 25| < 20, \\ 2 & \text{if} \quad 10 \leq |t - 25| < 15, \\ 3 & \text{if} \quad 5 \leq |t - 25| < 10, \\ 4 & \text{if} \quad |t - 25| < 5. \end{cases}$$

   The new global data gl′ *are given by*

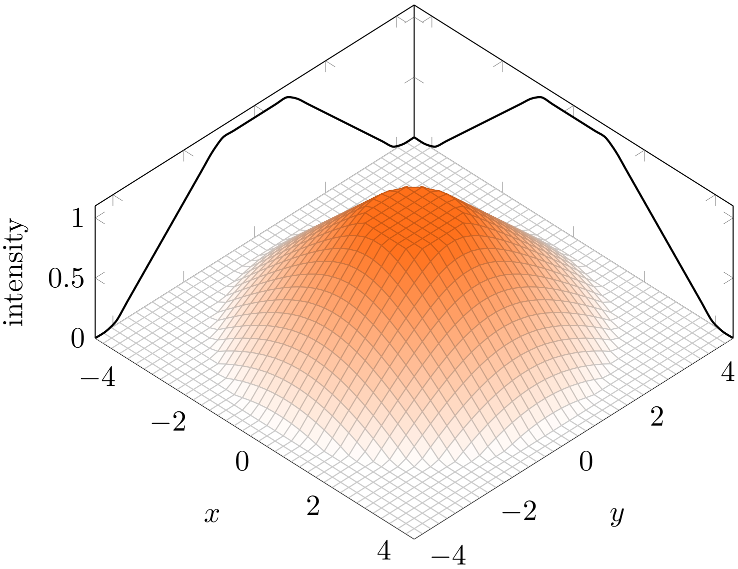


Figure 5.2: Graph of the intensity of the stench/breeze, as a function of the distance from a Wumpus/pit.

$$\begin{aligned}
\texttt{time}' &= \texttt{time}\ \mathrm{gl} + 1 \bmod 50, \\
\texttt{temperature}' &= \begin{cases} \texttt{Freezing} & \text{if } \texttt{light}\ \texttt{time}' = 0, \\ \texttt{Cold} & \text{if } \texttt{light}\ \texttt{time}' = 1, \\ \texttt{Temperate} & \text{if } \texttt{light}\ \texttt{time}' = 2, \\ \texttt{Warm} & \text{if } \texttt{light}\ \texttt{time}' = 3, \\ \texttt{Hot} & \text{if } \texttt{light}\ \texttt{time}' = 4, \end{cases} \\
\mathrm{gl}' &= \langle \texttt{time}', \texttt{temperature}' \rangle.
\end{aligned}$$

3. Wumpus behaviour: *Every Wumpus has three behaviours:*

- *If the Wumpus is adjacent to a player, it performs the* `attack` *action on that player.*
- *If there is a player reachable with at most* ($\texttt{light} \circ \texttt{time}$) gl *edges, move along the edge that minimises the distance to that player* (*in* $\mathbb{R}^2$)*. If there are multiple players, choose one at random as target. This target choice remains until the player is no longer within range.*
- *If there is no player within range, move in a random direction with probability*

$$0.2 \times (1 + (\texttt{light} \circ \texttt{temperature})\ \mathrm{gl}).$$

Whenever a Wumpus travels along an edge e *with* $\Delta\ e \neq 0$*, apply* 0.1 *damage with probability* `danger` e*.*

4. Agent behaviour: *Agents always act after Wumpuses and, depending on their implementation, may choose one of the following actions:*

- move*: Move along an edge e. If* $\Delta\ e = 0$*, restore* 0.1 *of the agent's fatigue, otherwise reduce it by* $0.05 \times$ `fatigue` *e. Additionally* ($if\ \Delta\ e \neq 0$)*, apply* 0.1 *damage with probability* `danger` *e.*

  If an agent's fatigue is below 0.2*, it cannot choose this action.*

- rotate*: The agent changes the direction into which it is facing to a value in* $x_1^+, x_1^-, x_2^+, x_2^-$.
- attack*: Move along an edge e to attack an agent or Wumpus.*
- give*: Give an item i from the agent's inventory to another agent a.*
- gather*: If there is a plant with a fruit on the agent's cell, take the fruit and put it in the agent's inventory.*
- butcher*: If there is a dead Wumpus on the agent's cell, remove it and add an item of meat to the agent's inventory.*
- collect*: If there is n gold on the player's cell, take an amount m* ($1 \leq m \leq n$) *of it an put it into the agent's inventory.*
- eat*: Eat a meat- or fruit-item i from the agent's inventory. Restore* 0.5 *health, to a maximum of* 2.0*.*
- gesture*: Expresses a gesture in the form of a string s. All other agents on the same cell receive s.*
- nothing*: Doing nothing this turn.*

5. Combat mechanics: *When two entities* `A`*,* `B` *attack each other, an entity being either an agent or a Wumpus, the health of* `A` *is subtracted from the health of* `B` *and vice versa. Any entity whose health thereby reaches or goes below 0 dies.*

Upon death in a fight, one meat is added to the cell. If the dead entity was an agent, the amount of gold, fruit, and meat in its inventory are added to values of the `gold`*,* `fruit`*, and* `meat` *fields of its cell.*

6. Movement mechanics: *Agents and Wumpuses may only move to another cell if that movement does not reduce their fatigue below 0. Neither an agent nor a Wumpus may move onto a cell that already has another agent or a Wumpus, or a plant. Any agent or Wumpus can move into a pit, but doing so deletes either of them from the world.*

7. Hunger: *If an agent does not eat a fruit or a meat item, its health declines by 0.01. If its health thereby reaches 0, it dies and the contents of its inventory are added to the values of the* `gold`*,* `fruit`*, and* `meat` *fields of its cell. However, no additional item of meat is created on the cell.*

It ought to be said that the formulae and constants used in the above definitions are, fundamentally, judgement calls and that there is no theoretical reason for choosing these over others. Nonetheless, we can give them an intuitive meaning:

1. *Environment:*

    - *Wumpus:* Wumpuses carry around them a wafting stench, the strength of which drops off linearly for three cells.
    - *Plant:* Fruits grow periodically on plants, although a plant can only bear one fruit at a time.
    - *Pit:* The breeze coming from pits works via the same mechanism as the stench of wumpuses, but as pits are immobile, the strength of a breeze does not change with time.

2. *Global data:* A day is segmented into 50 periods, where a time of 25 represents midday, and 0/50 represents midnight. The temperature is a function of the daytime, with midday being the hottest and midnight being the coldest.

3. *Wumpus behaviour:* Wumpuses are day-active and roam around randomly. At night, they are likely to sit still. When they sight an agent (depending on light conditions), they will invariably attempt to close the distance and attack.

4. *Agent behaviour:* Agents are free to do choose any action they wish. They may move around, attack wumpuses and other agents, gather items (fruit from plants, meat from dead wumpuses, gold lying around), consume food, give items to other agents, or communicate with them. They are limited by their health, which is depleted by travelling along dangerous paths and by fights, and by fatigue. They must thus periodically eat and rest to keep both up.

5. *Combat and hunger:* Agents must compete with each other for finite resources. They can either eat fruits from plants, get meat from killed Wumpuses and agents, or they can kill other agents for the contents of their inventories. Because their health slowly but steadily decreases, they must periodically eat food. In addition, fatigue limits their ability to move, forcing periodic rests.

5.2 Agents

The agents of our simulation are composed of two parts: their minds and their bodies. Their minds constitute their sensors and agents functions; their bodies, make up their actuators, although they are more than that. An agent's body can be damaged and healed, perceived by others, and it can hold items. As such, the bodies are actually part of the world. From the point of view of the agent's mind, they are external objects they happen to control.

5.2.1 Body and Percepts

As we saw in Definitions 27 and 29, agents (1) have a body composed of a name, health, fatigue, and an inventory of items they carry, and (2) can execute one of a fixed set of actions at each step. These data function in the obvious way: the name is publicly available information other agents can use for identification, the agent is killed when its health drops to zero, fatigue determines the effectiveness when attacking and prevents movement when low, and the inventory is used to store items which the agent can use for itself or give away to others.

What we are missing is the description of the agent's percepts in the world. As in the original Wumpus world, an agent can perceive everything on its cell:

1. the plant, if present,
2. the breeze,
3. the stench, and
4. the amount of fruit, gold, and meat.

In addition to this local information, the agent also has access to the global world state:

1. the temperature and
2. the current time.

The most important means of perception will be the agent's sight, however. The sense of sight is modelled via an approximately $\frac{\pi}{4}$ radians sight cone which is oriented in the agent's direction and is shortened or lengthened, depending or daylight. Formally:

Definition 30 (Sight cone). *Let $W = \langle G, \mathrm{gl} \rangle$ be a 2D grid world and let an agent be on vertex $v \in V(G)$, facing into direction d. Let further l_d be the line starting at v and extending infinitely into direction d, and $l_{v,w}$ be the line from v to w. Then, any other vertex $w \in V(G)$ falls into the agent's sight cone exactly if:*

1. *the angle between $l_{v,w}$ and l_d is less than or equal to $\frac{\pi}{4}$,*
2. $||v, w|| \leq 1.5 \times (((\texttt{light} \circ \texttt{time})\ \mathrm{gl}) + 1)$*, and*
3. *there is a path $v_1, v_2, \ldots, v_n$ from v to w in G such that the distance between v_i and the closest point along $l_{v,w}$ is less than or equal to $\frac{\sqrt{2}}{2}$ $(1 \leq i \leq n)$.*

Criterion 1 restricts the sight cone to $\frac{\pi}{4}$ radians; criterion 2 limits its length based on light conditions; criterion 3 demands rough line-of-sight, saying that the path in G may never deviate more than one cell from the line in $\mathbb{R}^2$. Figure 5.3 illustrates the working of this mechanism. If vertex w falls into an agent's sight cone, it perceives $\pi(w)$ and the following:

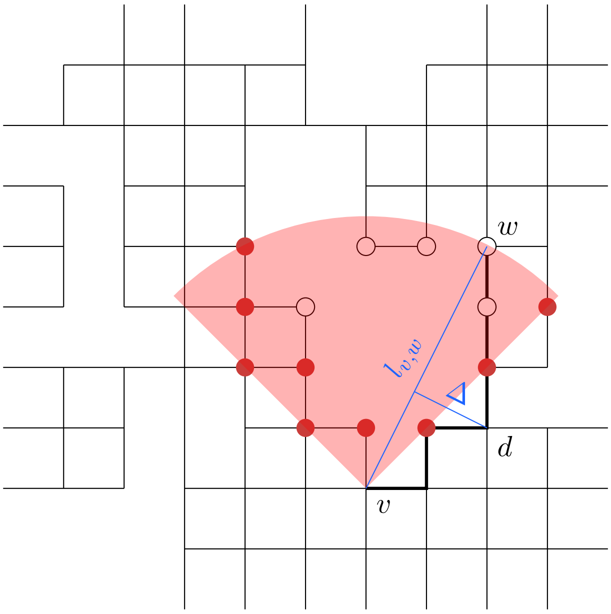


Figure 5.3: Sight cone of an agent at $\texttt{light}(t) = 2$. The cone with width $\frac{\pi}{4}$ signifies that agent's range of vision. Red vertices in it are perceived; the hollow black ones are not because they are blocked by holes in the world. The line $l_{v,w}$ illustrates why the vertex w is not visible from v: the shortest path from v to w runs through d, but the distance Δ between d and the closest point along $l_{v,w}$ is larger than $\frac{\sqrt{2}}{2}$.

1. the agent or Wumpus on w (excluding the agent's internal state),
2. the plant, it present,
3. the pit, if present, and
4. the amount of fruit, meat, and gold.

The breeze and the stench, being non-visual, are not thus perceived. As we can see from criterion 2 in Definition 30 and the formulae for breeze and stench in Definition 29, sight reaches farther, but is directed. The non-visual cues can tell an agent that it is in danger, but not from which direction that danger comes. If that agent consequently fails to look around, it may be attacked or wander into a pit.

5.2.2 Cognition

Our goal is the design of a reasonably effective type of agent which will be able to navigate $\mathcal{W}_{\text{jun}}$-type worlds. *Effectiveness*, in this context, simply means *survival*. There is no explicit performance measure; certain agents will survive, while others will not.

Relevant aspects. We have already seen what sort of data an agent must process if it is to perform well. It must first know or learn the geography of the world, of which it is a priori unaware. It must also be able to seek out resources in the form of plant and gold; it must be able to deal with the threat posed by Wumpuses, either by avoiding or defeating them. Most importantly, it must be able to interact with other agents in ways which avoid adverse behaviour towards the agent itself, and it must find ways to solicit beneficial behaviour from them.

In order to achieve this, three things are indispensable: (1) memory, (2) utility maximisation. If we don't impose a memory limit, it is quite easy to store everything that happens to an agent. In essence, such memories will be fragments of past states of the external which can be used to make decisions. Utility maximisation is the far more complex task: the agent must either perform individual fact synthesis or inherit certain predilections from its parents and must therewith exhibit useful behaviour. The fact synthesis can be done in a number of ways — machine learning, reasoning, heuristic —, but we must remember that knowledge, by itself, does not determine behaviour. In addition, the agent must possess a decision-making component which uses gained knowledge in whatever way it sees fit. Knowledge thus *allows* efficient decisions to be made, but fundamentally, an agent is free to disregard any fact it wants.

Design goals and dynamism. As with the world, the cognitive structure of agents is a compromise between intricacy and simplicity. Ideally, we would make every aspect of an agent's thinking dynamic and malleable under evolution, but this would necessitate a prohibitively high implementation effort. Instead, based on the description of *filters* in Section 3.2, we make the following compromise: the *evocation* of an emotion will be dynamic and different from agent to agent; the effects of emotions, however, will always be the same. As an example, different agents might become angry in different situations and to different degrees, but the behavioural consequences that follow from the emotion of anger will always be the same.

