A BOTTOM-UP APPROACH TO EMULATING EMOTIONS USING NEUROMODULATION IN AGENTS

Karla Parussel
Department of Computing Science and Mathematics,
University of Stirling,
Stirling,
FK9 4LA,
Scotland

This thesis has been submitted to the University of Stirling in partial fulfilment of the requirements for the degree of Doctor of Philosophy.
November, 2006

Abstract

A bottom-up approach to emulating emotions is expounded in this thesis. This is intended to be useful in research where a phenomenon is to be emulated but the nature of it can not easily be defined. This approach not only advocates emulating the underlying mechanisms that are proposed to give rise to emotion in natural agents, but also advocates applying an open-mind as to what the phenomenon actually is. There is evidence to suggest that neuromodulation is inherently responsible for giving rise to emotions in natural agents and that emotions consequently modulate the behaviour of the agent. The functionality provided by neuromodulation, when applied to agents with self-organising biologically plausible neural networks, is isolated and studied. In research efforts such as this the definition should emerge from the evidence rather than postulate that the definition, derived from limited information, is correct and should be implemented. An implementation of a working definition only tells us that the definition can be implemented. It does not tell us whether that working definition is itself correct and matches the phenomenon in the real world. If this model of emotions was assumed to be true and implemented in an agent, there would be a danger of precluding implementations that could offer alternative theories as to the relevance of neuromodulation to emotions. By isolating and studying different mechanisms such as neuromodulation that are thought to give rise to emotions, theories can arise as to what emotions are and the functionality that they provide. The application of this approach concludes with a theory as to how some emotions can operate via the use of neuromodulators. The theory is explained using the concepts of dynamical systems, free-energy and entropy.

Declaration

I hereby declare that this thesis has been composed by myself, that the work and results have not been presented for any university degree prior to this and that the ideas that I do not attribute to others are my own.

Karla Mattie Parussel
May, 2006

Acknowledgements

I am extremely grateful to the department of Computing Science and Mathematics at the University of Stirling for offering me the chance to study for a PhD. I am equally grateful for their departmental grant and to the EPSRC for funding my study.

I wish to thank my supervisor Professor Leslie S. Smith for his invaluable guidance and experience, and for proof-reading reams of text. I wish to thank my secondary supervisor Bruce Graham and all others in the department who offered helpful advice and references. The friendly and relaxed atmosphere of the department made studying at the University of Stirling a pleasure.

My parents deserve a special mention for believing in me when I did not believe in myself and for helping to fund my education. I also wish to thank my partner Stuart Forbes for his care and moral support.

Contents

List of Figures

Chapter 1

Introduction

This thesis argues that emotions are phenomena that emerge from the interaction of certain subcomponents within a natural agent and to properly understand what emotions are and recreate them, one must first model these subcomponents and their interactions in order to understand why they exist and how they are useful.

This contrasts with existing top-down approaches which give agents symbolic representations of, or discrete functions that emulate, the observed effect of emotions without considering how they were originally produced.

Biologically plausible models of the subcomponents are created in order to increase our understanding of the environmental and functional requirements for such emotions to emerge. Specifically the work attempts to explain the functional use of neuromodulation when applied to neural networks.

1.1 The bottom-up approach

The work does not attempt to mimic complex higher-level human emotions and feelings. These require consciousness and the awareness of emotions. Instead the work emulates and analyses the roles and interactions of the primary mechanisms suggested as being responsible for emotions and which allow animals to react and adapt to noisy and changing environments. This is achieved using a non-symbolic animat approach adapting via a self-organising system.

In her PhD thesis, Gadanho [Gad99] pp5 writes:

> Although the "emotions" [her quotation marks] used were much simplified, they were still named emotions as they tried to capture more functional aspects than those provided by a traditional reinforcement function. Moreover, calling them emotions enables this research to be identified with other emotion research so that developments in the field may be brought together and integrated to produce further richness of emotions functionality and added complexity of artificial agent's behaviour.

The use of quotation marks shows that she acknowledges that the term "emotions" is equivocal and that there is room for debate as to whether these really are emotions. Whilst it is true that her work can be identified as being relevant to other emotions research, it is not required for certain functions to be labelled as emotions to make this so. Presentation of evidence suggesting that these functions are somehow relevant to emotions in natural agents is sufficient.

It is also true that not labelling some part of the system as an emotion of some kind leaves one open to the danger that others fail to see how the work applies to emotions research. For example, the work presented here could be taken as little more than an exploration into the functional uses of neuromodulation with all discussion regarding emotions as a distraction. But something in the brain *produces* emotions. Not once has a neuro-anatomist been able to point to an emotion when cutting up a brain[1]. So if this does not happen in real brains then why make it so in artificial ones?

Nevertheless, certain neurotransmitters have been identified as having possible roles in emotion. For example, drugs that inhibit the re-uptake of serotonin are used to treat depressive disorders [ERKJ00] pp295 & pp1224.

---

[1] . . . or a symbol for that matter.

Emotions, as with intelligence and life, are emergent phenomena, with the label being applied by an observer[2]. Referring to a black box in an agent as an emotion is simply a useful way of avoiding a description of the functionality it provides. But if the raison d'être of the work is to describe this functionality, then the label is not required.

Grand et al [GCM97] state that the agents in the game "Creatures" can plausibly be argued to be instances of "strong" artificial life. As with Gadanho and her instances of "emotions", the authors have not formally argued that they *are* alive, because again, it is obvious that the argument will be a controversial one. For example, Boden dismisses this claim on the premise that life requires a metabolism for the budgeting and usage of energy for behaviour and maintenance of the body [Bod99] [3].

Being in the position to make a controversial scientific statement is acceptable, possibly even desirable, if the evidence backs it up. But the terms "Emotion" and "Alive" are subjective and undefined laymen terms that have not yet succumbed to scientific rigour, despite valiant attempts from researchers and philosophers such as Boden.

There is little scientific value in trying to convince others of a particular subjective viewpoint. Even if more were understood about what emotions are and how to create them, the problem could well remain unsolved indefinitely. The "Uncanny Valley" phenomenon [Mor70] could become relevant as we progress in this field. This is the situation whereby the closer an artificial model of some human quality is to being mistaken for the real thing, the more odd it feels to a human observer[4].

---

[2] One can argue that any neural functionality is emergent because it arises from the interaction of a number of neurons but is neither predictable nor reducible to those neurons. This though, is not a useful distinction of emergent phenomena in the brain. But at what other point should a distinguishable function of a number of neural cells be considered emergent? Emotions are described here as being emergent phenomena because unlike a single neuron or axon, there is not a single physical manifestation that can be pointed to and described as an emotion. It is a label applied to an overall effect.

[3] She does claim though that the model provides a good test bed for further research in artificial life, particularly with regard to implementing ideas on motivation and emotion.

[4] A plausible explanation for the uncanny valley is that when it is unmistakable that an agent is artificial, observers notice the human qualities it appears to have. As the agent becomes more similar to humans, observers tend to focus on the differences between it and

Instead the work presented here attempts to add to body of evidence of what emotions are and are not, so that in time others can review the larger picture. No matter how successful or otherwise the research turns out to be, never at any point will it be said that something is or is not an emotion. No parts of an implemented system will be labelled or described as an emotion.

Following the preaching of McDermott [McD81], labels will be as unambiguous and unequivocal as possible. When an anthropomorphic label is used[5], then it will be made explicitely clear that the label is nothing but a useful handle for the sake of convenience. The work is not concerned about semantics, except to argue that they are currently a distraction at this early stage of the field.

1.2 Why emotions?

Natural agents have evolved emotions. This has occurred because at some point in evolutionary history either emotions, or whatever gives rise to emotions, have provided an increase in evolutionary fitness for the species. Humans are the most intelligent and adaptive animals known to science, and they have emotions. Because emotions bear such an influence upon the behaviour of an agent, and consequently its fitness levels, they have so far provided an advantage. If they had not then there would have been evolutionary pressure to minimise their influence.

A significant research effort in the field of artificial intelligence has been spent in solving the action selection problem. How can an artificial agent be designed to select the actions that maximise resource gain, minimise resource cost and achieve certain goals given its environment?

The one constant throughout the evolutionary history of any species is the need to maintain certain resource levels by staying well fed and hydrated, and to achieve certain goals such as to survive and to breed. Considering the influence that emotions have on our behaviour and choice of action selection, and also

---

themselves.

[5] Namely ”Hunger” and ”Thirst”.

that they have been evolved, it is reasonable to suggest that emotions may have some relevance to solving the action selection problem.

Arzi-Gonczarowski [AG02] argues that emotions have evolved and therefore must be useful for the purposes of survival but warns that this by itself is not sufficient reason for giving emotions to artificial agents[6]. Instead a systems approach is proposed. Returning to first principles it should be recognised that an agent is primarily concerned with survival and the satisfaction of particular goals. The process of evolution in natural agents has resulted in intelligence and emotions. The proposed systems approach starts with basic reactive mechanisms and then repeatedly applies improvements and upgrades. Arzi-Gonczarowski asks in what form emotions would exist in artificial agents and whether they would be recognised as emotions.

1.3 Why neuromodulators?

The majority of research on neural networks applied to autonomous and intelligent agents is generally concerned with the interconnectivity of neurons. There are a number of reasons why neuromodulators should be researched when attempting to create artificial emotions. Neuromodulation exists in the brains of natural agents and may provide a number of functional advantages. Kaczmerak and Levitan [KL87] pp3 define neuromodulation as:

> ...the ability of neurons to alter their electrical properties in response to intracellular biochemical changes resulting from synaptic or hormonal stimulation. Defined this way, neuromodulation is one of the most important intrinsic properties of individual neurons. This property not only allows the nervous system to adapt its control of physiological functions to a continually changing environment, but it is also the basis for many long-lasting changes in animal behavior. Changes in behavior that can be related directly to changes

---

[6] After all, flies are not normally considered to have emotions, although this thesis will not deny the possibility that they have 'fly' emotions that we do not currently recognise.

> in the electrical responses of specific neurons include the triggering of long-lasting, but relatively fixed and innate, behaviors, such as feeding and reproductive behaviors . . . Because the modulation of neuronal electrical activity results in the choice of different patterns of behavior at different times, it is a fundamentally important aspect of neural activity.

This suggests that neuromodulation may provide some useful functionality that could be employed when creating artificial agents. This alone makes them a worthwhile phenomenon to study.

Emotions can be influenced by altering the levels of neuromodulators in a brain [Kel05]. Does this mean that neuromodulators are primarily responsible for giving rise to emotions? Or is it simply that altering the levels of neuromodulators in our brain is an easy and non-invasive way of altering its emotional state? Is Fellous [Fel99] correct when he proposes that emotion can be seen as continuous patterns of neuromodulation of certain brain structures?

In the brain, different levels of neuro-active substances modulate the sensitivity-to-input of neurons that have receptors for them [Koc99] pp94. Could this bring into play different maps from the *same* neural network? If so, this could allow the most effective and relevant behaviours and sensory interpretations to be adopted in order to adapt to the current environment.

Because neuro-modulators such as hormones are relatively long lasting, can they allow for a continuous transition between behaviours and internal states to occur? This would probably be similar to that of a dynamical system rather than that of a discrete change or flip between behaviours. Understanding an agent as a self-organising dynamical system removes the question of when to switch behaviours as the transition happens continuously over time. It also means that it is more appropriate to think of attractive and aversive external stimuli than positive or negative reinforcement. The system can therefore be self-organising and more autonomous. Self-organisation removes the question of when to teach the network and when to recall information encoded in it.

Can emulating neuro-modulators allow us to create neural systems that adapt over longer time frames than if we simply relied upon the topology of neurons and their connections? It is also possible that they could provide other functions that are not proposed here.

1.4 Why Artificial Intelligence?

The two fields of artificial intelligence and computational neuroscience have much to offer each other. Research in artificial intelligence provides the questions and an appreciation of the answers found by computational neuroscience.

By attempting to create and place in an environment the functionality that we see achieved by the brain, we can gain an appreciation of the need for that functionality and the key environmental factors for such functionality to be effective.

If we know what is required for an agent to carry out a certain computation then it will also help direct further research into how the brain is actually performing it. This is particularly important for emergent properties of the mind. How such properties emerge can not be adequately understood unless observed in-vivo or the underlying mechanisms are modelled. We can verify how the functionality is performed in the brain by modelling an explicitly stated set of underlying mechanisms that are theorised to give rise to it.

Scheutz [Sch04] uses an artificial life approach when investigating basic emotions. His framework allows analysis at the level of implementation, at the level of the individual and at the social level. In the framework, different emotion models can be compared and it is suggested that an artificial life approach may inform clinical researchers interested in lesion studies.

1.5 The unique contribution of this thesis

Emotions are emergent phenomena and they come in many different recognisable forms such as happiness, sadness, fear, anger etc. There are many aspects to emotions that are poorly understood and the similarities shared between each form of emotion are equivocal. Because of this, there are many ways of researching the emotions.

Probably the most established method is to observe the emotions exhibited by humans and animals [Dar72] and the consequence of emotional impairment or imbalence [Dam94]. Another approach is to research which parts of the brain are responsible for particular emotions. This would traditionally involve lesion studies, matching observed impairment of functionality to precise lesions. Advances in technology have recently provided non-invasive scanning techniques to show which parts of the brain are active during particular emotional states. For example, based on the knowledge that localised lesions to the human amygdala impair the recognition of fearful facial expressions, Morris et al [MFB$^+$88] have investigated the role of the amygdala more thoroughly using functional neuroimaging.

Once enough data has been obtained from observation of specific parts of the brain, computational models can be created to test specific hypotheses as to the functionality provided by these structures. An example of this is Morén and Balkenius [MB00]. A computational model that partially matches the characteristics of the amygdala and orbitofrontal cortex is presented. The functional model of emotional learning is tested using three classical conditioning experiments.

The ideal proof that the simulation matches the functioning of the real-world system would be obtained by substituting the real world system being modelled with the simulation. This is not practical, or for the most part even possible. But we can take a step closer to this ideal by placing the computational model within the context of an artificial agent and seeing how well it works in a real-world environment Prescott et al have taken this approach with a computational model of the vertebrate basal ganglia [GPG$^+$00][PGG$^+$02a]. They tested the theory that the basal ganglia provides functionality for solving the action selection problem in natural agents. The computational model was used to control a small mobile robot given the task of foraging within its environment. Varying the level of dopamine within the architecture results in similar behaviour as that observed in animals with insufficient or excessive dopamine levels. They suggest that the computational model could be used for testing theories regarding basal ganglia dysfunction [PGG$^+$02b].

This approach has demonstrated its worth for increasing our understanding of parts of the brain whose functionality can be observed directly in a physical environment. It is easier to determine whether an agent, either natural or artificial, is adapting successfully to an external environment than it is to determine whether an agent is acting emotionally.

This thesis proposes a complimentary approach. The parts of the brain that aid in the production of emotion are a result of an evolutionary process. The traditional approach tries to determine the function of these parts and the evolutionary pressures that produced them. We may settle upon a theory that seems to explain why evolution has given us what we observe, but we can never preclude alternative theories that we have not yet thought of.

The complimentary approach proposed in this thesis determines the functionality available to us as agent designers. Rather than work back from an end product left for us by natural evolution, the approach proposed here advocates working forwards and seeing what we can produce. This can be by a process of artificial evolution or by design.

This first requires seeing what can be achieved using the underlying building blocks of the brain and understanding the functionality that they provide us as agent designers. If we end up with a neural network that is functionally and structurally similar to observed parts of the brain, then we can state with more confidence the evolutionary pressures that produced those neural structures. If we end up with something different, then this also helps us refine our ideas as to how the evolutionary pressures have shaped the natural brain.

This thesis applies this proposed approach specifically to understanding the possible uses of neuromodulators in biologically plausible artificial neural networks. By doing so it is hoped that a number of questions can be answered. What functionality can neuromodulators provide us? How can we use them for creating artificially intelligent agents? And does the use of neuromodulation help explain the emergent phenomena of emotions?

1.6 Layout

Chapter two reviews the literature, explaining why natural agents have emotions and why artificial agents should also have them. It then discusses in more depth the problem of defining what emotions are and suggests a solution. The chapter then discusses the various theories regarding emotions and the functional roles ascribed to them before moving onto published research concerning neuromodulation.

The third chapter describes the tools and procedures used in this work. It starts by suggesting a role for the much maligned grid world before discussing the environment to be used to test various modulating agents. The agent, its neural network, actions and senses are described. This is then followed by a description of the tools and procedures used to optimise parameter sets. These allow fair comparisons to be made. The chapter ends with a description of a tool to visualise the structure and activity of the neural networks. This chapter explains the reasoning behind the design of the tools as well as technical details. Depending upon your intentions, it may prove useful to skim over certain details, returning to them if or when they are needed.

Chapter four contains details of the experiments that have been run and their results. The first experiments start with agents that cannot sense or act and are not situated within an environment. These are internal sensing agents with a single body and actions to immediately alter the internal state of that body. It is shown that these agents adapt. This is followed by experiments using extended tests, tests with increasing noise levels and an analysis of how the networks adapt. The experiments are then performed again, but this time for agents with a single modulator rather than with two modulators. The chapter ends by describing the performance of the agent when situated within the environment described in chapter three.

The work ends with chapter five. This chapter reflects upon the work and draws conclusions from it. Ways in which the work can be extended are described.

Chapter 2

Emotions in Natural and Artificial Agents

This chapter reviews research and theories regarding the use, implementation and functionality of emotions that occur in, or have been given to, natural and artificial agents.

2.1 Why natural agents should have emotions

2.1.1 The evolutionary pressure for emotions

Nesse [Nes90] defines emotions as specialised states of operation that give an evolutionary advantage to an agent in particular situations. Emotions in a natural agent are either pleasant or unpleasant, but never neutral. Natural selection is argued to shape emotions only for situations that contain threats or opportunities. Understanding this allows the characteristics of an emotion to be analysed as possible "design features" that increase an agent's ability to adapt to the situation that provided the evolutionary pressure for the emotion to develop.

2.1.2 Emotion in the society of the mind

When discussing whether agents should be given emotions, Minsky [Min88] is frequently cited as evidence that emotions are a fundamental property of intelligent and adaptive agents. In reality, Minsky poses a question as to how fundamental emotions are:

> In any case, no matter how neutral and rational a goal may seem, it will eventually conflict with other goals if it persists for long enough. No long-term project can be carried out without some defence against competing interests, and this is likely to produce what we call emotional reactions to the conflicts that come about among our most insistent goals. The question is not whether intelligent machines can have any emotions, but whether machines can be intelligent without any emotions.

In the hypothetical task of building an artificial animal [1] Minsky refers to how the different needs of the animal, such as thirst, hunger, warmth etc, need to be met. With the exception of multi-bodied 'animals' such as swarms of bees and ants, an animal must economise and share sensors and effectors between different needs. It is argued that an animal can only move in one direction at any one time and is therefore normally constrained to working towards only one goal at a time. If several needs are of an urgent priority then the competition between them needs to be resolved.

Each agent in Minsky's society of mind tries to inhibit all the other agents. The more inhibited an agent, the less able it is to inhibit others and therefore the more inhibited it becomes. This avalanche effect uses the same principle behind the competitive neural networks described in [HKP91] and [Gro92]. This cross-exclusion, and the other mechanisms that Minsky describes such as the conservation of global resources and the use of global inhibitory signals to regulate activity, are in effect global modulatory systems. If it proves useful to draw

---

[1] [Min88] was first published a few years before the pursuit of implementing artificial life properly emerged as a field in its own right.

inspiration from biological systems to create competitive agents and neural networks, then it could prove fruitful to examine how the brain broadcasts global signals using neuromodulators.

2.1.3 Emotions and consciousness

There is an area of common interest between those wanting to create artificial emotions and those wanting to create artificial consciousness. Shanahan [Sha05] implements a cognitive architecture inspired by the brain to anticipate and simulate future interactions with the environment. He implements analogies with the functional roles that have been attributed to consciousness, emotion and imagination.

Bosse et al [BJT05] explore Damasio's theory on core consciousness. Damasio describes an *emotion* as an unconscious neural reaction to a stimulus which leads to a change in body state. A *feeling* is an unconscious sensing of this new body state; and *core-consciousness*, or 'feeling a feeling', is the conscious recognition of this change, via a change to the *proto-self*, or representation of the body.

Aleksander et al [IML05] state that emotions and feelings seem to be a required property for being a conscious human being and that from this it can be inferred that it is also a fundamental property of machine consciousness. As a result, the claim of a machine being artificially conscious should be met with suspicion if it is does not have mechanisms emulating the role of emotions in living organisms.

2.2 Why study emotions?

There are many reasons why emotions should be studied with a view to implementation within an artificial intelligence or artificial life system. There are various reviews detailing these reasons and others such as [Sch02] and [Gad99]. Reasons that are relevant to the research carried out in this thesis are expounded here in more depth.

2.2.1 Emotions and rationality

The classical view of an agent or artificial intelligence is one consisting of logical inference and reasoning within a deterministic environment with clearly salient and defined variables and symbols. The history of the field has shown this to be inadequate for noisy and variable environments in which feature extraction is required before adaptation can occur. But even within these environments an agent must still act rationally inasmuch as its actions, behaviour and adaptation must have a high utility and low cost.

Ventura et al [VCPF99] state that apart from intellectual and academic curiosity, emotions should be studied and implemented if it leads to better performance than agents without emotions. Although emotions have a significant influence upon the behaviour of humans they claim that it is not clear whether emotions should be implemented in agents. Behaviours resulting from artificial emotions are argued to not be explainable[2], a crucial property when faced with convincing others about the decisions being made. This is a problem shared by the neural net community as a whole. But recognising that the mechanisms of emotions do play an essential role in human rationality, they may prove useful in agent adaptation and their benefit should be ascertained.

Evans [Eva02] discusses how emotions can help the reasoning process in the form of the search hypothesis. A rational agent makes a decision based upon the highest expected utility. A conditional probability is assigned to the outcome of each action that can be performed. But that requires all possible outcomes to be listed. In an open-ended and partially unknown environment this list can be indefinitely long. This process needs a cut-off point. The search hypothesis proposes that emotions constrain the range of outcomes to be considered and

---

[2] A controversial argument if one properly understands the underlying mechanisms and has adequate instrumentation. Although an emotional agent is clearly harder to explain than a purely deterministic one, this characteristic also applies to any non-trivial adaptive system situated in a noisy and dynamic environment; or indeed to any complex system whose behaviour depends upon the history of its input.

subjectively applies a utility to each. This avoids *analysis paralysis* and allows the agent to more rapidly settle upon a decision and act.

There are other ways in which irrationality is required in order to behave rationally. Nesse [Nes90] relies upon the concept of irrational behaviour in order to explain the rationality of the emotion of anger. In a long term, committed social partnership where one party is tempted to defect, the threat of an irrational and spiteful retaliation because of the betrayed partner's anger decreases the likelihood of a defection continuing or even taking place at all.

Rational is defined in the dictionary [SSH01] as:

> adj. 1 being based on reason; *a rational explanation.* 2 able to think sensibly or logically.

Given that the agent controllers implemented in this research are non-symbolic and do not perform any logical inference or deduction, it can be questioned as to whether the advantage of rationality conferred by emotions is relevant. A distinction must be made between an action being performed because of some logical reasoning or cognition, and an action being performed that can later be judged as being rational. As Ledoux points out [LeD98], the real reason for an emotion or action may be very different from the rational explanation that we later provide ourselves and others.

2.2.2 Artificial Life and Virtual Reality

The two fields of artificial life and virtual reality have much to offer each other. Artificial life attempts to explore the underlying biological principles unconstrained by the real environment. Rather than only studying life as we know it, it is open to life as it could be [Lan96]. This means that new, rich and complex environments can be explored.

Artificial intelligence also benefits from a bottom-up approach whereby agents should be physically embodied in noisy and realistic environments [Bro91b]. However what constitutes realism depends upon the target environment of the agent. Etzioni [Etz93] claims realism at every stage by using softbots in a working Unix operating system. Experiments using physical and mobile robots it is argued, are lengthy and difficult. They do not provide the only route to creating artificial intelligence.

The desired application of artificial intelligence should determine the environment in which it is created and the methods used, whether that be a robot maintaining an office, a softbot maintaining an operating system or an expert system in a doctor's office.

The introduction of [TT94] discusses the need to populate virtual reality environments with virtual plants, trees, animals and humans. Artificial emotions can be used to make animals and humans more convincing and can provide a mechanism by which designers can influence the emotional state of the user. Johnson and Jones [JJ99] discuss communication of affective information in virtual environments between all permutations of human and agent interactions. They stress the importance of affective states in making a virtual environment truly realistic. Gratch and Marsella [GM01] discuss the critical role of emotions in making virtual humans convincing and be able to hold the interest of the user. They increase empathy and attachment in the user to the virtual humans. Their application is the Mission Rehearsal Exercise system in which trainee users are immersed in a resource allocation scenario typically found in a military peace-keeping exercise.

Given that realism and convincing behaviour are common justifications for implementing artificial emotions, one can assume that it is also important to realistically emulate the underlying mechanisms that cause emotions.

2.3 The difficulty in defining emotions

When attempting to understand and recreate a phenomenon, it is intuitive to define it first. This focuses the research effort and helps to determine what is and is not worth the attention of the researcher. This increases the likelihood of some worthwhile results being obtained and conclusions drawn. Ordinarily this is a useful endeavour; that is, for phenomena for which definitions can be unequivocally decided upon.

Humankind has long attempted and failed to decide what emotions are and the function they fulfil, or indeed whether they do at all. Modern science has not yet produced a consensus on what emotions are. If this was achieved then the next step would be to create artificial versions of emotions. The lack of consensus suggests that we do not yet understand enough about emotions to create a single definition. Whilst very possibly true, this might not be the only reason.

It is futile to repeatedly ask the same question if it cannot yet be answered because of insufficient knowledge. More knowledge regarding the problem domain needs to be obtained before the question is repeated again. And so it is with asking what defines an emotion. Unfortunately this means that we have to find another way of understanding and recreating emotions.

2.3.1 Phenomena or epiphenomena?

Sloman [SC81] argues that the mechanisms that give rise to emotions are also the same mechanisms that an intelligent system requires in order to be flexible in a complex environment. The case argued is that if an intelligent system has multiple motives and limited powers then that system is likely to also have emotions. Conversely, emotions can be understood by examining the motives and types of processes that they can produce. An emotional state normally involves at least one strong motive.

Sloman attempts to create a grammar of artificial and natural minds by laying out the possible constraints on the design of intelligent systems, the internal processes involving motives and the types of motives that an intelligent system may have. This provides a spectrum of intelligent systems from the very simple to those containing the full complexity of the grammar. It is argued that within this spectrum it is pointless to draw a line as to what is and is not real intelligence. The analogy made is that if one player accepts the handicap of playing chess without a queen then it is still a game of real chess.

A seemingly opposing view is taken by [Sch02]. Scheutz reviews the emerging research on the design of emotional agents. A criticism is made of the common practice of labelling agents as having emotions when it is not clear whether or not they really do. It is argued that describing agents as having such properties without due justification is counter-productive to the quest to understand and model emotions. Highly abstracted causal descriptions lack the low-level details and mechanisms that give rise to emotions in natural agents in the first place. It may be that the missing low-level details are what is required for an agent to have actual emotions.

In McDermott's classic rant [McD81], it is argued that labelling some code as being a property of the larger system does not mean that it actually is that property. It is suggested instead that labels should be as unambiguous and unequivocal as possible and that the author should convince themselves and anybody else, that the code does indeed provide the property that is intended. So for example, Gadanho [GH98] did not implement the emotions of happiness, sadness, fear and anger in an agent. What was actually provided were four modules defining dependency and bias constants.

2.3.2 Classical concepts and cluster concepts

Scheutz makes a distinction between classical concepts and cluster concepts. Classical concepts have well-defined boundaries and a set of conditions that can be used to determine whether something is a member of that class. Emotions belong to cluster concepts. No one single definition of emotion can be applied to all different subspecies of emotion even though most emotions have enough in common to be grouped as a class. This shows itself in the field as a lack of agreement on what constitutes an emotion.

Bedau [Bed96] writes that a cluster concept provides no explanation as to why such properties should make up such ubiquitous and fundamental natural phenomena. But then neither will a single encapsulated definition; observation, hypothesising and evaluation will achieve that. Bedau's main argument against cluster concepts is that they make the properties of life seem arbitrary. Seeming is not the same as being. The fact that it is more difficult for us to reason about a cluster of properties that co-exist does not detract from the possibility that different aspects of the same cluster are involved with the different manifestations of the emergent phenomena.

The difficulty of the cluster concept does not just affect the field of emotions research. For example similar problems surround the definition of life and of intelligence. The first chapter of [Ste00] discusses the concept of intelligence, first from the lay person's point of view and then from an expert's point of view. The chapter starts:

> Looked at in one way, everyone knows what intelligence is; looked at in another way, no one does. Put another way, people all have conceptions . . . but no one knows for certain what it actually is.

The argument can be made that the concepts of lay-people are not relevant because they do not need to rigorously define or apply their concepts. For example Bedau, when reviewing the differing definitions of life in the literature, writes that "we want to know what life *is*, not what people *think* life is" [his emphasis]. An analogy is made with how chemists define glass as a liquid even though the everyday concept of it is as a solid object. But what is *life* but a handle that we use to refer to some natural phenomena? In which case the common usage of this handle is relevant. You can still study the nature of this phenomena regardless of the definition that you use.

In exploring the definition of a game, Wittgenstein [Wit58] considers the different examples of games; board-games, card-games, ball-games, the Olympic games, ring-a-ring-a-roses etc. and challenges us as to what common feature links them all. Instead of finding one definition to include all the examples there is a complicated network of similarities, best characterised in Wittgenstein's view as ”family resemblances”. If there is a special need for a boundary to be drawn then one can be, for that purpose only. A boundary can not be unambiguously defined. But a boundary is not required in order to make the concept usable.

If there is one underlying feature of many games that Wittgenstein does not consider then it is that the term ”game” is used to describe an activity which is relatively less consequential than the alternative. For example, a nation losing at the Olympic games is less consequential than if it had lost a war. For the purpose of teaching, a game can be less consequential than practising an activity for real. The child’s game of ring-a-ring-a-roses is less consequential than schooling, eating, drinking and sleeping.

By itself this feature is not complete enough to form a definition and is too contrived for some examples so Wittgenstein’s point is still valid. But it is interesting to point out that Wittgenstein is comparing all the examples together and each one out of context. When considering whether something fits a definition it may prove useful to take into account the context.

One might argue that a set of variables or lines of code are not emotions, intelligence or life. But then one could argue that neither are a set of neurochemicals, neurons or DNA sequences. It is the effect of these things within an entity and its environment that provides us with evidence of these phenomena.

2.3.3 Graded levels of phenomena

The problem of creating a definition does not just come from cluster concepts, but also because we are trying to apply a discrete classification to a continuous phenomena. Can an artificial agent be intelligent? No if we take intelligence to mean human intelligence because an artificial agent is not human. Yes if we use a working definition and see that the agent is effectively adapting within its own, partially unknown, environment.

Kak [Kak96] argues that we can test for levels of artificial intelligence rather than for a discrete property. Not all animals are equally as intelligent at all tasks. Experiments on the ability of pigeons to learn concepts have shown that gradations of cognitive function can be defined. In one experiment the pigeons were shown to have a similar performance to pre-school children. But all animals can be argued to be sufficiently intelligent to survive within their ecological niche. They are intelligent *within* their own environment. They are as intelligent as their environment, physical bodies and senses require them to be.

2.3.4 Working definitions

A useful approach is to find a working definition, such as Wang [Wan95]. This does not attempt to provide an unequivocal definition for everyone, but does provide a definition that can be used, and worked with, for a specific research effort. Wang points out that a concept is meaningless if it encompasses everything. Therefore a useful complimentary question is, what characterises the absence of intelligence?

Taylor [Tay98] argues the importance of being explicit as to what high-level phenomenon is being researched and the underlying mechanisms believed to be responsible for it. Referring specifically to the phenomenon of life, Taylor calls for an explicit definition of it when building investigative models. It is not required for the definition to be unequivocal.

Unequivocal definitions that are explicitely stated for the purpose of the research are working definitions. These are useful when employing a top-down approach. They provide a goal to aim for so that you know when you have made progress. If a bottom-up approach is used to identify and test each form of functionality that it is proposed that emotions gives us, then in essence we are using and testing a different working definition of emotions each time. No single working definition need then be decided upon to encompass the entire work. This does not solve the problem of how to unequivocally define the phenomena being emulated but it allows for different aspects to be researched until enough is understood to achieve this.

But how can high-level natural phenomenon be explicitly defined if we do not yet understand what it is? If a scientist is trying to answer whether a specific set of low-level mechanisms is directly responsible for a particular instance of some high-level phenomenon then Taylor's argument holds true. But understanding how the low-level mechanisms interact and can be used, without any specific result being intended, can also be a useful exercise. This is on-a-par with the way that mathematicians play with formulas without specific relevance to any real world phenomena.

It is useful to see what high-level phenomena can arise from those explicitly stated low-level mechanisms. This allows new avenues of research and engineering to be opened up that were not previously thought of. It cannot be assumed that all possible applications of the low-level mechanisms can be envisaged a-priori with limited knowledge.

2.3.5 Finding the boundaries of definitions of emergent phenomena

Fellous [Fel04] suggests that the focus of study should be on the function of emotions rather than on what they are. Seen in this way, animals can be seen functionally as having emotions, whether or not we empathise with them. Given this, robots can functionally have emotions as well. One function of emotions mentioned that has a robotic counterpart is to achieve a multi-level communication of simplified but high impact information.

So applying these lessons to the emergent phenomena of emotions, can we have graded levels of emotions? What is *not* an emotion? Can the concept of emotions be extended and applied to robots?

Even with humans we recognise that some people are more emotional than others. We also recognise when something obviously does not have emotions, for example an inanimate object such as a chair. The difficulty lies in the grey areas when there are no clear cut boundaries to the concept. The question becomes one of how far to extend the concept.

I would like to suggest a maxim:

> *When a scientific community cannot agree upon the scope of some emergent phenomena, they are only debating the boundaries of a definition.*

We must not be distracted by a need to create precise definitions. There is a saying relevant to the field of artificial life that if it looks like a duck, walks like a duck and quacks like a duck then in the absence of any compelling evidence to the contrary, it probably is a duck. We need to look beyond the question of whether this is a justifiable argument and recognise the true goal and achievement of creating such a duck-like agent in the first place.