Cognitive components. Based on the considerations outlines in earlier sections, we propose that agents be made out of the following six components:

- *Pre-social behaviour control (PSBC):* This controls aspects of an agents which, in principle, can work without other agents: fear, happiness, anger. These emotions are evoked in social situations, but in principle, they would be useful in a world without any other agents present.

- *Social judgement system (SJS):* Analogous to the PSBC, the SJS controls an agent's appraisal of other agents and thereby influences its decision-making.

- *Belief generation (BG):* Belief generation describes, in essence, the imagination of an agent. It allows reasoning and the internal simulation of parts of the world.

- *Attention-control* (*AC*)*:* Attention-control is the recognition of certain real or imaginary percepts as *important*, leading to the allocation of cognitive resources to them.

- *Decision-making* (*DM*)*:* This component captures the executive function of an agent and includes both internal decision-making (IDM) — *what to think* — and external decision-making (EDM) — *what to do.*

- *Memory:* Memory is a log of imagined and real events that happened to an agent. This log is utilised chiefly by the BG with the goal of providing world data.

As a side remark, these components make no claim to encompass the kind of intelligence humans have. In particular, there are no aesthetics, pure abstract reasoning, purely self-centered emotions like grief or remorse, etc. Providing such mechanisms is, however, not the goal her; we merely wish to make the agents complex enough to successfully navigate the world. For this purpose, a simple, social, and animalistic sort of intelligence suffices, one that, in complexity, is actually below even that of wolves and dogs.

Pre-social behaviour control. The PSBC is responsible for evoking the kinds of emotions that non-social animals have, in some form. Here "pre-social" does not refer to the current use of this system, but to its evolutionary history: past animals were able to experience anger and fear, or something analogous to anger and fear, before they developed social lives. The fight-or-flight instinct, and deciding when to engage in activity and when to abstain from it are necessary for survival even in solitary animals. A social system, of course, does impact these emotions, but a social system is not necessary for them to be there. We categorise the experienced emotions according to approach/avoidance and positivity/negativity, based on the work of Davidson and Irwin [DI99]. The four combinations are:

1. Anger, which is approach-related and negative. Anger causes attack-actions against Wumpuses and other agents, and gesture-actions with parameters the agent deems to be aggressive.

2. Fear, which is avoidance-related and negative. Fear causes flight and gesture-actions which the agent deems submissive.

3. Enthusiasm, which is approach-related and positive. Enthusiasm has a wide range of effects: gesture-actions with positive contents, fatigue-inducing activity, and the gathering and sharing of resources with other agents.

4. Contentment, which is avoidance-related and positive. Contentment is concerned primarily with the conservation of resources. Its chief effect is thus the is the cessation of action.

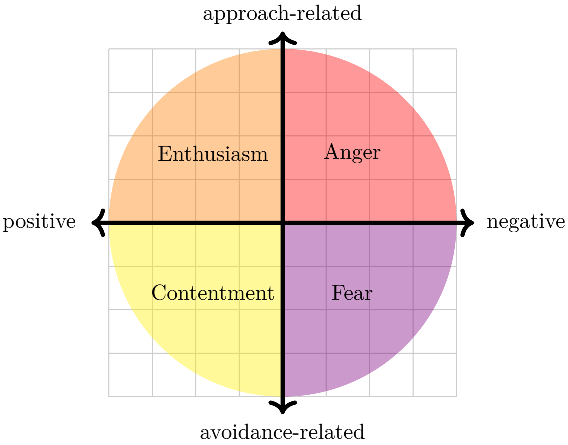


Figure 5.4: Emotions evoked by the PSBC.The left half contains the positive emotions of enthusiasm and contentment, whereas the right contains the negative emotions of anger and fear. Enthusiasm and anger are both approach-related, causing action, whereas contentment and fear are approach-related, causing flight or abstinence from action.

Figure 5.4 illustrates these four emotions. Each of them can be evoked with a *valence* within the interval $[-1, 1]$. Higher-valence emotions exert a greater pressure on decision-making and attention control. The figure, with its two axes, should not mislead us into thinking that emotions are just vectors in $\mathbb{R}^2$. There is, for example, weak/intense enthusiasm and there is weak/intense contentment, but there is no emotion halfway between contentment and enthusiasm. It *is* possible that a stimulus should activate two emotions at once, but those will actually be two emotions, not one "hybrid" emotion.

In terms of implementation, this is realised via the system we saw in Figure 4.7, Section 4.1: each of the four emotions has a *selector* that reads percepts and the *hormone storage*, using them to decide whether and how intensely to activate an emotion. Emotions, once active, flow into the *hormone storage* and send messages into the global message space. The scheme is illustrated in Figure 5.5: the filters of each emotions continually check the agent's percepts for relevant data. If a filter is activated, the message is passed the component's interpreter (to determine its urgency), which hands it to the processor. It then puts the message "I feel emotion E with intensity π_E" into the message space. In this, it takes the *hormone storage* into account: experiencing an emotion increases the corresponding hormone level, and, conversely, a high hormone level intensifies the emotion. Formally, the hormone storage is defined thus:

Definition 31 (Hormone storage). *Let $E_1, \ldots, E_n$ be the names of emotions. A* hormone storage *for the emotions $E_1, \ldots, E_n$ is the ADT $\mathtt{H}_n = \langle h_1 :: \mathbb{R}, \ldots, h_n :: \mathbb{R} \rangle$, together with the functions* `receive` $:: \mathtt{H}_n \to \mathbb{N} \to \mathbb{R} \to \mathtt{H}_n$ *and* `tick` $:: \mathtt{H}_n \to \mathtt{H}_n$*, given by*

$$
\begin{aligned}
\texttt{receive}\ h\ e\ \pi &= 2\pi * \log_2(1 - \texttt{get}\ h\ e),\\
\texttt{tick}\ h &= \langle \texttt{get}_1\ h - 2\log(\texttt{get}_1\ h),\\
&\quad \ldots\\
&\quad\ \texttt{get}_n\ h - 2\log(\texttt{get}_n\ h)\rangle.
\end{aligned}
$$

The idea is that hormone level increases and decreases logarithmically: whenever an agent receives a message about an experienced emotion e with intensity π, the corresponding level h_e is increased proportionally to π and the logarithm of the current level. The levels also decay at each time step, returning the agent to a neutral state over time if no stimuli are experienced.

One objection might be that, while an agent can experience conflicting emotions if multiple components are activated, different emotions cannot directly interact with each other. This is true; however, they can interact indirectly, through the message space: if a component C_X reads the message of component C_Y as a percept and, because of that, begins sending negatively-valenced messages, the emotion X is effectively shutting down the emotion Y — even though the process is controlled by C_Y. Of course, we do not claim that this mechanism accurately reflects nature, that being an empirical question, but at the very least, it gives us a way to implement both ambivalence and quick mood changes.

Social judgement system. The social judgement system (SJS) has the task of recognizing other agents as such and guiding friendly and hostile interactions with them. Real social behaviour is very complex and involves not only other agents as individuals, but the group itself. In the minds of tribal animals, the group exists as an entity unto itself, with its own will and mood. Our agents will not implement this group dynamic. Instead, they will appraise each other agent individually, according to three criteria:

- *Sympathy:* This determines how much an agent likes another one. Liked agents will receive friendly gestures, assistance in the form of food and protection from Wumpuses and hostile agents, disliked agents will be denied these benefits, receive hostile gestures and, if the dislike is sufficient, might be attacked.

- *Trust:* The trustworthiness of another agent influences the likelihood of two things: (1) the propensity to give out items in the hope of future reciprocation and (2) the aggressiveness if protection from the agent is present. The reasoning here is that the agent will be emboldened by the presence of trusted allies.

- *Competence:* Competence judges the capabilities of another agent. Competent agents will be respected, incompetent ones will be held in contempt. Similarly to trust, the presence of friendly, competent agents emboldens the agent.

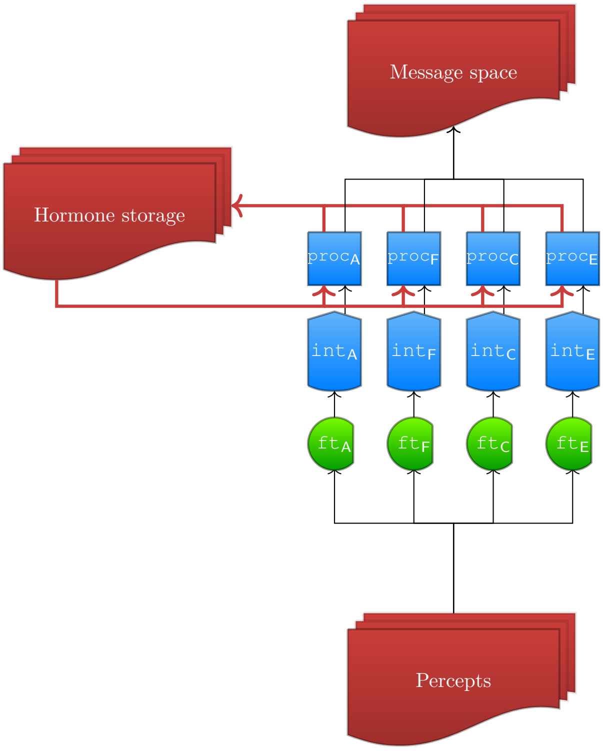


Figure 5.5: The PSBC as a collection of a hormone storage and four emotion selectors. The neural components are *anger* (A), *contentment* (C), *enthusiasm* (E), and *fear* (F).

Sympathy is the primary axis of judgement, since it determines whether others are seen as friends or enemies. Trust and competence are secondary and help an agents ascertain the quality of its allies an enemies. The three criteria are illustrated in Figure 5.6. Figures 5.7 and 5.8 list the different antagonistic and sympathetic judgements.

The evocative mechanism is structurally similar to that of the PSBC, as we saw in Figure 5.5, but with two crucial differences: first, social judgements are always attached to agents; second, the SJS models each of these three categories as a single emotions which can be positive or negative — that is, an agent cannot simultaneously experience trust and distrust for another one, but only a single emotion (trust). We see this system illustrated in Figure 5.9, which shows it to be largely analogous to the PSBC in Figure 5.5.

This system is a quite gross simplification of the real world. In reality, one does not simply possess an emotion called "trust", the value of which can go from -1 to +1, but rather, one possesses different kinds of trust, and trust with respect to different matters. One can, for instance, have a gut feeling that someone is generally unreliable and shady, but one can, through reason, come to the conclusion that this person will keep his word in a certain situation in which punishment would ensue. This does given an assurance of loyalty, but does not change the fundamentally negative appraisal of that person. Similarly, one can have judgements which seem to lie halfway between reason

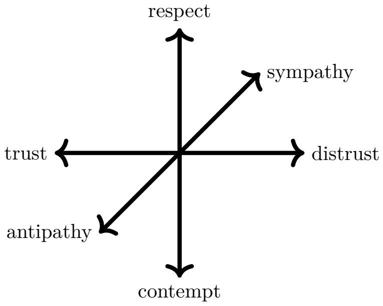


Figure 5.6: Emotions evoked by the SJS. The primary is axis is sympathy/antipathy, since it distinguishes friend from foe. Trust/distrust judges the loyalty/honor of another agent, whereas respect/contempt judges its competence.

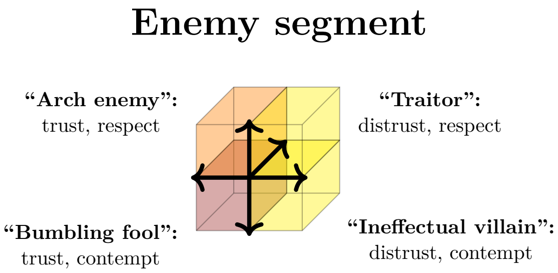


Figure 5.7: The four antipathic judgements. Enemies can be respected or held in contempt, and deemed trustworthy or untrustworthy. Respect for an enemy implies that an agent holds it to be competent. Trust implies that an agent knows its enemy to be basically honourable.

and emotion, and which pertain only to certain situations, such as trusting someone with money, with completing a task on time, or with one's child.

Our agents will not implement the nuances of such concepts directly, but they will not completely neglect them either. As we will see in the sections about memory and the relationship between components, the two affective systems will make use of memory and imagination in order to deliver situational judgements. To stay with our example about trust: if an agent imagines a situation in which another was loyal, or remembers such an event, it will be able to judge that other agent as trustworthy (in that situation.)

Belief generation. World-simulation is probably the most complex identifiable part of human cognition. Our version of it, therefore, will only be a minimalistic reproduction. Instead of constructing a system which is able to extensively utilise learning and construct its own ontologies and ways of thinking from scratch, we will reuse the actual ontology of the Wumpus world and generate beliefs about future world-states with its *semantics function* φ.

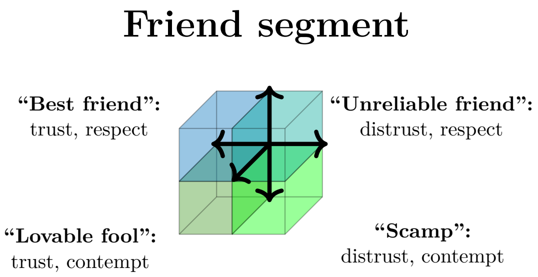


Figure 5.8: The four sympathetic judgements. Friends, like enemies, respected or held in contempt, and deemed trustworthy or untrustworthy. Distrust renders the sympathetic judgement tentative, since the agent cannot be sure of the assistance of an untrustworthy friend. Contempt works similarly, but doubts a friend's ability, rather than loyalty.