If we manage to isolate the various functions provided by the natural phenomena, the bottom-up processes that provide them and emulate these in an artificial agent to the same effect that they are found in natural agents; then we need not care whether or not people extend their own definitions to include them. Are we trying to evolve agents with artificial emotions, intelligence, consciousness, life etc? Or are we concerned only with convincing others that we have?

2.3.6 Phenomena relative to the environment and agent

Pattee [Pat96b] attempts to distinguish between computer simulations, and realisations of life. Computer simulations are stated to be metaphorical models and no matter how life-like, they will never be realisations of life. Realisations are defined as being literal and material. The simulation of a trajectory never results in a realisation of motion. It is proposed that the requirements for being an AL realisation of life are that the agent should be a situated model that is a phenotypic expression of a genotype that is subject to mutability, heritability and natural selection. Rather than just adapt to its environment, the realisation should also display emergent behaviour.

Sober [Pat96a] argues that a computer can be used to simulate stresses upon a bridge, but no matter how sophisticated the simulation, the computer simulation itself will never be a bridge. Computer simulations are at heart nothing more than representations and that there is no need for overstatement in the field of artificial life.

But Sober is applying a definition of a bridge, obtained from the real world and arguing that the definition should not be applied to the simulated world. *Within* that simulation and context, the simulated bridge has the definition of being a bridge. For example, it could be a bridge across a chasm in a virtual environment providing the same function as a real-world bridge. But it does not fit the definition of a real-world bridge because it does not match the exact subset of conditions for it to be so, even though both definitions are drawn from the same cluster concept.

If a metaphorical and a material model have the same list of properties, both of which fulfil Pattee's requirements above, then the former is only a simulation and the latter a realisation because Pattee has defined it to be so. Both are instances that fulfil a definition of artificial life. They are not instances that fulfil the definition of natural life.

Yet an instance of artificial life and an instance of natural life will both fulfil some of the conditions of the cluster concept. The scope of a definition should be decided by the environment of the agent or phenomena and the underlying materials and mechanisms which have produced it, whether it be lines of code or a biophysical expression of DNA. Definitions supply a context within which a condition can be determined as being true or false. The scope of the context supplied by the definition should match the scope of the environment of the agent or phenomena.

2.3.7 Emergence of precisely bounded definitions

As will be discussed later in section 2.5, this does not mean to say that we need to totally discard definitions. Definitions are a tool that allows us to discuss and reason about a subject, nothing else. Instead we should abandon the practice of attempting to apply precise constraints to an inherited layman term. Let new precisely bounded definitions arise through the bottom-up process of finding out what works and from the observations of natural phenomena *when they can be made.*

The new precise definitions should be created for the grey areas to which the layman terms cannot be easily applied. This leaves free the original layman terms to be applied to what everyone originally understood them to be meant. An example can be drawn from the relatively young field of Geology, the beginnings of which are attributed to James Hutton of Edinburgh in the late 18th Century. "Granite" used to be a layman term. The common usage of the term was clearly understood. But the characteristics that determine whether a rock is made up of granite can also be found to a greater or lesser extent in rock that is not granite. Yet the field now has a precise definition of what granite is, based upon the relative amounts of quartz, K-feldspar, plagioclase and feldpathoids that make up the rock [Nes00].

How did mineralogists manage this when practitioners in the field of artificial intelligence and artificial life argue how far to extend their definitions? The grey areas that surround the space of what is generally known as granite also have their own, newer, definitions derived from observation.

Why these definitions have arisen is beyond the scope of thesis and of course the comparison can only be taken so far. Mineralogists are observing quantifiable physical characteristics as opposed to emergent phenomena, the functions of which are still being determined. But the boundaries of the definition were not determined with a view to distinguishing between rock that was and was not granite. Instead the boundaries were determined with a view to distinguishing between rock that was granite, quartz syenite, anorthosite, alkali feldspar granite, granodiorite, quartz-rich granitoids etc. It became possible to determine the boundaries of the definition by observing and learning about the characteristics of the space outside of it.

What emotions researchers are doing is the equivalent of attempting to precisely apply the term "Granite" to a boundary definition such as "granodiorite", a rock that would originally have been classed as granite by laymen. But doing this ignores all the other boundary definitions. Instead each functional class of emotions should be given their own specific definition. We may not end up with a single definition of emotion but we will gain precise and unambiguous definitions that serve as a common point of reference.

2.4 The functionality provided by emotions in natural agents

2.4.1 The need to understand the functionality of emotions

Nesse [Nes90] argues that in order to understand emotions, we need to understand the functionality that they provide, the selective evolutionary forces that have shaped each emotion and the situations in which having these emotions have increased the Darwinian fitness of the agent.

In order to formulate explicit evolutionary explanations of emotions, Nesse advocates an adaptionist approach in which emotions are explained in terms of the resulting functionality that increases Darwinian fitness.

Rolls [Rol99] also advocates furthering our understanding of emotions by increasing our understanding of their function. Doing this increases our understanding of the brain mechanisms that give rise to emotions. But Nesse's stance is stronger on the importance of functionality, stating that no definition of emotions can be agreed upon until their function can be fully conceptualised. Scheutz [Sch02] also states that emotions should be defined in terms of their functional role.

If we fully understand the functionality of an emotion, the mechanism by which it fulfils its role, and can model this in an agent, then all that is different is the medium in which it is implemented. Maybe like Wang we should sometimes assume an inverse stance on our questioning. Instead of asking whether our models are also emotions, we should ask why they are not emotions and determine the significance of any differences between the two.

2.4.2 The functionality of emotion

Rolls describes ten functions of emotions in natural agents.

- Autonomic and endocrine responses
- Flexibility of behavioural responses to reinforcing stimuli
- Motivation
- Communication
- Social bonding
- Hedonism
- Cognitive evaluation of events and memories
- Facilitation of memory storage
- Persistence
- Facilitation of memory recall

2.4.3 Complex functionality

Some of the above functions have more prerequisites than others and are therefore more complex and time consuming to implement in an artificial agent.

Autonomic and endocrine responses require the implementation of an artificial body and an environment in which it can be situated. An example of an autonomic response would be a change of heart beat in a natural agent, or perhaps a change in performance and power consumption for a CPU in a robot. An example of an endocrine response in a natural agent would be the release of adrenaline. In an artificial agent this could be a global signal for the state of a resource, such as power level, which alters the behaviour and immediacy of the agent's goals.

Communication and social bonding requires a multi-agent system with a rich environment and senses, a means to communicate about the environment and a shared need for communication.

One suggested role of emotions is to provide a flexibility of behavioural responses to reinforcing stimuli. This assumes that all emotions are either rewarding and pleasurable, or punishing and unpleasurable, as advocated by Rolls and Nesse. Instead of learning a fixed behavioural response to a stimulus it is suggested that learning takes place in two stages. The first stage associates emotional states with the rewarding or punishing stimulus, and the second stage makes use of previously learned strategies to obtain the reward or avoid the punishment. Implementing a biologically plausible two-stage process will always be a significantly more complex task as both stages have to work together. This may have advantages to an adaptive agent and the fact that it appears to have evolved in natural agents suggests that this is so. Such an architecture could possibly be advantageous in a complex environment with many concurrent stimuli and goals.

Porr and Wörgötter implement a similar two-stage system in a physical robot [PW03] [PW02]. In their experiment, a robot can sense a collision with another object. This causes a reflex action in the robot, causing it to retract. The robot is also equipped with range finders. These respond to obstacles earlier than the collision sensors. Without supervision, the robot learns a causal relation between the two sensor modalities and reacts to the earliest incoming signal. Different strategies emerge to avoid obstacles depending on the initial conditions of the robot.

Implementing biologically plausible cognitive processes and selective memory storage and recall are ambitious projects in themselves and distracts from the aim of implementing the functionality of emotions in an artificial agent. It is therefore considered out of the scope of this thesis.

2.4.4 Simple functionality

A simple form of action selection can be substituted for cognitive processing. Instead of performing a fixed behavioural response to a stimulus, an agent can make an appropriate response according to its current emotional state. For example, a dark corner is attractive for prey that is being chased and is scared, yet may not be so attractive for an agent that is hungry and needs to explore.

Other functions are easier to implement. One suggested function of emotions is to motivate an agent to act to obtain a reward or avoid a punishment. This raises the question of whether an agent should only act because it is in an emotional state or whether emotions should function as a signal to prioritise certain actions.

Hedonism comes from the generalisation that anything that is pleasurable and is positively reinforcing has survival value. There are obvious exceptions to this. Brains can be deceived and manipulated into providing pleasure when it is detrimental to survival to do so. But when taking into account the environment that natural agents have evolved for, on the whole pleasure can be used to signal and to motivate the agent towards stimulus and behaviours that have a high survival value for the agent and its genes. In essence, emotion can be used to provide reinforcement learning for agents [Gad99] [SB98].

Some behaviours require persistence before their utility is realised even though the initiating stimulus is no longer present. For example, prey may still be in danger even though it no longer sees the predator that has narrowly missed catching it.

2.5 Previous definitions of emotions

The history of emotions research has been left until now so that it can be reviewed within the context of current scientific thinking on the subject.

As described above, Nesse [Nes90] defines emotions as specialised states of operation that give an evolutionary advantage to an agent in particular situations. LeDoux [LeD98] pp16 describes a distinguishing characteristic of cognitive processing as flexibility of response to the environment. Emotions provide a counter-balance to this by narrowing the response of an agent in ways that have a greater evolutionary fitness.

If each phenomenon that we may or may not recognise as an emotion has evolved to fulfil its own specialised role, then what makes us think that there is much in common between all emotions? Fear and joy fulfil vastly different roles and cannot be viewed as merely negative and positive emotions at opposite ends of a single spectrum. LeDoux [LeD98] writes that ”emotion” is nothing more than a label that allows us the convenience of discussing aspects of the brain and mind.

In the literature, different roles have been assigned to emotions. They have been explained as mechanisms to appraise the environment or to make salient features within it, or states produced by reward and punishment. Emotions have been described in terms of body responses or as a form of modulation. Each claim is plausible when discussed in terms of specific emotions, but none are applicable to all emotions. Given that each emotion has evolved for a specific reason, it may be that there is no single role that can be applied to all emotions. Instead it should be recognised that there are different classes of emotions. We should be wary of attempts to constrain the label of ”emotion” to one particular class. If some specific functionality is observed to be common to a class of emotions, then this class needs its own new label with an unambiguous meaning.

2.5.1 Body-centric emotions

Early ideas

Throughout the ages philosophers have been asking what emotions are but science first started to tackle the subject with an article by William James [Jam84] in 1884. This resulted in what is known today as the James-Lange theory of emotions because Lange [Lan85] proposed a similar argument a year later which attributed a greater role to autonomic feedback in creating the experience of emotions.

It was argued that emotions were generated by sensing changes occurring in the body such as the heart rate or changes in the skeletal muscles and responding to those changes.

In the scenario of meeting a bear and running away from it, it was proposed that we feel fear because we are running away rather than the running being a symptom of being afraid. It is because of the changes in the body that the experience of having an emotion is different to those of other states of the mind. Unfortunately it was never explained why we should be running away in the first place.

This view was largely accepted until challenged by Cannon [Can27] who researched the responses of the body when hungry or acutely emotional. Cannon argued that the responses were the same regardless of the emotional state and that these responses were also too slow to be the cause of the emotional state. Cannon denied that a response from the body was required for an emotional state, which instead originate within the brain. But Cannon did agree that emotional states were distinguished from non-emotional states because of the accompanying bodily responses.

Schachter and Singer [SS62] added to the debate by suggesting that cognitive processing determined what emotion was being experienced by interpreting the non-specific response of the body in the context of the information sensed from the environment. Their theory did not tackle the question of why the body was responding in such a manner in the first place.

The Somatic-marker hypothesis

Damasio [Dam94] also took a body-centric view when explaining how emotions are required for a human to be a rational agent, a view later espoused by the search hypothesis [Eva02].

Damasio describes how nearly every part of the body can send a signal to the brain via the peripheral nerve to the brain stem or spinal cord. The body can also communicate to the brain chemically by the bloodstream. This influences the brain directly or by activating the subfornical organ, a neuronal structure that senses low levels of angiotensin II, a blood-borne molecule, in order to regulate thirst [ERKJ00] pp1006. The subfornical organ then communicates this to the hypothalamus. The brain can also influence the body, either chemically or by the use of nerves.

The brain is grounded in the body. It senses and interacts with the environment from within the body. There are many requirements that an organism must fulfil in order to survive and to pass on its genes. It needs a supply of oxygen, water and nutrients. Predators need to be avoided and genes need to be passed on through sexual behaviour and care of kin. A brain cannot exist without a body but some simpler, evolutionarily older, organisms can exist without a brain. Damasio writes:

> I see the *essence* of emotion as the collection of changes in body state that are induced in [the] myriad of organs by nerve cell terminals, under the control of a dedicated brain system, which is responding to the content of thoughts relative to a particular entity or event.

A body-loop is described in which a signal from the brain is sent to the body and then received back by the brain. The brain waits for a report from the body. The report is an evaluation of the current, or predicted, situation. It frames rational searches. Damasio uses two examples. He asks us to imagine being the owner of a large business, approached by a lucrative potential client who is the arch-enemy of a long term friend. You have two options. Lose the client or lose your friendship. Before any reasoning and a cost / benefit analysis is mentally performed, when a negative situation is imagined as a result of a choice, an unpleasant gut feeling occurs in the body. This focuses the brain's attention and acts as an alarm signal. This may lead to an immediate rejection of that option.

In another example, a squirrel in Damasio's backyard sees a black cat, and before any cost / benefit analysis or mental reasoning is performed, the squirrel is jolted by a body state to run up the nearest tree.

The counter-argument to the Somatic-marker hypothesis

Rolls [Rol99] lists a number of reasons why the James-Lange theory of emotions is implausible and sees Damasio's somatic-marker hypothesis [Dam94] as a weakened version of this theory but one which still suffers from many of the same fatal flaws.

- What is the cause of the original body response? Why are we running from James' bear or the squirrel running up the tree?
- The peripheral changes are too indistinct to encode the information required to have subtly different emotional feelings to such a large range of different stimuli.
- Emotions evoked by imagery are accompanied by weaker and more indistinct peripheral responses than those evoked by external stimuli. The emotion may be just as strong for an evoked image but the need for a bodily response is reduced.
- Emotional responses are still possible and can still be perceived even when there is a disruption of peripheral responses and feedback, for example by spinal cord injury, surgery or blocked by drugs.
- Autonomic changes artificially induced by injections of substances and hormones such as adrenaline, do not produce particular emotions, they can only modulate the extent to which existing emotions are felt.
- Emotional behavioural expressions such as smiling, more often occur as a result of social communication rather than as an expression of one's emotional state.

- Once the need for a response has been determined, it is inefficient and noisy to have a peripheral response and then a transducer to measure that response in the execution of an emotion.

A resolution between the Somatic-marker hypothesis and its critics

Johnson [Joh04] questions why somatic-markers need to be confined to the body. The somatic-marker hypothesis is considered from an alternative perspective using the concept of Dawkin's extended phenotype [Daw82] to create the extended somatic marker. Johnson suggests that one convincing use of somatic markers is to allow simultaneous and co-ordinated communication with multiple brain regions. But non-somatic neuromodulation can achieve this too.

Johnson also questions why it is not sufficient for a marker to be a mental marker only and suggests the functional analogy of the hardware interlock, an independent safety-device outside of the main system that monitors one small aspect of it.

What I suggest is that somatic feedback is an epiphenomena of emotion rather than a prerequisite for it. If there is continuous feedback from the body to the brain as Damasio suggests, then it is not appropriate to think of the body-loop as a sequence with the brain communicating with the body and waiting for feedback before evaluating a response. Instead it is a continual stream of sensory information no different from the other external senses. The continual feedback from the body should be seen as updating the current emotional state. This is no different to any other new external sensory information updating the evaluation of a situation.

Upon seeing a cat, the squirrel becomes fearful and wants to keep a distance from it. The brain of the squirrel sends a signal to its body to *prime* itself. If the squirrel has plenty of energy then a jolt signifies that it is plausible to run away while it can. Indeed, a sufficient response from the body resulting in a jolt may make it necessary to run away as the movement will attract the attention of the cat.

But if the squirrel is tired after a day of foraging, the poor response of the body suggests that running away and drawing the attention of the cat is not guaranteed to save the squirrel. Instead, the squirrel may well freeze hoping not to be seen. It is because the squirrel is fearful that the signal and the response from the body occurs.

It makes sense that because the brain has evolved to maintain homoeostasis of the body there will in some cases be a signal / response loop between the two. But this does not mean that somatic feedback is essential to *all* emotions. Somatic feedback carries a cost and so why should it exist in situations that do not directly affect or rely upon the state of the body when the processing can occur fully in the brain? It may be that with some emotions, the body-loop only exists to be used in cases of extreme emotion, possibly acting much like a hardware interlock, but is not relevant for the normal case. There needs to be a distinction drawn between emotions that require immediate and strong somatic feedback, such as hunger, thirst, fear, anger, disgust and sexual arousal, and those which do not such as love and pity.

2.5.2 Environment-centric appraisal emotions

Not everyone agrees that emotions originate from the body, for example Rolls [Rol99] pp65 argues that emotional states are normally initiated by reinforcing stimuli that have their origin in the external environment. [3]

The body-centric theories of emotion in general ignore one fundamental question. What makes the body respond in the first place? For emotions that signal a particular state of the body, such as hunger, satiation or pain, this is not such an issue. But when the theories are applied to emotions that have their impetus in the environment that the organism is situated in, such as social emotions, fear and anger, then this question needs to be addressed.

The appraisal theory of emotions, as discussed by Magda Arnold [Arn60]

---

[3] He admits that there is room for further refinement to this idea of emotions and ignoring pleasure from satiation, that some people may say that they experience emotion when savouring food.

attempts to address this issue. The organism mentally and unconsciously appraises the desirability of its situation. This biases the actions of the organism to what it perceives as good and away from what it perceives as bad. It is this bias that the organism consciously feels as emotion. In the scenario of meeting a bear in the woods, the bias to run away is consciously felt as fear.

The idea that emotions are appraisals is still supported by researchers to this day. For example, Goldie and Döring [GD05] state that emotions necessarily involve evaluation but deny that this necessarily leads to a desire to act. Rolls [Rol99] pp60 proposes that emotions are states elicited by rewards and punishers and changes in reward and punishment.

Evaluation by itself is not sufficient to describe what an emotion is. The raison d'être of action selection is to increase the current or future well-being of an agent and performed in an unknown environment will involve evaluation. This does not necessarily involve what we recognise as emotion, but the ability to discern between what is beneficial or otherwise to an agent is a common characteristic of many theories about emotions.

2.5.3 The Interrupt theory of emotions

The Interrupt theory of emotions postulates that the role of emotions is to interrupt the on-going behaviour of an agent in its satisfaction of goals. New goals are then set according to the emotion that has been activated. The idea was first introduced by Simon [Sim67]. It was later developed by Oatley and Johnson-Laird [OJL87] who among other things, ascribe to emotions the role of non-propositional communication. These are non-symbolic signals that have no denotations. A similarity is drawn with hormones, which function purely causally.

Interruption can be seen as an extreme form of modulation; a particular instance where there is one overriding strong emotion or need, and a corresponding clear transition in behaviour and goals for the agent. For instances where there is more than one competing emotion or need, or the emotions and needs are less strongly felt, then the same mechanisms can be seen as modulating the agent's goals and behaviour.

To suggest that emotions function solely as an interrupt is an example of electrocentrism[4], inherited from the field of computer science from which the quest for artificial intelligence developed. It is a viewpoint grounded in discrete binary logic rather than one that sees natural computation as consisting of complex and noisy dynamical systems interacting with one another to produce emergent phenomena.

2.6 Neuromodulation

One way of studying the functionality of emotions, is to identify the extra functionality provided by the mechanisms that give rise to emotions in natural agents compared to solutions that do not have those mechanisms. One such mechanism is neuromodulation.

2.6.1 Neuromodulation in natural agents

Fellous [Fel99] proposes that emotion can be seen as continuous patterns of neuromodulation of certain brain structures. It is argued that theories considering emotions to emanate from certain brain structures and from non-localised diffuse chemical processes should be integrated. Three brain structures are considered in this way: the hypothalamus, amygdala and prefrontal cortex.

Given that functionality implemented using neuromodulation can also be implemented using only neurons and synaptic connections, this leads to the question of what practical benefits are gained from using one over the other.

For example Rolls [Rol99] refers to how hunger and satiety are signalled by the body. The duodenum contains receptors sensitive to the chemical composition of food draining from the stomach. Receptors responsive to glucose contribute to satiety by passing signals via the vagus nerve to the brain. Fat

---

[4] A term coined by Brookes [Bro91a], see section 2.6.2 for further discussion on this.

infused in the duodenum can also produce satiety but this is likely to be signalled by the use of hormones. The candidate hormone for this is Leptin, or OB protein. The hypothesis being that this is responsible for longer term regulation of body weight and body fat rather than regulation on a meal-to-meal timescale.

Kelley [Kel05] argues that in their broadest possible sense, emotions are required for any organism or species to survive. They allow animals to satisfy needs and act more effectively within their environment. If robots are to survive as effectively then they also need equivalent systems. She writes:

> The basic premise . . . is that emotions are derived from complex neurochemically coded systems, structured by evolution, that are present in one form or another from single-celled bacteria to primates.

It is discussed how the function of a peptide can change throughout evolutionary history and between species. For example serotonin biases dominant behaviour in lobsters, whereas in humans it is known to modulate impulsiveness and aggression. Yet peptides can have many functions throughout the same brain. Serotonin is a good example of a peptide that fulfils many different roles and for which a wide variety of receptors have been evolved. Oxytocin is another example. Three ways that chemical compounds can be used are described. As synaptic transmitters, as modulators of postsynaptic receptors and as neurohormonal signals released and acting at sites distal from one another.

Koch [Koc99] pp93 describes neuromodulators as being the brain's closest equivalent to a global variable. In a computer program, a global variable can be read or changed from any part of the code. In the brain a neuromodulator can affect any neuron that has receptors for it within a certain distance of the release site. Receptors for a variety of neuromodulators and neurotransmitters can be found on most neurons. In effect this allows each neuron to be addressed with some degree of specificity using a combination of neuromodulators.

2.6.2 Neuromodulation in artificial agents

The Von Neumann legacy

In the classic paper [Bro91a], Brooks argues that the Von Neumann model of computation has traditionally led research in the field of artificial intelligence in particular directions, even though the architecture of intelligent biological systems is completely different. Even when researchers in the field of artificial intelligence look to neuroscience for answers, the brain is seen as an electrical machine with electrical inputs and outputs to the sensors and actuators of the body. Brooks argues against this:

> The brain is situated in a soup of hormones, that influences it in the strongest possible ways. It receives messages encoded hormonally, and sends messages so encoded throughout the body. Our electrocentrism, based on our electronic models of computation, has lead *(sic)* us to ignore these aspects in our informal models of neuroscience, but hormones play a strong, almost dominating, role in determination of behavior ...

Perhaps it is unsurprising that this fundamental characteristic of the brain is largely ignored when computational neuroscientists also talk about assemblies of synapses as circuits, likening them to assemblies of transistors and microcircuits performing logic operations in computers [She98].

Artificial hormones

There has been little work exploring the dynamics of neuromodulators in artificial agents. Shen et al [SSW00] describe a distributed control system for use in self-reconfigurable robots inspired by hormones in biological systems. Hormones are used as global signals that select and trigger actions in different subsystems. This is used for motion co-ordination and re-configuration. Execution of the selected action is left up to the subsystem. This reduces the cost of communicating globally. Hormones are described as messages that are broadcast rather than have a specific destination. They have a finite lifetime and can trigger different actions at the different sites where they are received.

Shen and Chuong later generalise the control system to produce the Digital Hormone Model [SC02]. This is a cellular automaton and reaction-diffusion model which matches the self-organisation of homogenous skin cells into feather buds. Living cells release activator hormones and inhibitor hormones. Released hormones diffuse throughout the cellular automata and can react with one another.

Gadanho [GH01] tests whether emotions can be used for a continuous state and behavioural transitions. This is achieved by having the influence of the sensory input upon the system of an agent persist until after the cause of that sensory input has disappeared from the immediate environment.

She uses a non-symbolic bottom-up animat approach, although her emotion model is not based on any biological mechanism. Instead she uses black box functions that are described as emotions, feelings and hormones. These functions emulate the primitive emotions of happiness, sadness, fear and anger; and the feelings of hunger, pain, restlessness, temperature, eating, smell, warmth and proximity.

Sensory inputs to the system are described as sensations. These are summed with the outputs of the hormones and are inputted to the feelings. The feelings connect to the emotions. If an output from the emotions reaches a threshold then it increases the associated hormone, which is then fed back to be summed with the current incoming sensations. A selection threshold is also used to determine which is the dominant emotion of the agent.

"Creatures" [GCM97] is a computer game in which characters learn about their environment, interact with the player, breed and die. A simple biochemistry is used, consisting of chemicals, emitters, reactions and receptors. Chemicals are implemented as a number between 0 and 255 and an associated strength increased by emitters. Chemicals can be transformed using reactions and are genetically specified. Chemical strengths are read by receptors so as to allow parts of the creature to become responsive to that chemical. This simulated biochemistry influences the operation of the underlying neural network, and is also used to simulate a simple metabolism and immune system. The biochemistry is used to control the flow of information within the creature, synaptic atrophy and migration, drive-reduction and reinforcement learning.

Cañamero has applied the hormone concept to help solve the action-selection problem [Cañ97]. She uses a bottom-up "nouvelle AI" approach [Bro91b]. She also argues that in order to understand emotions, the underlying biological mechanisms that give rise to them must also be understood along with how they evolved.

Recognising that different emotions evolved for different reasons in different parts of the brain, an architecture inspired by Minsky's Society-of-mind [Min88], in which an agent is made up of a society of subagents [5], is used. Each subagent fulfils the role of a basic emotion; that is, an emotion that is necessary for survival. The mix of concurrently active basic emotions leads to complex emotions.

In this architecture, emotions are seen as modifiers and modulators of the agent's behaviour and motivations. Each emotion is implemented as a different subagent. Emotions can release hormones at a strength corresponding to their level of activation. Motivations are seen as drives that maintain the homoeostasis of the agent's body. Somatic variables that need to be maintained within a certain homoeostatic range have corresponding internal sensors. Internal sensors have receptors for hormones so that the variable can be acted upon. As an example, insulin decreases blood sugar levels and glucagon increases it.

Taking into account Kelley's description of neuromodulators fulfilling different roles depending on where in the brain they are received, and Koch's analogy of a bar-code like system, it can be argued that hormones often have fewer functions than neuromodulators [6]

---

[5] The term "subagents" rather than "agents" is used here to avoid confusion with the single agent whose 'mind' hosts the society

[6] Because hormones modulate the response of neurons they are technically neuromodulators too. An artificial distinction is drawn here between hormones and neuromodulators to dis-

This is because hormones are transported via the bloodstream and ordinarily signal a change of body state. Hormones can therefore be seen as having a far more reaching effect than neuromodulators. Neuromodulators can be seen as having a more localised effect because of the possibility of multiple functions being performed depending on where they are released and received in the brain. Therefore with neuromodulators, it is more appropriate to see them purely as a signal and the modulation of neuronal behaviour as a function of the receptor.

Cañamero also uses a second group of hormones that are not released by emotional subagents but which modulate the arousal of the agent. Because this second group is not based on a change of body state, they would probably more likely be described within a real, or biologically plausible model of a brain as neuromodulators rather than hormones.

Cañamero's work can be seen as experiments in achieving more complex dynamics within an agent controller by the addition of simple mechanisms inspired by hormones. This is an experiment with practical engineering benefits rather than an academic model of specific biological mechanisms. For example, if modulation were to be used in an agent situated in a computer game, then considering how efficiency of computation and memory capacity is more important than realism, Cañamero's architecture would be ideal. Nonetheless, such work can also improve our understanding of the natural phenomena that it is inspired by.

When discussing the scientific status of the field of artificial life (AL), Noble [Nob97] argues that AL simulations cannot prove theories concerning the real world. They are not tools for empirical study as the results only apply to the artificial worlds that have been run.

AL models can however clarify the logical implications of a set of assumptions. The function of an AL model then is to establish the plausibility of a theory. This theory can then be referred back to the relevant empirical science

---

tinguish between neuromodulators that primarily signal changes in somatic states and those that are local only to the brain. However some neuromodulators fulfil both roles depending on their release site.

in order to be proven in the "real" domain. AL is an analytical tool for science in the same way that mathematics is. It determines whether the axioms that make up the model lead to the predicted outcomes but is indifferent to the truth of the axioms in any other environment. But if the result of an AL simulation is corroborated by empirical study of the real world, the simulation may help to explain why.

Noble draws a distinction between two classes of research efforts in the field of artificial life. One uses biology to inspire new methods for engineering, as Cañamero has done, and the other aims to improve our understanding of natural phenomena.

It is this latter class of work that Noble discusses. But the technical class of research can also overlap with the empirical. As part of the process of referring the findings back to the relevant empirical science, more complex and plausible models can be created to further test our conclusions concerning the phenomena. For example, modulation could be applied to agent controllers that are more biologically plausible so that not only do we have an increased understanding of the phenomena of modulation, but we can also test whether it applies to what we currently understand of the brain. This in turn might lead onto better engineering techniques.

Avila-García and Cañamero [AGC05] later apply the hormone concept to modulating the salience of percepts. They demonstrate how modulating the salience of what the agent perceives can alter its behaviour. An architecture reactive to internal and external stimuli is used to maintain the homoeostasis of internal resources. A biologically plausible model of hormonal modulation is used to directly modulate variables within the architecture that correspond to the agents perception. These include cues from the external environment, such as the perception of obstacles and predators, as well as internal cues such as the integrity of its bodily structure.

Although internal variables are directly modulated instead of biologically plausible neural networks, it is not hard to imagine how neuromodulators could be used for the same purpose in a natural agent. Neurons that respond to certain internal or external stimuli could be excited by increased levels of a neuromodulator. Although the model of modulation may be biologically plausible, it cannot automatically be assumed that the underlying architecture works in the same fashion as a real brain. Consequenty, it can neither be assumed that a brain will need the same functionality provided by modulation as the architecture of the agent. But experiments using biologically plausible neural networks can provide further evidence as to whether or not this is the case. If this appears to be so, then these experiments increase our understanding of another aspect of how the brain can possibly work.

Artificial neuromodulators local to a neural network

A more biologically plausible model is used by Husbands et al [HSJO98], [HSO$^+$98] and [Hus98]. They artificially evolve robot controllers built from networks inspired by the modulatory effects of freely diffusing gases in real neuronal networks. The models are particularly inspired by the modulatory gas nitric oxide. Results show that gas modulation considerably speeds up evolutionary runs to produce successful controllers.

The evolutionary approach in the field of artificial intelligence assumes that agent control systems are too hard to design and therefore must be evolved [HHC93]. Therefore the above use of modulation for artificial evolution is indifferent to how the robot controller works internally or how it adapts.

Reynolds [Rey94] describes a limitation to the use of artificial evolution for the generation of robot controllers. They evolve to use brittle strategies that depend upon the subtleties of the environment that the evolutionary run takes place in. The difficulty lies in creating the right environment for the agent to be evolved in. This requires exactly modelling the features that the agent will encounter in its destination environment and substituting other features that would be present in your model with noise. This is described by detail by Jakobi in [Jak98a] and [Jak98b].

The complexity of the GasNet model is increased in [PSHO99], and [PHO98]. Previously, two gases were used to modulate a single parameter in the neuron transfer function. One to increase the parameter and the other to decrease, the change being dependent upon the concentration of modulator. Four gases were used in [Hus98]. The modulatory effects were gas-specific. These gases are replaced by a single gas whose effect depends upon the site where it is received. This is more analogous to the way in which secondary messengers can determine the effect upon a neuron. To allow this there is a wider range of parameters that have the potential to be modulated. Gas diffusion and decay rates are also modelled in a more realistic way leading to interesting time delay and reservoir effects of the gas.

Neuromodulation has been used solely for evolving neural networks. French and Cañamero used neuromodulation in a Braitenberg vehicle controlled by a manually crafted artificial neural network consisting of eleven non-spiking neurons and twelve synapses. Neuromodulation is used to provide a robot with foraging behaviour.

2.6.3 Modulation emerging as a new definition of emotions

In section 2.5.3 the Interrupt theory of emotions was discussed with regard to how it was a particular, yet extreme, instance of modulation. So the idea of emotions fulfilling the function of modulating the agent's goals and behaviours is not entirely new, albeit not necessarily recognised by proponents of the Interrupt theory of emotions. But there is now an increasing body of recent research that has embraced the idea that emotions modulate the behaviour, needs and goals of the agent. Whether this becomes the predominantly attributed role of emotions, or whether it becomes yet another avenue of ideas that is researched and bears influence upon the future course of emotions research remains to be seen. This shall be largely determined by how widely the function of modulation can be attributed to being provided by the heterogenous set of emotions; each one having been evolved because of a distinct and specific need.

Both emotions and neurochemicals are fundamental to how living organisms have evolved. They are also fundamental to how they operate within rich and unknown environments. It is well known that a change in the level of a neurochemical can have a significant effect upon the emotional state of a living organism. In order to understand what emotions are, we first need to understand the functionality that emotions provide. One way of doing this is to understand the functionality that neuromodulation provides to an agent.

2.7 Summary

This chapter started by explaining that each emotion has evolved to satisfy a particular need or to guide living organisms in particular situations. The discussion of how emotions, or a mechanism similar to them, are relevant to an agent satisfying needs within a rich and unknown environment has been explored. Emotions are mechanisms that are fundamental to many aspects of both natural and artificial agents. Research in this area has relevance to our understanding of consciousness, rationality and helps in creating virtual realities.

Recognising the importance of emotions in creating adaptive agents, the chapter then moved onto the issue of what emotions actually are. There is an assumption that in order to create emotions you first need to know what they are. The problem of how to define an emotion has plagued the literature and different solutions to this issue have been proposed. None have been unanimously agreed upon.

Because of this there have been concerns that the wrong question is being asked. We should instead be increasing our understanding of the functionality that emotions provide. Taking this further, I have proposed that not only is the wrong question being asked but that creating definitions before we understand what we are trying to define leads to a mire of semantics and is a distraction from our original intention. Definitions are nothing more than a tool, not an end in themselves. If we increase our understanding of the domain then we are more likely to decide upon many useful definitions that map the space of functionality provided by emotions. This is in contrast to current efforts to apply one precise definition to the whole of the domain.