The agent generates the immediate future world-state by reconstructing a segment of the present world from the facts stored in its memory. This will result in an internal world that might contains less information than the external, real one, but the internal world will not contradict the external one on any fact. Importantly, a representation of the agent itself — let us call it `I` to distinguish it from the real agent — will be a part of this generated world. The belief generator then reads each action that the decision-maker has proposed (see below), configures `I` to perform it, and calls φ to advance the time by one step. `I` will perform its action, its consequences will be simulated by φ, and `I` will receive corresponding perceptions. The belief generator then reads out `I`'s message space and inserts these messages into the real agent's message space. It has thus utilised the agent's memory and φ to infer the consequences of certain actions to the best of the agent's knowledge.

This simulated future world-state is then read out by the affective components, which emotionally evaluate it, as well as the decision-maker, which either proposes another action for the belief-generation to simulate, finalises the plan if it is satisfied with the predicted, or orders that a certain number of steps be retracted if the situation is deemed unfavourable.

Memory. While real-world memory is complex phenomenon, for expediency's sake, our agents will possess only a simple analogue to it, in the form of a private database of world data which they perceived in the past. These data are of type $\texttt{TV}_{\text{jun}}$, $\texttt{TE}_{\text{jun}}$, which were given in Definition 27. We store them on a per-cell and per-edge basis and update them whenever we perceive them anew. This gives rise to the following definition:

Definition 32. *Let $\langle G, \text{gl}\rangle$ be a $\mathcal{W}_{\text{jun}}$-type world and let A be an agent. The memory database of A has type*

$$\texttt{Memory} = \texttt{Memory}\ (\texttt{Map}\ V(G)\ \texttt{TV}_{\text{jun}})\ (\texttt{Map}\ E(G)\ \texttt{TE}_{\text{jun}})$$

and is accessed through the functions

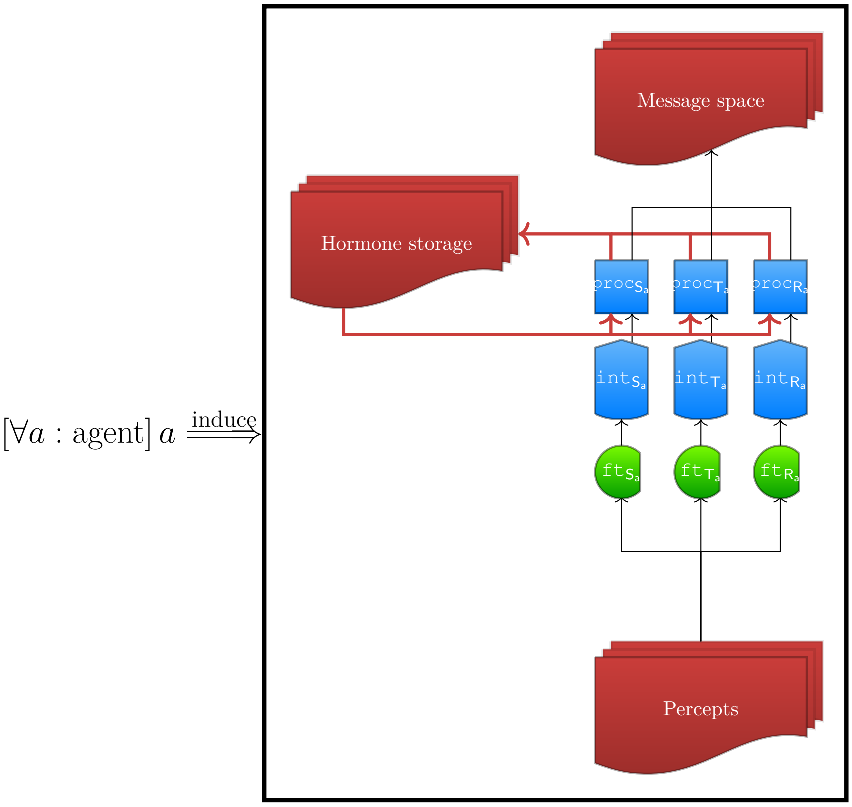


Figure 5.9: The SJS *for one other agent* as a collection of a hormone storage and three emotion selectors. The neural components are *sympathy* (S), *trust* (T), and *respect* (R). Every agent which is encountered has its own SJS instance.

$$\begin{aligned}
\texttt{store} &:: \mathcal{W}_{\text{jun}} \rightarrow \texttt{Memory} \rightarrow \texttt{Memory},\\
\texttt{retrieve}_{\texttt{V}} &:: \texttt{Memory} \rightarrow \texttt{V(G)} \rightarrow \texttt{Maybe TV}_{\text{jun}},\\
\texttt{retrieve}_{\texttt{E}} &:: \texttt{Memory} \rightarrow \texttt{E(G)} \rightarrow \texttt{Maybe TE}_{\text{jun}},\\
\texttt{retreive}_{\texttt{A}} &:: \texttt{Memory} \rightarrow \texttt{String} \rightarrow [\texttt{Action}],
\end{aligned}$$

where `store` *updates the database with the edges and cells of the world which the agent can perceive;* $\texttt{receive}_{\texttt{V}}$ *and* $\texttt{receive}_{\texttt{E}}$ *return the values associated with a given cell or edge, provided that data for the given cell/edge is stored. The function* $\texttt{receive}_{\texttt{A}}$ *takes the name of an agent B as a key and returns the list of actions A has observed that B performs.*

We should note that a number of justified criticisms can be levelled against it. For one, it does not deal with uncertain data that are either old, or were not inaccurately perceived. It only records past states, but not sequences of events. Most direly, it does not provide enough information to contextualise the actions of other agents. Suppose that A observes

B attacking C. A may infer that B is powerful or aggressive, but the list of actions returned by $\texttt{retrieve}_{\texttt{A}}$ are not enough to construct a theory of mind for either B or C. A thus does not know whether B's attack was revenge, opportunism, betrayal, or plain hostility.

Nonetheless, this database is valuable for the agent. The functions $\texttt{retrieve}_{\texttt{V}}$ and $\texttt{retrieve}_{\texttt{E}}$ can provide actionable information about the static aspects of the world such as the location of plants or dangerous paths. Even the information about its changing aspects, such as the location of wumpuses, will be reasonably good, since wumpuses, in the absence of agents, tend to stay in place over time[1].

Attention-control. Attention-control serves as a prioritisation mechanism for the decision-making process. In addition to the emotional evaluation of the whole of the agent's perceptions, we also group incoming messages by cell and evaluate each of those groups separately to determine which locations in the world evoke the strongest emotions. We thereby have a method of prioritising cells and tasks that require immediate attention and to prevent, colloquially speaking, aimless deliberation on unimportant ones.

For our agents, paying attention means to focus on a cell and making it the target of its decision-making. The cell can be another one with say, food or a Wumpus on it, or it can be the agent's own cell, which would be important in case of low health, say.

The AC component then emits messages of type `EmotionOnCell` and is modelled via the functions

$$\begin{aligned}
\texttt{Cell} &= \langle \mathbb{N}, \mathbb{N} \rangle, \\
\texttt{EmotionName} &= \texttt{Anger} + \texttt{Fear} + \texttt{Enthusiasm} + \texttt{Contentment}, \\
\texttt{EmotionOnCell} &= \texttt{EmotionOnCell}\ \texttt{Cell}\ \texttt{EmotionName}\ \mathbb{R}, \\
\\
\texttt{attention} &:: \texttt{s} \to \mathcal{W}_{\text{jun}} \to [\texttt{EmotionOnCell}], \\
\texttt{ac} &:: \texttt{s} \to \mathcal{W}_{\text{jun}} \to \texttt{s},
\end{aligned}$$

where `s` is the internal state of the agent. The function `ac` is just a wrapper around `attention` which puts the latter's message into the agent's message space.

We should note that the AC does not prescribe any specific action in relation to a cell or an emotion. Its role thus merely consists of noticing, so to speak, emotionally important places in the world and to communicate these to the DM which uses the AC's messages to guide its own planning process, described in detail below. The general scheme consists of selecting the globally most strongly felt emotion and then going through the messages of the AC, in descending order according to that most strongly felt emotion, and seeing what actions it might take.

[1] They approximately perform 2-dimensional random walks over time. The expectation $\mathsf{E}(W)$ of a random walk is the null-vector $\langle 0, 0, \ldots, 0 \rangle$. Given that they have a disproportionately high chance of just staying in place, depending on light conditions, their positions are even quite densely clustered around that.

Decision-making. Decision-making is split into two components: external decision-making, which controls the agent's actions, and internal decision-making, which controls the BG and thus drives the planning process. Aside from the difference in target, both are modelled via a function

$$\texttt{choice} :: \texttt{s} \to \mathcal{W}_{\text{jun}} \to \langle \texttt{Action}, \texttt{s} \rangle$$

where `s` is the internal state of the agent. The function `choice` evaluates a world and the previous state of the agent and then gives a new internal state, together with a proposed action from the list in Definition 4 — that is, one of the following: move, rotate, attack, give, gather, butcher, collect, eat, gesture. The function `choice` is wrapped into another function

$$\texttt{dm} :: \texttt{s} \to \mathcal{W}_{\text{jun}} \to \texttt{s}$$

which inserts the messages of `choice` into the agent's message space. If `choice` non-imaginary action, its is marked accordingly. The actions proposed by the internal decision-making component (IDM) are instructions for the BG and, in principle, can go on as long as the agent wishes to deliberate. Those of the external decision-maker are translated into the real world. Once the simulation program receives the return value of an agent's EDM, that agent is done, so to speak: it has performed its action for that tick and is no longer consulted until the next one.

An agent's decision-making begins by observing the intensity of each of its four emotions (anger, fear, enthusiasm, contentment), and choosing the most intensely felt as its *dominant emotion.* The agent then evaluates each cell it perceives separately to determine which evokes its dominant emotion most strongly — as described above, this cell is then designated the *target* of its planning in the current step, and we select one of the actions appropriate to the dominant emotion and the target. Each emotion has associated with it the following set of hard-wired — instinctive, if you will — actions:

- *Anger:* moving/rotating towards the target, sending a hostile gesture, and attacking;
- *Fear:* moving away from the target;
- *Enthusiasm:* moving/rotating towards the target, sending a friendly gesture, giving an item, collecting an item, harvesting a plant;
- *Contentment:* doing nothing, i.e., resting.

If the dominant emotion is sufficiently strong, the chosen action is communicated directly to the EDM. Otherwise, it is marked as hypothetical and remains within the IDM, where its consequences can be evaluated by the belief generator and the whole process can repeat.

Lastly, we note the corner case of no emotion being sufficiently strong to serve as the dominant one. In this case, the agent judges that nothing need be done and forgoes the planning process altogether, delivering a nothing-action instead.

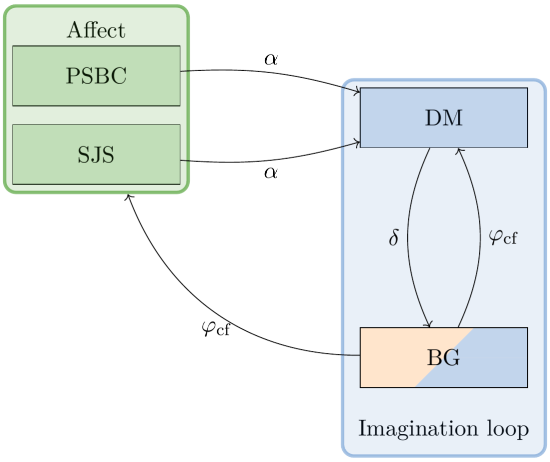


Figure 5.10: Imagination loop, influenced by affect. The edge labels denote the type of signal: α for affective information, δ for control signals, and φ_{cf} for imagined perceptions.

Finalizing and aborting plans. Once an agent begins taking hypothetical actions, it will continue to do so until its dominant emotion drops sufficiently — we conceive of this event as the emotion being satisfied, in which case we begin executing the first step of the plan — or until a conflicting emotion becomes stronger, in which case we retract a random number of steps from the plan and continue evaluating hypothetical actions. This entire process is subject to an upper bound which decreases each time a new hypothetical action is proposed. Should the agent not commit to a plan before this bound reaches, 0, it simply terminates the process and takes whatever action is associated with its dominant emotion.

From the affective subsystem, the BG, and the DM, we can thus put together the imagination simulator loop described in previous chapters. In Figure 5.10, we see the DM issuing commands to the BG, which generates data for the affective systems and the IDM. The affective systems treat this data as if it were coming from the external world and generate affective messages, which are consumed by the IDM and inform its commands to the BG.

Relationship between components. Having defined the agent's components, we now put them together into a functioning whole. The core of the agent's cognition will consist of the interplay between perception and decision-making, with the affective systems and attention control influencing the latter. We see the system sketched in Figure 5.11.