The chapter then discussed the functions that have been attributed to emotions and the different theories regarding how they work. In the literature, emotions have been reasoned as being body-centric, environment-centric and also attributed the function of being an interrupt mechanism. Problems with these theories were also discussed. The chapter ends by reviewing a new area in emotions research, emotions as a form of modulation of an agents behaviour, needs and goals. In the same way that emotions are fundamental to how natural agents have evolved, chemical and neuro- modulation is also fundamental to how they operate. This requires a significant shift in thinking compared to the mathematical and logical heritage of computer science and mathematics that has allowed us to so far reason about complex systems.

Chapter 3

Tools and procedures

As explained in the previous chapter, emotions and neuromodulation are fundamentally linked. To understand the functionality provided by emotions we need to understand the functionality provided by neuromodulation. In order to do this we first need to isolate the effects of neuromodulation.

3.1 Requirements

3.1.1 The agent approach

We are trying to understand why natural adaptive agents have emotions. It is impractical, if even possible, to implement in an artificial system the complexity of real-world natural agent-environment interaction[1]. Doing so achieves little while we are still postulating the functionality of emotions. An agent approach allows us to model and observe the interaction of properties that we consider fundamental to a phenomenon. These properties can be used to test the logical implications of what we postulate.

Because adaptation can take so many forms the tasks set for the agents need to be well-defined. This allows us to be specific about the functionality provided by neuromodulation as well as its costs and limitations. The simpler the agent

---

[1] Animals can be observed in the wild if such complexity is required.

and task that it is set, the easier it will be to understand any functionality that we discern. The simplest tests may not by themselves reveal the full nature of neuromodulation but they do provide a good starting point from which to make further, more sophisticated investigations.

3.1.2 Discerning the functionality of neuromodulation

In order to isolate the functionality that is provided by neuromodulation, we must compare how an agent adapts using modulation with one that adapts without it. Sheutz advocates this approach to systematically determine the utility of emotions in artificial agents but does not apply it to biologically plausible models [She04].

Modulation must not be seen as an external property that can be applied to an agent. Rather, the use of modulation must be fundamental to how the agent adapts if the mechanism and the functionality that it provides is to be used to its fullest extent. Comparing the performance before and after removing a mechanism intrinsic to how an agent adapts will inevitably show the mechanism to be useful. This by itself provides little information if we already know that the mechanism is relied upon for adaptation. We need to compare two agents, one that is designed to use modulation and the other designed to adapt without the use of modulation. Comparisons between two such agents will be meaningful.

There is a fundamental difficulty in comparing two agents that are otherwise equivalent except for the way in which they adapt. How do we test for equivalence? If they are equivalent because of their performance levels, how else do we compare the two agents? Comparing the behaviour of two agents leads to the difficult task of quantifying how they act. Or perhaps the use or otherwise of modulation stops the two agents performing equally well in which case how can we say that they are equivalent?

If we use a working definition of intelligence as being the ability to adapt to an unknown environment[2], then the agents can be compared by testing them

---

[2] If an agent never has to adapt to an unknown environment then it can perform equally

in an environment different to that for which they were designed. This allows us to test the scope, scalability and robustness of any adaptive mechanism.

3.1.3 Requirements of an agent

There are a number of basic requirements that agents must commonly satisfy. They are listed here in order of importance. The latter requirements often help to fulfil the former ones, if they are applicable.

- Maintain homoeostasis and keep replete the critical resources.
- Adapt to an external environment and learn the value of properties within it.
- Learn and adapt to the temporal properties of an environment.

If an agent fails to maintain its critical resources then it will effectively die and not be able to satisfy any other requirements[3]. Adapting to an external environment aids in maintaining homoeostasis and resource levels, and in all but the simplest environments is required to keep the agent alive[4]. If an agent is to be anything other than a complex set of sensorimotor associations then it needs to learn cause and effect within its environment. This is required if the agent is to prepare for future situations before they occur, or to deliberately move into desirable states of the environment other than by chance. The environment in which an agent is placed will determine how well the model of adaptation is tested[5].

---

well, or better, by merely following hard coded rules. It is difficult to argue that such an agent is acting intelligently.

[3] In some agents this requirement is not applicable, for example, because it is in a virtual environment or because it has a guaranteed or non-rechargeable power source.

[4] Adapting to an external environment is taken here to include learning to sense and act within that environment. Because doing this concerns different parts of the brain to that of action selection, and because such functionality requires testing in a rich, preferably real-world environment, this is beyond the scope of this thesis.

[5] The term environment here is used in its loosest sense to include all uses that an agent can be put to.

3.1.4 How the agents adapt

The underlying architecture

There are a myriad examples existing in the natural world of emotional and intelligent agents using neuromodulation as an adaptive mechanism. There are two possible reasons why the brain might have evolved to consist of neurons, synapses, neurotransmitters, neuromodulators etc. Either because this form makes the most effective and efficient use of the material that is available, or because they allow for the most effective adaptive techniques to be performed.

Computers use silicon circuit boards and most commonly implement the Von Neumann architecture. The most effective and efficient model of adaptation for a computer may be unlike any form found in wetware. But increasing our understanding of the functionality being performed by the brain will eventually allow us to translate its fundamental adaptive properties into a computational model most suitable for the computer architectures that we currently use.

The choice of underlying model to be modulated can affect the functionality, costs and benefits that come from using modulation. Modulating a model of adaptation that has no known equivalent in wetware will tell us little about the logical implications of any theory regarding emotions in natural agents. Therefore the agents implemented here adapt using biologically plausible neural networks.

As with most simulations and artificial life experiments, there is a compromise between resources[6] and the complexity of the model to be implemented. The model must capture all the properties of the natural system that allow for the desired functionality to occur, while at the same time abstracting over properties that add little or nothing to our understanding of the phenomenon being modelled.

---

[6] i.e. implementation effort, processing power and capacity of the computer.

Biologically plausible use of the model

Not only must the model be biologically plausible, but so too must the way in which it is used. Applying where possible the same requirements and constraints as faced by adaptive models in nature will allow us to more greatly appreciate the method by which the functionality is provided and the benefits afforded by its use.

Therefore it was known from the outset that supervised learning techniques were not adequate. This meant that the agent would adapt using unsupervised learning techniques. But this by itself does not necessarily mean that the agent must use a self-organising system.

Reinforcement learning or a self-organising system

When introducing the topic of Reinforcement learning, Sutton and Barto [SB98] describe how an infant learns by interacting with its environment; by observation and play. The infant has a sensorimotor connection to its environment rather than to an *explicit* teacher. Reinforcement learning is described as being different from supervised learning because the latter entails learning from examples provided by an external teacher.

It is commonly known that animals can be trained using reinforcement learning. Thorndike's 'Law of effect' describes the effect of reinforcement on animal action selection [Tho11]. The classic experiments by Pavlov [Pav27] show us that temporal sequence learning exists in natural agents, that they can learn cause-and-effect and associate environmental cues preceding significant occurrences. But the biological mechanisms of reinforcement learning are still not fully understood. For example, there is debate as to whether dopamine is used as a reinforcement signal in the brain, or as a preparatory 'Go' signal. Rolls [Rol99] chapter 6 provides an overview of the supporting evidence for both sides of the argument.

The efficacy of reinforcement learning in training animals is by itself insufficient evidence to infer that this is the sole or primary means by which animals learn. Kaelbling et al [KLM96] conclude their overview of reinforcement learning by remarking on the scalability issues that affect the majority of the presented algorithms. These issues arise because of the difficulty of solving arbitrary problems using a generalisable technique.

Although reinforcement learning may not require an explicit supervisor external to the agent, some part of the agent must act as an *implicit* supervisor in order to determine the utility of the environmental state and to produce a reward or error signal. This signal cannot come directly from the environment but must be deduced from it. This sentiment was first expressed by Klopf [Klo88] with regard to Temporal Difference learning. Natural autonomous and adaptive agents cannot normally rely on evaluative signals from the environment.

Regardless of the environment, an animal has constant needs. It must eat, drink, breed and survive. Instincts and biophysical mechanisms have evolved to recognise when those needs are satisfied. The failure to satisfy these needs can form the largest determinant of the evolutionary fitness of an agent. Instincts, the need for play, excitement, love etc, have all evolved because they directly or indirectly satisfy our base needs. In terms of evolutionary fitness, the utility of all acts can ultimately be determined according to how well they allow an agent to satisfy its base needs[7].

An explicit reward signal that is derived solely from how well a base need or goal is likely to be satisfied will be reduced for acts less directly involved with the satisfaction of that need. This by itself is a limiting factor in how well such a model can scale. But some natural agents, at least with humans, can find intermediate goals and needs more attractive than their base needs; sometimes even foregoing them as a consequence. Can reinforcement learning model such phenomena? Or can it be better understood as an attractor in a dynamical system? If an internal reinforcement signal is present in an agent with only environmental and somatic stimuli as input and choice of motor action as

---

[7] For example, obtaining money is essentially a worthless act, except when used as part of a monetary system whereby it becomes a very desirable act because of its effectiveness at satisfying many base needs. But if you give a dog some cash it will probably try to eat it.

output, are we just labelling it as such? The reinforcement signal can be seen as merely part of a larger self-organising system. As Rolls suggests when weighing up the argument about a functional role of dopamine [Rol99] pp178:

> ...the dopamine projection to the striatum may act as a 'Go' or 'preparation' signal to set the thresholds of neurons in the striatum, and / or as a general modulatory signal that could help to strengthen synapses of conjunctively active pre- and postsynaptic neurons. In such a system, what is learned would be dependent on the presynaptic firing of all the input axons and the postsynaptic activation of the neuron, and would not be explicitely guided by a reinforce / teacher signal that would provide feedback *after* [his emphasis] each trial on the degree of success of each trial as in the reinforcement learning algorithm.

Rolls also suggests a similar role for dopamine in the basal ganglia in which it is used to set the learning threshold, pp197. The functioning of the brain can be understood as a process of self-organisation, Kelso [Kel95].

Although it is important for the underlying model of adaptation to be biologically plausible, the primary concern here is with the functionality provided by modulation. An in-depth comparison between reinforcement learning techniques and self-organising systems is out of the scope of this thesis. Therefore both models of adaptation were tried and the agent environments were created with this in mind.

3.2 The initial test environment

The objective of any experiment performed here is to isolate the effects of neuromodulation in order to discover its uses and costs. The intention is to increase our understanding of the nature of emotions and its relationship with neuromodulation. This will be achieved by comparing one agent that adapts with the use of modulation, with another that adapts without. In essence this is the equivalent of asking whether apples are better than oranges. The proposed solution is to compare the agents in environments, and under constraints, that they were not optimised for.

A decision needed to be made as to how complex and realistic the adaptive models needed to be. The choices were:

- Non-embodied neural networks.
- Embodied neural networks.
- Agents in an artificial and virtual environment.
- Agents in a physical environment.
- Agents in a virtual reality environment.

Comparing neural networks without the need to maintain the resources of a body allows their application to a wide range of non-biologically plausible uses. But this tells us little about the nature of emotions.

Section 3.1.3 listed some basic requirements commonly asked of an agent. These were to maintain homoeostasis and to adapt to both the immediate and temporal properties of an external environment. Constraining adaptation to tasks more commonly demanded of wetware requires the networks to first be able to maintain the resources of a body. This by itself does not require the use of an external environment inasmuch as a body, sensors and actions to read and change the state of the body are external to the neural network.

Once this is achieved an agent approach can used to apply further constraints. This allows for the possibilities of testing in a physical, artificial or virtually real environment. Choosing to situate an adaptive agent in a physical environment is more costly and creates many problems that require solving before any comparisons can be made[8]. While this is an important and necessary endeavour for more proven adaptive algorithms, it is particularly risky when it

---

[8] Problems such as learning to sense and perform actions within the environment etc.

is not known from the outset whether adaptation will occur at all using modulation. Situating an adaptive algorithm in a physical environment also carries an increased cost in time, effort and hardware which can be better spent checking whether or not the algorithm adapts in the first place. The added complexity involved in adapting to a physical environment means that it is harder to isolate and understand any functionality provided by modulation.

This left two options. To implement in either a simple artificial and virtual environment such as a grid world, or to use a more complex virtual environment such as a virtual reality model or simulation of a physical environment. It was decided that all the reasons that apply to implementing in a physical environment also apply, albeit to a far lesser extent, to implementing in a simulation of a physical environment. The first step was to develop the modulating and non-modulating neural networks, and *then* find interesting comparisons.

3.2.1 Grid worlds

The term 'grid world' is commonly used to describe a simple environment in which adaptive agents are situated. The world forms a grid, normally toroidal to avoid edges. It is made up of discrete squares in which an agent can be placed. Because of the simplicity of the environment, the agents are normally given magic actions and magic sensors[9]. Grid worlds normally run in discrete rather than continuous time and this removes some temporal problems for the agent, such as when to choose a new action.

History of grid worlds

Grid worlds were initially used in the field of artificial life to demonstrate basic principles of single-life adaptation and artificial evolution for agents, self organisation using cellular automata, group social dynamics and language acquisition of multiple agents. While useful for pedagogical reasons, these worlds

---

[9] The functionality of sensors and actions is implemented externally to the agent, therefore from the agent's perspective they happen by 'magic'.

will always be 'toy' applications. Due to their simplicity, the agents evolved or adapted within them are not transferable to a more complex environment.

Yaeger [Yae94] adds complexity to the grid world format in the form of PolyWorld. He attempts to emulate all the principle components of real, living systems and to bring them together in a single artificial environment. Rather than aiming to transfer agents to a more complex environment, Yaegar claims that PolyWorld may serve as a tool for further research into evolutionary biology, behavioural ecology, ecology and neurophysiology.

Tools such as evolvable hardware and Khepera robots were later used to allow adaptation and evolution to occur in a real world environment. This made the limited use of grid worlds, even complex ones, less attractive. The relative ease with which an agent could be made to adapt within these simplistic environments meant that if work was based solely on a grid world then it made for a less convincing argument.

The advantages of using an initially simple environment

It is argued here that, while of limited use, the simplicity of grid worlds is still useful in modern research for the purpose of prototyping.

- If a new idea or technique to solve the action-selection problem cannot work in a simple environment then it is almost certain not to work in a more complex target environment.
- A simple environment with fewer parameters is more easily understood by the designer than a real-world environment, and more easily adapted to by the agent. If an agent is initially developed for a complex target environment and it does not adapt, then there are more parameters that need to be looked at and controlled before it is understood why adaptation does not occur.
- Behaviour that arises from a complex environment can be an emergent phenomenon, which can be better understood when the agent has previ-

ously been seen acting in a simpler environment.

- Because an agent adapts to an environment, the environment influences the behaviour of the agent. Therefore finding the right environment to adapt to can unambiguously demonstrate the validity of a new idea. A grid world can be used to prototype the environment to adapt to.

- The mechanisms and variables that are fundamental to some emergent phenomenon must be isolated from any extraneous features if the phenomenon is to be truly understood. It is more practical to start with simple systems and to add complexity than to start with more complex systems that then need to be simplified.

A simple grid world is inadequate for artificially evolving an agent intended for a complex target environment, or for exploring techniques of complex sensory interpretation or motor co-ordination. They can be useful to understand abstract intelligent functions such as action selection or emergent phenomena such as social group dynamics.

Physical robots, such as Khepera robots, in simple environments may be used instead of grid worlds for research into abstract intelligent functions and social group dynamics. They may seem more impressive but they are still placed in abstract and simplified environments and may be given semi-magic actions. Keijzer [Kei98] discusses how the use of wheeled robots still side-steps a core problem of understanding adaptive behaviour due to a tendency to simplify the sensory-motor control of robots.

One major difference between virtual environments and simple physical ones containing wheeled agents is that the robot is acting within the environment in continuous rather than discrete time. But robotic agents are still digital and still perform binary logic. Therefore they are still, at their core, discrete. They sample the world at a finer granularity than agents in a grid world with magic sensors and actions. Software environments can be implemented to be more realistic so as to emulate this finer granularity.

There is one fundamental difference in that even though the agent is discretised, real environments are not. But from the agent's perspective, this difference equates to uncertainty and noise. Noise can be artificially added to the senses and actions of an agent in a virtual environment, or the environment can be modelled to be more realistic beyond a level that an agent can sense. Taken to its logical conclusion this leads on to the field of Virtual Reality. Such environments can be used for second stage prototyping.

3.2.2 HelloWorld

HelloWorld is a framework to create environments designed for the initial testing of adaptive agents. The key features of HelloWorld are *simplicity* and *expandability*. It allows the creation of environments intended to prototype new ideas that might help solve the action-selection problem before greater effort is expended in applying them to a more complex target environment.

It was written from the very beginning not to solve one particular problem but to be expandable. The motivation was to quickly and easily create environments and to rapidly prototype new ideas. It is not intended to be used as a substitute for the final adaptation in a complex target environment.

The aims of HelloWorld are to:

- Rapidly and initially test new ideas or techniques that require high level and abstracted sensor input and motor output.
- More easily deduce the key features that are required for a particular emergent phenomenon or adaptation to occur.
- Allow for greater understanding of these key features through visualisation and parameter manipulation.
- Allow for the initial searching of high level system parameters before adaptation takes place in a more complex environment.

3.2.3 The environment of agents with external senses and actions

In HelloWorld an object can either be an entity or an agent, the difference being that an entity does not sense the outside world. Everything within a HelloWorld environment has three attributes; energy, water and density. These are merely labels for scalar values and are not meant to model any real-world complexity[10]. Everything in HelloWorld has these three attributes, whether or not they adapt. Setting a high density is useful when creating entities that are not meant to be eaten. Energy and water levels are useful as resources that the agents must replenish. Two resources are provided so that independent needs can be intelligently balanced. The standard agent body constantly loses energy and water over time through the actions the agent undertakes. When one of these levels drops to zero the agent dies. An agent can choose an action every turn. If it fails to do so then the default inactive action is chosen for it. This carries the maximum cost. See section 3.2.5 for further details about how these attributes are used within the environment.

3.2.4 Requirements of the environment

So as to be rich enough to test for non-trivial adaptation rather than mere sensor-motor association, it was decided that an initial test environment would need to meet the requirements set out below. If adaptation does not make full use of the opportunities that the environment provides then this shows the limit of the adaptive technique in its current form. This knowledge can help decide the focus of future research. Therefore it is useful to have an environment slightly richer than that expected to be required.

- There must be multiple contexts for the objects within the environment so that the agent can learn to react to them in different ways according to its needs.

---

[10] Labels such as this shall not be written in quotes for the sake of intelligibility but they will always be explicitely defined when first introduced.

- Intermediate states must exist that allow an agent to gain access to many more directly beneficial states. An intelligent agent can then distinguish these from intermediate states with low utility.
- The environment must allow the opportunity for an agent to learn a specific sequence of actions, that when performed in certain states, can lead to a long term increase in resources. This demonstrates temporal action selection.

3.2.5 Description of environment

The environment proposed for comparing modulating and non-modulating agents will consist of three types of objects. They shall be referred to here as 'food', 'rock' and the agent itself. The agents shall be tested alone within the environment. The grid world used in this research is 30 x 30 squares and is toroidal.

Food

The term 'food' is used here to describe an entity that an agent is able to gain energy and water from given that the correct actions are used at the correct time. Food has a low density level.

The food entity cannot act and does not spread or produce more food. It has constantly increasing energy and water levels which saturate at a point, after which they will no longer increase. If either level is decreased below a threshold, in this environment by an agent performing an eat or drink action on it, then no more of that resource shall be gained from that entity until the level has climbed back past the threshold again. The food entity waits for a period of time before starting to increase its resource level again. This means that it is more beneficial for the agent to search for food elsewhere rather than subsist upon a safe bet by continually eating or drinking from the same depleted food entity.

Because a food entity can release either energy or water according to whether an eat or drink action is performed on it, this allows the agent to react to its environment in different ways according to its current needs. If an agent is hungry but not thirsty and the food in front of it will only release water, then if the agent moves to another food object to eat then this demonstrates the ability to arbitrate between needs. If the agent is hungry but not dangerously so, and is also somewhat thirsty and drinks from the food in front of it, then this demonstrates that the agent is responding to its environment intelligently.

An eat or drink action performed on the food shall release a percentage of that food entity's available resource for the agent. This means that if an agent moves elsewhere in search of new food that has not been eaten for a long period of time, then it will probably benefit more in the long run. This satisfies the third requirement of the environment that the agent has the opportunity to learn a specific sequence of actions for certain states to increase resources in the long term.

Rocks

The term 'rock' is used to describe an entity that has a high density and low energy and water levels. Surrounding each food object, except for a single square which will allow access for an agent, will be a wall of rock objects, see figure 3.1. This ordinarily hides the food from the view of the agent but does allow the agent the chance to initially find some food via a random walk. This affords the agent the chance to learn to move into indirectly beneficial states that can lead to multiple directly beneficial states. Learning this will result in behaviour such as moving towards rocks, then following them round until locating the entrance to the food.

3.3 The agent

The agent is entirely software based and is made up of a body, actions, senses and a neural network, see figure 3.2. The body is used to simulate an internal

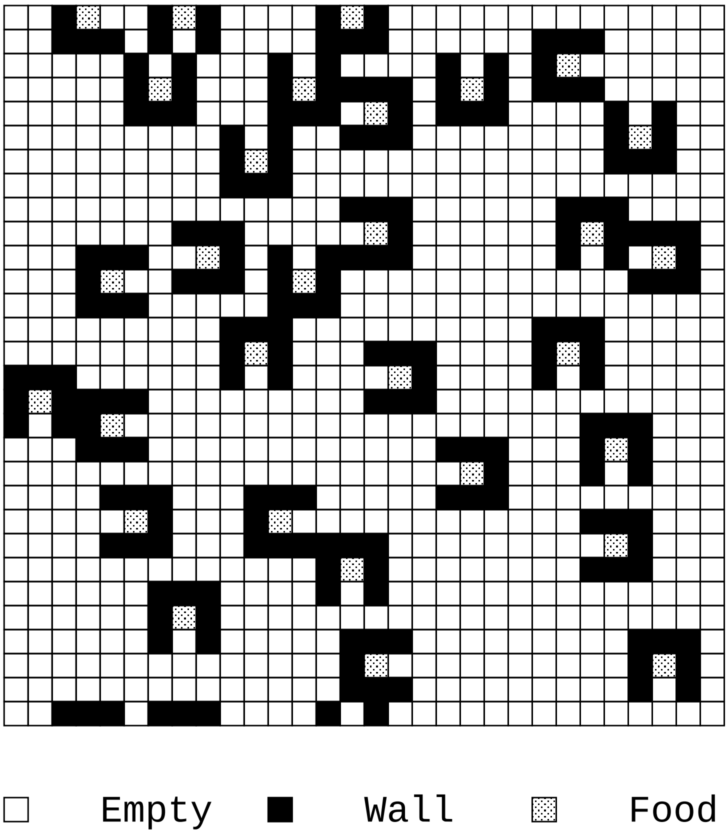


Figure 3.1: The grid world is a toroidal grid of squares. Each square can have only one object in it at any one time. Black squares consist of 'rock' entities which are inedible. They hide 'food' entities from the view of an agent (grey squares). White squares are empty.

state that needs to be maintained in order for the agent to survive. The neural network decides which actions to call based upon the inputs it receives through its senses and body. Actions directly alter the body state and can also change the environment if the agent is situated within one. If this is the case then the actions can also indirectly change what the agent senses in the next cycle.

3.3.1 Body

An adaptive agent needs a reason to adapt in order to do so. A common reason is to maximise and retain resources. In this context a resource is a single continuous scalar value that can correlate with a characteristic of the state of

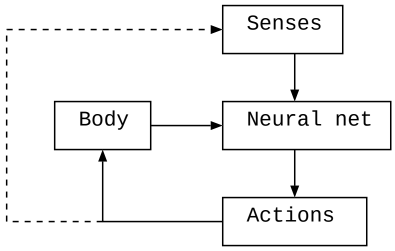


Figure 3.2: The neural network takes as input signals from the body and senses and performs one action. This directly alters the body state and the environment of the agent if it is situated within one. Actions can therefore indirectly alter the senses of the agent.

the agent or environment. A resource can correlate with a single quantifiable level such as the charge of a battery for a physical robot, or be an estimation of a virtual non-measurable level such as utility or safety. An adaptive agent is faced with two tasks when maximising these resources, that of learning to perform actions which result in an increase in a resource level and that of learning *not* to perform actions which result in a decrease of resource.

The agent has a body that requires two resources, energy and water. It keeps track of the largest increase and decrease of each resource. The current change in resource level is then scaled to these maximums to be within the range [0,1] before being passed to an adaptive controller.

3.3.2 Actions

In order to change its body state, an agent needs to trigger an action. These are 'magic' actions, as defined in footnote 9. In any network topology, there will be an output layer of neurons. Each neuron in that layer corresponds to a different action. The action corresponding to the winning neuron in that layer is then performed.

The agent uses a feed-forward neural network of spiking neurons. The network can be iterated over a number of times within a single turn. The number of iterations is determined by evolution and is constrained below a certain specified maximum. Hard coding the network to only have a single iteration would involve making an a-priori assumption about how it would best adapt.

After iterating through the network, the winning output neuron is chosen. Which neuron wins is determined by summing up the total activation of each neuron over all the iterations and choosing the neuron with the greatest sum. This stops a neuron with strong inputs from losing because it has just spiked and thus has low activity or is in a refractory period.

Situated agents

What actions were provided to an agent depended on whether it was evolved to be situated within a grid-world environment, or whether it was evolved to adapt to internal sensory input only.

If the agent was evolved to be situated within an environment, then the actions carried a cost in bodily resources. This meant that an agent needed to adapt in order to replenish its resources and survive. This cost is carried regardless of whether the net effect of the action benefitted the agent's body state.

- Do nothing
- Move (one cell forward that the agent is facing)
- Turn right
- Turn left
- Consume energy
- Consume water

Non-situated agent

Agents that were *not* evolved to be situated within an environment were given actions that had an immediate effect on either the energy or water resource level of the agent's body. Actions either increased or decreased the resource by a one, two or zero points.

- -2 Energy
- -1 Energy
- +0 Energy
- +1 Energy
- +2 Energy
- -2 Water
- -1 Water
- +0 Water
- +1 Water
- +2 Water

3.3.3 Sensors

If the agent is situated within an environment then it is given a sensor. The sensor used in this research is uni-directional and the agent 'view' spreads much like the light from a torch, see figure 3.3. If a square is inhabited by an entity then information regarding anything that inhabits the squares behind this is filtered out by the sensor.

3.3.4 The neural network

The agent adapts using a feed forward network of adaptive leaky integrate-and-fire neurons based on the model described in [WDKvE89] and [Koc99] pp339.

Each neural network is made up of three distinct layers; input, middle and output layer. Where an agent has two or three input layers (hunger, thirst and sometimes ian external stimuli), this is no different to grouping all the neurons into a single input layer. See figure 4.1 for an example of a neural network.

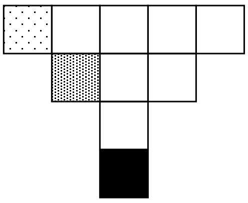


Figure 3.3: Agents that are evolved to adapt to a grid-world environment are given a uni-directional sensor. This sensor provides information to the agent regarding the contents of the cells it can sense. In the example above, the black square is inhabited by the agent that is using the sensor and is shown here only to show where the agent is relative to what it is sensing. The dark grey square is inhabited by another entity viewed by the agent. The light grey square behind it is obscured from the view of the agent and information regarding this square is filtered out by the sensor before being passed to the agent.

Spiking neurons were used in the neural network, each one acting as a capacitor to integrate and contain the charge delivered by synaptic input. This charge slowly leaks away over time. The neurons have a fixed voltage threshold and base leakage which are genetically determined.

The neurons also have an adaptive leakage to account for how frequently they have recently spiked. If a neuron spikes then its leakage is increased by a genetically determined amount. If the neuron does not spike then the leakage is decreased by that same amount [11]. Resistance is constrained within the range [0, 1].

The spiking threshold is the same for all neurons in the network and is constant. The neurons are stochastic so that once the spiking threshold has been reached, there is a random chance that a spike will be transmitted along the output weights. Either way the cell loses its activation[12].

The neurons send out a stereotypical spike. This is implemented as having a binary output. The weights connecting the neurons are constrained within the range [0, 1]. The learning rule employed uses spike timing-dependent plasticity

---

[11] It was not known whether separate increment and decrement parameters were required. To keep the number of attributes to a minimum it was decided that two parameters would be used only if it was found to be required. The network evolved well with only one parameter.

[12] This model of stochastic firing is a simplified one. In real neurons there is both both a possibility that a spike is passed down the axon to the target cells and a possibility of vesicle release once a spike has reached a synapse.

[BW02] [13].

3.3.5 Spike timing-dependent plasticity (STDP)

STDP rules increase or decrease weights according to the order of pre- and post-synaptic spikes. The timing of the spike dictates the change that takes place. Weight increase occurs if the spiking of a post-synaptic neuron is preceded by the spiking of a pre-synaptic neuron. Decrease occurs if the order is reversed[14]. As described by Bi and Wang [BW02], this is close to the original postulation of Hebb that the synapse from cell A to B is strengthened only if A takes part in firing B.

Izhikevich et al [ID03] relate STDP learning to a standard implementation of the LTP / LTD learning rule called the BCM formulation (Bienenstock-Cooper-Munro). Rather than update the synaptic weight due to the timing of single spikes, BCM compares spike-trains. If pre-synaptic input results in high post-synaptic firing rates then the synaptic weight increases. Input that results in low firing rates decrease the synaptic weight.

Properties

Some STDP rules such as those modelled by Turrigano and Rossum [MCWvR00] are based on experimental evidence that the amount of change also depends on the initial synaptic size. They observe that strong synapses are potentiated relatively less than weak ones but that depression is independent of synaptic strength. They also make the assumption that only the first synaptic event after a given spike depresses the synapse. Depression does not result from further synaptic events until a postsynaptic spike occurs.

Although it is useful to understand how such rules can be used, it is also

---

[13] It was unknown from the outset which hebbian learning algorithm to use. Evolutionary runs initially consisted of a choice of different algorithms. At the time the agents were not adapting and it was felt that the choice was reducing the likelihood of convergence and subsequent optimisation. It was decided that one algorithm that could be acted upon by evolution in a number of ways would be better used throughout the population.

[14] This order is set by a genetically encoded parameter so that it is possible for networks to be evolved that use anti-hebbian learning rules.

useful to understand what can be done without such constraints. This may in future help provide clues as to why these rules are constrained in such ways. The rule employed in this model is relatively free of such constraints. It is not based upon the current synaptic size, whether for depression or potentiation, but solely on the timing of the spiking of the pre- and post-synaptic neurons. Nor does it involve activity-dependent scaling [MCWvR00], -adjustment of the synaptic weights used to regulate post-synaptic activity.

Implementation

The STDP rule used here is implemented using a two-coincidence-detector model [KNB02] [KB02] based on [SMA00] and later evolved for use in robots by [DP03]. Each neuron has its own post-synaptic recording function that is incremented when the neuron spikes and which decays over time in-between spikes. This is compared to the pre-synaptic recording function of the neuron that has transmitted the activation. Each layer of neurons has its own increment and decay rates determined prior to testing via automated parameter optimisation.

3.3.6 Modulation

The network can be influenced by modulators. A modulator is a global signal that can influence the behaviour of a neuron if that neuron has receptors for it. The signal decays over time, as specified by the re-uptake rate, and can be increased by firing neurons that have secretors for it.

Which receptors the neurons have is stochastically determined according to genetically set parameters. Depending on whether it is excitatory or inhibitory, a receptor can accentuate or attenuate the total input of a neuron, or increase or decrease the probability of an activation being transmitted when a neuron is fired. Whether the receptor modulates the input or probability of transmitting an activation is genetically determined.

3.4 The evolutionary mechanisms

3.4.1 The need for automated optimisation

A complex neural network has a number of parameters which must be set correctly for it to adapt successfully. These are parameters that have no obvious value to which they should be set. When manually designing a neural network architecture, it can be difficult to know whether the new design is an improvement upon another design unless it is also known whether both networks are performing at maximum efficacy.

Hooker [Hoo96] argues against the competitive testing of algorithms. One of the many issues raised is the problem of how much to tune the parameter set of each algorithm. Rather than manually adjust the set of parameters until the network performs as intended, it was decided to automate the search using evolutionary techniques. Applying the same automated approach to different versions of the same network to optimise the parameter sets allows all versions to be compared fairly. Meaningful comparisons can be made as a result. Another requirement for automating the search was due to the large number of parameters that would otherwise need to be manually adjusted when time could be better spent evaluating and analysing the performance of the model.

The downside of automating the optimisation of the parameter set is that all the disadvantages of artificial evolution can also occur if the agent is too configurable. For example, the search may settle upon a solution that relies upon unforeseen characteristics of the fitness function or test procedure.

The alternative to designing a neural network architecture and automating the optimisation of the parameter set is to rely solely upon unconstrained artificial evolution. If successful, then this approach provides the architecture of the neural network and its optimal parameter set. The disadvantage of this approach is that the resulting neural network can be too large and complex to easily understand.

If modulating and non-modulating versions of an agent controller were to be meaningfully compared so as to increase our understanding of the functionality provided by modulation, then the agent controller had to be fully understood. This is frequently not the case with agent controllers whose design is the result of artificial evolution. If we fully understand how an agent controller works and we see some new functionality being provided by modulation, then we have a chance of understanding how that new functionality came about. This increases our understanding of the possible roles of neuromodulation in real brains.

The design process used in this thesis was a mixture of the two. The architecture was manually designed and the parameter set optimised using artificial evolution. Automated optimisation was constrained so as not to alter the original architecture. If the neural network performed poorly then an alternative design was tested.

Alongside this, unconstrained artificial evolution was used as a form of exploration only. It helped determine what could possibly be achieved with the underlying neural building blocks and how the agent could function. The agents resulting from unconstrained evolution were not otherwise used. All the agents tested in this thesis were manually designed and optimised using constrained artificial evolution.

3.4.2 Methods of evolutionary adaptation in HelloWorld

Artificial evolution is used to search the space of parameters for the most effective set of values. This is then hard-coded into a single agent so that comparisons can be made fairly between modulating and non-modulating agents. There are three methods of evolutionary adaptation within the HelloWorld framework. These are listed in the order in which they were developed:

- Endogenous evolution.
- Genetic algorithm using an explicit fitness function.
- Mutation-based online evolution.

Endogenous evolution

The original method was by use of endogenous evolution without the use of an explicit fitness function. Although it was not expected to be useful for parameter optimisation *per se*, it had previously been used for open-ended evolution and had demonstrated that solutions could be developed.

Evolution occurs when a breeding action is given to the agents. The agents that do not initially choose to breed do not pass on their genes. The only genes that survive are those of the agents who attempt to breed and survive using other actions. The symptom of artificial evolution successfully starting in the first few iterations of a world is a population that grows massively, reduces to a minimum and then gradually proceeds to grow.

Any fitness function present is therefore implicit in the configuration of the environment and the actions given to the agents. Although useful for toy and pedagogical applications this form of evolution can only be used for spatial environments. It is also less easy to tailor an environment to produce a specific behaviour or solution than it is to hard-code an explicit fitness function.

Because no explicit fitness function is present, the following information is not available to an agent developer:

- Fitness of the population.
- How well evolution is proceeding.
- When to stop the evolutionary run.
- Which is the fittest solution within the population.
- How often the fittest solution has been tested.

If the environment is a dynamic one, which it needs to be if the adaptation is to be interesting, then an implicit fitness function will change throughout the course of an evolutionary run. This is in contrast to an explicit fitness function.

Cliff and Miller [CM95] describe the *Red Queen effect* where the fitness landscape[15] of a population can be altered by other populations that it is interacting with. Rather than leading to smooth evolutionary progress, the Red Queen effect can result in the repetition of evolutionary strategies long since discarded. An example would be in the co-evolution of predator and prey. The population of prey develop a strategy to evade predators. Predators adapt to this and the benefit to prey following this strategy is negated. The prey population then evolve a second strategy, only for the predators to adapt to this too. Because the prey no longer use the first strategy, predators no longer need to adapt to it and it soon becomes forgotten by them. This means reverting back to the first strategy becomes viable again for the prey.

Genetic algorithm

The second method of evolutionary adaptation is via a genetic algorithm. The agents can be spatially situated within the world and tested for a specified finite time. The result of their interactions with the world are used by an explicit fitness function to determine their fitness. For non-spatial genetic algorithms, agents are stripped of all senses and only given actions that have an effect upon the internal state of the agent.

Although genetic algorithms have been proven to be useful as a search technique, it was felt that their use here should not be taken for granted. Aside from the problem of premature convergence, the classic genetic algorithm showed itself to have two major disadvantages. It's use led to ambiguity as to the average fitness of each solution and also to computational inefficiency.

The assumption underlying genetic algorithms is that a useful gene, or set of genes, provides an increase in the fitness of an agent allowing it to breed more. The gene is then more likely to spread throughout the population and

---

[15] The fitness landscape model is described in Kauffman [Kau93] and Depew and Weber [DW95]. The space of potential genotypes that may occur in a population can be mapped onto a hyper-dimensional landscape. The axes correspond to particular qualities that the genotype may provide for the phenotype. The height of the land denotes the fitness that a genotype will possess if it is mapped to that location.

the average fitness of the population increases.

But when we are trying to find the optimal solution or parameter set we are concerned with the fitness of a solution in its entirety and not with the fitness of the individual genes that it is formed from. If there is a stochastic mapping between genotype and phenotype, as there is for the agents used in this research[16] then we also want to know the average fitness of the solution over multiple expressions. This is also the case for any environment in which the starting conditions of each evaluation may differ[17].

This by itself is an insufficient reason to abandon the classic genetic algorithm as once it is assumed that fitness levels will no longer increase, each solution in the final generation can be tested an equal number of times with the fittest solution being picked. The assumption here though is that we know when the genetic algorithm has finished because the population has converged. As shall be seen later, this is not necessarily the case.

Despite the widespread use of genetic algorithms, they have been accused of being computationally expensive in certain domains, such as when applied to the MAX-CLIQUE problem [CP93]. Moriarty and Miikkulainen [MM99] describe how premature convergence reduces the speed of a genetic algorithm as it moves from performing parallel searches to a random walk using mutation.

Convergence, whether premature or otherwise, results in the majority of agents within a generation being identical so that the use of recombination and selection becomes a computationally expensive form of duplication.

Efforts to maintain population diversity to prevent premature convergence can also prove to be computationally expensive. Miller and Shaw [MS96] created dynamic niching to reduce the computational cost of conventional sharing techniques. The sharing technique alters the fitness of genotypes depending on how dissimilar they are to each other. The aim is to reduce the redundancy of highly converged populations by increasing the fitness of novel genotypes. The

---

[16] See section 3.5.6.

[17] Such as with online evolution whereby a single agent sequentially evaluates each solution in a generation.

computational cost lies in comparing the similarity of multiple members of a population.

Leung et al [LGX97] prove that when a genetic algorithm is run with a zero mutation rate, population diversity will inevitably decrease to zero variation and premature convergence will occur. Premature convergence is directly caused by a decrease in population diversity. Even if the genes required to create the global optimum are available in the initial population, it is possible that they will be lost by the selection and crossover operators during the course of convergence. Adapting the probability of crossover provides no help in avoiding premature convergence. It is discussed how existing methods to avoid or delay premature convergence are heuristic in nature. They propose further research to investigate adapting the population size and mutation rate in relation to population diversity so as to prevent premature convergence.

Harvey [Har94] describes how convergence normally occurs early on in an evolutionary run. When premature convergence occurs it is often seen as undesirable. Harvey has a different attitude when discussing variable length genotypes. He describes how progress can continue afterwards via the spread of beneficial mutations throughout a population. Mutation acts as a counterbalance to convergence and increases the variability of the gene-pool. It allows a population to spread out along the fitness landscape and to find new peaks. But if the mutation rate is too high then the search becomes a random one and any convergence upon hill-tops will be lost.

This attitude is at odds with that of Moriarty and Miikkulainen who see a highly converged population as redundant and slow. They maintain diversity by using symbiotic evolution to encourage speciation. This is achieved by applying the fitness function to individual parts of an agent to determine how well it performed, on average, with other available parts. Which parts are evaluated together is randomly determined.

Harvey and Thompson [HT97] argue that in practice most landscapes, particularly for the evolution of hardware, contain neutral networks that allow for escape from local maxima. They advocate encouraging the formation of neutral networks by allowing for the equivalent of junk DNA in the genotype. While a population searches the ridges of a neutral network within a fitness landscape, it can be seen as being highly converged and having little or no discernable increase in fitness. Convergence occurs naturally, even when using a function that assigns random fitness scores. Even if the landscape consists of multiple peaks, each of an equivalent fitness, a genetic algorithm will converge upon only one [MS96]. It is after convergence has occurred that the evolutionary run can properly get underway. Mutation is seen by Harvey and Thompson as the primary operator.

Barnett [Bar01] also views mutation as the genetic operator of primary importance in an evolutionary run. As for the purpose of crossover, Barnett opines:

> . . . at best recombination will function as a *macro-mutation* [Barnett's emphasis] operator - a "leap in the dark" across the fitness landscape.

But even this function of recombination can be questioned. When the operator is applied to genotypes from different niches of the population, the resulting genotype often has a low fitness [MS96].

What are we trying to do when we use a genetic algorithm that does not prematurely converge? We are trying to have it converge upon the highest peak of a fitness landscape. The alternative is to produce the optimal solution by a purely random process. But until the genetic algorithm reaches that optimum there is no way of knowing whether the peak that it is converging upon is a global maxima. Therefore some method of escape from local maxima is required, whether by backtracking to a previous col and trying another peak, or by a random walk along a neutral network.

If we assume from the outset that a search method needs to climb to the top of a peak in order to determine whether it has found the global maxima, and also that a search method relies upon mutation to find new slopes and move up them, then a more efficient form of search can be implemented.

Barnett describes an algorithm consisting of a single agent that makes no use of the recombination operator. The genotype of the agent is replicated and mutated. If this results in a higher fitness then the mutated copy is kept, otherwise the agent reverts back to the original genotype. The process is then repeated. By this method the netcrawler performs a random walk along a neutral network until it finds access to a network of a higher fitness. Two mutation modes are described for use with the netcrawler. Each point on a genotype can either have an equal chance of mutation, or there can be a constant number of mutations applied to the genotype as a whole.

Unfortunately the netcrawler is not applicable to the optimisation of genotypes that stochastically map to phenotypes. A symptom of this mapping is that the average fitness of the most successful genotype often rapidly decreases through repeated trials until it falls below that of a previous genotype originally deemed less fit. The reason is that the often genotype only becomes the most successful because the mapping, or the starting conditions of its evaluation, happens to be particularly good by chance.

Online evolution

The third form of adaptation is the one that is predominantly used in the research described here. It is described here as 'online evolution' because a single agent is situated within an environment and interacts with it while the search progresses.

Barnett refers to an ethos at Sussex university whereby artificial evolution is used for the solving of real-world problems. This results in an alternative understanding of artificial evolution. The third method described here is still being developed. It has been applied to the solving of practical problems as this provides a reliable indicator of its usefulness. The method is described not to advocate a certain method of parameter optimisation or evolutionary adaptation, but for completeness and to describe how the parameter sets were

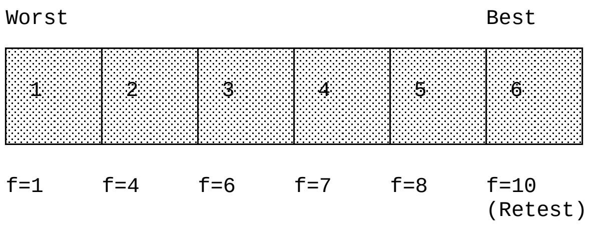


Figure 3.4: After a new solution has been added to the queue and evaluated, the fittest solution is then re-evaluated and the queue re-ordered if necessary.

obtained.

A mutation-based strategy is used to find the optimal solution. As with a genetic algorithm, a fitness function and mutation genetic operator is used, but there is no sexual reproduction or cross-over[18]. How the genetic operators are applied to the genotypes is described in more detail in section 3.4.4.

The agent has a queue of genotypes that encode neural networks. The queue is ordered according to average fitness. The agent generates a neural network from the fittest genotype and uses it to control its behaviour for a period of time. The average fitness of the genotype that the network was generated from is then updated accordingly. If the average fitness decreases to below that of another solution then the queue is re-ordered (see figure 3.4).

This backtracking is required to prevent a genotype that would normally lead to a high average fitness being replaced by another genotype, that by chance, happens to be mapped to a fitter phenotype. When it is discovered that the search has moved downhill, backtracking allows the search to revert back to a previous position.

If the genotype of the current neural network still has the highest average fitness then the agent copies and mutates a new version of it. The new version is then tested. It deletes the least fit network in order to save memory (see figure 3.5).

The mutation and recombination operators in a genetic algorithm are commonly seen as two opposing forces acting upon the diversity of a population.

---

[18] Other genetic operators such as duplication were not required to search for the optimal parameter set. It is not known how online evolution copes with open-ended search problems.

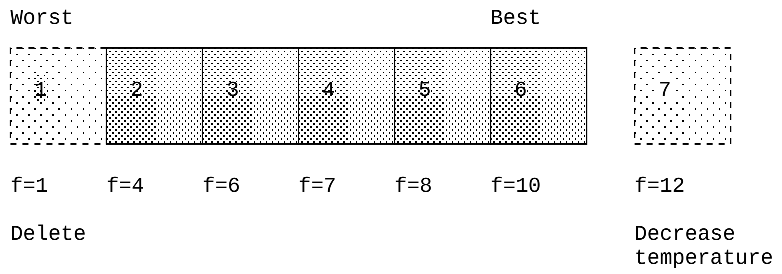


Figure 3.5: A new solution has been copied and mutated from the current fittest solution with a fitness of 10. It uses the temperature of the solution 6 but decreases it by a random amount. After evaluation the fitness of the new solution is determined to be 12 so it stays at the top of the queue. It is likely that more evaluations will bring this average fitness down. The worst solution, with a fitness of 1, is deleted.

Mutation increases diversity whilst recombination decreases it. If the use of the recombination operator in the absence of mutation inevitably leads to convergence, then it can be seen as acting in a similar way to the temperature rating used in the simulated annealing algorithm [RN95].

A genetic algorithm can be seen as a process of convergence to a single, hopefully optimal solution, whereby the run is finished and the best solution is picked. A temperature rating is used here much to the same purpose. It determines how great a percentage of the new solution is mutated. The search starts off with a temperature rating of 100% so that each new solution is completely random [19]. A minimum for the temperature rating is set so that each new genotype will have at least one new mutation to test.

Inspiration for a temperature rating came from the simulated annealing algorithm [RN95], a computational analogy of the process of freezing a liquid. With simulated annealing, the temperature rating allows a search to move slightly downhill when faced with the prospect of converging upon a local maxima. This is an alternative to starting the whole search again at a random position. Simulated annealing is complete and optimal if the temperature rating decreases slowly enough.

---

[19] In order to improve the effectiveness of the search technique, the search starts with the queue filled with completely random solutions that are evaluated once each.

Unlike the temperature rating for simulated annealing, the temperature rating for online evolution can revert back to a previous level. Each new solution uses the temperature rating of the previous solution that it is copied and mutated from, but decreases it by a random amount. This allows the search to backtrack to higher temperature ratings when the queue needs re-ordering. This means that the setting of the cooling schedule is not dependent upon how many times the search has to backtrack. Because a variable percentage of the solution can be mutated at each turn, there always exists the possibility that the search can find a spot higher up on another peak elsewhere on the landscape. As the search progresses and the temperature decreases, less of the genotype is mutated, effectively decreasing the search radius. New genotypes will be more similar to their parents. If they are found to be an improvement then they will more likely be an optimisation of their parents, higher up the same peak.

At the end of the search, the queue is iterated over so that each solution has been evaluated the same number of times. The fittest solution can then be picked by the designer.

The use of online evolution provides the following practical advantages:

- It is easy to record how often an entire genotype has been evaluated and its average fitness. This is of particular importance for stochastic genotype-phenotype mappings.
- The temperature rating indicates the progress of the evolutionary run.
- It is known exactly which genotype in the final population has the highest average fitness.
- The memory requirements of online evolution are lower. Only the current best genotype, and at times a mutated version of it, need to be present in the physical RAM of a computer. It is more practical to swap the rest of the queue to non-transient memory[20].

---

[20] Compare this to a genetic algorithm in which many members of the population need to be accessed multiple times to breed with each other so as to create the next generation. Only after the selection and replacement is completed can the previous generation be deleted.

3.4.3 Fitness function

Agents were allowed to perform for a set period of time, after which their fitness level was determined. The fitness of an agent is determined by

$$Energy + Water + Age - \mid Energy - Water \mid$$

The values for Energy and Water are those held by the agent at the end of its evaluation period. This provides information as to the agent's efficacy over the course of its entire evaluation period. It was decided that this was the simplest method of determining an agent's fitness level. Adding in complexity before it was deemed to be necessary would introduce more opportunity for artificial evolution to converge upon a local maxima. Evaluating the agent upon its average resource levels was also considered but this raised the question as how best to calculate an agent's average resource level at any point in time.

The absolute difference between the energy and water resource is subtracted from the fitness as both resources are essential for the agent to stay alive. This decreases the chance of an evolutionary run settling into a local maxima whereby a strategy is discovered that reduces the difference between the two resources at the expense of increasing both resource levels. Subtracting the absolute difference between resources also decreases the chance of an agent evolving to only concentrate on one resource.

3.4.4 Hierarchical genotypes

The problem

For an agent controller to be evolved artificially, it needs to be represented in two different forms. The first form consists of a set of instructions or parameters which define the characteristics of the agent controller (the genotype). This maps onto a second form, the phenotype, which is the implementation of the agent controller. Coding a phenotype is like coding any other computer program, but there is often very little variety when it comes to how the genotype is coded.

The conventional form that a genotype takes is that of a one-dimensional string separated by markers, such as [HHC93]. This requires a complex mapping from a single string of real or boolean values to a multi-dimensional aggregate of functional components, each with their own unique set of parameters.

Operations performed on a string of values may not equate to a valid configuration in the corresponding phenotype. Or a small change in the genotype may result in a large, disruptive change in the phenotype. Constraints applied to the phenotype may also be difficult to encode in the one-dimensional genotype.

Rules governing such constraints must be coded as part of the mapping process from genotype to phenotype. The more rules there are the more complex and computationally expensive the mapping process becomes. Unless the rules can be isolated from one another then the mapping process will suffer from poor scalability in such regard. If such rules are not encoded then the evolutionary process may well be hampered as alterations to the genotype may be too disruptive. This raises the question. If biological plausibility is not a requirement, are there better ways to represent a genotype?

A solution

Computer programs are most often hierarchical in nature. Functions are encapsulated as small chunks of code, which are called alongside other functions by higher level client. The same applies to machines and other everyday objects. A car may have four wheels, but each wheel consists of a tyre, inner tube, wheel cap, bolts etc. Each subcomponent performs a particular function so that the larger, encapsulating component can perform its function.

The same can be argued of the brain. As humans we do not just 'see'. We detect colours, edges, we recognise objects regardless of what angle they are to us and their position within our field of view. We also interpret what we see and apply emotional salience to objects.

An artificial or real neural network can also be seen as being hierarchical to some extent. It consists of layers or areas, with different types of neurons pertaining to particular areas and connecting to neurons elsewhere. Neurons using different learning rules, releasing and responsive to different neuromodulators.

So if what we are evolving is hierarchical in nature, why represent it using a one dimensional genotype? The genotypes used in this research are both hierarchical and recursive. Hierarchical so that each part of the genotype more closely and more intuitively matches the part of the phenotype being evolved. And recursive to allow a function or component to be used multiple times while only needing to be evolved once.

Genotypes are represented as tree structures where each gene contains further genes that can be evolved. This allows genetic operators to generally be used in a progressively less disruptive way as the evolutionary run progresses and the fitness of the agents increases. The benefits of this approach are two-fold; primarily to how the genotypes are engineered but it is also hoped that it will allow for both exploration of the fitness landscape and optimisation of the genotype to the peak that is being climbed.

Engineering benefits

What a hierarchical genotype structure can evolve, a single dimensional genotype can do too. The advantage of using a hierarchical structure is that it is easier to engineer and use than a one dimensional genotype. This is because it conforms to certain established principles of software engineering; namely encapsulation, one-point-of-maintenance etc. A hierarchical genotypic structure can also be more easily compared to the structure of an expressed hierarchically structured phenotype than can a genotype consisting of a string of bits or real numbers. This simplifies the genotype-phenotype mapping process.

Because each gene of a genotype can itself contain multiple genotypes and parameters, it is ideally suited to being implemented using a object-oriented programming methodology and can benefit from all the advantages that such an approach brings. The genotypic encoding is more easily expandable when new parameters or new genes are required as changes to the structure are contained. Because the gene is written as an encapsulated class, the effect of mutation and duplication on the gene's parameters can be written in a type- and gene-specific way. The encoding of types and values can also be re-used for other applications.

Hierarchical design

Yoshimura and Izui [YI02] propose the use of a hierarchical genetic representation for the artificial evolution of machine products. They recognise that design problems for machine products are generally hierarchically expressed and their system matches the structure of the genotype to that of the phenotype. They note that because the search space is larger and optimisation is more complex when using hierarchical genotype representations, that the search is computationally more expensive and superior solutions are not always obtained. This requires the use of parallel processing in order to shorten calculation times.

Phenotypes are often aggregates of components and subcomponents. Each subcomponent needs to be expressed from its own section of a genotype. Making the genotype hierarchical allows each gene to contain other sub-genotypes to encode its parameters. These sub-genotypes can be used to map directly to sub-components in the phenotype. The higher in the tree the genes are, the larger the effect they have on the expressed phenotype.

The genotype encodes, amongst other things, a topology of the neural network using recursion [21]. This is achieved by having the genes that encode the information that is needed to express a layer of neurons, contain other genes of the same type. The crossover, mutation and duplication genetic operators can be applied to these structures. The structures are of variable length and depth. The genotype therefore allows for both the use of sexual reproduction, such as in a genetic algorithm, and the mutation of copies, as used in the mutation

---

[21] But without specifying neuron-to-neuron connections

based search described in section 3.4.2.

Evolutionary progress

There has been much discussion in the literature concerning the rate of evolutionary change. Green et al [GNK00], when discussing neo-Darwinism, describe the assumption of what is known as the *gradualistic theory* of a population slowly and gradually changing over time. An alternative form of gradualism is *Punctuated equilibria* [EG72]. This recognises the transient periods of extreme change that occur infrequently during the course of evolution. Harvey [Har94] discusses the need for small mutation rates in artificial evolution to balance between exploration by the population and dispersal away from a local maxima.

Progress in evolution is seen as slow and gradual and is commonly understood to be a form of hill climbing. In keeping with this is the common assumption that changes to the genotype need to be small. The larger the change the more disruptive and less likely it will be to lead to an increase in fitness. On the whole this is correct, except at the beginning of an evolutionary run when only the base of the hills in the fitness landscape have been climbed and little or no optimisation has taken place. At such stage in an evolutionary run there is a far greater probability that a random placement in the fitness landscape will lead to an increase in fitness than if the genotype is already on top of one of the highest peaks.

There are two assumptions underlying the use of hierarchical genotypes. First, that optimisations works most effectively by making gradual changes, the extent of which is proportional to how much more a genotype can be improved. Second, that in the absence of neutral networks, exploration of the fitness landscape may require larger changes so that the search can leave the local maxima that is on.

Optimisation

When used with online evolution, see section 3.4.2, the hierarchical genotype is used in much the same way as a flat genotype would be. Each gene has the same chance of mutation. When used with classic genetic algorithms or endogenous evolution which rely upon the crossover operator, the genetic operators need to be adapted for use with hierarchical genotypes.

The hierarchical structure of the genotype means that evolution initially progresses via the changes that have a large effect on the resulting phenotype while the fitness of the population is still relatively low. Large changes will be more likely to survive and spread at the beginning of an evolutionary run than near the end when genotypes are being more greatly refined. As the evolutionary run progresses further refinements that have a smaller effect upon the expressed phenotype are allowed to be made. This increases the likelihood of future genotypes surviving because they are not significantly weaker than the rest of the population.

This is achieved in a genetic algorithm by applying operators to the tree in a top-down manner, stopping at the level in which the genotypes differ in such a way that crossover is meaningless. For example, a genotype for a layer of neurons may encode a gene for the type of hebbian learning rule to be employed, a decision with a relatively large effect on the phenotype's fitness. The genes that encode the hebbian learning rules may need to encode different parameters specific to each rule. It would therefore be meaningless to apply crossover to the parameters of different hebbian learning rules. The genetic operators would only be applied to the parameters if both parents were expressing the same hebbian learning rule.

Exploration

This technique allows a population to climb gradually up a slope of a fitness landscape, but what about local maxima? It was reasoned that a hierarchical genotype would allow both exploration and optimisation to occur depending on where a mutation occurred in the structure.

Even though genotypic convergence through the use of crossover may have refined the genotype down to the lowest level in the tree, other genetic operators such as mutation and duplication can still occur at any level. This means that the possibility will always exist that a mutation will occur at a high level in the tree, causing a change large enough for a new genotype to be produced on another part of the fitness landscape.

The probability of this high level change being beneficial depends upon the height of the current local maxima compared to the other maxima in the fitness landscape, and how far the evolutionary run has progressed. Because convergence occurs only at the lowest level of the genotype in which crossover is meaningful, the high level change can still spread throughout the population.

Exploration and optimisation occur contemporaneously. As convergence progresses, mutations can occur at a lower level of the hierarchy, therefore having a smaller effect and allowing the genotype to climb the peak it is on.

3.4.5 Mutation and duplication

The occurrence of mutation or duplication is tested for during crossover. If it occurs, a gene will mutate any of the parameters on which further crossover cannot be applied. If mutation or duplication were to be applied to parameters that were also subject to the crossover genetic operator, then the chance of mutation and duplication occurring would be doubled.

Duplication occurs according to the neutral gene model advocated by [THH96]. It must have little effect on the fitness of the genotype and should produce a largely redundant extension to the phenotype. The redundant part should only duplicate the functionality of the original. This allows mutation to then be applied to one of the copies of the genotype in further generations so that extra functionality can be evolved from it.

3.4.6 Crossover

A sequence of genes is stored in a single dimensional container class. Because of the variable length of these genotypic strings and because of the hierarchical nature, crossover is more complex than merely slicing at a random position and copying part of one parent's genotype and part of the other's.

The size of the genotype is chosen according to the following rules:

- If both parent's genotypes are empty then an empty genotype is expressed in the offspring.
- If one parent's genotype is empty and the other is not then the non-empty genotype is expressed in full. This is so as to encourage the growth of genotypes within a population.
- If both parent's genotypes happen to be the same size then a random position is chosen. The first parent copies their genes from left of that position and the second parent copies their genes from right of it.
- If the genotypes are of different lengths then crossover occurs with the smallest genotype being examined first. A random position on this genotype is chosen and those genes are copied. A random length of the remaining longer genotype is then expressed from after that split. This may result in a genotype being expressed that is smaller than both parent's or up to as long as the longest parent's genotype.

As discussed above, if the genes from both parents are compatible, i.e of the same type, then crossover occurs on the parameters contained within these compatible genes and a hybrid of the two is copied instead.

3.4.7 Control genes

Information specifying how other genes are expressed can also be evolved. This is encoded using control genes.

There is only one type of control gene used by the genotype. This specifies how many times a single gene is expressed. This information is stored in a control gene as many different genes make use of this information, albeit in different ways.

The control gene contains a single real value. The integral part of the value specifies how many times the gene that the control gene applies to is expressed. How the fractional part of the value is expressed, if at all, is dependent upon the gene that uses this control gene. It is normally used as a probability of the gene being expressed. This allows a smoother evolutionary transition when increasing the number of times that a gene is expressed. This is especially so for genes that can be partially expressed. An example being a fibre gene that specifies how two layers connect.

3.5 Implementation

The genotype has been implemented in C++ with each gene defined as a separate class. This allows for code re-use and common code has been shared between different agents in HelloWorld. Rather than a single dimension genotype consisting of only floating point values, the most apt primitive types were chosen according to their intended use. The genotype hierarchy is shown in figure 3.6.

3.5.1 TreeGene

Each class inherits from the TreeGene base class which specifies the functions and genetic operators that each gene must implement.

3.5.2 GeneTree

The GeneTree class is a container class that stores a sequence of genes of the same type in a vector. It is this class that implements the functionality, described in section 3.4.6, of applying genetic operators to variable length gene sequences.

3.5.3 BioNetGene

At the top of the hierarchical tree is a BioNetGene object. This gene contains the rest of the genotype and encodes the neural network.

There are three global parameters that can be optimised.

- Neuron threshold.
- Modulator re-uptake rate, ignored for non-modulating networks. (stored in the EndocrineSystemGene.)
- Number of iterations made through the network each turn.

3.5.4 EndocrineSystemGene

The BioNetGene contains an EndocrineSystemGene which contains information global to the entire neural network. The only parameter the EndocrineSystemGene has is how fast the modulator levels decrease in the absence of any more being produced; the re-uptake rate.

3.5.5 BioLayerGene

The BioNetGene contains a single BioLayerGene. The BioLayerGene encodes the values for the parameters of a group of neurons as well as genes to encode the hebbian learning rule, receptors and production of modulators. It also contains a sequence of genes specifying the inputs to it.

The following parameters are genetically determined. All neurons within the same layer will share the same values for these parameters.

- Base leakage of the leaky integrate-and-fire neuron.
- Spike-dependent change to leakage.
- Base probability of firing if spike threshold is reached.

- Length of refractory period
- Learning rate applied to the STDP level to determine the change in synaptic weight
- Increment to STDP recording function.
- Decay over time for STDP recording function.
- Whether the STDP rule is hebbian or anti-hebbian.

3.5.6 BioFibre

The BioFibre class encodes the information used to express the pattern of connections between two layers. As these are stored in a BioLayerGene class the layer to output to is already known. Which layer that the fibre connects from and any offset that is used to connect to other parts of the neural network is encoded in this gene

Connections expressed by BioFibre genes are uni-directional and have a genetically encoded value that specifies how long a spike will take to reach the other neuron. This is implemented as a queue of a specified length. In practise it was found that this feature was not required for any of the implemented agents. Therefore all connections are of equal length so there is no delay with an activation being transmitted.

All connections between layers are excitatory and modifiable. Parameter searches were conducted to see whether inhibitory connections would be useful but they were not. Non-modifiable connections were avoided to help minimise the risk that evolution would hard-code the network topology to increase the average fitness.

Real value connectivity

The multitude of connectivity between two layers is open to optimisation. This is a continuous value whereby the fractional part determines the chance of a connection between two neurons being made. So for example, a multitude of 1.5 would mean that every neuron in a source layer was connected to every neuron in the target layer at least once, but with a 50% chance of being made a second time. Parameter optimisation most often selected multitudes of less than 1.

Stochastic mapping

Because the multitude of connectivity between layers is a continuous value the mapping between genotype to phenotype becomes stochastic. This means that the same genotype is able to produce a different phenotype each time it is used. As a result a fit genotype cannot be guaranteed to express a fit phenotype.

There may be many reasons why a stochastic mapping process has advantages over a deterministic one. For example, it may allow for an overlap in the space of possible phenotypes between two different genes, thus allowing mutations to be less disruptive. But only one reason demanded its use here. The majority of what the genotype encodes can be deterministically mapped to a corresponding phenotypic component. The only part that does not is the encoding of the connectivity between neurons. It was crucial that any resulting behaviour was the result of the agent learning rather than it being due to evolution hard-coding the network topology. Hard-coding of the network topology would mean that it could not be guaranteed that the modulating and non-modulating agents were an otherwise equivalent design.

If the connectivity between neurons was specified at a neuron-to-neuron level, then not only would this result in a very large and inefficient genotype, but it would also be very easy for brittle, hard coded agent controllers to be produced by artificial evolution whose inner-workings were not fully understood. It could also lead to different strategies being evolved for modulating and non-modulating agents making comparisons between the two meaningless.

A stochastic mapping also has disadvantages. A genotype must be judged according to its average fitness so as to prevent a fit genotype being lost, or a poor genotype being retained because the phenotype it was mapped to was unrepresentative of its quality. As a consequence, evolutionary searches become computationally more expensive and it becomes harder to be sure which genotype is overall the best one to use.

An alternative to using real valued connectivity would be to enforce that every neuron in a destination layer mapped to every neuron in the target layer once and once only. But the connections would still require to be randomly weighted at the beginning of the evaluation and the mapping would still effectively be stochastic. Real valued connectivity between layers, in effect, is merely a more extreme form of stochastic weighting.

Less disruptive mutations

Another reason for a layer-to-layer mapping to be encoded in the genotype, and as a continuous value, was to decrease the probability that mutations would be disruptive. Thompson et al [THH96] describe mutations as "often deleterious, and occasionally advantageous." Mutations to the genetic encoding of a neuron-to-neuron mapping could potentially have very large effects if evolution had produced a brittle strategy by hard-coding the network topology. This would reduce the probability of those mutations increasing the fitness of the genotype. A layer-to-layer mapping means that if two layers are specified as being connected, then they will continue to be connected throughout the evolutionary run. Evolution merely decides *how* connected the layers should be. The connectivity being encoded as a real value allows for smaller and less disruptive mutations to be made. A minimum connectivity of 0.3 was specified to further constrain the parameter search.

3.5.7 ConnectivityGene

The ConnectivityGene is stored in the BioFibre class. It specifies the pattern of connectivity between the neurons of two layers. In practise a control gene may specify that there is only a probability of each connection being made, or that they should be made multiple times. Therefore the actual connectivity may develop differently to that which is specified in the genotype. This allows for a smoother transition between types of connectivity genes. The only connectivity found to be useful in this research is full connectivity, where every neuron connects to every other neuron in the other layer.

3.5.8 ModulatorGene

ModulatorGenes are used to encode the types of modulators available for use by the system. In practise a type is specified using a single integer value that can be incremented or decremented by mutation. A ReceptorGene or EndocrineGene has a ModulatorGene to specify which modulator it will react to or produce. If any of these ModulatorGenes contains an integer value that no other ModulatorGenes have then a new modulator is added to the system.

There exists a coupling between EndocrineGenes and ReceptorGenes, in that they both need to have mutated the same ModulatorGene in order for a new modulator to be usefully added to the system. If either gene, but not both, mutate a new modulator gene then it has no effect on the system. Even though it will not have a direct benefit to the agent, the mutation may have a benefit to future generations. For example, a layer in one agent is expressed with the ability to produce a new type of modulator but no other parts of the system have receptors for that modulator. An offspring in a future generation may develop a receptor elsewhere in the system that responds to that modulator in a beneficial way. So mutating a new modulator is the equivalent of branching into a new area of search space. It is similar to the functionality of duplicated neutral genes in that the benefit comes from being able to use future mutations. Open-ended evolutionary runs with unconstrained mutation and duplication result in many different modulators being evolved.

The identifier of the modulator is a one-dimensional value; an integer. Increasing the dimensionality of the identifier, such as using a string of values, will decrease the chance of independent modulator genes being mutated twice the same way and therefore will decrease the chance that a new modulator will be added to the system.

3.5.9 EndocrineGene and ReceptorGene

An agent has a set of modulators that any neuron in the system can be influenced by if it expresses a ReceptorGene. A neuron can also increment the level of a particular modulator if it expresses an EndocrineGene. Both the EndocrineGene and ReceptorGene carry a single floating point value. The EndocrineGene specifies the increment that it makes to the strength of the modulator when the host neuron spikes. The ReceptorGene specifies the modulation rate, or the influence that the modulator has on the activation of a neuron with a receptor for it.

A neuron will pass a value describing its current activation to its receptors and add up all of the returned values before applying that change to itself. A receptor determines the change that needs to be made to the activation by modulating the passed value with the modulation rate. It then modulates the level of change by the level of the modulator in the system. This modulated change to the passed value is then returned.

A number of receptors and secretors can be genetically specified for all the neurons within a layer. These contain the following parameters:

- What to modulate (probability-of-firing or sensitivity-to-input).
- Modulation rate.
- Secretion rate.