At the very heart of the agent lies its decision-making component, which controls both the agent's actions and its belief generation. The DM and the BG form the *imagination loop* ι which develops plans by exploring the likely consequences of certain actions. In that capacity, the DM evaluate the BG's simulated worlds for desirability and chooses

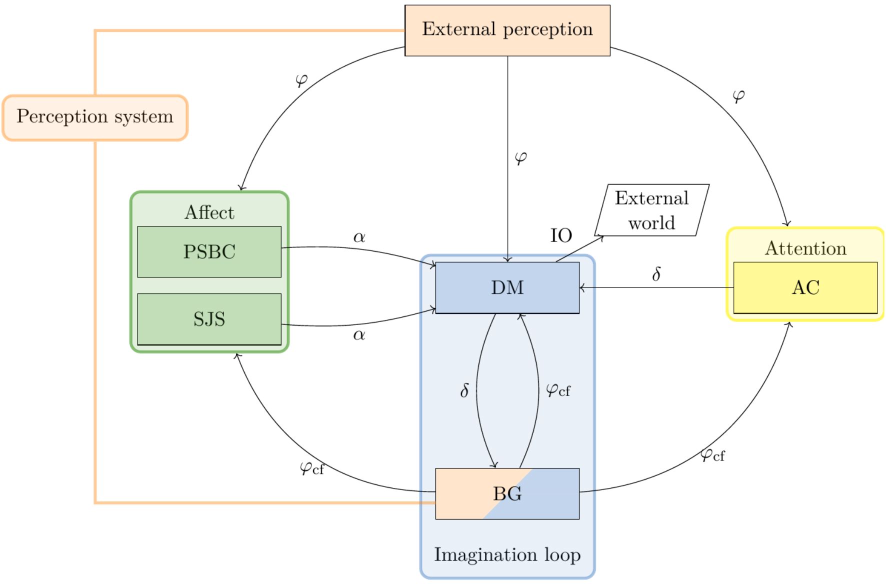


Figure 5.11: High-level view of the cognitive structure of agents, with groups of systems shown in coloured boxes. The PSBC and the SJS comprise the affective group; the DM and BG the imagination loop responsible for planning. The BG, with external perception, makes up the perception system. As in the previous figure, the edge labels show which kind of message the system sends out: α for affective information, φ and φ_{cf} for (imagined) perceptions, and δ for control signals. IO corresponds to real actions in the world.

which imagined steps to take next. These evaluations are influenced by the second group of systems: the affective ones. The PSBC and SJS process perceptions and feed their resultant emotional states into the DM. Through this colouring of its decision-making, agents with different emotional dispositions will act and think differently from each other.

The third part of the system is the attention-control, which also evaluates real and imagined emotions and outputs its data for the DM's usage. It's only purpose is to alert the agent to important or shocking information which demands immediate action. Its alerts cause the DM to cease its current course of action and re-plan based on the piece of information deemed important.

We now have all the pieces we need to create the agent function `agent`:

Definition 33 (Agent function). *Let S be a type. Then an* agent function *with internal data of type S has type*

$$\texttt{agent} \ :: \mathcal{W}_{\text{jun}} \to \texttt{S} \to \langle \texttt{S}, \texttt{Action} \rangle$$

and `agent` *is defined as:*

$$
\begin{aligned}
\texttt{agent}\ w = \ & \texttt{fromJust} \\
& \circ\ \texttt{getActionMessage} \\
& \circ\ \texttt{head} \\
& \circ\ \texttt{dropWhile}\ \texttt{noResult} \\
& \circ\ \texttt{iterate}\ \texttt{loop} \\
& \circ\ \texttt{perception}\ w,
\end{aligned}
$$

where

$$
\begin{aligned}
&\texttt{perception} :: \mathcal{W}_{\text{jun}} \to \texttt{S} \to \texttt{S}, \\
&\texttt{psbc}, \texttt{sjs}, \texttt{ac}, \texttt{dm}, \texttt{bg} :: \texttt{S} \to \texttt{S}, \\
\\
&\texttt{loop} :: \texttt{S} \to \texttt{S}, \\
&\texttt{loop} = \texttt{memory} \circ \texttt{bg} \circ \texttt{dm} \circ \texttt{ac} \circ \texttt{sjs} \circ \texttt{psbc}, \\
\\
&\texttt{getActionMessage} :: \texttt{S} \to \texttt{Maybe Action}, \\
&\texttt{getActionMessage}\ S = \text{the first message in the message-space of } S \\
&\qquad\qquad\qquad\qquad\qquad \text{which is a non-imaginary action}, \\
&\texttt{noResult} :: \texttt{S} \to \texttt{Bool}, \\
&\texttt{noResult} = \texttt{not} \circ \texttt{isJust} \circ \texttt{getActionMessage}, \\
\\
&\texttt{iterate} :: (\texttt{a} \to \texttt{a}) \to \texttt{a} \to [\texttt{a}], \\
&\texttt{iterate}\ f\ x = x : \texttt{iterate}(fx, x), \\
\\
&\texttt{dropWhile} :: (\texttt{a} \to \texttt{Bool}) \to [\texttt{a}] \to [\texttt{a}], \\
&\texttt{dropWhile}\ p\ xs = \begin{cases} h : \texttt{dropWhile}\ p\ t & \text{if } xs = (h : t) \wedge (p\ h = \texttt{True}), \\ xs & \text{otherwise.} \end{cases}
\end{aligned}
$$

That is, `agent` takes the current world and its current internal state, and returns its new internal state, together with the action it wishes to perform. Note that ∘ is function concatenation; the list of functions in `agent` has to be read bottom-to-top.

This agent function can now be plugged into the standard semantics we defined back in Definition 29: the function sem calls every agent with the world and its last internal state and receives a new internal agent state, together with the action the agent has chosen to perform at that time step.

5.3 Implementation Details

We implemented our agent architecture, together with a world simulation that realises the standard semantics of Definition 29, in the functional programming language *Haskell*. Although we will not endeavour to describe it in detail, the prototype, roughly, consists of three major components:

1. the *world simulator*,
2. the *agent intelligence*, and
3. the *world generator*.

Save for minor technical details, the world simulator and the agent intelligence closely follow the specifications in this chapter. The only component which we did not mention before, as it falls outside the scope of this thesis, but shall find brief mention here, is the world generator; we implemented a way of creating and loading worlds in a convenient way, using bitmaps. Each world consists of

- a *topography map*, wherein white pixels indicate accessible cells and black cells indicate inaccessible ones;
- an *entity map*, wherein red pixels indicate Wumpuses, green pixels indicate plants, and blue pixels indicate agents;
- an *item map*, wherein the red value of a pixel indicates the number of meat items on that cell, the green value indicates the number of fruit items, and the blue value indicates the number of gold items; and, lastly,
- an *an agent file*, which stores the personalities of the world's agents in *comma-separated value* (CSV) format.

Agent personalities shall be discussed in greater details in Chapter 6, but suffice it to say that emotional reactions like anger, fear, enthusiasm, contentment, and sympathy can be either weak or strong in each agent. The purpose of such personalities was to evaluate and compare different emotional strategies, and determine which would perform best in a large world.

The source code is available at the URL

`https://github.com/jtapolczai/wumpus.`

The program can be compiled and run with the standard Haskell compiler, *GHC*.[2]

[2] GHC is available as part of the Haskell Platform from `https://www.haskell.org/platform/` or as part of the development tool *Stack*, from `www.haskellstack.org`.

CHAPTER 6

Experimental Evaluation

In this chapter, we present the results of our evaluation, which consists of an assessment of individual behaviour in small test-worlds as well as a population-based evaluation the performance of various populations of agents was measured over time.

6.1 General Considerations

Our goal was to evaluate our agents in specific scenarios, and to compare different kinds of agents to each other. To this end, we introduced parametricity in their emotional reactions. While the interplay of affect, decision-making and belief generation was the same in all agents, as was the schema of their emotional reactions, the strength of these reactions varied — while all agents feared dying, for instance, not all feared it to the same degree. Similarly, while all felt fear in the proximity of a Wumpus, they felt it with different intensities.

We parametrised our agents through five criteria, with two possible values for each:

- anger, with the possible values *strong* and *weak*;
- fear, with the possible values *strong* and *weak*;
- enthusiasm, with the possible values *strong* and *weak*;
- contentment, with the possible values *strong* and *weak*;
- hostility, with the values *hostile* and *friendly*.

Note that each emotion of the PSBC is represented by one criterion, while the emotions of the SJS were rolled into one for the sake of simplicity. For each emotion or *personality* *fragment*, we constructed a graph consisting of approximately 10^4 nodes by hand[1], with output nodes that had configurable significances. These graphs are far too large for explication here[2] and there was, indeed, no special theory behind their construction, but we did use common-sense assumptions which we will illustrate by listing a few output nodes:

- High health reduced fear.
- The presence of a Wumpus with low health increased anger. Closer Wumpuses generated more anger than distant ones.
- Dying significantly increased fear.[3]
- Items lying on the ground increased enthusiasm.
- Low health also increased enthusiasm, so as to induce the enthusiasm-related action of eating,
- Empty cells increased contentment.
- Eating fruit or meat decreased enthusiasm as a way of signalling satisfaction.
- Killing a Wumpus decreased anger.
- Receiving a gift increased sympathy.

Each output node had a variable significance which we varied according to whether we wanted the emotion to be strong or weak, or hostile or friendly, respectively. The concrete values for these significances were, again, the product of intuition. For an agent with strong fear, dying increased the value by 0.8 out of a possible 1, making it almost certain that a “You have died”-message would lead fear to override all other emotions. For weak fear, the value only increased by 0.5 — still very high, but considerably lower, and possible to override if the agent's anger was strong enough. The five criteria induced 32 possibly combinations of values, with each combination representing a possible personality for an agent.

For convenience, we will use a shorthand notation as defined next to specify an agent's personality.

Definition 34. *Let A be an agent and let $P_A = \langle \mathtt{X}_a, \mathtt{X}_f, \mathtt{X}_e, \mathtt{X}_c, \mathtt{X}_h \rangle$ with $\mathtt{X}_a, \mathtt{X}_f, \mathtt{X}_e, \mathtt{X}_c \in \{\mathtt{W}, \mathtt{S}\}$ and $\mathtt{X}_h \in \{\mathtt{H}, \mathtt{F}\}$. Then, P_A is a specification for a A's personality, with $\mathtt{X}_a$, $\mathtt{X}_f$, $\mathtt{X}_e$, $\mathtt{X}_c$, and $\mathtt{X}_h$ representing the agent's personality fragments for anger, fear, enthusiasm, and hostility, respectively. The values* W *and* S *stand for a* weak *or* strong *fragment, while* H *and* F *stand for a* hostile *or* friendly *value for hostility.*

[1] Of course, we generated large parts of these graphs through templating as well, as there had many repeating structures.

[2] The code of the implementation is accessible at `https://github.com/jtapolczai/wumpus`.

[3] While such a node might seem useless, agents can receive “You have died”-messages from their belief generators.

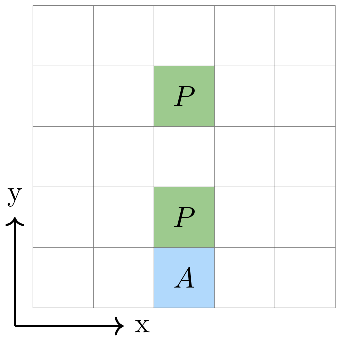


Figure 6.1: The world of Scenario 1. In the south, we have a single agent labelled A. The green squares labelled P indicate plants.

```
A at (2,0) moved north to (2,1).
A at (2,1) harvested a plant.
A at (2,1) ate fruit.
A at (2,1) did nothing.
A at (2,1) did nothing.
```

Listing 6.1: Actions in Scenario 1.

6.2 Evaluation of Individual Behaviour

To evaluate the general fitness of our architecture, we placed one or two agents in a number of scenarios with a clear expected outcome. Unless otherwise noted, we expected all personalities to perform in the same way.

Scenario 1: Harvesting a single plant. The first scenario was a 5x5 world with two ripe plants and an agent with a health of 0.6 that had food in its inventory. The agent's health was low, but eating one plant would restore it to a level of 1.1. We expected the agent to move to the closest plant, harvest it, eat the fruit which is now in its inventory, and then rest. We can see the world in Figure 6.1 and the actions of the agent in Listing 6.1. Coordinates are given in the format (x, y). As we can see, the agent performed according to our expectations.

Scenario 2: Harvesting all plants. This scenario was identical to the previous one, save for the agent's health, which was set at 0.1. Very close to dying, we expected the agent to harvest both plants and eat both fruits in succession to increase its health above 1. We see the actions of the agent in Listing 6.2. The agent again performed according to our expectations.

```
A at (2,0) moved north to (2,1).
A at (2,1) harvested a plant.
A at (2,1) ate fruit.
A at (2,1) moved north to (2,2).
A at (2,2) moved north to (2,3).
A at (2,3) harvested a plant.
A at (2,3) ate fruit.
A at (2,3) did nothing.
A at (2,3) did nothing.
```

Listing 6.2: Actions in Scenario 2.

```
A at (2,2) did nothing.
A at (2,2) did nothing.
A at (2,2) did nothing.
A at (2,2) did nothing.
A at (2,2) did nothing.
```

Listing 6.3: Actions in Scenario 3.

Scenario 3: Resting. This scenario is the simplest: we place an agent in good health into an empty 5x5 world. We expect it to simply stay put for a short while, avoiding exertion, or to look around to ascertain its surroundings. Listing 6.3 shows the results. Since we had not built in any notion of curiosity into our agents, they simply remained in place, which we deemed acceptable behaviour.

Scenario 4: Killing a wounded Wumpus. Here we put a Wumpus with 0.1 health at some distance from a healthy agent, with the expectation that any agent would attack such a weak enemy, regardless of personality. The world is shown in Figure 6.2 and the agent's actions are shown in Listing 6.4. We can see that the agent indeed approached and killed the wounded Wumpus. However, enthusiasm then replaced anger as its dominant emotion and the agent collected the meat from the corpse. This was both surprising and beneficial behaviour, as there was nothing else to do and an agent can always use additional food in its inventory.