3.5.10 Automated parameter set optimisation in practise

If a new idea for an agent controller failed to work then it was unknown whether the fault lay in the design or the parameter set. Genetic algorithms were initially used for parameter optimisation but they proved too slow for use in rapid

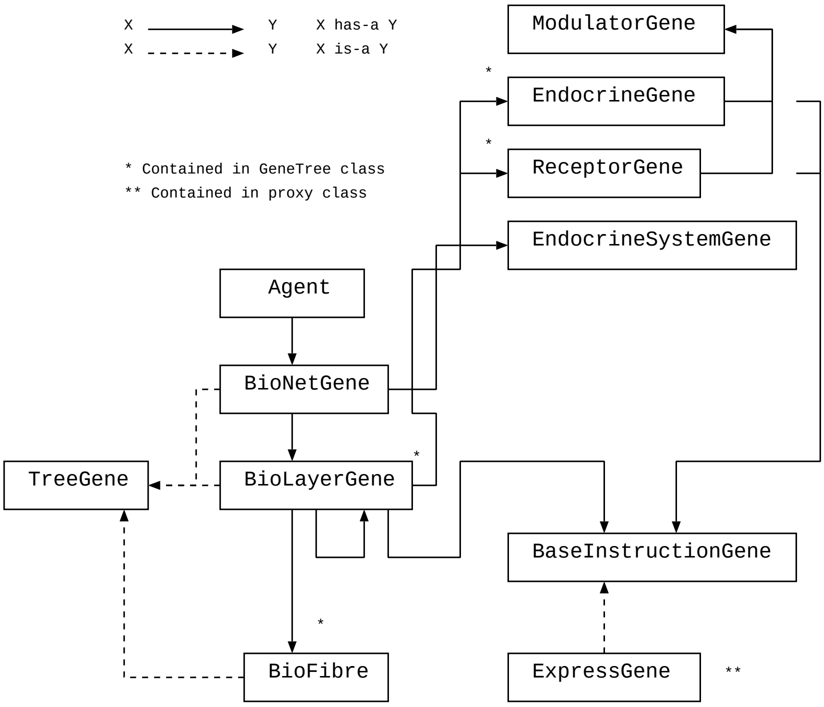


Figure 3.6: The BioNet is the agent controller and contains the neural net which is created from the BioNetGene. Each neuron in a layer is the same and is genetically encoded in the BioLayerGene. The BioFibre genotype maps stochastically to its phenotype. The EndocrineGene encodes secretion rate of a modulator, the type of modulator is encoded in the ModulatorGene. The ReceptorGene encodes modulation rate and the whether the probability of a neuron firing is modulated or its sensitivity-to-input. The EndocrineGene encodes the system-wide re-uptake rate of all modulators.

prototyping of agent controllers. Genetic algorithms were also used for open-ended evolution whereby they were given a free-reign as to the size and type of agent controller produced. Although the fitness levels increased to a certain extent, very little interesting behaviour could be discerned.

Because the duplication operator was used during unconstrained evolution, the neural networks would grow in complexity over time regardless of whether the fitness levels increased. So even though the population size remained constant, the memory requirements of the evolutionary run would increase. The runs were stopped when the agent controllers grew too complex and the memory requirements of the whole population exceeded the capacity of the computer that it was run on. Once the computer began to swap physical memory to disk progress slowed down sufficiently that the evolutionary run had to be prematurely ended.

These difficulties prompted the development of online evolution. It was immediately found to be significantly faster than using a genetic algorithm for parameter optimisation. And no matter how large the queue was, only two agents ever needed to be stored in physical RAM at any one time.

The first attempt at online evolution resulted in an agent that worked in a similar way to Barnett's netcrawler [Bar01]. Only two solutions were stored, the current best solution and the newly mutated solution being evaluated. The temperature rating initially mutated 100% of the genotype and slowly decreased to 1% over the course of the run, decremented each time a better agent was discovered. Although the technique was initially fast, the final genotype could not be relied upon to be the fittest genotype because of the stochastic mapping being used between genotype and phenotype. If the agent did find a good solution then it was likely to be replaced by a poorer solution that happened to be mapped to a more successful phenotype. It was realised that because of this, genotypes needed to be evaluated multiple times and an average fitness recorded. This meant some form of backtracking was required as the average fitness fell below that of a previous version.

This slowed down the technique, but it was still faster than using a genetic algorithm. Setting a higher decrement to the temperature rate allowed for faster runs but meant that the agent was less likely to settle upon a good solution. Several evolutionary runs were made in parallel to find the optimal parameter sets, each using a different random seed. Some runs were stopped shortly after starting because their progress was poor. If the decrement to the temperature rating was reduced then this was less likely to happen. The downside to this was that all the runs would then take longer to complete. The runs that were not stopped were run to completion and then compared. These runs all reached similar levels of fitness.

Knowing how to set the temperature rating was the key to using online evolution most effectively but other techniques were found to help as well. Filling the queue full of completely randomised solutions at the beginning of the run before the temperature was lowered, each evaluated once, helped reduce the likelihood of reaching a local maxima. Using a larger population meant that the agent was more likely to find a good solution, but it took significantly longer for the temperature to decrease to the final point whereby the run could be ended[22].

At the time of writing, the temperature only decreases when mutating a new solution. If backtracking occurs, the temperature rating returns to that of the current best solution. The current best solution often requires many copies to be made from it and mutated before a better solution is found. But a certain number of improvements need to be made before the temperature drops to a cut-off point and this is dependent upon the chosen decrement to the temperature rating. If ran to completion, the majority of the processing is spent reducing the temperature rating the last few percent. Therefore reducing the temperature rating of the current best solution each time it is used for copying and mutating a new solution may increase the chance of finding a better solution and also

---

[22] Runs were finished when the average temperature of the queue was $\approx 1\%$, i.e. each member of the queue mutated on average 1% of it's genotype. A larger population meant that it took longer to reach this average.

shorten the length of the evolutionary run.

It is not known how the use of hierarchical genotypes affected the performance of the evolutionary techniques, but they proved useful on a practical level for prototyping new ideas.

3.6 The need for a visual display

One problem that was foreseen as potentially occurring, was the lack of information about how a neural network was adapting. Printing out logs of run-time values can show that a particular component of a network is working within a set range, but not whether it is providing the functionality intended of it. Run-time values were displayed as graphs and proved invaluable for debugging components, such as the models of neurons or levels of modulators. The graphs did not show whether the components were working together as intended or how they were interacting within a larger system. The dimensionality of the data was high because of the large number of interacting components that made up the network. This had to be presented in a way that was simple and intuitive for a human to understand.

The original hope was that established evolutionary techniques, such as genetic algorithms could be used for finding the correct set of parameters of the system in order to make it work as intended. This would save spending significant time and effort developing a visual display. It was found that this alone was not sufficient and could only be used to optimise what already worked to some degree[23].

A visual display allowed unintended bugs and incorrect assumptions made during development to be discovered. This provided an understanding of the functioning of the system as a whole. This is particularly important for demonstrating the system to others and in allowing them to also understand it. But

---

[23] This is because evolution had to be constrained to reduce the risk that it would produce hard-coded solutions that were difficult to understand and which did not allow fair comparisons to be made between modulating and non-modulating agents.

most importantly, the visual display allowed verification that the simulation was functioning as intended.

3.6.1 The visual display system

The run-time data was collected and used to create an animated three dimensional view of the model, output in VRML [ANM97]. User configuration allows the model view to be animated to varying degrees. If very little is being displayed then a non-animated view is useful to display the structure of the neural network. This can determine whether the model is small in size or whether it is lacking activity. Connections between neurons can be displayed either according to their weight, or only when they are activated by a firing neuron. The former can indicate adaptation throughout the lifetime of the agent, the latter can indicate how much input a neuron receives.

It was not possible to easily show the level of modulation that each neuron receives without compromising the presentation of other information. The size of a neuron indicates its current activity and its colour helps identify it as being part of a particular layer. Because there can be more than one modulator exciting or inhibiting a neuron in various ways, presenting this information would clutter the display and make it more complex. It was easily possible to show the level of a modulator on a global scale. How the modulator is used can be inferred from understanding of the model and other information gleaned from the display.

3.6.2 Model view consistency

A significant effort was spent in making the view consistent regardless of the size and complexity of the model given to it. This was important as it was not initially known whether the designed models were to be compared against those produced by open-ended evolution.

The two main problems were how to spatially position the layers and how to colour their weights and neurons so that they were distinct from those of other

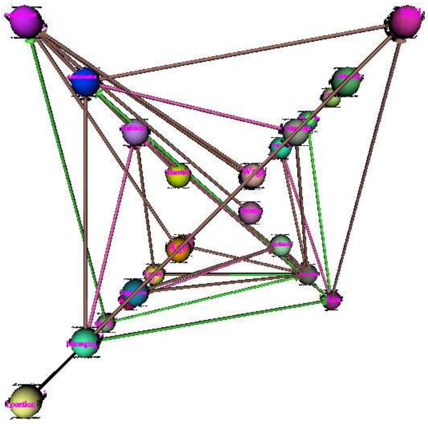


Figure 3.7: Java implementation. Random graph before sorting. Nodes are of a constant size. Although the nodes are positioned here in an orderly fashion, spreading out along each dimension in turn, they could also be positioned randomly. Nodes are coloured randomly by the algorithm. The edges represent connectivity present in the data. In this example, which nodes are connected is randomly determined. The length of an edge between nodes carries no meaning.

layers. If the spatial co-ordinates of the layers were chosen at random then it would mean that it would be more difficult to make sense of the view. The same applies to weights from different layers having the same colour.

Spatial distribution

The spatial layout was computed using a variant of the Spring-Embedder model [Che99]. The model view can be seen as a graph whereby the edges are of a variable length. All nodes are repelled away from each other, but an attractive force exists between nodes that are connected by an edge. These forces are applied until the entire structure reaches an equilibrium.

The algorithm was first developed for an existing Java program previously written by the author. This produces graphs similar to [WBDH95] whereby the forces act in all three dimensions. Once it was shown to be scalable and consistent, it was re-implemented in C++ and applied to visualising the modulated networks.

In the final implementation, the forces only act in two dimensions. Each layer

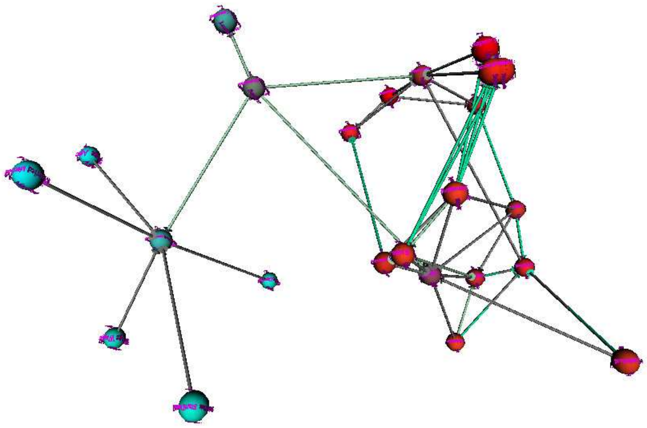


Figure 3.8: Same graph after sorting in three dimensions. Nodes that are connected together are drawn together. All nodes repel all other nodes. This results in connected nodes forming clusters, and clusters distancing themselves from other clusters. Nodes within the same cluster share a similar colouring. Nodes that link clusters share the colouring of both clusters.

acts as a single node and all the connections between two layers count as one edge in the graph. The x-axis is used to spatially distribute the neurons within a layer as a row. Because the attractive and repulsive forces are constrained to act in only two dimensions, along the y- and z- axis, there is an added requirement to apply a repulsive force between nodes and edges that has not yet been implemented. This is especially important when considering that the edges of the graph are displayed as intersecting planes through space because they extend along the x-axis. This application can lead to the development of significantly novel variants of the algorithm if work continues on this. It is an interesting possible line of research in data visualisation but is not followed up further in this thesis.

Node and edge colouring

Colour encoding is one way in which information can be easily represented without increasing the complexity of a visual display. Or it can be used to make some information that is already represented more obvious. In the model view, drawing connected nodes closer together shows the clusters that they form.

Having all nodes within a cluster coloured the same makes the clusters more obvious to the human eye. It can also show how related partial clusters are if there are many links between them as the colouring can change subtly.

There were two requirements for colouring when visualising the neural network. These were that layers that were connected, and also different connections to the same layer, should be coloured as differently as possible. If not, then if a layer of neurons is connected to multiple layers elsewhere in the network then it becomes difficult to discern which connection is output from which layer. Colouring connected layers as differently as possible also helps discern which neuron belongs to which layer for small networks as the layers are drawn closer together.

Essentially this is a variant upon the classic map-colouring problem [Pre00]. How do you colour all the adjacent bounded areas on a map using as few colours as possible? The difference here being that using as few colours as possible is not a constraint required for visualising the neural network.

This was achieved by diverging the red, green and blue values (dyes) of the colours of two layers connected by weights, or weights that were connected to the same layer. A percentage of the difference between the same dyes of two RGB values is added or subtracted from each in order to drive them further apart.

3.6.3 The visual display saves the day

One major disadvantage with relying solely upon black box adaptation occurred when artificial evolution persisted in converging upon local maxima in the fitness landscape. It was necessary to constrain the ways that the network architecture of the agents could be adapted.

The three dimensional model view proved invaluable for debugging the code, testing every assumption and for showing new ways that evolution needed to be constrained. This led to improved confidence in the software and tools, and a better understanding of how they can be used.

One example of this was that the number of neurons in the output layer was reduced to zero through mutation so as to make it more probable that the agent would choose the default 'Do Nothing' action which conserved resources and allowed the agents to live longer. [24] Once the population converged upon this easiest solution for increasing their fitness, it was difficult for the population to use a different evolutionary strategy without lowering their fitness first.

When it was noticed that this was happening, adaptation was constrained so that the number of neurons in a layer could not be mutated during the course of an evolutionary run. The number of neurons in the input and output layers were fixed. For middle layers, which did not require a specific size, each agent was given a random number of neurons and the population was allowed to converge upon a single number.

The result of this was that the population converged upon middle layers having a very small number of neurons between one and three. This meant that information from the input layer was lost so as to allow the agents to evolve pattern generators that produced the same output regardless of sensory input. This suggested that the sensory input was too complex and not being interpreted because it was too difficult to do so. This conclusion would not likely have been made had the system not been visualised as a whole[25].

It was noticed that the neural networks would only evolve or adapt to perform the 'Do Nothing' action. This lowered certain internal sensor levels. It was because of this that it was discovered that the network worked best as a minimal disturbance system. Rather than change the pattern by which the sensory signals were applied to the input layer, their strength were instead increased or decreased according to the result of previous actions.

Developing a useful visual display carries a significant cost in time and effort that distracts the researcher from their original interest. As adaptive systems

---

[24] To avoid this agents who do not decide on an action now have a default inactive action chosen for them. This carries with it the maximum cost in resources

[25] Incidentally, these seem to fit the description of central pattern generators that occur in natural brains, as described by Kelso [Kel95] pp239–243. Even though sensory input was required to drive the CPGs, information from the input was lost.

become more complex this will become more of an issue as there will be a greater need for appropriate information visualisation. Yet developing them will carry a greater cost as they need to present ever more complex systems in a simple way that is intuitive for a human to understand. This suggests that as research into computational intelligence develops, there will be a growing need for generic and re-usable tools for information visualisation.

3.7 Summary

In order to understand the functionality provided by neuromodulation, we must first isolate it. This endeavour calls for simplicity where possible and is the primary reason for the initial use of non-situated software-based agents. The isolation of the functionality provided by modulation shall be achieved by comparing equivalent modulating and non-modulating agents in environments that they were not designed for.

Requirements commonly satisfied by using an agent approach were described, and then discussed in relation to the agents to be used in this research. The chapter then proceeded to discuss the need for biological plausibility when understanding the role of emotions and neuromodulation in natural agents. Understanding the functionality provided by neuromodulation may help us create artificial mechanisms more suited to the silicon-based computer architectures that underly artificial agents. This discussion led onto the question of what the primary mechanism is that allows agents to adapt to an environment. Should the agents be seen as learning by reinforcement? Or should they be seen as self-organising systems within an environment? This question is relevant to the understanding of functionality provided by certain forms of neuromodulation, such as the role of dopamine in the basal ganglia.

The discussion returned to the need for simplicity, this time in the context of the environment that the agents are to be evaluated in. The use of a grid world was justified on the grounds that it helped for the purposes of prototyping and isolating observed functionality. A simple environment was then described that agents could be situated within.

The agent controller was then described in detail. Biologically plausible neural networks are to be used for this purpose. These have many parameters that must be correctly set for fair comparisons to be made between equivalent modulating and non-modulating agents. The optimisation of agent parameter sets is automated using evolutionary techniques. Three methods of optimisation were discussed; endogenous evolution, genetic algorithms and online evolution. This was followed by discussion of the use of hierarchical genotypes.

The chapter ended by describing the visual display developed for use in understanding the functioning of the neural networks. An example of its use and the practical benefits that it provides was then described.

Chapter 4

Experiments and results

4.1 The plan and its rationale

As described in section 2.6, this thesis proposes increasing our understanding of emotions by studying the the underlying mechanisms that help give rise to them in natural agents. The mechanism that this thesis focuses on is neuromodulation. This will be achieved by identifying the extra functionality provided by neuromodulation by comparing modulating and non-modulating agents that are otherwise equivalent.

Functionality provided by neuromodulation can also be implemented using only neurons and synaptic connections. This leads to the question of what practical benefits are gained from using one over the other. The experiments in this chapter will seek to help answer this question.

As described in section 3.1.2, the agents will be tested for how well they adapt to environments that they were not specifically designed for. This allows us to compare attributes of the adaptive agents such as the robustness and scalability of the solutions employed by them.

4.2 Implementation

For the first experiments, the agents shall be tested on the first requirement listed in section 3.1.3. This is to maintain homoeostasis and keep replete its critical resources. The artificial life animat concept shall be abstracted to provide the simplest possible context for testing the effect of neuromodulation when applied to an artificial neural network. Therefore the agents shall not be situated within an environment.

4.2.1 Actions and senses

The agent can neither sense an environment nor be affected by one. The only thing that it interacts with is a body with two resources, energy and water. The choice of output directly and immediately alters the level of a resource. This consequently alters the strength of the input signal to the network.

The agent is given a set of actions that increase or decrease by one or two resource points [1], or are neutral to, either the energy or water level in the body. There is one action for each permutation, effectively making 9 in total even though there are ten possible actions to choose from. There are two neutral actions, one for energy and one for water, but as neither have any effect on the state of the agent's resources they are effectively the same. It is useful to have one, preferably neutral action, duplicated because if either action is used differently to the other then it throws doubt on how well the agent is adapting. The actions provided to the agents are listed in section 3.3.2,

The inactive action is used by default when an agent does not choose for itself. This results in the resources of the agent being reduced by two points each. The effect of this is more costly to the agent than if it deliberately chose the most costly action available to it as that would only result in a reduction of one resource. This effectively encourages the agent to make a choice.

---

[1] Points are used as it is an arbitrary level that has no correlation with any real physical quantity.

4.2.2 The system

The agent has a body that requires two resources, labelled 'Energy' and 'Water'. Each change in resource level is passed to a feed-forward neural network as an input signal. Before being input, they are scaled to the largest increase and decrease that has occurred to each resource so as to be within the range [0,1]. They are then inverted so that desirable changes such as increases to a resource level, result in a reduced signal to the neural network.

The network learns which outputs should be most frequently and strongly fired to minimise the level of input signal. There is one output per action and this has an effect on the internal state of the agent. The change in internal state determines the strength of the input signal to the network in the next turn. If the action has a desirable effect[2] then decreasing the strength of the input signal the next turn increases the likelihood of that action being used again.

The network consists of three layers of adaptive leaky integrate-and-fire neurons (section 3.3.4) learning via spike timing-dependent plasticity [BW02]; see figure 4.1 and 4.2.

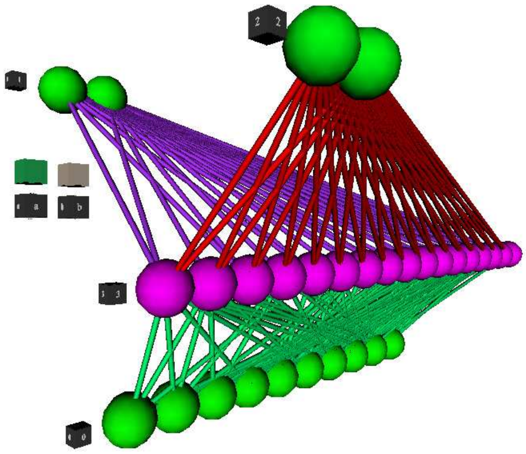


Figure 4.1: The structure of a typical modulating network. Clockwise from top left; (top left) hunger signal layer, (top right) thirst signal layer, (middle) hebbian middle layer, (bottom) anti-hebbian output layer, (left box) hunger modulator, (right box) thirst modulator.

---

[2] . . . for the agent or for whatever other purpose the network is put to.

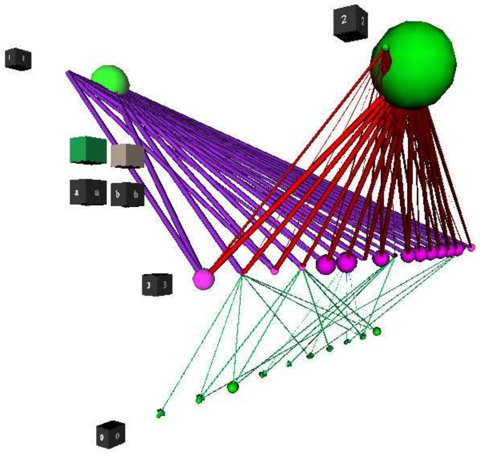


Figure 4.2: A modulating network in action. Neuron size shows level of activation, width of weights show synaptic strength. Each upper cube corresponds to a modulator, the height of which is animated to show the changing global strength of that modulator. The non-animated numbered cube underneath is used to identify the modulator.

For each resource the input layer has two neurons that output to the middle layer. One neuron signals the need for the resource and the other neuron signals the satisfaction of that need. If a previous action performed by the agent results in a decrease in hunger or an increase in resource satiation, then the corresponding input signal is decreased until the next turn.

There are situations in which an effective behaviour for an agent may decrease a need but not satisfy it. For example, if it is in an environment which is temporarily bereft of resources then waiting and conserving its current levels may be the optimal behaviour. Alternatively there may be situations in which an agent needs to store more resources than it is used to doing. In this case the need for the resource will be signalled despite that need being signalled as satisfied. An example would be an agent expecting to soon find itself in an environment bereft of resources.

For the modulating network, the input layer neurons increase corresponding modulator strengths when fired, while the middle layer neurons have receptors for those modulators.

The network is iterated over a number of times within a single turn, after which the winning output neuron is chosen. The agent then executes the action corresponding to the winning neuron. Which neuron wins is determined by summing up the total activation of each neuron over the iterations and choosing the neuron with the greatest sum. This stops a neuron with strong inputs from losing because it just has spiked and thus has low activity or is in a refractory period.

The number of iterators of the network is determined by the automated parameter optimisation. There is an upper limit of ten iterations. The modulating agent performed nine iterations each cycle and the non-modulating agent performed six.

4.2.3 Modulators

Two variants of the network were created; modulating and non-modulating. They were the same except that the modulating network had in addition two modulators, one used to signify energy and water levels.

The secretors increase an associated modulator. Each input layer neuron has one secretor for the modulator that corresponds to the resource being signalled. Neurons within the middle layer are given a random subset of receptors. These can therefore be modulated by neurons in the input layer. The receptors modulate either the neuron's sensitivity to input or probability of firing. The extent of this is determined by the level of the associated modulator and whether the receptor is inhibitory or excitatory. The modulation rate of the receptors and the increment rate of the secretors is determined by the initial parameter search. For a schematic of the modulating system, including body and actions, see figure 4.3.

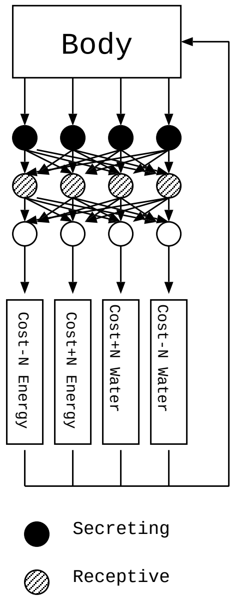


Figure 4.3: The agent consists of a body that contains water and energy levels. The change to these, made by the previous action performed by the agent, is fed to the input layer of the network. This consists of two neurons for each level. One neuron signals the need for an increase to a resource, the other signals the satisfaction of that need. The strongest output neuron fires the corresponding action. There is one output neuron for each action (not all are shown for the sake of simplicity). Neurons in the input layer secrete a modulator corresponding to the resource they represent. Neurons in the middle layer have a random subset of excitatory and inhibitory receptors for these modulators.

4.3 Initial tests

As discussed in section 3.4.1, there were certain parameters that needed to be set; for example the number of neurons in the middle layer. It was not obvious what values these parameters should be given so a parameter search was performed.

Once completed, a modulating and a non-modulating genotype were picked for comparison against one another. The fitness of the genotypes were similar and both were typical of the kind of solutions that were evolved. Because there was a stochastic mapping from genotype to phenotype, and to provide multiple evaluations, the genotypes were hard-coded so that they could be tested as a population.

Examination of the genotypes showed that parameter optimisation had converged upon a fully hebbian network for the non-modulating network, and a hybrid anti-hebbian / hebbian network for the modulating network. The modulating and the non-modulating genotype produced agents of similar performance. The first thing to check was whether the agents were learning and adapting, and if so, how well they were adapting. This was to make sure that their behaviour was not hard-coded.

4.3.1 Initial results

Figure 4.4 is an example of how frequently the different actions were performed by a population of phenotypes throughout their lifetime. The example is typical of both modulating and non-modulating agents regardless of the random seed they are run with.

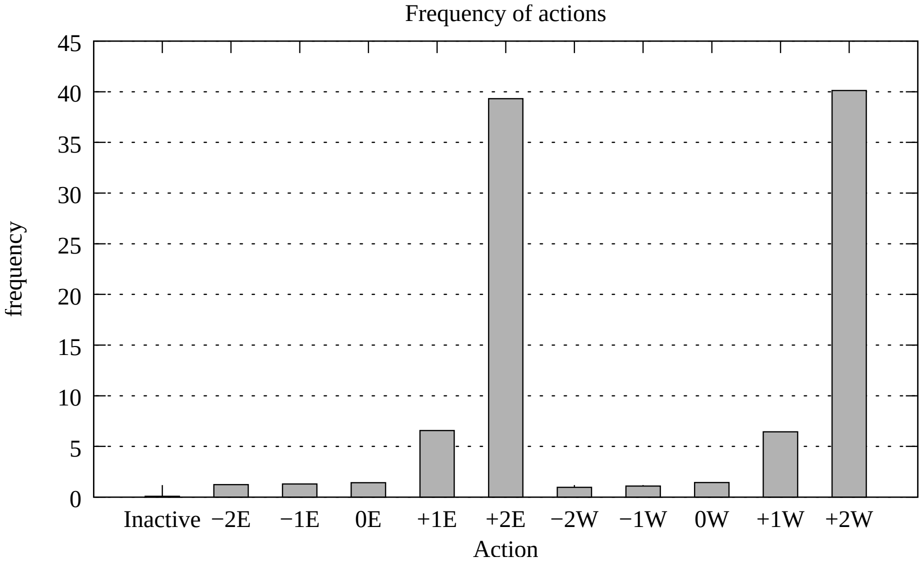


Figure 4.4: Frequency of each action chosen by a population of hard-coded modulating agents.

When viewed over the course of the agent lifetime it can be seen that a typical agent learns which actions result in reduced input levels. In the example shown in figure 4.5, the agent initially settles on a neutral action before settling on the most rewarding water action. The agent then alternates between this and the most rewarding energy action. Figure 4.6 shows the initial learning process before one output neuron wins over all the overs.

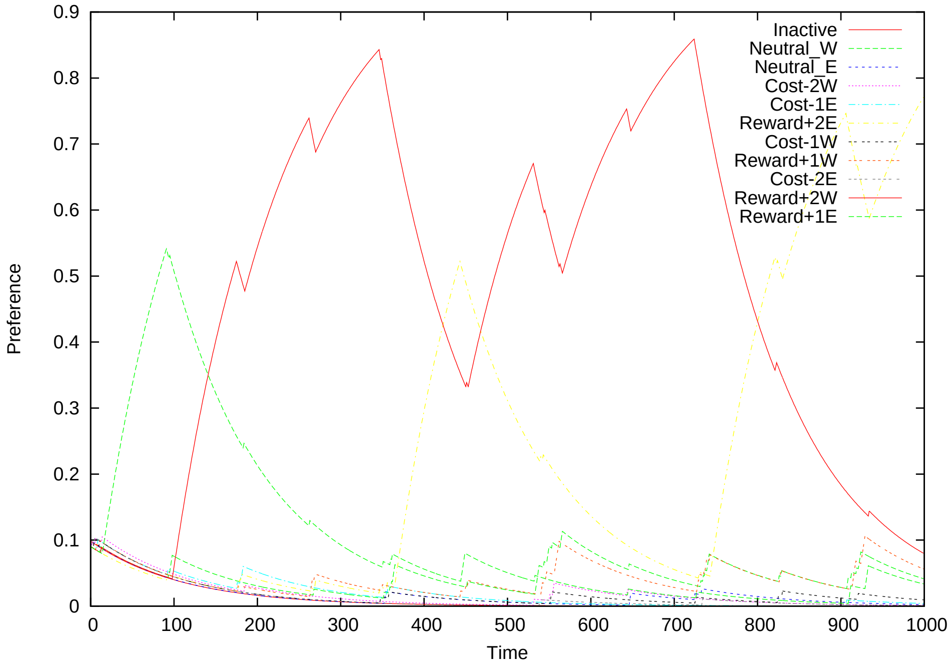


Figure 4.5: Actions chosen over the lifetime of a typical single modulating agent. This is typical of most of the agents in the population.

The performance of the non-modulating and the modulating agents were similar, although on average the non-modulating network would reach higher levels of fitness and would be optimised by the parameter search more quickly. The initial tests showed that both agents were learning and adapting which actions would minimise cost and maximise reward of their resource levels.

Even though the agents were optimised for the full set of actions, they were also tested with a subset of actions to show that they could perform cost-minimisation as well as reward-maximisation. The results of the cost-minimisation show that the network has difficulty minimising cost but nevertheless does consistently manage it to some degree.

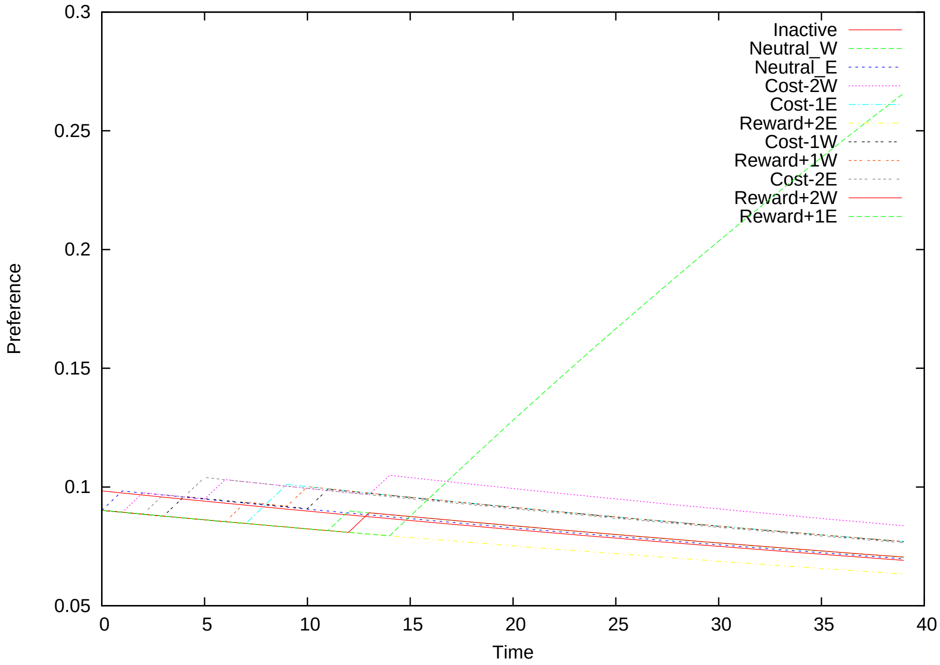


Figure 4.6: The first 40 cycles of the run in figure 4.5 showing the initial learning process.

```
Inactive...0.0714989%

Cost-2Enery...14.6167%

Cost-1Enery...15.3653%

Neutral_Enery...17.7995%

Cost-2Water...16.5317%

Cost-1Water...17.4343%

Neutral_Water...18.181%
```

Reward maximisation is significantly more successful:

```
Inactive...0.0715556%

Neutral_Enery...2.16822%

Reward+1Enery...5.61711%

Reward+2Enery...42.5876%

Neutral_Water...2.04333%
```

```
Reward+1Water...6.39622%

Reward+2Water...41.116%
```

4.3.2 Hebbian and anti-hebbian

Two variants of the model were found by evolution. An entirely hebbian variant, and another in which an anti-hebbian rule was used by the output layer. The non-modulating network used an entirely hebbian solution whereas the modulating network used the hybrid.

Anti-hebbian STDP is used here to describe the increase of a synaptic weight if the spike of a pre-synaptic neuron is *preceeded* by the spike of a post-synaptic neuron. If the order is reversed then a decrease occurs. Therefore if cell A does *not* take part in strengthening cell B then the synapse is strengthened.

Lisman [Lis89] describes a model of the mechanism underlying hebbian and anti-hebbian learning rules, although different terminology is used. The hebbian process used increases the synaptic weight when presynaptic and postsynaptic activity occur together. The anti-hebbian process decreases the weight when activity does not occur together. This is further defined as post-not-pre anti-hebbian process and pre-not-post anti-hebbian process. These decrease the synaptic weight according to the absence of either pre- or postsynaptic activity, but not the other.

4.3.3 Extended tests

During parameter optimisation, each genotype was tested for 1,000 turns before being evaluated by a fitness function. The evaluation was cut short if the genotype proved to be of exceptionally poor performance. This was determined by allocating a small level of resources at the beginning of each evaluation period[3]. If either the energy or water resource declined to zero then evaluation of the genotype was halted. The early death results in a lower age value when calculating the fitness level.

---

[3] In practise this was 500 points per resource.

During the initial examination of how well the agents performed, it was noticed that network activity would decrease over time for the non-modulating agents. This did not happen for the modulating agents. Further tests were made to compare the performance of the modulating and non-modulating agents for evaluation periods exceeding that for which they were evolved for.

The average performance of two populations were compared. One population was made up of modulating agents, the other of non-modulating agents. All members of a population were created from a single optimised hard-coded genotype.

When testing a population of non-modulating agents for longer than 1,000 turns, spiking-activity in the network would cease over time. This led to the weights freezing because the STDP learning rule only updated the weights when spikes occurred. The activation of the output neurons would slowly decay over time with the winning action remaining the same in the absence of any change in the effect of that action; see figure 4.7. The limited use of artificial evolution for parameter optimisation had settled upon a brittle strategy which depended on how long each agent was evaluated for.

A population of modulating agents were then tested for the same extended period of time. They were shown to continue transitioning between the same two winning output neurons that caused a maximum increase in energy and water, with other neurons very occasionally being chosen; see figure 4.8.

Unlike the non-modulating agent, spiking activity did not cease. Consequently the weights continued to be updated. Modulation had prevented artificial evolution, used for the parameter optimisation, from overfitting the test environment. It had achieved this by exciting the neurons and causing them to continue spiking even when the absence of any modulators would have caused spiking-activity to cease.

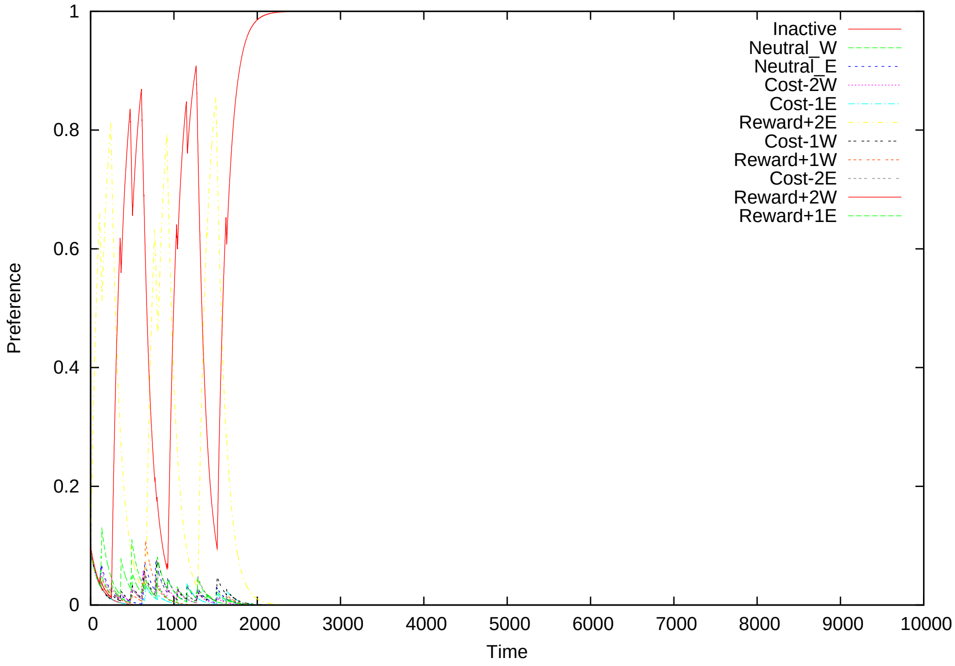


Figure 4.7: Non-Modulating agent run over an extended period of time (10,000 turns). The x-axis denotes the number of turns that the agent was evaluated for, at each turn the agent could choose a different action. The y-axis denotes the frequency of that action. Choosing an action increases its frequency resulting in the decrease of frequency for the other actions.

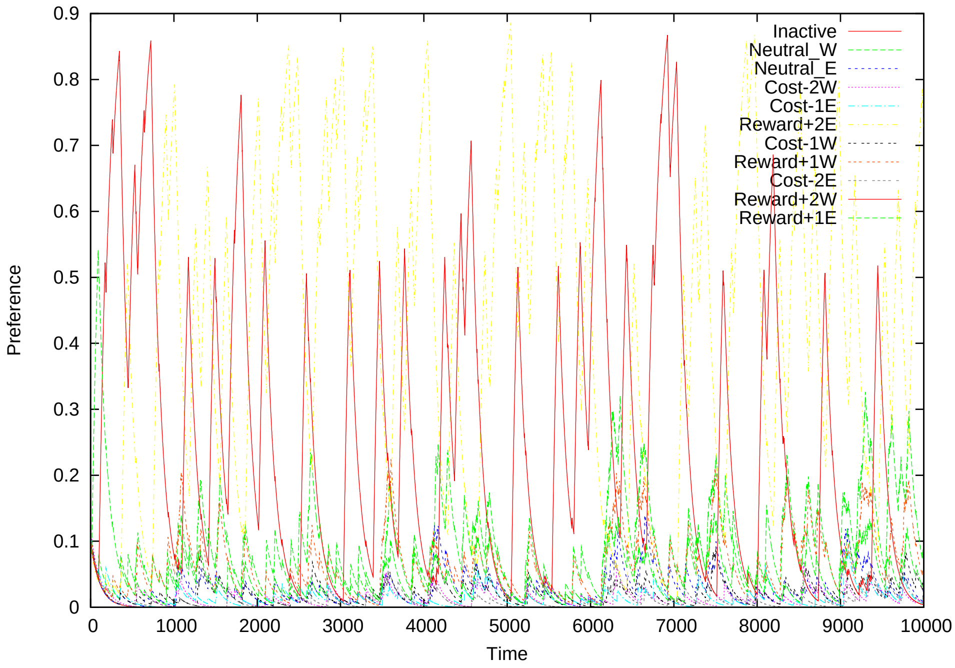


Figure 4.8: Modulating agent run over the same extended period of time and using the same axes as in figure 4.7.

4.3.4 Re-optimisation

In the previous experiment it was shown that optimising parameter sets using fixed length evaluations led to a difference between how the modulating and non-modulating agents functioned. In order to make the agents more similar, new evolutionary runs were started, this time using variable length evaluations. Variable length evaluations would allow further comparisons to be made between modulating and non-modulating agents that could not be ascribed to the time period of the evaluation.

During automated parameter optimisation, a genotype would be mapped to an agent and run for a randomly determined period of time up to 1,000 cycles. At the end of the evaluation period the fitness of the genotype would be determined. The score was then divided by the number of cycles that the genotype was evaluated for so as not to give an unfair advantage to genotypes that were evaluated for longer.

With the non-modulating genotype, this produced a hybrid anti-hebbian / hebbian solution as originally evolved for the modulating agent. Both the modulating and non-modulating agent performed the maximum of ten allowed iterations of the network per cycle. Comparisons between the newly evolved genotypes showed that the non-modulating version slightly out-performed the modulating version.

This shows that although neuromodulation can decrease performance it stopped evolution from overfitting the environment in which it occurred. Modulation had stopped the agent from settling into a stable state and caused it to constantly re-learn its environment. But when both modulating and non-modulating agents were evolved for environments that required them to constantly re-learn throughout their lifetimes, modulation was no longer needed to provoke activity within the network.

4.3.5 Analysis of initial results

It was discovered that the network performed most effectively when the actions it chose could minimise input activity. Wörgötter and Porr [WP04] provide an overview of the field of temporal sequence learning. They discuss how the learning paradigm of disturbance minimisation, as opposed to reward maximisation, removes the problem of credit structuring and assignment. The two paradigms are not equivalent. Whereas maximal return is associated with a few points on a decision surface, minimal disturbance uses all of the points. In a minimal disturbance system, every input into the system drives the learning process. If there is no signal then the system is seen as being in a stable state. Rewards and maximal return are not sought, as is the case with credit assignment learning. Instead, any disturbance-free state is satisfactory.

With the non-modulating network, a decrease in input activity eventually leads to a decrease in output activity. If an action results in the lowest possible strength of input signal and spiking-activity in the network has already declined to the minimum threshold required for learning, then the network settles into a stable state. This occurs in the absence of any changes external to the system, such as the effect of an action changing or noise being added to the input signals.

Modulation agitates the network, stopping it from settling into a stable state where activity declines to a point whereby the network stops alternating between actions. When tested using an extended run, the modulating network, unlike the non-modulating version, continues to alternate between the actions causing the least input disturbance throughout its lifetime. Figure 4.8 shows that other actions always have a chance of being selected.

When comparing the modulating and non-modulating agents in environments that they were not evolved for, in this case evaluated for a variable or extended length of time, then it is shown that modulation makes the agent more robust. This robustness carries with it a performance cost.

This suggests that one functional use of neuromodulation is to provide agitation to the agent in order to not let it settle into a stable state; even though the environment may allow for it or make this the optimal behaviour. An explanation for this could be that natural agents have not evolved for such environments because they rarely exist and cannot be relied upon to last.

4.3.6 Discussion of initial results

The use of hunger and thirst input signals are not intended here to be analogous to emotions. They should instead be seen as inputs that drive the agent to perform certain actions. Each drive increases the level of a corresponding modulator. Each modulator decays slowly over time and therefore temporally regulates the initiating drive by inhibiting and exciting the neurons that receive those input signals. The modulator accentuates activity and continues to agitate the agent when it would otherwise be too low to fire neurons. It also attentuates activity when it would otherwise saturate the network. In effect modulators are acting here as a narrow band activity filter.

Minsky argued that an agent will eventually suffer from goals that compete with one another if they persist for long enough [Min88][4]. Section 2.6.3 discusses how emotions can be seen as fulfilling the function of modulating the goals, needs and behaviours of an agent. It has been shown here that modulators can arbitrate between competing actions, letting less successful actions be executed when the agent would otherwise perform only the most successful ones. It is also plausible to suggest that as modulators persist over time, they could also be used to focus the agent upon one set of actions or behaviour to the exclusion of others when they would otherwise need to compete.

4.3.7 Variance of fitness

The mapping from genotype to phenotype is stochastic. Therefore an average fitness derived from an equal number of evaluations is required in order to make fair comparisons between different agents or different genotypes. But how do the fitness levels vary throughout the population? For example, is it the

---

[4] The full quote by Minsky can be found in section 2.1.2.

case whereby there are only a few good genotypes in the population but their fitness levels are sufficiently large to increase the average fitness by an inordinate amount? Or are all the agents within the population able to adapt and survive in the environment to some extent? The latter case is preferable to the former so that the correct selection of the fittest agent from the population becomes less critical.

To test for this, rather than use a population of 450 phenotypes, a single agent was tested 450 times. Each time a different random seed was chosen and the genotype was mapped to a new phenotype and run. The resulting fitness level was plotted each time. The genotype chosen had been optimised using variable length evaluations [5]. Both the modulating and non-modulating agent were tested, see figure 4.9. In order to more clearly discern the distribution, the fitness levels were sorted in ascending order and re-plotted, see figure 4.10. The graph shows an even distribution of fitness throughout the population with only a slight difference between the non-modulating and modulating agent. The genotype of the non-modulating agent has a slightly higher fitness on average than the modulating agent, but it also produces phenotypes of poorer fitness levels. The genotype of the modulating agent will more likely be expressed as a phenotype with an average fitness level than that of the non-modulating agent. Although the difference is minimal and by itself not sufficient evidence to draw any conclusion, it is in keeping with the idea of modulation acting as a form of regulation; stopping an agent from overfitting its environment or letting activity die out.

4.4 Tests with noise

4.4.1 Reasons for testing

To show that the network was adapting to its inputs and not merely creating a standard pattern generator to poll particular outputs, noise was added to

---

[5] See section 4.3.4.

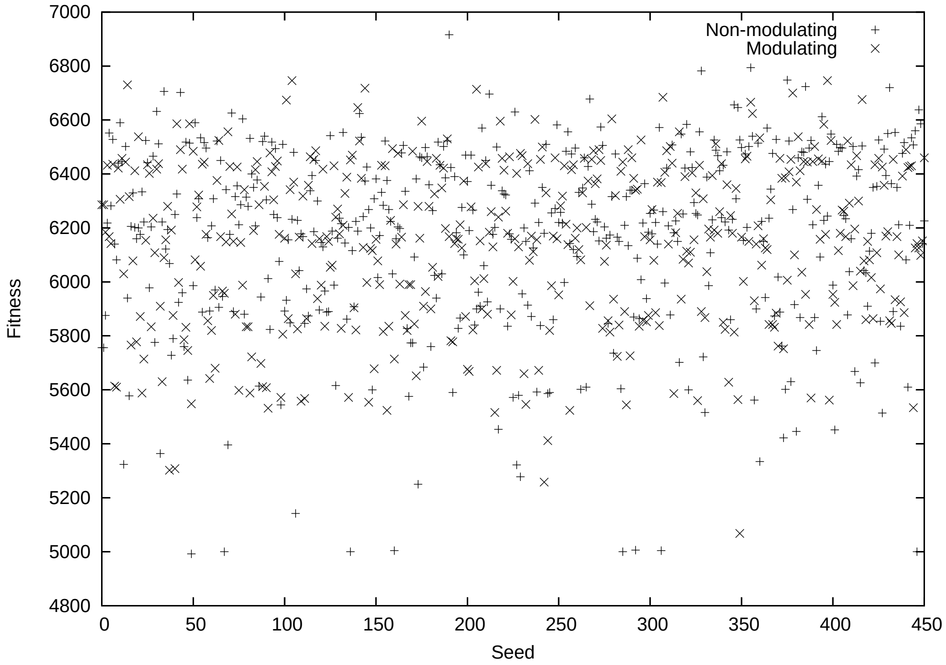


Figure 4.9: A single modulating and non-modulating agent were both tested 450 times, each time with an incremented random seed. The fitness of the agent was recorded.

the input signals. A gradual degradation in performance correlating with an increase in noise would be further proof that learning was occurring.

Because the networks were optimised in a noise-less environment, testing them in this way would show how robust they were. If they were not robust to noise at all then case selection using programming logic would be simpler and more efficient.

Another reason was to test whether modulation made an agent more or less robust to noise. Although the initial tests show that modulation agitates the network to stop evolution from overfitting the environment, this does not necessarily mean that the effect of modulation upon the network is the same as that of noise applied to the input signals. Noise is random whereas modulator strengths are increased for a reason and decay consistently over time. Because neuromodulators have a longer lasting effect than neurotransmitters and influ-

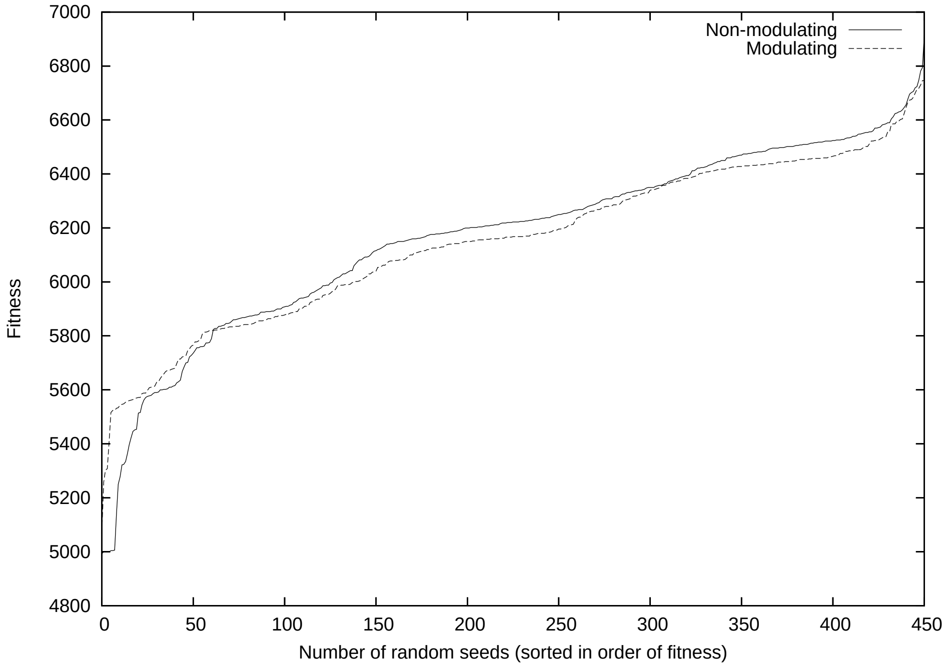


Figure 4.10: Figure 4.9 sorted in order of ascending fitness.

ence a larger area, it was originally suspected that they may provide a stabilising influence with regard to erroneous input signals. Alternatively, if the function of modulation is to agitate and destabilise the neural network so as to stop it settling into a stable state, then modulation could be largely redundant within a noisy environment. Comparing the two agents would show which, either or both, is the case.

4.4.2 The tests

In the following tests, noise is added to the input signals before being passed to the network. Internal sensory signals are constrained within the range [0:1].

The level of noise to be added to the internal sensory signal is specified by a single parameter to be within the range $[-P{:}+P]$. A random number is drawn from this range and applied to the input signal. This results in the error either decreasing or increasing the signal.

Four agents were tested; modulating and non-modulating agents evolved using fixed length evaluations and the same again evolved using variable length evaluations. All four are the same agents that were tested in sections 4.3.1 and 4.3.3.

An agent is tested 101 times with the noise level P incremented each time by 0.01 in the range [0:1]. At the end of the run the total resources gained or lost by the agent at the end of its lifetime, is plotted against the noise level. Because the genotype is stochastic the results are averaged over a population of 450 phenotypes[6].

4.4.3 The effect of increasing noise levels

The total resources of an agent, gained by the end of its lifetime and averaged over a population of 450 phenotypes, can be seen in figure 4.11. The two graphs compare the modulating and non-modulating agents first evolved using fixed length evaluations and discussed in section 4.3.1.

With little or no noise the non-modulating phenotypes perform better on average than the modulating phenotypes. But when subjected to higher levels of noise[7] the modulating phenotypes perform better than the non-modulating ones. It can clearly be seen that there is a more graceful degradation of performance with the modulating phenotypes when faced with increasing levels of noise than that of non-modulating phenotypes.

Although the benefit of this only applies to noisy environments with the costs being incurred when in the absence of noise, this is still a useful feature of modulation. It is plausible to design an agent that can increase its level of modulation to cope with increasing levels of noise encountered in the environment. An increase in modulation by the agent may even appear to an observer as being similar to an emotional response as the agent increasingly ignores the subtleties sensed in its environment.

---

[6] 'Agent' is used here not to describe a specific instance but the architecture and set of parameters encoded in the genotype. A phenotype is an instance of this.

[7] Much higher in the case of the modulating agent evolved using fixed length evaluations.

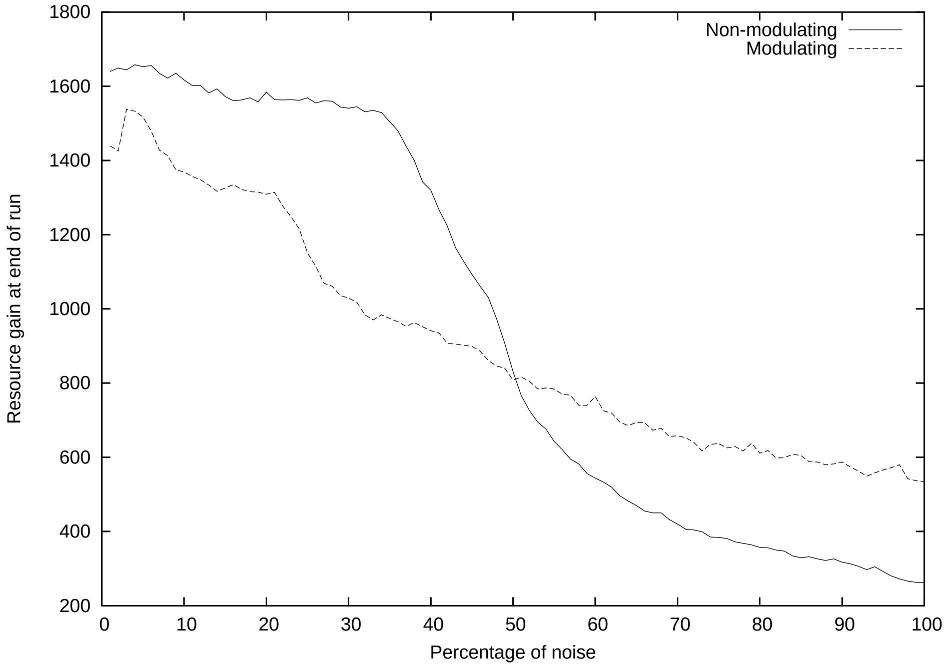


Figure 4.11: Noise applied to agents evolved using fixed length evaluations. Non-modulating agent (Highest=1658, Lowest=262, Range=1396) and Modulating agent (Highest=1538, Lowest=533, Range=1005). The total resources gained by the agent (energy + water), averaged over a population of 450 phenotypes, is read at the end of each run. The y-axis corresponds to resource level, the x-axis corresponds to noise. Even though the modulating agent performs worse in the absence of noise, it copes significantly better than the non-modulating agent as the noise increases significantly.

An interesting feature gleaned from the graph is that both agents, especially the modulating agent, perform at their best when noise is added to the input signal in small amounts. Better than in the absence of any noise at all. This is an example of self-organisation being facilitated by random perturbations [Hey00].

Although the degradation of the modulating agent for variable length evaluations is not as graceful as that for fixed length evaluations, it still performs significantly better than the non-modulating agent for almost all levels of noise. The non-modulating agent performs marginally better than the modulating agent in the presence of low levels of noise.

The exact values can be compared below. Both modulating agents, evolved using fixed and variable length evaluations, outperform the non-modulating agents at higher levels of noise. The agent most robust to noise, and the best performing in the presence of high noise levels is the modulating agent evolved using fixed length evaluations.

	Highest	Lowest	Range
Non-modulating fixed	1658	262	1396
Modulating fixed	1538	533	1005
Non-modulating variable	1552	227	1325
Modulating variable	1535	334	1201

The performance of both the modulating and non-modulating agents degrades gracefully under the increasing presence of noise. This is further proof that the network is adapting and that learning is taking place. The tests provide evidence of another function that can be ascribed to modulation, that of increasing robustness to the effect of noise. And if modulation increases robustness to noise, this rules out the idea that it stops a network overfitting its environment by providing noise. Instead it is better to think of modulation as providing agitation to a system that seeks to minimise disturbance to its input.

4.5 Tests with rates set at zero

4.5.1 The original investigation

Up to this point the modulators had been constrained to modulating, by use of receptors, either the sensitivity of a neuron to its input or its probability of firing. Doya describes how adaptive agents rely upon a number of parameters in order to adapt successfully [Doy00] [Doy02]. These are referred to as metaparameters because of how they regulate the way that other parameters of the system change as learning takes place. In artificial agents, these metaparameters need to be tuned by a human. Doya argues that this is not the case with natural agents

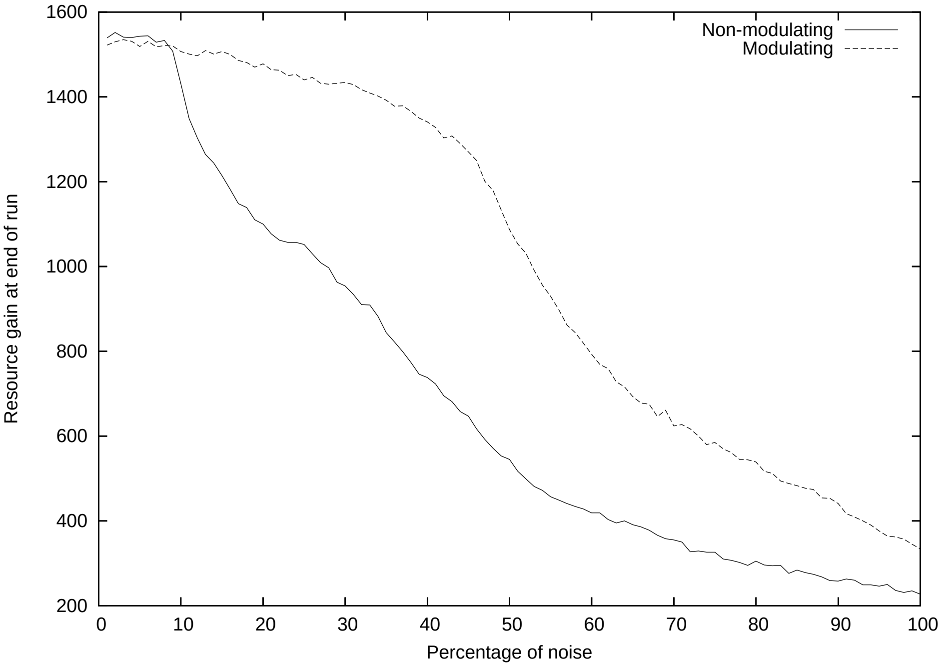


Figure 4.12: Noise applied to agents evolved using variable length evaluations. Non-modulating agent (Highest=1552, Lowest=227, Range=1325). Modulating agent (Highest=1535, Lowest=334, Range=1201). The modulating agent performs significantly better than the non-modulating agent for almost all levels of noise, only performing marginally less well in the presence of low levels of noise. Performance degradation is more graceful for increasing levels of noise than with the non-modulating agent.

which adapt to unknown environments without supervision.

The metaparameters used in the experiments in this thesis were tuned using automated optimisation. The closest equivalent of this occurring for a natural agent would be the evolution of its species. Doya proposes that the brain has the capability to adapt its metaparameters and that its neuromodulatory systems fulfill this role. The roles of dopamine, seratonin, noradrenaline and acetylcholine are theorised about within the reinforcement learning framework. If this is true then not only is it an important area of research that needs further exploration, but it might also simplify the process of automated optimisation.

The STDP hebbian learning rate was picked as the first metaparameter to try modulating. If successful this would provide a means for using emotion to determine the importance of what an agent should learn. Hasselmo proposes a way that acetylcholine might modulate memory formation in the hippocampus [Has99].

The experiments failed, but for a surprising reason. The network still adapted even with a zero hebbian learning rate. To make sure, the tests were re-run with all code updating the synaptic weights commented out. The result was confirmed.

Up to this point it had been assumed that the agents adapted by setting their synaptic weights using hebbian learning. If this was not the case then how the agent functioned was misunderstood. This situation needed to be rectified if the functionality provided by neuromodulation was to be properly understood. Further tests were carried out to explore how well an agent could adapt with a hebbian learning rate set to zero.

4.5.2 Hebbian learning rate

In these tests the hebbian learning rate of the agents were set to zero. Cost actions were tested each turn to see if their effect would be randomly changed. So for example, an action that previously increased the energy resource of an agent by one amount could change to decreasing the water resource by another

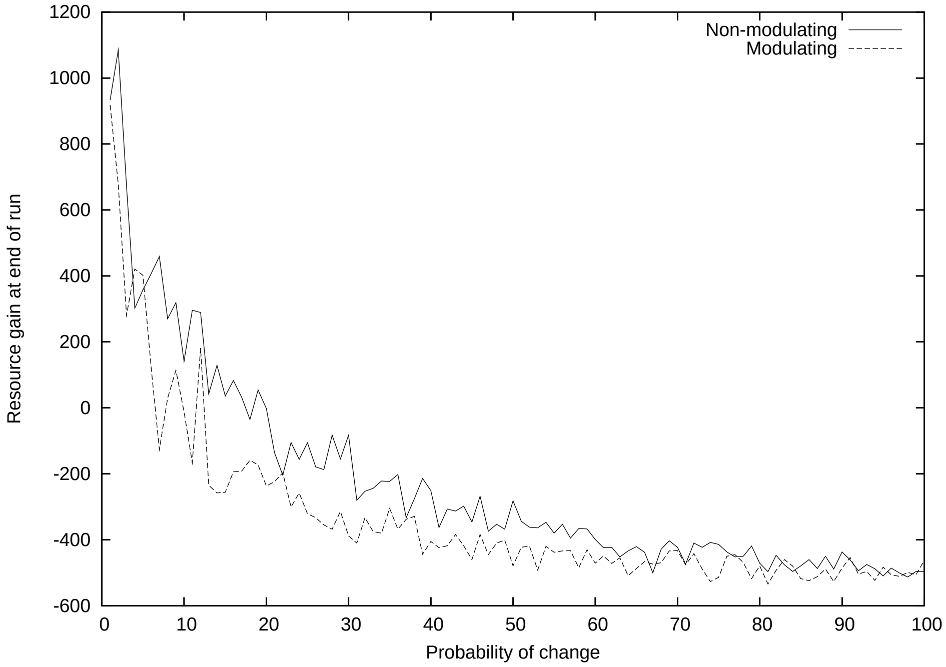


Figure 4.13: Random cost actions in tests using a population of 450 phenotypes. Hebbian learning rate is fixed at zero. Agents were evolved using fixed length evaluations. The non-modulating agent copes better with having to re-learn the effect of an action than the modulating agent.

amount. This change applied to the entire population of 450 phenotypes. This was a difficult test for the agent. Given that there were 10 actions to choose from, if there was a 10% probability of a change occurring then there was on average one action every turn that changed its effect.

The modulating agent evolved using fixed length evaluations showed a decrease in performance compared to its non-modulating counterpart, see figure 4.13. The performance of both modulating and non-modulating agent evolved using variable length evaluations were comparable, see figure 4.14.

4.5.3 All rates set to zero

In order to understand how the networks were still adapting despite not being able to update their synaptic weights, it was decided that all possible learning

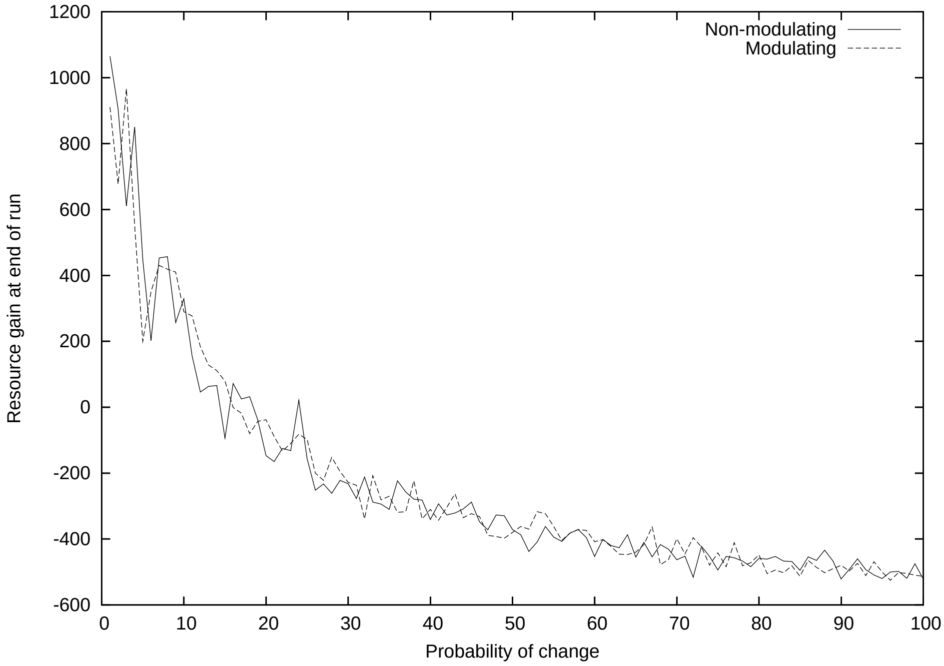


Figure 4.14: Random cost actions in tests using a population of 450 phenotypes. Hebbian learning rate is fixed at zero. Agents were evolved using variable length evaluations. The performance of the modulating agent and the non-modulating agent are comparable.

mechanisms should be switched off so that adaptation could no longer occur. Once this was achieved, the different mechanisms could be switched on in order to discern the exact effect that they had. The STDP hebbian learning rate, the increment / decrement rate to the spike-dependent leakage and the modulation rate were identified as potential learning mechanisms. Ordinarily each layer of neurons would use their own evolved rate.

Tests resetting individual rates to zero were inconclusive and suggested that the agents were learning via more than one mechanism. A further experiment was run whereby all the rates, the hebbian learning rate, spike-dependent leakage rate and the modulation rate, were set to zero.

Again the results were surprising in that the agents still showed signs of adaptation, albeit with a significantly reduced level of performance and consistency. The networks consisted of fixed synaptic weights and neurons with leakage levels fixed regardless of how frequently they spiked. Synaptic weights and leakage levels were randomly set when the connections or neurons were initially created and remained unchanged throughout the lifetime of the phenotype.

Disabling modulation by setting the modulation rate to zero showed that even though to some extent the modulating agent still adapted, the modulators were relied upon to sustain activity in the network. The disabled modulating agent performed worse than a control agent would that chose actions purely at random. This is because approximately 50% of the time, the disabled modulating agent did not choose an action and therefore the maximum cost default action was chosen for it instead. When re-tested with the evolved modulation rates, the agent had a performance similar to that of the non-modulating agent.

This use of modulation to sustain activity in the network was seen earlier when the agents evolved using fixed length evaluations were tested for an extended period of time, see figures 4.7 and 4.8.

Yet for agents evolved using fixed length evaluations the non-modulating agent is more likely to cope better than the modulating agent in changing en-

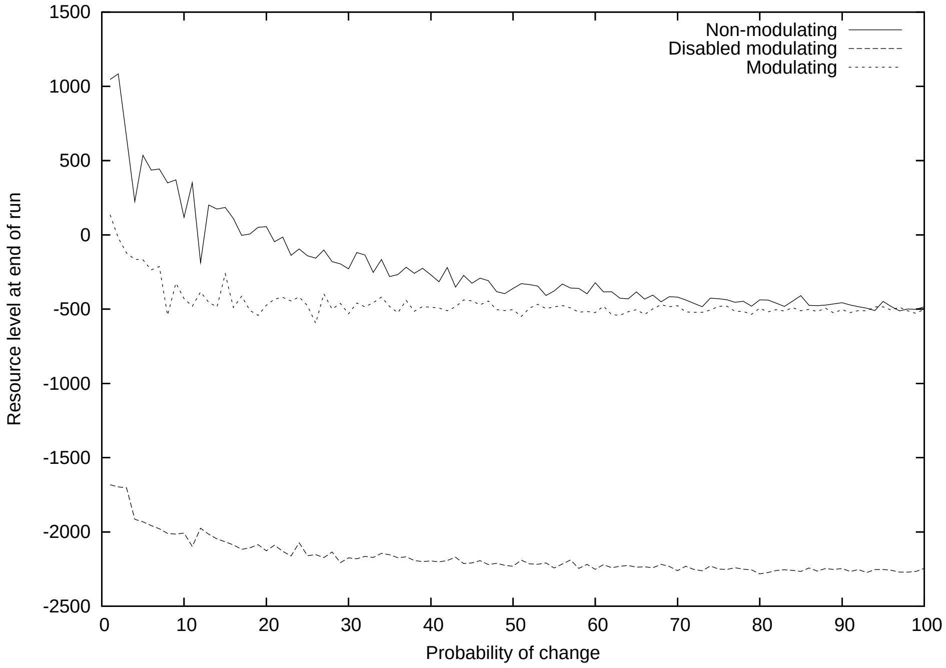


Figure 4.15: Random cost actions in tests using a population of 450 phenotypes. Hebbian learning rate, spike-dependent leakage and modulation rate are all fixed at zero. Modulation rate is set to those evolved for each layer for the enabled modulating agent. Agents were evolved using fixed length evaluations. The non-modulating agent copes better with having to re-learn the effect of an action than the modulating agent.