Scenario 5: Picking up items. An agent with low health was placed into a world in which two cells had items: one had two pieces of meat and another, most distant one, had three pieces of fruit. We expected the agent to collect the items and eat as many as necessary to restore its health to at least 1. Figure 6.3 shows the world and Listing 6.5 the results. We can see that the agent was killed in both cases, though for different reasons: in the first, the agent killed the closer Wumpus, but this reduced its health to

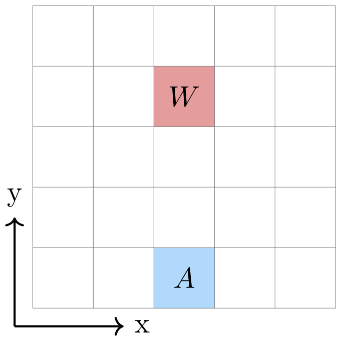


Figure 6.2: The world of Scenario 4. In the south, we have a single agent labelled A. To its north, we have wounded Wumpus labelled W.

```
W at (2,3) moved south to (2,2).
A at (2,0) moved north to (2,1).
A at (2,1) attacked w1 to its north.
A at (2,1) moved north to (2,2).
A at (2,2) picked up meat.
A at (2,2) ate meat.
A at (2,2) did nothing.
A at (2,2) did nothing.
```

Listing 6.4: Actions in Scenario 4.

0.4. When the second Wumpus came close, the agent began moving backwards until it reached the edge of the world, whereupon the Wumpus killed it. In the second case, the agent was controlled by anger, not by fear. It had picked up the meat from the first Wumpus as the second approached it but, instead of eating, the agent immediately and suicidally attacked the Wumpus. In both cases, the agent's behaviour was suboptimal, as it could have eaten the meat from the first Wumpus and thereby regain enough health to survive the second's attack — however, if we conceive of the first agent running away in fear and the second attacking in blind anger, we can at least make intuitive sense of their actions.

Scenario 6: Fight or flight. Similarly to Scenario 4, we placed an agent together with two Wumpuses, each with a health of 0.6. An agent was able to win the fight against both by first killing one, eating its meat to restore its health, and then killing the second. Here, we wanted to test the interplay between anger and fear — accordingly, we expected agents with different personalities to react in different ways. Figure 6.4 shows the world; Listing 6.6 shows the actions of an agent with the personality $\langle \texttt{W}, \texttt{W}, \texttt{W}, \texttt{W}, \texttt{F} \rangle$, and Listing 6.7 the actions of an agent with the personality $\langle \texttt{S}, \texttt{W}, \texttt{S}, \texttt{W}, \texttt{F} \rangle$. Our expectation was that the first agent would flee, whereas the second would try to fight.

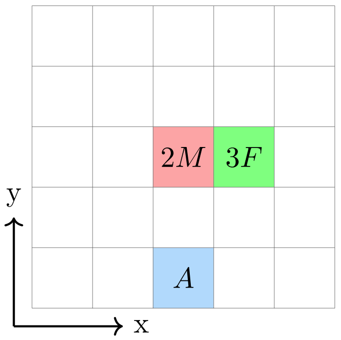


Figure 6.3: The world of Scenario 5. In the south, we have a single agent labelled A. To its north we have a cell with two pieces of meat, labelled $2M$, and another cell with 3 pieces of fruit, labelled $3F$.

```
A at (2,0) moved north to (2,1).
A at (2,1) moved north to (2,2).
A at (2,2) picked up meat.
A at (2,2) ate meat.
A at (2,2) ate meat.
A at (2,2) did nothing.
A at (2,2) did nothing.
```

Listing 6.5: Actions in Scenario 5.

Scenario 7: Searching for food. We placed an agent with 0.5 health into a world with one pile of fruit outside of its sight cone. This scenario was interesting because, while the goal was clear, there was no obvious series of actions which the agent was supposed to take, nor was a goal immediately obvious. We expected it to search its surroundings in search of a source of food. The world is shown in Figure 6.5 and the results in Listing 6.8. It should be noted that the agent's actions are largely random, as agents may randomly select a possible action if several equally beneficial are possible. As we can see, the results were somewhat unsatisfying. While the agent did eventually find the food, it spent many round aimlessly wandering and turning around. Due to the lack of any dedicated algorithm responsible for exploration and the lack of stimuli to guide it, the agent essentially moved around randomly until it eventually stumbled upon the cell with meat on it.

Scenario 8: Giving a gift to a friend. Here we placed two agents, first with the personalities $\langle \texttt{W}, \texttt{W}, \texttt{W}, \texttt{W}, \texttt{F} \rangle$ and then $\langle \texttt{W}, \texttt{W}, \texttt{S}, \texttt{W}, \texttt{F} \rangle$ in a 5x5 world and added fruit to the inventory of A. Although their disposition towards each other was neutral, we expected them to interact in a friendly way, by giving items and sending sympathy-related gestures. Figure 6.6 shows the world and Listing 6.9 shows somewhat interesting results. In the case

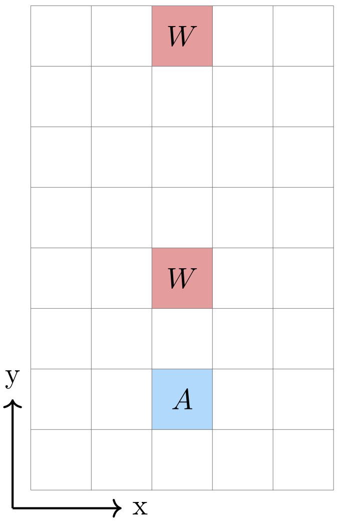


Figure 6.4: The world of Scenario 6. We have an agent labelled A and two Wumpuses labelled $W1$ and $W2$.

```
A at (2,0) moved north to (2,1).
W1 at (2,3) moved south to (2,2).
W2 at (2,7) moved south to (2,6).
A at (2,1) attacked W1 to its north.
W2 at (2,6) moved south to (2,5).
A at (2,1) moved north to (2,2).
W2 at (2,5) moved south to (2,4).
A at (2,2) moved south to (2,1).
W2 at (2,4) moved south to (2,3).
A at (2,1) moved south to (2,0).
W2 at (2,3) moved south to (2,2).
A at (2,0) did nothing.
W2 at (2,2) moved south to (2,1).
A at (2,0) did nothing.
W2 at (2,1) attacked A to its south.
```

Listing 6.6: Actions of an agent with the personality ⟨`W`, `W`, `W`, `W`, `F`⟩ in Scenario 6.

of weak enthusiasm, instead of approaching B, A just ate the food and then remained in place, content. B, on the other hand, approached A and started sending the gesture *love*, which we hard-coded as the friendly one. A's contentment overwhelmed its enthusiasm once its health was sufficiently high. Though this behaviour was unexpected and we could have changed it by adjusting emotional filters, we did deem A selfishness in its interaction with what was essentially a stranger efficient. In the second case, in which

```
    A at (2,0) moved north to (2,1).
    W1 at (2,3) moved south to (2,2).
    W2 at (2,7) moved south to (2,6).
    A at (2,1) attacked W2 to its north.
    W2 at (2,6) moved south to (2,5).
    A at (2,1) moved north to (2,2).
    W2 at (2,5) moved south to (2,4).
    A at (2,2) picked up meat.
    W2 at (2,4) moved south to (2,3).
    A at (2,2) attacked W2 to its north.
```

Listing 6.7: Actions of an agent with the personality $\langle \mathtt{S}, \mathtt{W}, \mathtt{S}, \mathtt{W}, \mathtt{F} \rangle$ in Scenario 6.

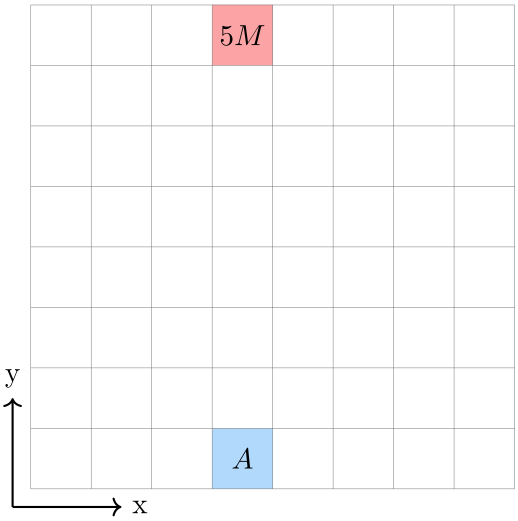


Figure 6.5: The world of Scenario 7. We have an agent labelled A and a cell with five pieces of meat labelled $5M$.

both agents had strong enthusiasm, A was more charitable. As we seen in Listing 6.10, A chose to share after eating one item of meat.

Results

Our agents performed the basic tasks set to them reasonably well. Agents managed to acquire food if they knew the location of a food source, kill enemies if it seemed easy to do so, and have positive interactions with friendly agents. As we saw in Scenario 6, their personalities were also able to meaningfully influence their behaviour. Somewhat disappointing was the performance in Scenario 7, where it became clear that the agents would have benefited from a drive to systematically explore their surroundings.

```
A at (3,0) moved north to (3,1).
A at (3,1) turned west.
A at (3,1) moved west to (2,1).
A at (2,1) moved west to (1,1).
A at (1,1) moved west to (0,1).
A at (0,1) turned south.
A at (0,1) turned north.
A at (0,1) turned south.
A at (0,1) turned north.
A at (0,1) turned south.
A at (0,1) turned north.
A at (0,1) moved east to (1,1).
A at (1,1) moved east to (2,1).
A at (2,1) moved north to (2,2).
A at (2,2) moved north to (2,3).
A at (2,3) moved north to (2,4).
A at (2,4) moved north to (2,5).
A at (2,5) moved east to (3,5).
A at (3,5) moved north to (3,6).
A at (3,6) moved north to (3,7).
A at (3,7) picked up meat.
A at (3,7) ate meat.
A at (3,7) did nothing.
```

Listing 6.8: Actions in Scenario 7.

6.3 Population-Based Evaluation

For the second part of our evaluation, we created a 30x30 world with 30 agents. We randomly distributed these, along with $30/3 = 10$ Wumpuses uniformly and randomly over all cells. We also placed plants and gold uniformly and randomly in the world such that, for all cells (x, y) in the world,

$$\begin{aligned} P(\text{A plant is added to cell } (x, y)) &= 0.05 \\ P(\text{1 gold is added to cell } (x, y)) &= 0.01 \\ P(\text{A pit is added to cell } (x, y)) &= 0.01 \end{aligned}$$

The aim of these parameters was the creation of a world in which food was somewhat readily available, gold was a rare item, and Wumpuses presented a threat, but not such an overwhelming one that they could kill all the agents. Figure 6.7 depicts the world created in the manner.

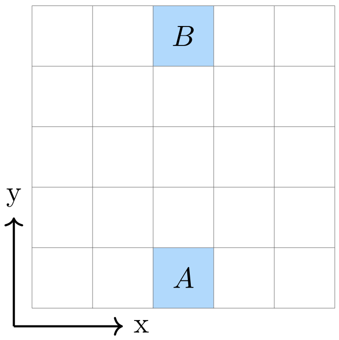


Figure 6.6: The world of Scenario 8. We have two agents labelled A and B. A has three pieces of fruit in its inventory.

```
A at (2,0) ate fruit.
B at (2,4) moved south to (2,3).
A at (2,0) did nothing.
B at (2,3) moved south to (2,2).
A at (2,0) did nothing.
B at (2,2) moved south to (2,1).
A at (2,0) did nothing.
B at (2,1) gestured 'love' to A to its south.
A at (2,0) did nothing.
B at (2,1) gestured 'love' to A to its south.
```

Listing 6.9: Actions in Scenario 8 when both agents had the personality $\langle \texttt{W}, \texttt{W}, \texttt{W}, \texttt{W}, \texttt{F} \rangle$.

We then simulated this world for 32 times, giving our population a different one of the 32 possible personalities each time. The simulations ran for 50 rounds or until at least half of the agents died — whichever came sooner. During the simulation, we recorded the survival rate of each population and that of the Wumpuses, along with a number of other metrics. The results are listed in Tables 6.1-6.16.

Results

We ran the trial with each personality and captured the following data: the number of