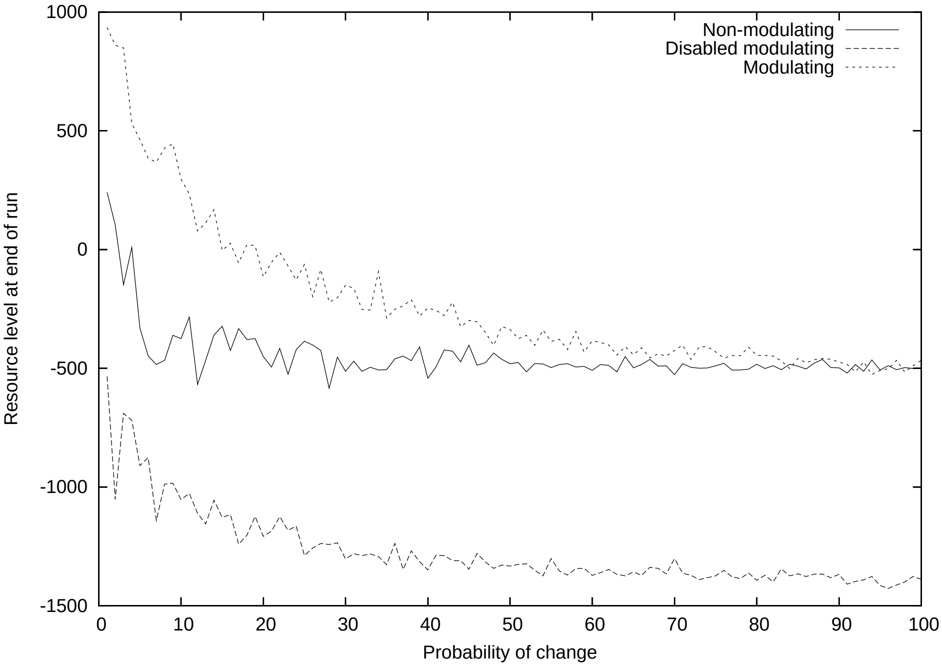


Figure 4.16: Random cost actions in tests using a population of 450 phenotypes. Hebbian learning rate, spike-dependent leakage and modulation rate are all fixed at zero. Modulation rate is set to those evolved for each layer for the enabled modulating agent. Agents were evolved using variable length evaluations. The modulating agent copes better with having to re-learn the effect of an action than the non-modulating agent.

vironments, see figures 4.13 and 4.15[8]. This suggests that the non-modulating agent is constantly perturbed by the changes in the environment, forcing it to re-learn. Modulation perturbs the network regardless of how frequently the environment changes and can be used as a tool to stop agents overfitting their sensory inputs.

For agents evolved using variable length evaluations, the modulating agent was either comparable to (figure 4.14) or better than (figure 4.16) the non-modulating agent. Figure 4.17 shows how the agent evolved using variable length evaluations relies more upon modulation than its counterpart evolved using fixed length evaluations. This suggests that modulation has been useful in coping with unpredictability in the environment. This is probably because activity was sustained within the network, perturbing the agent to switch more rapidly between the two actions required to maximise resources. This would reduce the probability of an evaluation being stopped while one resource was significantly larger than the other, thus reducing fitness.

When the probability of change is too high for the agents to adapt, they perform worse than a purely random agent that chose an action every turn. If an agent does not choose an action then the inactive action is chosen for it by default. The default inactive function is more costly than if the agent chose an action itself. The base performance of the agents when tested with random cost actions at the highest probability of change is approximately a net loss of 500 resource points throughout the lifetime. This shows that the the agents are not adapting in such an environment and also not always choosing an action.

4.5.4 Stopping adaptation

Weight distribution

If the agents could still adapt without updating their synaptic weights and without using a spike-dependent leakage, then this raised the question of how

---

[8] A negative resource rating shows a net loss of energy and / or water over the course of the lifetime.

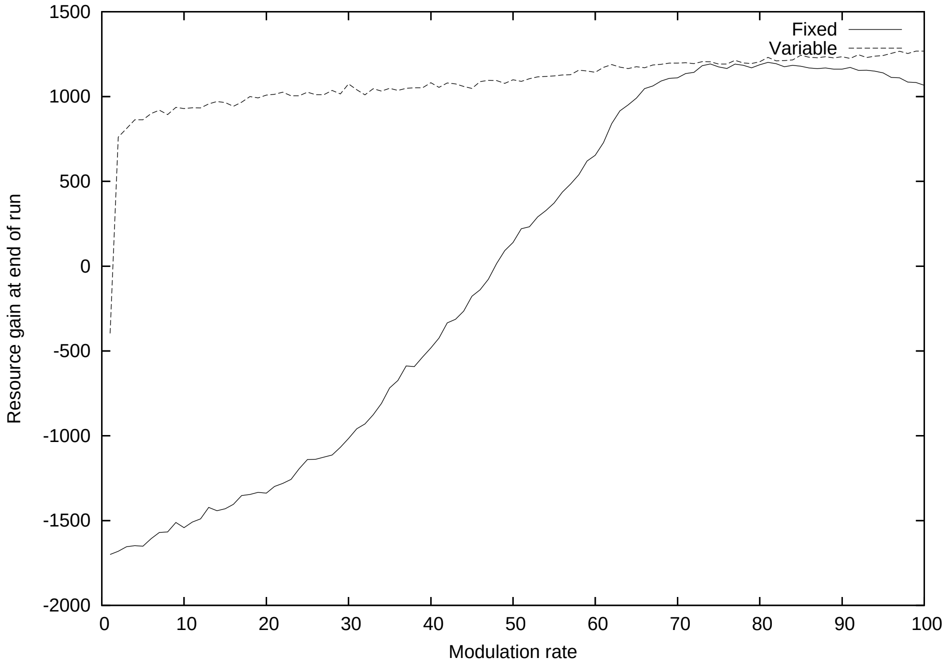


Figure 4.17: Agents evolved using variable length evaluations rely more on modulation than agents evolved using fixed length evaluations.

they were adapting. Finding out what stopped the agents from adapting would answer this question.

It was found that adaptation stopped when all the synaptic weights were initialised to the same fixed value, in this case 1. Ordinarily they would be initialised to a random value. Each action was performed approximately the same number of times, much the same as if a control agent chose its actions at random.

The performance of the agents could also be significantly compromised by disabling partial connectivity between layers. This meant that every neuron in a layer connected the same number of times as its neighbours to every neuron in a target layer. This did not fully stop the agent from adapting but did inhibit it from performing both cost minimisation and reward maximisation for both resources. This is more evidence that the randomness of the weight distribution is fundamental to adaptation occurring.

Isolating the adaptive mechanisms

Once adaptation had been stopped, each potential adaptive mechanism was used in isolation. This showed whether they were enabling the agent to adapt, whether they were ignored or whether they provided another function. The weights were fixed at 1 for each test and the initial leakage was fixed at 0. When testing the effect of STDP in isolation, the weights were only initially set to 1 and were allowed to change thereafter. The results were averaged over a population of 450 agents. Further tests using just a single agent showed that although the results were less consistent, adaptation still occurred through the use of these mechanisms.

Figure 4.18 shows the result of how frequently each action was performed throughout the course of a run. The results are from tests involving a population of non-modulating phenotypes mapped from a genotype evolved using fixed length evaluations. The sum frequency of all the actions totals 100% or less. The frequency of the default inactive action, chosen for the agent when it does

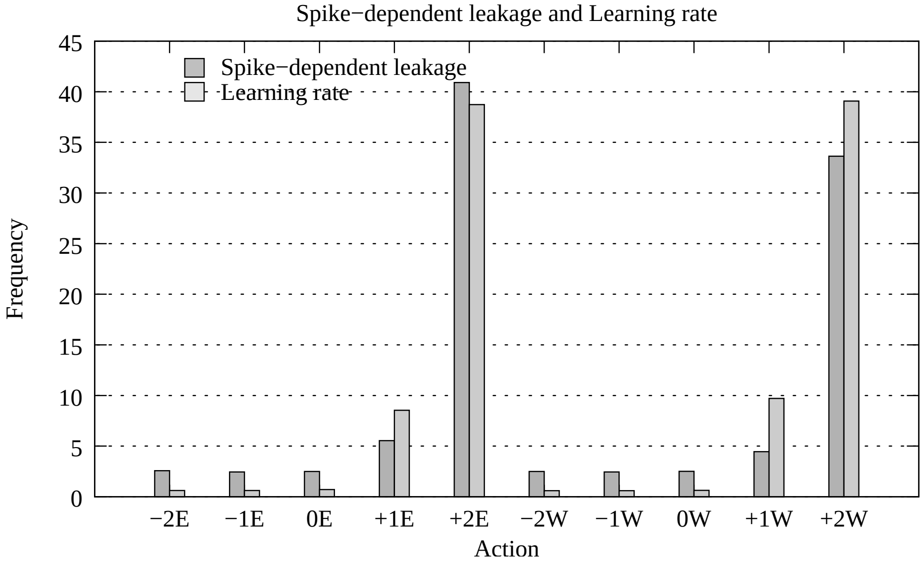


Figure 4.18: Non-modulating agent evolved using fixed length evaluations. Average proportion of actions performed over a population of 450 phenotypes. Both mechanisms are used for adapting to both resources. Synaptic weights and initial leakage are fixed in order to stop adaptation without these mechanisms.

not choose itself, is not shown.

The agent uses STDP hebbian learning and spike-dependent leakage to adapt. This shows that both hebbian learning and spike-dependent leakage provide the same function. Spike-dependent leakage allows adaptation to occur as much as with hebbian-learning even though all the weights are fixed at 1 throughout the test.

The non-modulating agent evolved using variable length evaluations adapts using spike-dependent leakage; see figure 4.19. Hebbian learning makes no difference to how well it adapts. The agent probably evolved to not make use of hebbian learning so as to stop it adapting too much and overfitting the environment.

The converse is the case with the modulating agents. For the agent evolved using variable length evaluations, both mechanisms are used for adapting and increasing both resources, see figure 4.20. Interestingly, for the modulating agent evolved using fixed length evaluations, both mechanisms are still used, but each

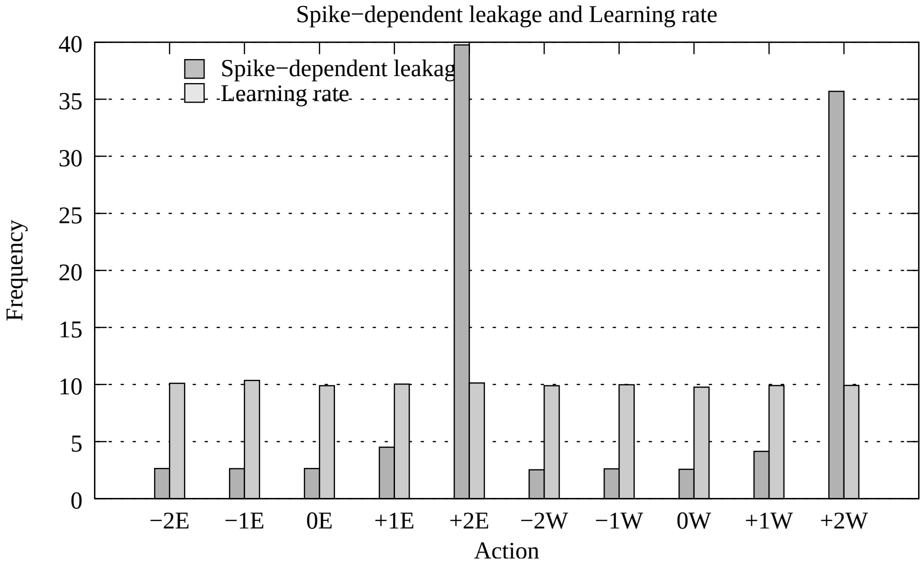


Figure 4.19: Non-modulating agent evolved using variable length evaluations. Average proportion of actions performed over a population of 450 phenotypes. The agent adapts using spike-dependent leakage. Hebbian learning makes no difference to how well it adapts. Synaptic weights and initial leakage are fixed in order to stop adaptation without these mechanisms.

mechanism provides adaptation for only one resource. For this agent, hebbian learning is used for adapting which energy actions are used, and spike-dependent leakage is used for adapting which water functions are used; see figure 4.21. At first glance it seems as if modulation is allowing the agent to adapt more in a dynamic environment than compared to a static one, but this might not be the case.

The non-modulating agent evolved using fixed length evaluations was allowed to overfit its environment. Evolution using variable length evaluations prevented the other non-modulating agent from doing the same. That agent evolved to only use spike-dependent leakage and to not use hebbian learning. Fixed length evaluations allowed for the full use of all adaptive mechanisms, whereas evolution using variable length evaluations required the adaptive mechanisms to be partially inhibited. There is no reason to suggest that this was any different for the modulating agents.

It is plausible to argue that a mechanism can allow for more effective adaptation if it only has to learn to increase one resource rather than two. In this case, the modulating agent evolved using fixed length evaluations can be seen as adapting more fully because it can devote each mechanism to a different resource. The fact that the non-modulating agent does not do this as well suggests that modulation is performing the function of context-switching. This raises the question as to why the non-modulating agent performs better than the modulating agent. This is probably because modulation also acts to agitate the network and this stops it from adapting fully to its input signals.

This still leaves the question as to why the modulating agent evolved using variable length evaluations can use both mechanisms for adaptation, unlike its non-modulating counterpart. Modulation can regulate spiking-activity, both exciting it when it would otherwise decrease, or inhibiting it when the network threatens to overreact to the input signals. Because both the adaptive mechanisms, hebbian learning and spike-dependent leakage, rely upon spiking activity, it can be argued that modulation inhibits both mechanisms from overfitting the input signals by increasing or decreasing spiking-activity.

Figure 4.22 shows the effect of modulation by itself in the absence of any other adaptive mechanisms. As with the other tests, all synaptic weights are fixed at 1 in order to stop adaptation. This is compared to the same agent being used without modulation. Modulation excites spiking-activity in the network and makes the agent more likely to choose an action rather than have the default inactive action chosen for it (not shown on the graph). Unlike hebbian learning and spike-dependent leakage, modulation does not enable the agent to learn the effect of its actions. This shows that modulation is used purely for regulation of a system that is already adapting.

Adaptation through inhibition

How can the network adapt without any of the adaptive mechanisms, spike-dependent and hebbian learning, being enabled? The answer lies in asking

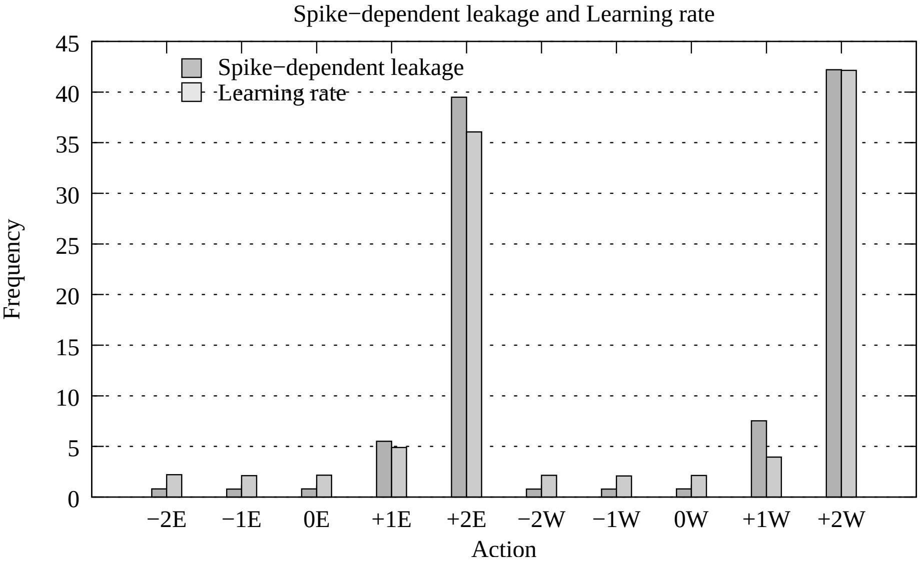


Figure 4.20: Modulating agent evolved using variable length evaluations. Average proportion of actions performed over a population of 450 phenotypes. Both mechanisms are used for adapting to both resources. Synaptic weights and initial leakage are fixed in order to stop adaptation without these mechanisms.

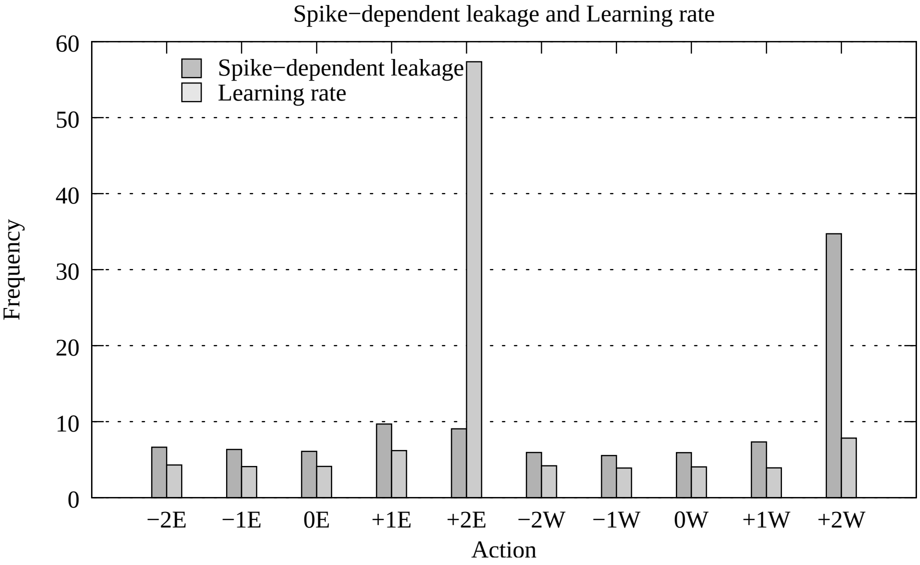
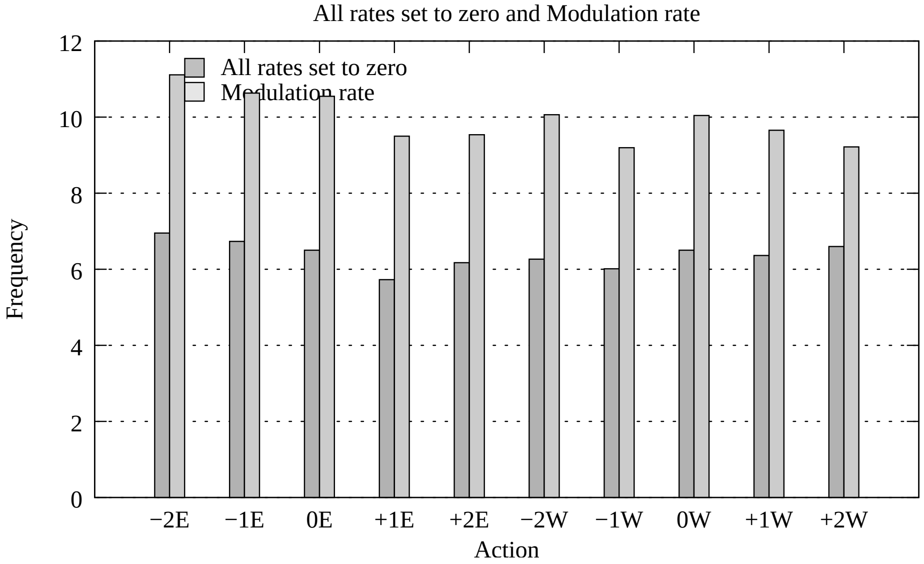


Figure 4.21: Modulating agent evolved using fixed length evaluations. Average proportion of actions performed over a population of 450 phenotypes. Each mechanism provides adaptation for only one resource. Hebbian learning is used for adapting which energy actions are used, spike-dependent leakage is used for adapting which water functions are used. Synaptic weights and leakage are fixed in order to stop adaptation without these mechanisms.



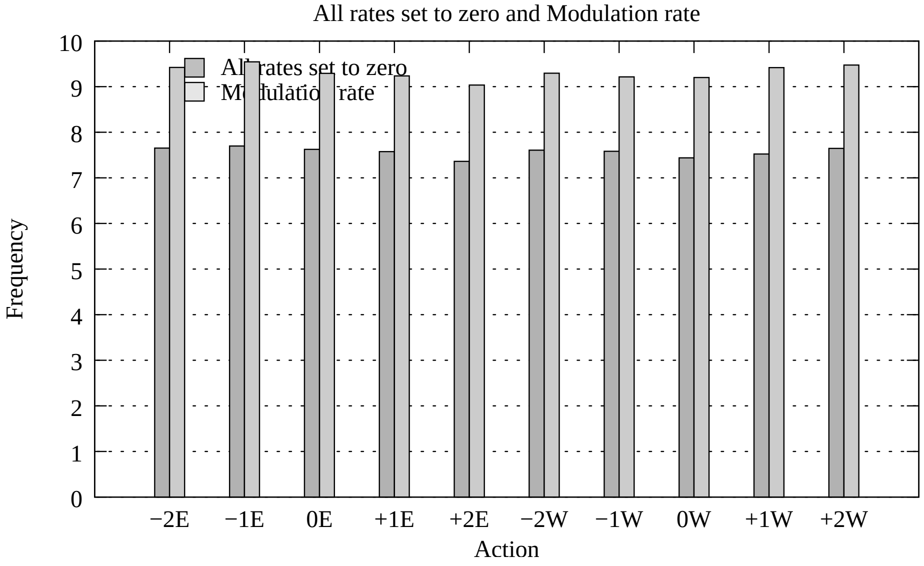


Figure 4.22: Top: Modulating agent evolved using fixed length evaluations. Bottom: Modulating agent evolved using variable length evaluations. Average proportion of actions performed over a population of 450 phenotypes. The effect of modulation by itself in the absence of any other adaptive mechanisms. Synaptic weights and initial leakage are fixed in order to stop adaptation. Modulation excites spiking-activity in the network and makes the agent more likely to choose an action rather than have the default inactive action chosen for it (not shown on graph).

another question; what does it mean to adapt?

Adaptation requires at a minimum, some internal state or variable to change in response to the external environment. Hebbian learning changes the state of the synaptic weights, and spike-dependent leakage changes the leakage of the neuron to carrying a charge. But these are not the only internal variables. Input signals also change the voltage levels of the neurons and the timing of the spikes can vary.

Adaptation only stops when all the weights are fixed at 1. This means that all the neurons reach the voltage threshold and fire at the same time. When the voltages of the neurons are varied, whether by randomising or updating the synaptic weights or by adapting the level of leakage, the neurons fire at different times. Some neurons have less leakage of their synaptic inputs and can reach the voltage threshold sooner, some later because of greater leakage.

When input signals are first applied to the network, because the neurons fire at different rates due to their varying resistances, some output neurons will fire before the others. If these do not lower the input signals in the next round then eventually the signals from the other neurons will filter through the network and cause other output neurons to fire. If an output neuron fires and subsequently causes the input signals to be reduced, then this reduces the input signals for other, possibly faster firing neurons. Adaptation is performed by indirectly inhibiting the other output neurons. Faster firing neurons that do not lower the input signals may have an initial advantage, but because of the connectivity between the three layers, are also likely to activate other weaker output neurons as well.

If the voltages of the neurons in the network were to be mapped onto a hyper-dimensional landscape with resulting spiking-activity leading to higher or lower peaks and basins, then adaptation can be seen as the process of moving into the lowest part of the landscape. Modulation agitates the network sufficiently so as to stop it settling into a single basin.

Using an analogy of a ball rolling along a peak, ridge or plateau, then given sufficient energy it will eventually roll down a slope. The ball will not be able to later return unless its kinetic energy is first increased. This process will continue until the ball comes to a stop at the bottom of the landscape, or within a local depression that requires too much kinetic energy for it to escape from.

Modulation can agitate the network by increasing or decreasing the sensitivity of a neuron to input or its probability of firing. This is the equivalent of increasing the kinetic energy of the ball by shaking it out of local minima and pushing it off course until it finds the lowest point on the landscape[9]. At this point the ball would come to a rest when using a non-modulating network, but with a modulating network the ball is still being shaken around within the depression [10].

This is not a new concept in neural network theory, an energy function was first used with Hopfield networks [HKP91] pp21. This allows an 'energy landscape' to be imagined whereby patterns memorised, being attractors in the system, can be seen as local minima in the landscape. As with the analogy of the ball, assuming the influence of gravity, a particle placed anywhere on this imaginary surface will roll down to the nearest basin.

Heylighen describes the two concepts of a fitness landscape and an energy-potential landscape as being equivalent [Hey00]. Biology conventionally uses the former concept whilst physics uses the latter. The energy landscape can be used to describe the dynamics of self-organising systems. Valleys correspond to attractors in a dynamical system[11], the speed that the system moves into them being determined by the steepness of the slope. If the landscape remains unchanged then the system will remain in the first minima that it settles in unless random perturbations or noise is used to push the system temporally upwards and out of it. Noise is generally more able to push a system out of local minima than out of the deepest valleys and therefore enables a system

---

[9] Modulation can be seen to increase the activity of a network in figures 4.15 and 4.16.

[10] This effect can be seen in figures 4.7 and 4.8. Activity in the non-modulating agent decreases to zero, whereas the modulating agent continues to try other actions throughout the course of the evaluation.

[11] Kauffman uses the landscape concept to describe dynamical systems[Kau93] pp176. Attractors, disjoint from each other in state space, can be seen as lakes with drainage basins.

to find the global minima. But noise also stops a system from settling at the bottom of a global minima once it has been found. If this is desired then the level of noise has to be reduced over time to allow the system to settle.

This helps explain why modulating agents perform optimally when a certain amount of noise is added to their input signals, see section 4.4.3. The optimal level of noise allows the system to find the global minima in the energy landscape, possibly by temporarily overcoming any dampening effect of modulation, yet not be sufficiently strong to escape from it. The non-modulating agent performs optimally at the same level of noise but the increase in performance is not significant. The difference between the performance of the modulating and the non-modulating agent decreases at this point. This suggests that at lower noise levels, the non-modulating agent can settle within the global minima more easily than the modulating agent, but at higher noise levels the modulating agent is better able to regulate its activity and not be pushed out of the global minima completely.

Adaptive mechanisms

The middle layer can be thought of as providing 'activity diffraction' to allow the signals to filter through the system at different speeds. The output layer can be thought of as providing 'activity integration', integrating those signals back into combinations that allow particular output neurons to fire more frequently than others.

How well adaptation occurs in a single agent without the use of adaptive mechanisms depends heavily upon the random seed that is used. This is due to the stochastic mapping from genotype to phenotype and how the weights are randomly initialised. Spike-dependent leakage and hebbian learning removes this dependency.

How the adaptive mechanisms improve the performance of the network depends upon where they are used and for which agent. They generally provide two different roles; to either strengthen the activity patterns that have been learned (exploitation) or to counter-balance this so that other neurons have a chance to fire (exploration).

For example, hebbian learning from the input to the middle layer strengthens the effect of 'activity diffraction' by moving individual weights towards either end of the synaptic weight range. Spike-dependent leakage counter-balances this effect with the levels staying at maximum most of the time and occasionally dropping to allow a spike to occur.

Spike-dependent leakage and anti-hebbian learning between the middle layer and output layer inhibit faster firing output neurons. But if these output neurons subsequently reduce the input signals and are less likely to fire, then the adaptive mechanisms will decrease the leakage of the activation delivered by synaptic input in order to make them more likely to fire in future. Synaptic weights between these two layers are more likely to be evenly spread over the synaptic weight range.

From the perspective of modulating neuronal activity

When introducing the electrical properties of neurons, Koch [Koc99] starts by discussing the RC circuit as the simplest model of a neuron. This is a unit that consists of a single leakage and capacitance. If given a synaptic input then the RC circuit can be used to perform the nonlinear functions of divisive normalisation and gain control. Keeping this perspective it can be argued that synaptic update, spike-dependent leakage and global neuromodulation all perform the same function; that of modulating the charge contained in this capacitance. The difference between these methods of modulation lies not just in how widespread their sphere of influence is but in how widespread their sphere of receptiveness is.

The modulatory effect of a spike-dependent leakage is influenced by the spiking of a single neuron. The modulatory effect of synaptic update is influenced by the spiking of all the pre-synaptic neurons that connect to the post-synaptic neuron. The modulatory effect of global neuromodulators is influenced by any part of the system that can increase the level of that neuromodulator.

When an animated display is produced of the neural network in action, it can be seen that spike-dependent leakages, synaptic weights and levels of neuromodulator can change rapidly and continuously. This is not consistent with the idea of synaptic update acting as a slow and permanent method of learning. It may be the case that learning in an agent can be implemented using slow changes to synaptic weights, but it must not be assumed that that this is the only function provided by synaptic update.

All three forms of modulation, spike-dependent leakage, synaptic update and global neuromodulation can regulate the firing rate of a neuron based upon the history within its sphere of receptiveness. A spike-dependent leakage can be used to regulate the firing of a neuron to stop it firing too much or too little. These experiments have shown how the use of neuromodulators stop a network from overfitting the sensory input signals, and also to sustain activity within the network when it would otherwise desist. In these networks synaptic update performs a similar regulatory function as spike-dependent leakage. The difference between the three forms of activity regulation lie in the scale of their use. Spike-dependent leakage, synaptic update and neuromodulation inhabit different parts of the same spectrum.

4.6 Other agents

The agents in this section are more complex than the internal-sensing agents described earlier in this chapter. There are various reasons why they have not been written up in-full. Some were successful but time constraints meant that they could not be fully analysed. Others adapted but have not yet produced results interesting enough to be written up. Some ideas were simply not fruitful. It is important to write up failed scientific endeavours if it can be shown why they are failures because this stops other researchers making the same mistake. But if it is not known why the experiment has failed, it may still be of interest to other researchers, if only to warn that the endeavour may be more difficult than first anticipated.

4.6.1 Context switching

It was originally envisaged that if neuromodulators can be seen as the brain's closest equivalent of global variables [Koc99], then they might be useful as a form of context switching. For example, if an agent's energy level reaches dangerously low levels then the hunger modulator is increased and the agent changes its behaviour in order to find food. This is why the internal-sensing agents were given two modulators instead of one. With these agents, the modulator levels were observed to change rapidly and continuously, as with the synaptic weights. This did not match with what one would expect from a neural network that used modulators as global signals and synaptic weights to encode learnt patterns.

Real neuromodulators diffuse in the brain and are relatively slow. To allow for this a reuptake rate was used which decreases the strength of the modulator by a fixed ratio over time. Evolution decided upon reuptake rates that were faster than expected. If spike-dependent leakage, hebbian learning and modulation can be seen as regulating the activity of the network using different scales of influence and receptiveness, then it can be expected that their levels will change continuously over time.

Although the diffusion of neuromodulators, and any messengers that are triggered by metabotropic receptors, is *relatively* slow compared to neurotransmitters, their rate of diffusion can still range over several orders of magnitude. This can be from hundreds of milliseconds to minutes, Koch [Koc99] pp97. Evolution was constrained to the use of only two modulators in the experiments described in this chapter. The strength of these modulators were increased only by input neurons, and receptors for them were used only in the middle layer. When unconstrained evolution was used, the number of modulators per agent grew dramatically. It is possible that adding more modulators to the system would allow them to be optimised by evolution to work over a wider range of time scales, especially if their use is not fixed to two specific sites in the network.

To test for whether modulators could be used for context switching, an agent was created that could accept signals as to which resource to increase. After a set period of time, it was randomly determined which would be the next resource to signal. The parameters for the agent were optimised using a modified fitness function. If the agent increased the resource that it was told to, then it would consequently be given a higher fitness rating.

After the parameter sets were optimised and hard-coded, the agents were tested by plotting 1,000 runs of 1,000 turns each, with each run averaged over a population of 450 agents. The first run randomly determined each turn which resource to signal. The period of time that a resource was signalled for was increased for each run. The last run had only one resource being signalled.

The results were not interesting. It is likely that more interesting results would have been achieved had a better fitness function been used. But with complete evolutionary optimisation taking a significant time to complete there was not enough time left to re-start it[12]. Penalising the fitness level of agents for increasing the wrong resource would probably have helped.

The lesson learned from this endeavour was that the agents should first be roughly evolved to see if they perform as expected. Once the concept is proven, the evolutionary run can be re-started with significant processor time expended to evolve the optimal parameter sets.

4.6.2 Single modulator agent

Once the role of modulators in internal-sensing agents was better understood, the agents were re-evolved using one instead of two modulators. Is the only use of modulators to agitate the network and stop it settling into a stable state? If so then only one modulator is required in an agent. Although there was evidence

---

[12] It is expected that reducing the temperature rating of a solution each time it is re-evaluated will dramatically speed up the search process. This stops the search from having to find a high number of improvements even though it has already found a very fit solution relative to the rest of the fitness landscape.

gained from unconstrained evolutionary runs that agents could benefit from more than one modulator, comparing the performance of a single modulator agent to a two modulator agent would help isolate the functionality provided by modulation. An increase in performance from using two modulators instead of one would provide further evidence that an agent would benefit from being given more than two modulators.

The single modulator was increased by both hunger and thirst neurons in the input layer. Neurons in the middle layer had receptors only for this modulator. Two versions of the single modulator agent were evolved, one using fixed length evaluations and another using variable length evaluations. Both agents evolved to use hybrid hebbian / anti-hebbian learning. The agent evolved using fixed-length evaluations performed the maximum of ten allowed iterations of the network per cycle. The agent evolved using variable length evaluations performed eight iterations per cycle. The experiments carried out using the two modulator agents were repeated using the single modulator agent.

The first experiment consisted of extended evaluations of 10,000 cycles, even though the single modulator agents were evolved using a maximum of 1,000 cycles. As with the two modulator agents, spiking activity continued throughout the evaluation for both single modulator agents. But there was a marked difference in how the two single modulator agents performed. The agent evolved using fixed length evaluations performed significantly better than the agent evolved using variable length evaluations, see figure 4.23.