- harvests,
- attacks,
- gifts given (split by gift type),
- gestures sent,

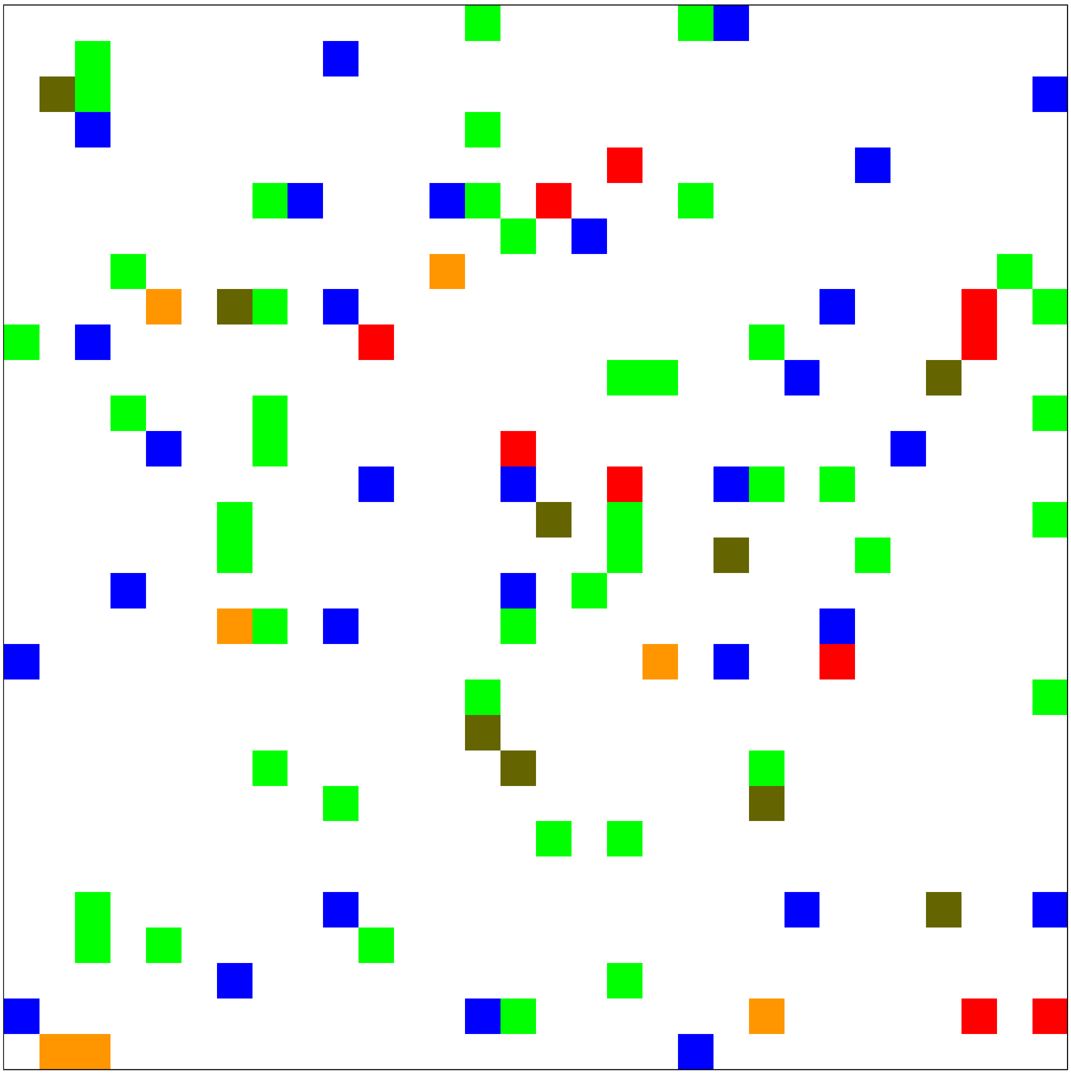


Figure 6.7: The world of the quantitative evaluation, consisting of 30x30 cells. Green pixels represent plants, ochre ones pits, red ones Wumpuses and blue ones agents, where the value of the blue channel is the agent's ID. Orange pixels represent gold. All entities and items were distributed randomly.

```
A at (2,0) moved north to (2,1).
B at (2,4) moved south to (2,3).
A at (2,1) moved north to (2,2).
B at (2,3) gestured 'love' to 1 to its south.
A at (2,2) gave meat to 2 to its north.
B at (2,3) gestured 'love' to 1 to its south.
A at (2,2) gestured 'love' to 2 to its north.
B at (2,3) gave meat to 1 to its south.
A at (2,2) gestured 'love' to 2 to its north.
B at (2,3) gestured 'love' to 1 to its south.
```

Listing 6.10: Actions in Scenario 8 when both agents had the personality $\langle \mathtt{W}, \mathtt{W}, \mathtt{S}, \mathtt{W}, \mathtt{F} \rangle$.

- items eaten,
- surviving Wumpuses, and
- surviving agents.

In addition to recording the raw numbers, we also calculated their averages by personality fragment to see whether there was a clear relation between certain personality types and success.

Number of harvests. We see the number of harvests by personality type in Table 6.1, which shows a rather large difference between the personalities. Table 6.3 shows the average number of harvests by personality fragment. As expected, strong enthusiasm and weak contentment led to the largest increase in harvest frequency, with fear, anger and hostility having negligible effects.

Number of attacks. Table 6.2 shows the number of attacks and Table 6.4 the averages. All personality fragments seemed to play a role in the agents' aggression: anger predictably increased it and fear lowered it, but strong enthusiasm and contentment lowered the number as well. Surprisingly, hostile agents actually performed slightly less attacks, which might be explained with the fact that they spent more time sending hostile gestures than friendly, as we shall see below.

Gifts given. Tables 6.5-6.7 show the number of gifts given and Tables 6.8-6.10 show the averages. We can see immediately that meat was the most popular gift, followed by gold and fruit, which almost no agents gave. Most interesting is the case of meat: we see quite starkly that strong anger, weak fear, high enthusiasm and low contentment resulted in a population of apex predators that spent much of their time killing Wumpuses (and each other) and built up inventories that they shared with the survivors. Gold, too,

Personality	*Harvests*
⟨S, W, W, S, F⟩	8
⟨S, W, S, S, H⟩	9
⟨S, W, W, S, H⟩	9
⟨S, W, W, W, H⟩	11
⟨S, S, W, S, H⟩	11
⟨S, W, S, S, F⟩	11
⟨W, W, S, S, F⟩	11
⟨W, W, W, S, H⟩	11
⟨S, S, W, S, F⟩	12
⟨W, W, S, S, H⟩	12
⟨S, S, S, S, H⟩	13
⟨S, S, W, W, F⟩	13
⟨W, S, S, S, H⟩	13
⟨W, W, W, W, H⟩	13
⟨S, S, W, W, H⟩	14
⟨S, W, W, W, F⟩	14
⟨W, W, W, S, F⟩	14
⟨W, W, W, W, F⟩	14
⟨S, S, S, S, F⟩	15
⟨W, S, W, S, F⟩	15
⟨W, S, W, S, H⟩	15
⟨W, S, W, W, F⟩	15
⟨W, S, W, W, H⟩	15
⟨W, S, S, S, F⟩	16
⟨W, W, S, W, F⟩	17
⟨S, W, S, W, H⟩	18
⟨W, S, S, W, F⟩	18
⟨S, W, S, W, F⟩	19
⟨W, W, S, W, H⟩	19
⟨S, S, S, W, H⟩	20
⟨S, S, S, W, F⟩	21
⟨W, S, S, W, H⟩	24

Table 6.1: Number of plant harvests after 50 rounds.

Personality	*Attacks*
⟨S, S, S, W, H⟩	1
⟨W, S, S, W, H⟩	2
⟨W, W, S, S, F⟩	2
⟨W, W, S, S, H⟩	2
⟨S, S, S, S, F⟩	3
⟨W, S, S, S, H⟩	3
⟨W, S, W, S, F⟩	3
⟨W, S, W, S, H⟩	3
⟨S, S, S, S, H⟩	4
⟨S, S, W, S, H⟩	4
⟨W, S, S, S, F⟩	4
⟨W, S, W, W, F⟩	4
⟨S, S, S, W, F⟩	5
⟨W, S, W, W, H⟩	5
⟨W, W, W, S, F⟩	5
⟨W, W, W, S, H⟩	5
⟨S, S, W, W, H⟩	6
⟨S, W, S, S, H⟩	6
⟨S, W, S, W, H⟩	6
⟨W, S, S, W, F⟩	6
⟨W, W, S, W, H⟩	6
⟨W, W, W, W, F⟩	6
⟨W, W, W, W, H⟩	6
⟨S, S, W, S, F⟩	7
⟨S, W, S, S, F⟩	8
⟨S, W, S, W, F⟩	8
⟨S, W, W, S, F⟩	8
⟨S, S, W, W, F⟩	9
⟨W, W, S, W, F⟩	9
⟨S, W, W, S, H⟩	10
⟨S, W, W, W, F⟩	11
⟨S, W, W, W, H⟩	12

Table 6.2: Number of attacks after 50 rounds.

Personality fragment	*weak/hostile*	*strong/friendly*
Anger	15.125	13.625
Fear	13.125	15.625
Enthusiasm	12.75	16
Contentment	16.563	12.18
Hostility	14.188	14.563

Table 6.3: Average number of plant harvests, by personality fragment.

Personality fragment	*weak/hostile*	*friendly/strong*
Anger	4.438	6.75
Fear	6.875	4.313
Enthusiasm	6.5	4.688
Contentment	6.375	4.813
Hostility	5.063	6.125

Table 6.4: Average number of attacks, by personality fragment.

is of note: as its acquisition did not require killing, high contentment and friendliness significantly increased the chance that it would be given as a gift. The unpopularity of fruit as gift is somewhat contrary to our expectations and shows that agents did not guard plants as reliable sources of food and that they did not infer that the plants would regularly regrow.

Gestures. Table 6.11 shows the number of gestures sent and Table 6.17 shows the averages. We can clearly see the dominant role of strong enthusiasm, followed distantly by weak contentment. This is very much according to our expectations, as gestures are primarily enthusiasm-associated. Anger, fear, and hostility only had relatively minor effects on the number of social interactions.

Items eaten. We see the results and the averages in Table 6.12 and Table 6.18, respectively. Clearly, the two most significant factors here are strong enthusiasm and weak contentment. As eating is enthusiasm-related, it makes sense that agents with strong enthusiasm would eat more, as would agents with weak contentment, since food has to be acquired through action.

Surviving Wumpuses. We see the absolute number of surviving Wumpuses and the averages in Tables 6.14 and 6.16. Similarly to the number of meat gifts given, we see that strong anger, weak fear and low contentment all led to more Wumpuses being killed. Enthusiasm and hostility somewhat increased the Wumpuses' chances of survival — here we might hypothesise that agents with strong enthusiasm spent more time gathering fruit and engaging in interactions with other agents, and that hostile agents spent more time attacking each other than they spent attacking Wumpuses.

Personality	*Gifts (fruit)*
⟨S, W, W, W, H⟩	0
⟨S, S, S, S, F⟩	0
⟨S, S, S, S, H⟩	0
⟨S, S, S, W, F⟩	0
⟨S, S, W, S, F⟩	0
⟨S, S, W, S, H⟩	0
⟨S, S, W, W, F⟩	0
⟨S, S, W, W, H⟩	0
⟨S, W, S, S, F⟩	0
⟨S, W, S, S, H⟩	0
⟨S, W, S, W, F⟩	0
⟨S, W, S, W, H⟩	0
⟨S, W, W, S, F⟩	0
⟨S, W, W, S, H⟩	0
⟨S, W, W, W, F⟩	0
⟨W, S, S, S, H⟩	0
⟨W, S, S, W, F⟩	0
⟨W, S, W, S, F⟩	0
⟨W, S, W, S, H⟩	0
⟨W, S, W, W, F⟩	0
⟨W, W, S, S, F⟩	0
⟨W, W, S, S, H⟩	0
⟨W, W, S, W, F⟩	0
⟨W, W, S, W, H⟩	0
⟨W, W, W, S, F⟩	0
⟨W, W, W, S, H⟩	0
⟨W, W, W, W, F⟩	0
⟨W, W, W, W, H⟩	0
⟨S, S, S, W, H⟩	1
⟨W, S, S, W, H⟩	1
⟨W, S, W, W, H⟩	1
⟨W, S, S, S, F⟩	3

Table 6.5: Number of fruit items given as gifts after 50 rounds.

Personality	*Gifts (meat)*
⟨S, W, W, W, H⟩	0
⟨S, S, S, S, F⟩	0
⟨S, S, S, S, H⟩	0
⟨S, S, S, W, H⟩	0
⟨S, S, W, S, F⟩	0
⟨S, S, W, S, H⟩	0
⟨S, S, W, W, F⟩	0
⟨S, S, W, W, H⟩	0
⟨S, W, S, S, F⟩	0
⟨S, W, S, S, H⟩	0
⟨S, W, W, S, H⟩	0
⟨W, S, S, S, F⟩	0
⟨W, S, S, S, H⟩	0
⟨W, S, S, W, F⟩	0
⟨W, S, W, S, F⟩	0
⟨W, S, W, S, H⟩	0
⟨W, S, W, W, F⟩	0
⟨W, W, S, S, H⟩	0
⟨W, W, W, S, F⟩	0
⟨W, W, W, W, F⟩	0
⟨W, W, W, W, H⟩	0
⟨S, W, W, S, F⟩	1
⟨S, W, W, W, F⟩	2
⟨W, S, W, W, H⟩	2
⟨W, W, W, S, H⟩	2
⟨W, W, S, W, F⟩	5
⟨W, W, S, S, F⟩	7
⟨S, S, S, W, F⟩	11
⟨W, W, S, W, H⟩	12
⟨W, S, S, W, H⟩	17
⟨S, W, S, W, F⟩	19
⟨S, W, S, W, H⟩	19

Table 6.6: Number of meat items given as gifts after 50 rounds.

Personality	*Gifts (gold)*
⟨S, S, S, S, F⟩	0
⟨S, S, S, S, H⟩	0
⟨S, S, W, S, F⟩	0
⟨S, S, W, S, H⟩	0
⟨S, W, S, S, F⟩	0
⟨S, W, S, S, H⟩	0
⟨S, W, S, W, H⟩	0
⟨S, W, W, S, F⟩	0
⟨S, W, W, S, H⟩	0
⟨W, S, S, S, F⟩	0
⟨W, S, S, S, H⟩	0
⟨W, S, S, W, F⟩	0
⟨W, S, S, W, H⟩	0
⟨W, S, W, S, F⟩	0
⟨W, S, W, S, H⟩	0
⟨W, W, S, W, F⟩	0
⟨W, W, S, W, H⟩	0
⟨W, W, W, S, F⟩	0
⟨W, W, W, S, H⟩	0
⟨S, W, W, W, H⟩	1
⟨S, S, W, W, F⟩	1
⟨S, W, W, W, F⟩	1
⟨W, S, W, W, F⟩	1
⟨W, S, W, W, H⟩	1
⟨W, W, S, S, H⟩	1
⟨W, W, W, W, H⟩	1
⟨S, S, S, W, F⟩	2
⟨S, S, W, W, H⟩	2
⟨W, W, S, S, F⟩	2
⟨W, W, W, W, F⟩	2
⟨S, S, S, W, H⟩	5
⟨S, W, S, W, F⟩	6

Table 6.7: Number of gold items given as gifts after 50 rounds.

Personality fragment	*weak/hostile*	*strong/friendly*
Anger	0.313	0.063
Fear	0.0	0.375
Enthusiasm	0.063	0.313
Contentment	0.188	0.188
Hostility	0.188	0.188

Table 6.8: Average number of fruit gifts given, by personality fragment.

Personality fragment	*weak/hostile*	*friendly/strong*
Anger	2.8125	3.25
Fear	4.1875	1.875
Enthusiasm	0.4375	5.625
Contentment	5.4375	0.625
Hostility	3.25	2.8125

Table 6.9: Average number of meat gifts given, by personality fragment.

Personality fragment	*weak/hostile*	*friendly/strong*
Anger	0.5	1.125
Fear	0.875	0.75
Enthusiasm	0.625	1.0
Contentment	1.438	0.188
Hostility	0.688	0.938

Table 6.10: Average number of gold items given, by personality fragment.

Surviving agents. The absolute and average number of surviving agents are listed in Tables 6.13 and 6.15. While all other metrics were important as well, in the end, survival by whatever means is the measure of success in our evaluation. While all personality types managed to survive reasonable well, we can see significant differences between them. In Table 6.15, we can see that agents with weak anger survived better than those with strong anger, as did agents with strong fear. This does make sense, as fighting was a dangerous tactic, despite the meat with which it provided the victor. High enthusiasm had a moderately beneficial effect on survival, which again makes sense, as it is associated with low-cost, high-reward behaviour like the gathering of food and items and the sending of friendly gestures that might lead to gifts in the future. Contentment had relatively little effect and hostility actually increased survival. The case of hostility is somewhat odd, as we would have expected friendliness to have the same beneficial effect that enthusiasm in general had, but it seems that not being too friendly by perhaps giving away all of one's inventory to others might also be advantageous.

Personality	*Gestures*
⟨W, W, W, S, H⟩	30
⟨S, W, W, S, H⟩	32
⟨S, W, W, S, F⟩	34
⟨W, W, W, S, F⟩	40
⟨W, S, W, S, H⟩	46
⟨S, S, W, S, H⟩	61
⟨W, S, W, S, F⟩	66
⟨S, S, W, S, F⟩	69
⟨S, W, W, W, F⟩	92
⟨S, W, W, W, H⟩	116
⟨S, S, W, W, F⟩	120
⟨W, S, W, W, F⟩	124
⟨W, S, W, W, H⟩	129
⟨W, W, W, W, F⟩	129
⟨W, W, W, W, H⟩	132
⟨S, S, W, W, H⟩	136
⟨S, W, S, W, F⟩	309
⟨S, S, S, S, H⟩	335
⟨W, S, S, W, F⟩	359
⟨W, S, S, W, H⟩	361
⟨W, W, S, W, F⟩	373
⟨S, S, S, S, F⟩	388
⟨S, W, S, S, F⟩	394
⟨S, W, S, S, H⟩	397
⟨W, S, S, S, F⟩	403
⟨S, S, S, W, F⟩	405
⟨S, S, S, W, H⟩	411
⟨W, W, S, W, H⟩	439
⟨S, W, S, W, H⟩	451
⟨W, W, S, S, H⟩	452
⟨W, S, S, S, H⟩	494
⟨W, W, S, S, F⟩	495

Table 6.11: Number of gestures sent after 50 rounds.

Personality	*Meals*
⟨S, W, S, S, H⟩	10
⟨S, W, S, S, F⟩	11
⟨S, W, W, S, F⟩	11
⟨S, W, W, S, H⟩	11
⟨S, S, W, S, H⟩	12
⟨W, W, S, S, F⟩	12
⟨W, W, S, S, H⟩	12
⟨W, W, W, S, H⟩	12
⟨S, W, W, W, H⟩	13
⟨S, S, S, S, H⟩	13
⟨S, S, W, S, F⟩	13
⟨W, S, S, S, H⟩	13
⟨S, S, W, W, H⟩	14
⟨W, S, W, S, H⟩	14
⟨W, S, W, W, H⟩	14
⟨W, W, W, W, H⟩	14
⟨S, S, S, S, F⟩	15
⟨S, S, W, W, F⟩	15
⟨W, S, W, S, F⟩	15
⟨W, S, W, W, F⟩	15
⟨W, W, W, S, F⟩	15
⟨W, W, W, W, F⟩	15
⟨W, S, S, S, F⟩	16
⟨S, W, S, W, F⟩	18
⟨S, W, W, W, F⟩	18
⟨S, S, S, W, H⟩	19
⟨W, S, S, W, F⟩	19
⟨S, S, S, W, F⟩	20
⟨S, W, S, W, H⟩	20
⟨W, W, S, W, F⟩	20
⟨W, W, S, W, H⟩	20
⟨W, S, S, W, H⟩	24

Table 6.12: Number of items eaten after 50 rounds.

Personality	*Agents*
⟨S, W, W, W, H⟩	20
⟨S, W, W, S, H⟩	20
⟨S, S, W, W, F⟩	21
⟨S, W, S, S, H⟩	21
⟨S, W, S, W, F⟩	21
⟨S, W, W, S, F⟩	21
⟨W, S, S, W, F⟩	21
⟨W, W, S, W, F⟩	21
⟨S, S, S, S, H⟩	22
⟨S, S, W, S, F⟩	22
⟨S, S, W, W, H⟩	22
⟨S, W, S, S, F⟩	22
⟨S, W, W, W, F⟩	22
⟨W, W, W, S, H⟩	22
⟨W, W, W, W, H⟩	22
⟨W, W, W, S, F⟩	23
⟨W, W, W, W, F⟩	23
⟨S, S, S, S, F⟩	24
⟨S, S, S, W, F⟩	24
⟨S, S, W, S, H⟩	24
⟨W, S, W, W, H⟩	24
⟨S, W, S, W, H⟩	25
⟨W, S, W, W, F⟩	25
⟨W, W, S, S, F⟩	25
⟨W, W, S, S, H⟩	25
⟨S, S, S, W, H⟩	26
⟨W, S, S, S, F⟩	26
⟨W, S, S, S, H⟩	26
⟨W, S, S, W, H⟩	26
⟨W, S, W, S, F⟩	26
⟨W, W, S, W, H⟩	26
⟨W, S, W, S, H⟩	27

Table 6.13: Number of surviving agents after 50 rounds.

Personality	*Wumpuses*
⟨S, S, W, W, F⟩	3
⟨S, W, W, W, F⟩	3
⟨S, W, W, W, H⟩	4
⟨S, W, S, W, H⟩	4
⟨W, W, S, W, F⟩	4
⟨S, W, W, S, H⟩	5
⟨W, S, W, W, H⟩	5
⟨S, S, S, W, F⟩	6
⟨S, S, S, W, H⟩	6
⟨S, S, W, S, F⟩	6
⟨S, W, S, S, F⟩	6
⟨S, W, S, S, H⟩	6
⟨S, W, S, W, F⟩	6
⟨W, S, S, S, F⟩	6
⟨W, S, W, W, F⟩	6
⟨W, W, S, W, H⟩	6
⟨W, W, W, S, F⟩	6
⟨S, S, W, S, H⟩	7
⟨S, S, W, W, H⟩	7
⟨S, W, W, S, F⟩	7
⟨W, S, S, W, H⟩	7
⟨W, S, W, S, F⟩	7
⟨W, S, W, S, H⟩	7
⟨W, W, S, S, F⟩	7
⟨W, W, W, S, H⟩	7
⟨W, W, W, W, F⟩	7
⟨S, S, S, S, F⟩	8
⟨S, S, S, S, H⟩	8
⟨W, S, S, S, H⟩	8
⟨W, S, S, W, F⟩	8
⟨W, W, S, S, H⟩	8
⟨W, W, W, W, H⟩	8

Table 6.14: Number of surviving Wumpuses after 50 rounds.

Personality fragment	*weak/hostile*	*strong/friendly*
Anger	24.25	22.313
Fear	22.438	24.125
Enthusiasm	22.75	23.8125
Contentment	23.063	23.5
Hostility	23.625	22.938

Table 6.15: Average number of surviving agents, by personality fragment.

Personality fragment	*weak/hostile*	*friendly/strong*
Anger	6.688	5.75
Fear	5.875	6.5625
Enthusiasm	5.9375	6.5
Contentment	5.625	6.813
Hostility	6.438	6.0

Table 6.16: Average number surviving Wumpuses, by personality fragment.

Personality fragment	*weak/hostile*	*strong/friendly*
Anger	254.5	234.375
Fear	244.688	244.188
Enthusiasm	84.75	404.125
Contentment	255.375	233.5
Hostility	251.375	237.5

Table 6.17: Average number of gestures sent, by personality fragment.

Personality fragment	*weak/hostile*	*friendly/strong*
Anger	15.625	14.563
Fear	14.5	15.688
Enthusiasm	13.813	16.38
Contentment	17.375	12.813
Hostility	14.688	15.5

Table 6.18: Average number items eaten, by personality fragment.

CHAPTER 7

Conclusion

In this thesis, we proposed an agent architecture that combined affective reactions with reasoning about future states of the world to achieve efficient behaviour. We began in Chapter 2 by listing related work in the fields of artificial intelligence and biology. Here we also made certain preliminary hypotheses that would undergird our architecture and our implementation. We proposed a model in which brains are semi-independently evolved collections of components which listened in on each other's activity.

From there, we proceeded to formalise these notions in the form of a component model in Chapter 3. Our components had the ability to filter messages relevant to them from a larger message space, to interpret them, and to put back messages of their own. Although components were thus able to communicate, they were not, as such, aware of the existence of other components. In Chapter 4, we used the component model to construct our affective architecture, consisting of emotional components as well as components for reasoning. Agents evaluated their perceptions to generate pre-social emotions like anger and fear, as well as social emotions like sympathy or trust for other agents based on their positive or negative interactions with them. Guided by the agent's emotions, a decision-maker proposed hypothetical actions and a belief-generator generated future states of the world, which were again submitted to emotional evaluation. In this way, the agent constructed and evaluated plans until it was satisfied with the predicted outcome.

Chapter 5 detailed our implementation. We put our agents into a moderately complex world filled with plants, pits, items, and hostile Wumpuses. This world was round-based and in each round, each agent had to choose one of a pre-defined set of actions to perform.

In Chapter 6, we submitted our agents to both an evaluation of individual behaviour, comprising eight simple test worlds with clear goals, as well as a population-based evaluation in which we put 32 different populations into a large, complex world and measured their performance over time. In the test worlds, all agents fulfilled their set tasks and almost all did so identically, showing the basic fitness for purpose of the artificial intelligence we designed.

In the case of the population-based evaluation, we were interested in two things: would the agents manage to survive in a complex world and would we see differences between various personalities. To both questions, the answer was "yes". We saw that all agents survived reasonably well, although there were marked differences in the strategies they chose. Aggressive agents killed many of the Wumpuses in their environment, but doing so was costly to their numbers. More peaceful agents that avoided conflict and spent more time harvesting plants survived better, even if they did not clear their worlds of hostile Wumpuses.

These results show that our agents perform as well in their toy world as a simple animal like a crab or a small fish would in the real one. The agents, however, did not fully meet all expectations: despite their ability to predict the future, they failed to make inferences like the fact that plants regrow in 10 time-steps. Their lack of a theory of mind also meant that, while they were capable of liking other agents, they did not coordinate with them for hunting and for protection.

Future work. In the course of the implementation and evaluation of the proof-of-concept accompanying this thesis, a number of possible improvement arose, which were not explored further but which can form the basis of fruitful future investigation. Specifically:

- *Causality-based world simulation:* Presently, the agents create plans by taking hypothetical actions and simulating the world state as a result of these. As a consequence, the lengths of plans and the number of time-steps required to perform them correspond one-to-one. This schema is functional, but has apparent drawbacks when we compare it to the way in which humans plan actions: If, say, one wanted to go 100 steps in a straight line to get a glass of water, one would not consider each required step individually. Rather, one would summarise the required 100 steps as the single action "walk in a straight line towards the glass". Similarly, if one had to wait ten minutes for a train, one would not consider what to do during each second of the wait; one would simply resolve to "sit there". Clearly, not all actions or series of actions are explicated to the same degree in the minds of humans when they make plans.

  It thus stands to reason that, during the planning process, one ought to consider a sort of *causal distance* — that is, the number of actions which the agent regards as qualitatively distinct. As soon as we begin to group actions together and distinguish temporal from causal distance, the question during planning ceases to be "how long will it take to achieve X?" and becomes "how complicated is it to achieve X?"

- *Goal-based planning:* Our planning scheme first selects an emotion to serve as the guiding one and then proceeds to create hypothetical steps until the guiding emotion is either satisfied, leading the the plan's execution, or until a conflicting emotion overpowers it, leading to the plan's abortion. This is, once again, basically functional, but one could improve upon it by associating certain outcomes — e.g., sating one's hunger or killing a Wumpus — with certain emotions and selecting one of these as goals to reach. Agents would thus no longer seek to satisfy their dominant emotions by any means possibly, but by working towards specific goals.

- *Emotional learning:* In conjunction with goal-based planning, one might also make the association of outcomes with emotions a dynamic one. Instead of outcomes being permanently associated with this or that emotion, agents would be able to learn what constitutes a "good" or "bad", or a "pleasurable", "painful" outcome.

- *Inference about world-states and forgetting:* The agents' memory is merely a perfunctory fact-storage which remembers past perceptions about the world. Importantly, it does not incorporate inferences about likely changes which an agent might reasonably learn, such as the fact that plants regrow or that an agent which was last seen surrounded by ten Wumpuses is likely to be dead now. The learning and application of such inferences about the likely, but not directly observed, changes in the world is an open-ended area of improvement, but carries the possibility of much-optimised behaviour.

- *Concept synthesis:* Although agents are able to experience individual facts about their surrounding world, they do not create larger concepts from these facts to serve as cognitive shortcuts. An agent might perceive three Wumpuses in front of it, say, but it has no concept of "three Wumpuses" or "a horde of Wumpuses". One can think of many other macro-concepts which would directly aid in the creation of efficient plans and reduce cognitive load: "a dangerous area", "an aggressive agent", "a gathering-place for Wumpuses", etc.

- *Evolution of neural nets:* Emotional reactions are currently hand-crafted; the personalities of agents customised by inserting different nets for individual emotions. One might instead allow emotions to evolve by applying genetic algorithms to the neural nets, selecting the best-performing agents in each generation and creating the agents of the next one through recombination and mutation of their parents.

- *Theory of mind:* Our belief generator currently makes no attempt at predicting the actions of other agents; it merely models them as completely passive entities. It would be an interesting, if difficult, addition to utilise levels of trust, sympathy, and respect felt towards certain other agents, as well as some general theory of mind to reason about likely actions that other agents will take.

Bibliography

[Alb93] James S. Albus. A Reference Model Architecture for Intelligent Systems Design. In *An Introduction to Intelligent and Autonomous Control*, pages 27–56. Kluwer Academic Publishers, 1993.

[Alb96] James S. Albus. The Engineering of Mind. In *Information Sciences*, pages 23–32. John Wiley & Sons, Inc., 1996.

[Arb89] Michael A. Arbib. *The Metaphorical Brain 2: Neural Networks and Beyond.* John Wiley & Sons, Inc., 2nd edition, 1989.

[Arb02] Michael A. Arbib. *The Handbook of Brain Theory and Neural Networks.* MIT Press, 2002.

[AV13] Jorge Amory and Patrik Vuilleumier. *The Cambridge Handbook of Human Affective Neuroscience.* Cambridge University Press, 2013.

[Bar91] Henk Barendregt. Introduction to Generalized Type Systems. *Journal of Functional Programming*, 1:125–154, 1991.

[Ber96] Alain Berthoz. The Role of Inhibition in the Hierarchical Gating of Executed and Imagined Movements. *Cognitive Brain Reseach*, 3(2):101–13, 1996.

[BJ87] Kenneth P. Birman and Thomas A. Joseph. Exploiting Virtual Synchrony in Distributed Systems. *SIGOPS Operating Systems Review*, 21(5):123–138, 1987.

[BM09] František Baluška and Stefano Mancuso. Deep Evolutionary Origins of Neurobiology: Turning the Essence of “Neural” Upside-Down. *Communicative & Integrative Biology*, 2(1):60–65, 2009.

[Bra87] Michael E. Bratman. *Intention, Plans, and Practical Reason.* Harvard University Press, 1987.

[Bre03] Cynthia Breazeal. Emotion and Sociable Humanoid Robots. *Internation Journal of Human-Computer Studies*, 59:119–155, 2003.

[Bro86] Rodney A. Brooks. A Robust Layered Control System for a Mobile Robot. *IEEE Journal of Robotics and Automation*, 2(1):14–23, 1986.

[Bro91] Rodney A. Brooks. Intelligence Without Reason. In *Computers and Thought, IJCAI-91*, pages 569–595. Morgan Kaufmann, 1991.

[Car05] Sean Carroll. *Endless Forms most Beautiful: The New Science of Evo-devo and the Making of the Animal Kingdom.* Norton, New York, 2005.

[CD94] Thierry Coquand and Peter Dybjer. Inductive Definitions and Type Theory an Introduction. In P.S. Thiagarajan, editor, *Foundation of Software Technology and Theoretical Computer Science*, volume 880 of *Lecture Notes in Computer Science*, pages 60–76. Springer Berlin Heidelberg, 1994.

[CG99] John T. Cacioppo and Wendi L. Gardner. Emotion. *Annual Review of Psychology*, 50(1):191–214, 1999. PMID: 10074678.

[Col05] Robin G. Collingwood. *The Principles of Art.* Oxford University Press, London, 2005.

[Cop] Jack Copeland. What is Artificial Intelligence? `http://www.alanturing.net/turing_archive/pages/Reference%20Articles/what_is_AI/What%20is%20AI11.html`.

[Cre93] Daniel Crevier. *AI: The Tumultuous History of the Search for Artificial Intelligence.* Basic Books, Inc., New York, NY, USA, 1993.

[Dam98] Antonio R. Damasio. Emotion in the Perspective of an Integrated Nervous System. *Brain Research Reviews*, 26:83–86, 1998.

[Den91] Daniel C. Dennett. *Consciousness Explained.* Penguin, 1991.

[DI99] Richard J. Davidson and William Irwin. The Functional Neuroanatomy of Emotion and Affective Style. *Trends in Cognitive Sciences*, 3(1):11–21, 1999.

[FL07] Jörg Fromm and Silke Lautner. Electrical Signals and their Physiological Significance in Plants. *Plant, Cell & Environment*, 30(3):249–257, 2007.

[FN71] Richard E. Fikes and Nils J. Nilsson. STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving. In *Proceedings of the 2nd International Joint Conference on Artificial Intelligence*, IJCAI-71, pages 608–620, San Francisco, CA, USA, 1971. Morgan Kaufmann Publishers Inc.

[Fon09] Johnny Fontaine. Self-Reflexive Emotions. In *The Oxford Companion to Emotion and the Affective Sciences*, pages 357–359. Oxford University Press, New York, 2009.

[GH01] Sandra Clara Gadanho and John Hallam. Robot Learning Driven by Emotions. *Adaptive Behaviour*, 9(1):42–64, 2001.

[GL88] Michael Gelfond and Vladimir Lifschitz. The Stable Model Semantics For Logic Programming. In *ICLP-88*, pages 1070–1080. MIT Press, 1988.

[Gra18] Henry Gray. *Anatomy of the Human Body.* Lea & Febinger, Philadelphia, twentieth edition, 1918.

[Gra94] Jeffrey A. Gray. Three Fundamental Emotion Systems. In *The Nature of Emotion: Fundamental Questions*, pages 243–247. Oxford University Press, 1994.

[GS01] Gerd Gigerenzer and R. Selten. *Bounded Rationality: The Adaptive Toolbox.* Cambridge: The MIT Press, 2001.

[Hai03] Jonathan Haidth. The Moral Emotions. In *Handbook of Affective Sciences*, pages 852–870. Oxford University Press, New York, 2003.

[HBS73] Carl Hewitt, Peter Bishop, and Richard Steiger. A Universal Modular ACTOR Formalism for Artificial Intelligence. In *Proceedings of the 3rd International Joint Conference on Artificial Intelligence*, IJCAI-73, pages 235–245, San Francisco, CA, USA, 1973. Morgan Kaufmann Publishers Inc.

[HCE$^+$13] Linda Z. Holland, João E. Carvalho, Hector Escriva, Vincent Laudet, Michael Schubert, Sebastian Shimeld, and Jr-Kai Yu. Evolution of Bilaterian Central Nervous Systems: A Single Origin? *EvoDevo*, 4, 2013.

[HNR68] Peter E. Hart, Nils J. Nilsson, and Bertram Raphael. A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968.

[Hof79] Douglas R. Hofstadter. *Godel, Escher, Bach: An Eternal Golden Braid.* Basic Books, Inc., New York, NY, USA, 1979.

[Hof96] Douglas R. Hofstadter. *Fluid Concepts and Creative Analogies: Computer Models of the Fundamental Mechanisms of Thought.* Basic Books, Inc., New York, NY, USA, 1996.

[IGR92] Francois F. Ingrand, Michael P. Georgeff, and Anand S. Rao. An Architecture for Real-Time Reasoning and System Control. *IEEE Expert: Intelligent Systems and Their Applications*, 7(6):34–44, 1992.

[IPY06] Mihai Ionescu, Gheorghe Păun, and Takashi Yokomori. Spiking Neural P Systems. *Fundam. Inf.*, 71(2,3):279–308, 2006.

[JR97] Bart Jacobs and Jan Rutten. A Tutorial on (Co)Algebras and (Co)Induction. *EATCS Bulletin*, 62:62–222, 1997.

[Kem12] Gary Kemp. Collingwood's aesthetics. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy.* Fall 2012 edition, 2012.

[LBC90] Peter J. Lang, Margaret M. Bradley, and Bruce N. Cuthbert. Emotion, Attention and the Startle Reflex. *Psychological Review*, 97:377–398, 1990.

[LHZ11] Benjamin J. Liebeskind, David M. Hillis, and Harold H. Zakon. Evolution of Sodium Channels Predates the Origin of Nervous Systems in Animals. *Proceedings of the National Academy of Sciences*, 108(22):9154–9159, 2011.

[LME$^+$99] Martin Lotze, Pedro Montoya, Michael Erb, Ernst Hülsmann, Herta Flor, Uwe Klose, Niels Birbaumer, and Wolfgang Grodd. Activation of Cortical and Cerebellar Motor Areas during Executed and Imagined Hand Movements: And fMRI Study. *Journal of Cognitive Neuroscience*, 11(5):491–501, 1999.

[LMM99] Sally P. Leys, George O. Mackie, and Robert W. Meech. Impulse Conduction in a Sponge. *Journal of Experimental Biology*, 202(9):1139–1150, 1999.

[Mac90] Paul MacLean. *The Triune Brain in Evolution: Role in Paleocerebral Functions.* Plenum Press, New York, 1990.

[ME09] David Matsumoto and Paul Ekman. Basic Emotions. In *The Oxford Companion to Emotion and the Affective Sciences*, pages 69–73. Oxford University Press, New York, 2009.

[Men88] Paul Francis Mendler. *Inductive Definition in Type Theory.* PhD thesis, Cornell University, Ithaca, NY, USA, 1988.

[Min88] Marvin Minsky. *The Society of Mind.* Simon & Schuster, New York, 1988.

[Min06] Marvin Minsky. *The Emotion Machine.* Simon & Schuster, New York, 2006.

[MKC$^+$14] Leonid L. Moroz, Kevin M. Kocot, Mathew R. Citarella, Sohn Dosung, Tigran P. Norekian, Inna S. Povolotskaya, Anastasia P. Grigorenko, Christopher Dailey, Eugene Berezikov, Katherine M. Buckley, Andrey Ptitsyn, Denis Reshetov, Krishanu Mukherjee, Tatiana P. Moroz, Yelena Bobkova, Fahong Yu, Vladimir V. Kapitonov, Jerzy Jurka, Yuri V. Bobkov, Joshua J. Swore, David O. Girardo, Alexander Fodor, Fedor Gusev, Rachel Sanford, Rebecca Bruders, Ellen Kittler, Claudia E. Mills, Jonathan P. Rast, Romain Derelle, Victor V. Solovyev, Fyodor A. Kondrashov, Billie J. Swalla, Jonathan V. Sweedler, Evgeny I. Rogaev, Kenneth M. Halanych, and Andrea B. Kohn. The Ctenophore Genome and the Evolutionary Origins of Neural Systems. *Nature*, 510(7503):109–114, 2014.

[OLCC88] Andress Ortony, Gerald L. Clore, and Allan Collins. *The Cognitive Structure of Emotions.* Cambridge University Press, Cambridge, 1988.

[PRS10] Gheorghe Păun, Grzegorz Rozenberg, and Arto Salomaa. *The Oxford Handbook of Membrane Computing.* Oxford University Press, Inc., New York, NY, USA, 2010.

[RMPG95] Anand S. Rao and Michael P. Michael P. Georgeff. BDI Agents: From Theory to Practice. In *ICMAS-95*, pages 312–319, 1995.

[RN10] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Pearson Education, New Jersey, 2010.

[Rob05] Jenefer Robinson. *Deeper Than Reason: Emotion and Its Role in Literature, Music, and Art*. Oxford University Press, New York, 2005.

[SGS05] David Sander, Didier Grandjean, and Klaus R. Scherer. A Systems Approach to Appraisal Mechanisms in Emotion. *Neural Networks*, 18(4):317–352, 2005.

[Slo] Aaron Sloman. The SimAgent TOOLKIT. `http://www.cs.bham.ac.uk/research/projects/poplog/packages/simagent.html`.

[Slo91] Aaron Sloman. Developing concepts of consciousness. *Behavioral and Brain Sciences*, 14:694–695, 1991.

[Slo93] Aaron Sloman. The Mind as a Control System. *Royal Institute of Philosophy Supplement*, 34:69–110, 1993.

[Slo97] Aaron Sloman. What Sort of Control System is Able to Have a Personality? In *Creating Personalities for Synthetic Actors, Towards Autonomous Personality Agents*, pages 166–208, London, UK, 1997. Springer-Verlag.

[Slo99] Aaron Sloman. What Sort of Architecture is Required for a Human-like Agent? In Michael Wooldridge and Anand Rao, editors, *Foundations of Rational Agency*, volume 14 of *Applied Logic Series*, pages 35–52. Springer Netherlands, 1999.

[Slo01] Aaron Sloman. Beyond Shallow Models of Emotion. In *Cognitive Processing: International Quarterly of Cognitive Science*, pages 177–198, 2001.

[SSBHH08] Chun Siong Soon, Marcel Brass, Hans-Jochen Heinze, and John-Dylan Haynes. Unconscious Determinants of Free Decisions in the Human Brain. *Nature Neuroscience*, 11:543–545, 2008.

[TD08] Fabrice Teroni and Julien A. Deonna. Differentiating Shame From Guilt. *Consciousness and Cognition*, 17(3):725–740, 2008.

[Uni] Carnegie-Mellon University. 4CAPS Cognitive Neuroarchitecture. `http://www.ccbi.cmu.edu/4CAPS/index.html`.

[Woo] Denise Woodward. Animals I – An Overview of Phylogeny and Diversity. `https://wikispaces.psu.edu/display/bio110/Animals+I+-+An+Overview+of+Phylogeny+and+Diversity`.