It was not that the agent evolved using variable length evaluations was inconsistent in deciding which actions were beneficial and which were otherwise. Unlike the agent evolved using fixed length evaluations, the agent did not distinguish between actions that were the most beneficial and exploit those to the exclusion of all others. This is a valid evolutionary strategy when one considers that the fitness of the agent was partially determined by how similar the resource levels were to one another and that it could not rely upon the evaluation lasting a fixed length of time. It is interesting to note though that the two

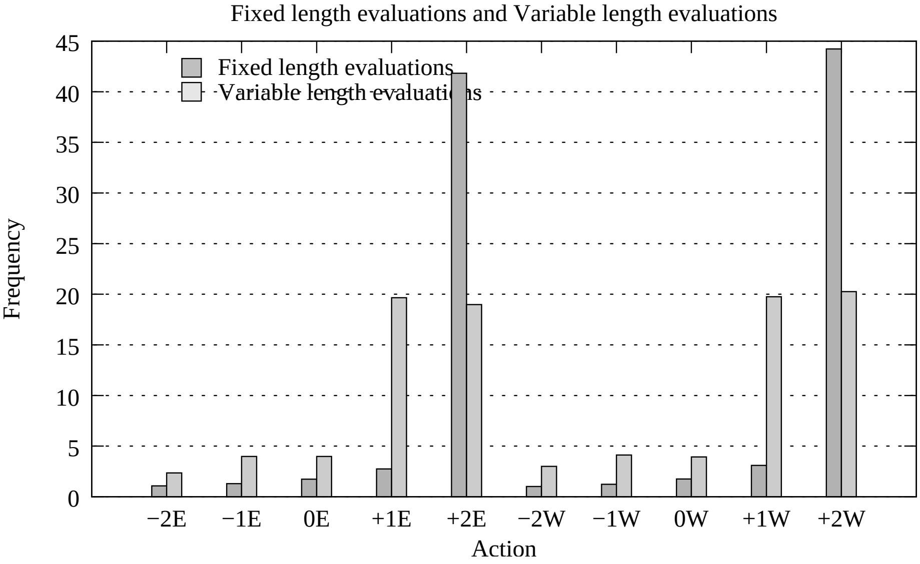


Figure 4.23: Frequency of each action chosen by a population of hard-coded single modulator agents.

modulator agent evolved using variable length evaluations did not rely upon the same strategy. This suggests that having a modulator for each resource is useful in signalling the level of that resource. This further suggests that modulators can be used for context switching and have more uses than to merely agitate the network.

The performance of the fittest phenotypes of each agent was then examined in further detail. The agent evolved using fixed length evaluations can be seen performing the two optimal actions, once learnt, throughout its evaluation to the exclusion of all other actions, see figure 4.24. In contrast, the agent evolved using variable length evaluations continues to try all the other actions throughout its evaluations 4.25. Although these are the fittest examples of these two agents, their behaviour matches the rest of their respective populations. It is particularly interesting to note though, that the fittest example of the agent evolved using variable evaluations still excludes the two costliest actions throughout the course of its evaluation. This further suggests that the strategy is indeed the optimal one for the agent, rather than because the evolutionary parameter search was unable to escape a local maxima.

This was backed up by evaluating the ability of the two agents to re-learn the effects of their actions. Because the agent evolved using variable length evaluations continued to try other actions throughout its evaluation, it was suspected that it would perform better in this test. This was indeed the case, see figure 4.26.

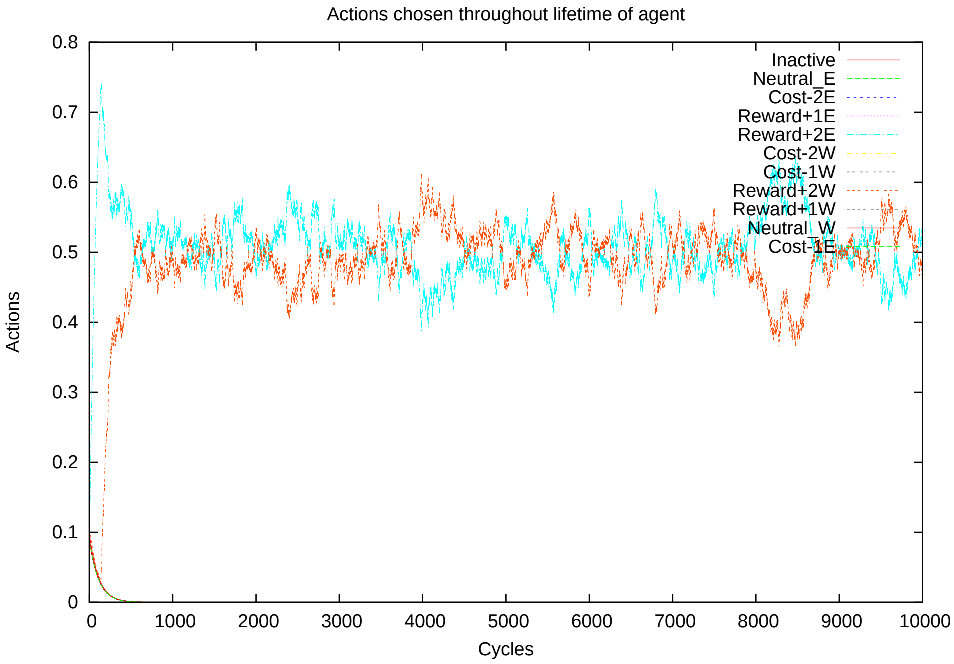


Figure 4.24: Actions chosen over the extended lifetime of the fittest example of a single modulator agent evolved using fixed length evaluations.

The different adaptive mechanisms were then isolated to examine which ones the agents were using. Both agents used both spike-dependent leakage and hebbian learning for adaptation, see figures 4.27 and 4.28. This is in contrast to the two modulator agent evolved using fixed length evaluations. This agent used spike-dependent leakage to maximise its water resource and hebbian learning to maximise its energy resource. Both agents evolved using fixed length evaluations had the chance to fully exploit their environment. The only difference between

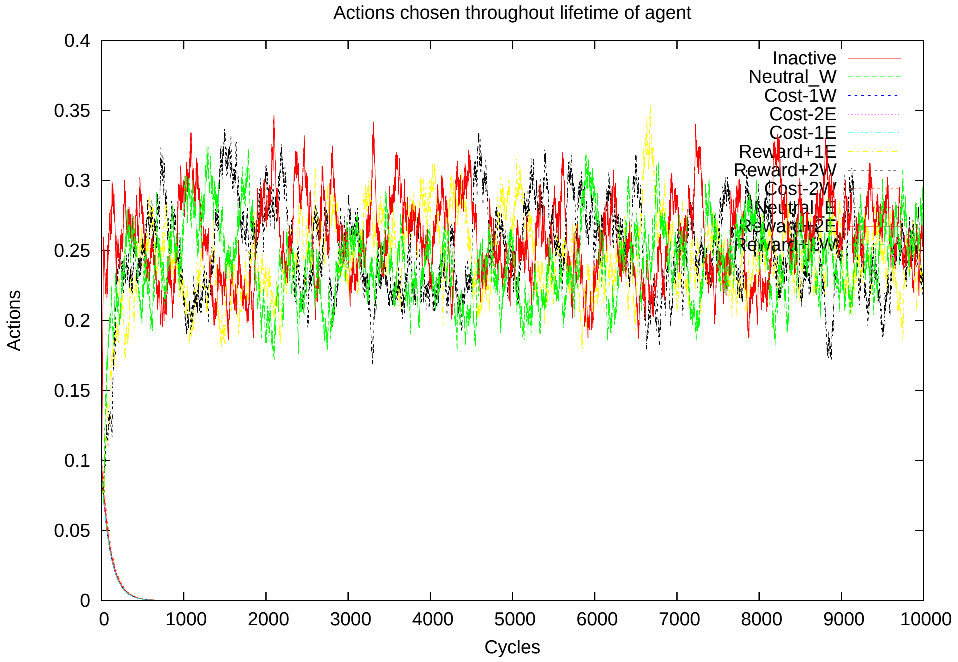


Figure 4.25: Actions chosen over the extended lifetime of the fittest example of a single modulator agent evolved using variable length evaluations.

the two agents was that the two modulator agent increased the strengths of its modulators by using different input neurons for each one. It is not known whether this would allow for resource-specialisation of adaptive mechanisms.

4.6.3 External-sensing agent

The agents were originally intended to be run whilst situated in the environment described in section 3.2.5. The second requirement of an agent listed in section 3.1.3 is that an agent should be able to adapt to an external environment and learn the value of properties within it. This is a common requirement of an agent and therefore it was important to determine whether modulation could provide any extra functionality for this purpose.

As the agents were situated in an external environment, they were provided with the actions described in section 3.3.2. Most of the agents that were initially

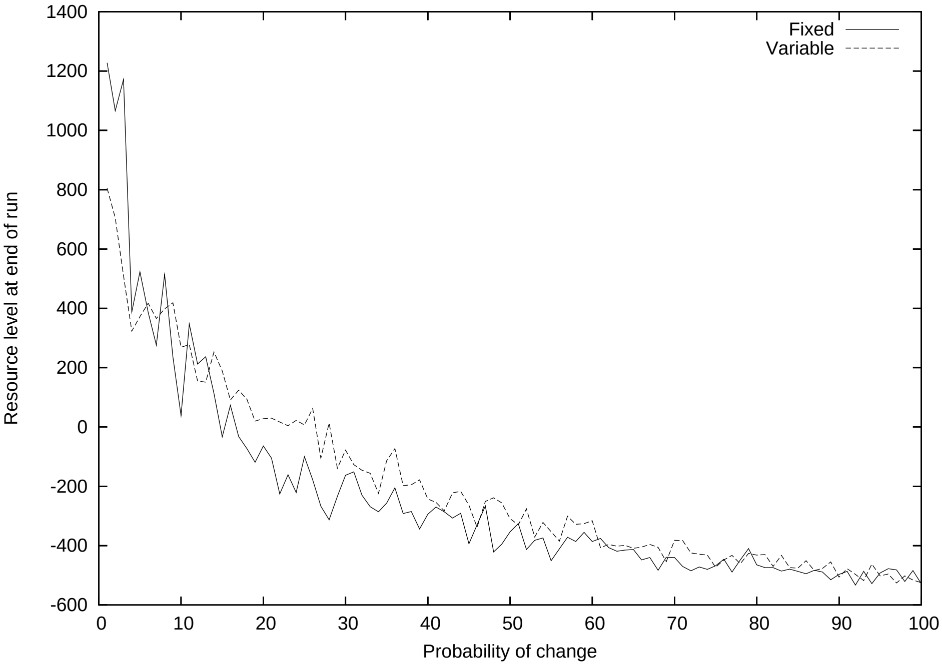


Figure 4.26: Random cost actions in tests using a population of 450 phenotypes. The single modulator agent evolved using fixed length evaluations is less able to re-learn the effects of its actions compared to its counter-part evolved using variable length evaluations.

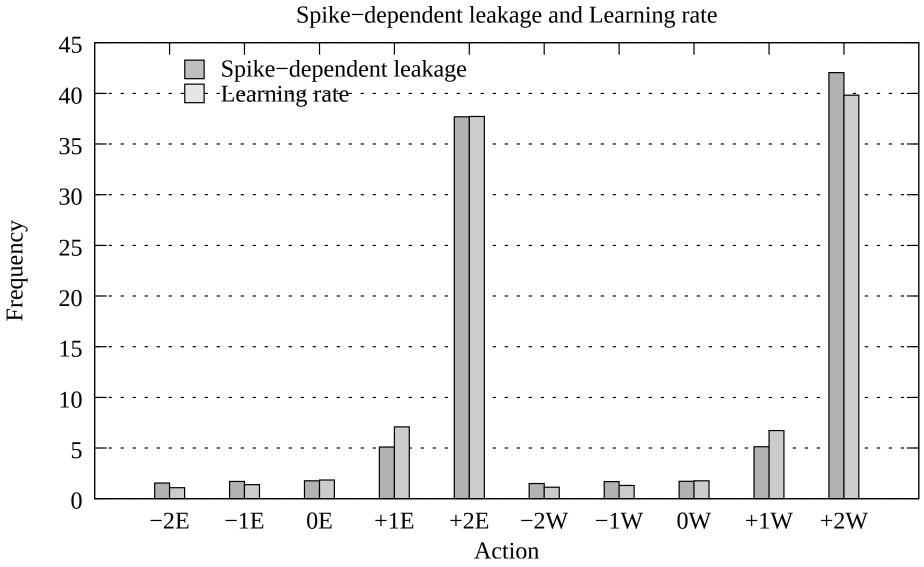


Figure 4.27: Single modulator agent evolved using fixed length evaluations. Both mechanisms are used for adapting to both resources. Synaptic weights and leakage are fixed in order to stop adaptation without these mechanisms.

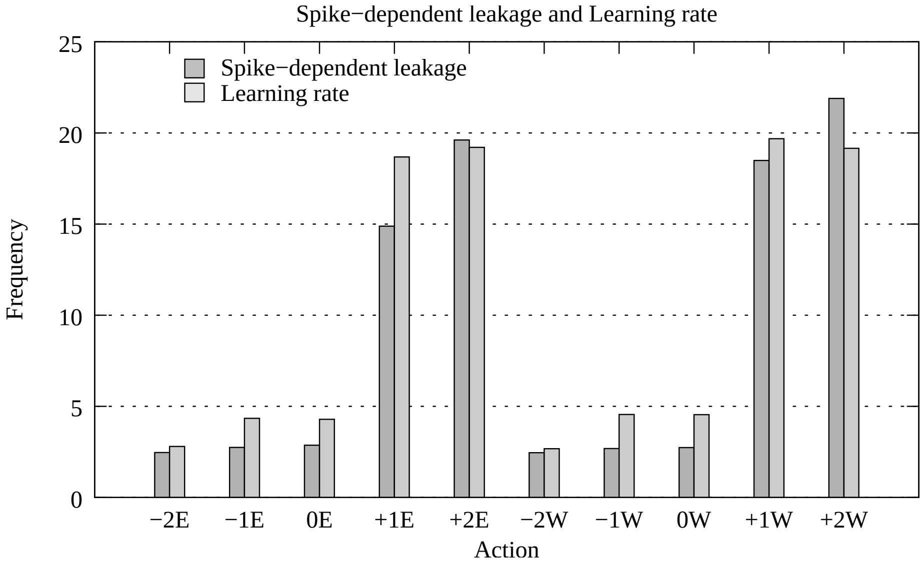


Figure 4.28: Single modulator agent evolved using variable length evaluations. Both spike-dependent leakage and hebbian learning independently provide adaptation for all the beneficial actions available to the agent. Synaptic weights and leakage are fixed in order to stop adaptation without these mechanisms.

designed rarely adapted to the environment. They mostly performed actions at random. The few times an agent did adapt, it was to perform the 'DoNothing' action slightly more often than any other action. This action conserved resources and led to a decrease in input signal to the neural networks. Yet the agents did not perform either the 'ConsumeEnergy' or 'ConsumeWater' action any more than any other action. These actions increased the strength of input signals. As a result the internal-sensing agent was created to investigate further and this avenue of research proved the most fruitful.

After it was found that the agents performed as minimal disturbance systems, new agents were later evolved to adapt whilst situated in the grid world. They were given a layer of 48 neurons to accept the external senses. Senses were input into the network as a vector of binary values that corresponded to the type of entity observed and its position relative to the agent. The distance the agent could scan determined the number of neurons in this layer.

The external-senses layer fed into the middle layer. Because the agents were understood to increase the strength of output neurons that subsequently minimise input signals, it was assumed that the agent would not be able to adapt to a layer of 48 neurons that were fed binary values. Therefore the internal-senses layer[13] fed into the external-senses layer as well as the middle layer in order to apply a value to the external senses. With the modulating agents this connection was implemented by giving the external-senses layer receptors for the modulators produced by the internal-senses layer.

Three agents were evolved, a non-modulating agent, a modulating agent with a single modulator to be increased by both hunger and thirst neurons, and another with two modulators. The agents adapted to the environment, performing the 'DoNothing', 'ConsumeEnergy' and 'ConsumeWater' actions more often than actions to move or turn.

It was then discovered that the agents did not need to have the internal-senses layer connect to the external-senses layer, whether by synaptic weights

---

[13] Previously referred to as the input layer when talking about the internal-sensing agent.

or modulators. It was sufficient for both layers to connect to the middle layer. Because a full analysis was not performed on the agent it is not known whether the agent was adapting to both the external and internal senses, or simply the internal senses. It is suspected to be the latter because although the performance of the agents was better than a random agent, it was not impressively so. The performance of the agents with and without the internal-senses layer being connected to the external-senses layer was equivalent. It is not yet clear how value can be applied to the external-senses with such a system. This is a necessary step for the agent to be situated in an environment.

4.7 Evaluation of modulation

Modulation, when applied to local learning neural networks, provides some benefits but also carries some costs. Overfitting is a problem common to many techniques that seek to emulate an unknown function or data set; whether this technique be curve fitting, neural networks or genetic algorithms etc. This happens when an adaptive process adapts more than is necessary to approximate the use of the underlying variables and proceeds to match characteristics specific to the training data.

Modulation inhibits a local learning neural network from fully adapting to its sensory inputs or environment. Such a network will more likely be able to generalise over other environments or data sets. A non-modulating neural network may find it difficult, impossible or be slow to adapt to other environments or data sets. The network may be relying upon non-existent characteristics specific to the training data or environment.

But this increase in robustness carries a cost. The same mechanism that stops a neural network from overfitting its training data or environment, also inhibits performance when there is no need to. For example, the training data or environment can be processed to only include the underlying characteristics that require adaptation, such as [Rey94]. A rapidly changing environment in which an agent needs to quickly re-learn the effects of its actions may be best adapted to by a non-modulating network. But this advantage disappeared for the agents evolved using variable length evaluations; a characteristic specific to their initial training environment was removed.

But is it always possible or practical to constrain a neural network to such generalised training data? We may not be able to train an agent to an environment if it is largely unknown to us. Or the environment may be noisy. In such cases, the increased robustness provided by modulation may outweigh the cost of how well the agent adapts.

Modulation can not be seen as merely a compromise to a network's ability to adapt. It also acts as a form of regulation; stopping the network from adapting too much to noise, but also stopping adaptation from ceasing all-together when the network threatens to enter a stable-state.

When seen in relation to spike-dependent leakage and synaptic update via spike timing-dependent plasticity, global modulation can be seen as another way to regulate the activity of a neuron. All three mechanisms can be seen as differing by degrees of locality in their areas of influence and receptiveness. Spike-dependent leakage is local to a single neuron, STDP is local only to connected neurons, and modulation relates groups of neurons that have secretors or receptors for it. Spike-dependent leakage and STDP allow for learning to take place, whereas modulation regulates the overall activity of the network. Modulation is of little practical use without other more local forms of adaptation in place. The effects of modulation are too widespread for it to be of much use by itself; it should be used for regulating a network at a greater scale than can easily be done using intra-neuronal mechanisms or inter-neuronal connectivity.

But maybe this is a perspective limited by the use of a single neural network. Spike-dependent leakage regulates the activation of a single neuron according to how frequently it has spiked in the past. Hebbian learning regulates the activation of a neuron according to the historical activity of the neurons connecting to it. These two mechanisms when combined in a greater system allow for learning to take place. Modulation regulates the activity of a single neural network. Perhaps modulation can also allow for a form of learning to occur when combined with other modulated networks?

There is also evidence that modulation can be used as a selection mechanism, as postulated by Koch [Koc99]. Not necessarily by bringing into play different maps from the same set of neurons, but by selecting different adaptive mechanisms or parts of the network. This can be used to promote exploration over exploitation. Output neurons or actions that would normally be ignored may be selected by modulation.

The evidence suggests that local learning neural networks can benefit from multiple modulators. This is plausible when modulators are seen as acting in a similar way to hebbian learning and spike-dependent leakage, but over a larger area of influence and receptiveness. Future research should experiment with modulator sets ranging over different time courses. Or the neurons producing modulators could themselves be modulated so as to regulate the benefits and costs of modulation according to whether the agent needs to exploit or explore its environment.

4.8 Summary

The chapter presented a set of experiments to isolate the functionality provided by neuromodulation. This was achieved by comparing the performance of a modulating agent alongside its non-modulating counterpart. In order to make fair comparisons, the fitness of both agents needed to be the same, or similar, when run in the same way as when they were originally evolved. But in order to discern the functionality provided by neuromodulation, the agents needed to be tested in novel ways.

The agents were originally evolved with a set of cost and reward actions that had the direct effect of increasing or decreasing either the energy or water level of the agent's body. This change was then passed as an input signal to the network in the next turn. The agent learnt how to both choose the actions that led to the biggest increase in a resource, and to avoid choosing actions that led to the biggest decrease.

The first difference that was noted between the modulating and non-modulating agent was when they were both run for an extended period of time. The spiking-activity of the non-modulating agent ceased early on in the run whereas it continued throughout the run for the modulating agent. Two new agents were then evolved, this time using variable length evaluations, so that both agents could adapt regardless of how long it was tested for.

All four agents were then tested to see how they adapted to increasing levels of noise added to the input signals. It was discovered that modulation made the agents more robust to increasing levels of noise. Even though the modulating agents performed worse than the non-modulating agents for low levels of noise, the performance of the modulating agent declined less as the noise increased. The difference was more marked for the agents evolved using fixed length evaluations than for those evolved using variable length evaluations.

It was then discovered that adaptation could continue, albeit at greatly reduced levels of performance, in networks in which synaptic update, spike-dependent leakage and modulation were disabled. Adaptation ceased when all synaptic weights were fixed at 1. Enabling each of these three mechanisms in turn for each agent showed the effect that the learning mechanisms had on the network. Modulation increased the probability of an action being chosen by the network and stopped the network from reacting too strongly to the input signals. Spike-dependent leakage and synaptic update both allowed for adaptation to occur. All three adaptive mechanisms can be seen as performing similar functions but at different scales of specificity.

Further tests using single modulator agents suggested that agents could benefit from being evolved with multiple modulators and that modulation can be used as a selection mechanism rather than to merely agitate a network.

To summarise the key functionality found to be provided by modulation:

- Modulation acts as a form of regulation, preventing the network from fully adapting to its sensory input or environment. It also stops adaptation from ceasing all-together when the network threatens to enter a stable-state.
- Modulators prevent a neural network from overfitting its training data or environment. Modulation inhibits performance when there is no danger of overfitting.
- Modulation allows for arbitration between explorative and exploitative actions.
- A modulating agent can devote spike-dependent leakage and hebbian learning to maximise different resources.
- Modulation can make an agent more robust to increasing levels of noise but decreases performance for low levels of noise.
- Without other more local forms of adaptation in place modulation is of little practical use.
- Neural networks can probably benefit from multiple modulators.

Chapter 5

Discussion

5.1 Overview

The fields of artificial intelligence and artificial life have their own unique problem not normally found in other scientific fields. They often attempt to both understand and then recreate a natural phenomenon for which there is no unequivocal definition. Examples include intelligence, life, emotions, consciousness etc.

A field that attempted only to understand such phenomena would proceed by studying it and distinct definitions would emerge as more was understood. An example of this can be found in the field of geology regarding granite, as described in section 2.3.7. A field that sought only to engineer phenomena, without regard to how well it matched the closest natural equivalent, would have their own definitions emerge to describe the technology as it developed.

In order to solve this problem, a bottom-up approach has been proposed and tested in this thesis when attempting to understand and create artificial emotions. This approach not only advocates emulating the underlying mechanisms that are proposed to give rise to the natural phenomenon, but also advocates delaying for as long as possible the temptation to define the phenomenon.

Neuromodulation is one mechanism that has been proposed as being responsible for emotions. This thesis has researched some of the functionality that this mechanism can provide. This has been achieved by comparing modulating and non-modulating agents that adapt using self-organising biologically plausible neural networks.

5.2 Understanding natural phenomena in terms of computation

5.2.1 Historical comparisons between computers and biological phenomena

It is relatively easy to look back on the history of a scientific field and understand how the paradigms established throughout it were eventually found to be inadequate in some regard. When a new paradigm becomes established, its improvements upon the previous paradigm are well understood. It is also relatively easy to envisage future developments based upon the current paradigm. While an established paradigm still has much to offer, and until its limitations are recognised and understood, it is often difficult to envisage the next paradigm to replace it.

Because the brain can compute the same problems as a computer, it would at first seem plausible that given enough processing power and capacity, human-level intelligence can be created using conventional programming techniques. Initial success makes it seem plausible to reverse the translation process and apply an understanding of silicon-based computing back to the brain, as with [Sim67]. While understanding the brain in such computational terms has its uses, the limitations of this approach lay in a failure to appreciate the difference in underlying architectures, as well as how and why the brain developed.

When the limitations of classical artificial intelligence using a top-down approach were better understood [Bro91b] [Bro91a], the new paradigm of bottom-up development was established. Inspiration was taken from natural and biological phenomena to create neural networks, evolutionary algorithms, artificial immune systems [Das98] etc. This bought further success and again our understanding of the original natural phenomena has increased as a result[1].

In section 1.4, research in artificial intelligence was described as providing the questions for, and an appreciation of, the answers found by computational neuroscience. It was also argued that conclusions reached in modelling and verifying the underlying assumptions of a function performed by the brain can be used to help direct further research into the brain. While to some extent this is true, are biological phenomena best understood in this way? They are at heart physical systems subject to the same laws of thermodynamics as they were when they first evolved as a small collection of cells. They can be understood using the same theories of complexity that can be applied to all other natural phenomena. Can this perspective be applied to the computational models described here?

5.2.2 Minimal energy networks

Boden [Bod99] proposes metabolism as a defining criterion for life. Although the validity of using such a definition of life is of no concern here, it is relevant to note that the flow and use of energy can be used as an explanation of life and intelligence.

Boden refers to how researchers in the field of artificial life predominantly think in terms of information and computation rather than in terms of energy and matter[2]. She describes three senses of metabolism, ranging from the weakest to the strongest. The first sense defines metabolism simply as energy dependency. The second sense specifies energy consumption, storage and budgeting for driving the behaviour of an agent. The third sense adds energy usage for the maintenance of a body to the list. Metabolism, when referring to it in the third sense, continuously exchanges energy between the agent and its environment.

---

[1] A more thorough review of how the paradigms of connectionism, artificial life and dynamical systems relate to the previous paradigm of understanding the human brain in terms of computer programming can be found in [Elm99].

[2] This could be ascribed to the influence of computer science because computers and electronics currently provide the primary means of implementation.

External energy is converted into 'currency' so as to deliver it to the internal processes of the organism.

The idea of a currency in energy is pertinent to the use of modulators for the agitation of a neural network. To recapitulate the findings of this research, designing an architecture to adapt using unsupervised learning proved harder than first envisaged. A number of different architectures, encodings and hebbian learning rules were used in the attempt. It was discovered that the neural network worked best as a minimal-disturbance system. This required the input levels to be reduced according to how beneficial the previous action was to the agent. Modulation was understood to function primarily by providing agitation and what can be seen as an increase in 'energy' to these minimal disturbance systems. Spiking activity was shown to globally decrease when the use of modulators was removed. Receptors responding to modulators were shown to agitate the network so as to stop it settling into a stable state. But receptors activated by modulators can also inhibit neurons. This causes the firing of neurons with inhibitory receptors to be delayed, and in effect, the activity or 'energy' of the neural network is budgeted.

Boden proceeds by making references to the work of Maturana and Varela [MV80], one of which is of particular relevance here. Her succinct summary is quoted rather than "Maturana and Varela's often tortuous prose":

> Maturana and Varela's avoidance of informational concepts leads them to deny also that organisms have any inputs from or outputs to the environment. They speak only of "perturbations" of the system itself. They grant that an observer may find it useful to distinguish between "internal" and "external" perturbations, but insist that for the autopoietic system itself these are indistinguishable. A state of the system is a state of the system: It does not carry a label announcing its causation. In reality, then, they are all internal perturbations. (Moreover, they are all perturbations in the present tense: an observer may say that the system has "learnt" something,

> but the system merely does what its state at that moment leads it to do.)

There are several reasons why this passage is of relevance here. It describes perfectly the functioning of the neural networks developed here and can be used to explain how the agent can continue to adapt even without the use of hebbian learning or spike-dependent leakage. It may seem like the system has learnt, but to re-use the analogy of the ball on a mountainous landscape, the ball has rolled down a slope and settled into a basin.

It is also interesting to note the similarity to McDermott's rant concerning the indiscriminant use of labels [McD81]. But Maturana and Varela go further by saying that the function that we ascribe to a system is also merely a label. Therefore, if we are to describe the functionality provided by neuromodulation then we must also understand and describe the system being modulated.

5.2.3 Information and energy

Avery [Ave04] (chapter 4) describes how information theory was derived from an understanding of free-energy and entropy in physical systems. A short summary is provided here.

James Clerk Maxwell established a relationship between entropy and the concept of information by proposing a now famous thought experiment. This involved imagining a closed box partitioned into two halves and sorting faster and slower moving particles from a gas of a uniform temperature into either side of the partition. Information regarding each molecule of gas is needed to sort the gas. Removing the partition separating the two halves of a box mixes the sorted particles resulting in a loss of information. Szilard later analysed the relationship between energy and missing information using Boltzmann's constant. The concept of entropy was used by Shannon to explain his formula quantifying missing information. Shrödinger used the concept of entropy and Gibbs free-energy to explain life. He started by questioning why food is required to keep us alive and explained how life increases entropy by the act of feeding. Food is relatively ordered and after being used to produce energy for the organism is returned in a relatively simple state to the environment.

So although information theory was derived from an understanding of free-energy and entropy in physical systems, information is nothing more than a human concept. Even though it is an extremely useful concept, we cannot assume that it will always provide the most apt explanation of a naturally evolved physical system. As with definitions, it is a tool that we should not feel compelled to immediately use without question.

Avery finishes the chapter with an excellent summary of the process to which Shrödinger refers to:

> A flood of information-containing free energy reaches the earth's biosphere in the form of sunlight. Passing through the metabolic pathways of living organisms, this information keeps the organisms far away from thermodynamic equilibrium (”which is death”). [His parenthesis] As the thermodynamic information flows through the biosphere, much of it is degraded into heat, but part is converted into cybernetic information and preserved in the intricate structures which are characteristic of life. The principle of natural selection ensures that as this happens, the configurations of matter in living organisms constantly increase in complexity, refinement and statistical improbability. This is the process which we call evolution, or in the case of human society, progress.

Heylighen describes how variety can be encouraged within a system by keeping it far from thermodynamic equilibrium [Hey00]. The modulating agent continued to try other actions throughout the extended evaluations whilst spiking activity ceased for the non-modulating agent, see section 4.3.3. Modulators were used to keep the agent sufficiently far from equilibrium.

5.2.4 Neuromodulation understood in context

Chaisson [Cha03] proposes that life be defined as:

> . . . an open, coherent spacetime structure kept far from thermodynamic equilibrium by a flow of energy through it - a carbon-based system operating in a water-based medium with higher forms of metabolizing oxygen.

Chaisson does not merely explain why life should be defined in terms of the flow of energy, he also describes why life exists because of it. The principles behind the formation of life are proposed to be the same as those that create and maintain all other structures in the universe [Cha05] [Cha01] [Cha06]. He explains how order emerges from energy and how complexity has developed since the very beginning of the universe[3]. Even though the mechanisms involved may be different, the same process underlies the vastly more complex environments that we are more familiar with.

Open structures, complex and ordered relative to their environment, produce entropy by maintaining order. Resources rich in energy flow in and low-energy waste flows out[4]. Entropy local to the system decreases at the expense of the entropy of the environment. Order is created because of the flow of energy. Describing the process as self-organisation implicitly and inappropriately suggests that the system is closed. The process is better understood as the flow of energy into an open system and its dissipation out of it. Perturbations happen by chance to any natural open system that consists of many degrees of freedom. The ordinary case is for these perturbations to dissappear over time but if the perturbations are too large to be damped by the system and not too large to completely destroy the structure, then the system will become re-ordered. Energy acts as a selective force, pruning away weak structures and leaving behind those that are able to utilise the energy. Continual reordering in this manner

---

[3] Kauffman devotes the second part of his book to the possibility that life ". . . crystallized as a self-reproducing metabolism in a space of possible organic reactions" [Kau93] pp285.

[4] As with a metabolism.

results in increased complexity. This results in greater instability and therefore increases the chance of further re-ordering occurring. This process increases local complexity and energy consumption in the presence of optimal energy flow; too little and the system will starve, too large and it will be destroyed.

If one were to set about designing components that could be ordered using such a process for the purpose of adaptively controlling an agent, then it is likely that they would be very similar to the ones used in this research. If a single artificial cell were to be designed for this purpose then it would probably adapt to temporary saturation from excess activation by attenuating its input signals. This conversely requires decreasing the attenuation during periods of lesser activity effectively accentuating the input signals. Such a cell would improve the survival rate of a host organism for environments that at times resulted in excessive input signals. In the networks used in this research, spike-dependent leakage would function adequately in this manner.

For a system to break apart and re-order itself because of an excess inflow of energy, the components need to connect to each other in patterns that sufficiently dampen, or make optimal use of the energy. The system must have the ability to prune away connectivity that does not allow for this. This would require a mechanism such as STDP. The effective existence or strength of a connection would be determined by whether the activation of both components help to dampen or make optimal use of the excess activation. This may require an effect upon the environment of the system in order to regulate the inflow of energy. In an agent this would take the form of an action.

In the same way that a single cell may be either insufficiently sensitive to input or continuously saturated and therefore not able to aid in the survival of the host organism, the activity of the entire network may need to be regulated. This would require mechanisms such as modulators. For an agent, the effect of this is to regulate its behaviour.

More complex environments might require different orderings of the system to be frequently selected and used. Rather than break down the ordering of the system each time, i.e. relearn the environment, the system may need to select different subsystems to cope with the excessive inflow of energy. This would require a global signal to activate or deactivate, excite or inhibit, parts of the system. For an agent, such a mechanism would in effect be a global modulator.

An agent that could select a subset of subsystems rather than just one at a time would be more adaptable. Such an agent would also make more efficient use of its brain. It could have a greater variety of subsystems, each performing a more specialised function. The subset of subsystems that provides the optimal behaviour to aid the survival of the agent given a particular environmental state could be selected using global modulators[5]. This suggests that modulators are most suited for selection of parts of a network or agent controller, as suggested by the results in section 4.6.2, instead of activating different maps from the same network, as questioned in section 1.3.

If this theory is found to hold true in natural agents, then it explains why emotions are so difficult to define by merely observing them. This suggests that "emotions" is a label that we as observers (whether conscious observers of our own internal state or of the external behaviour of others) effectively apply to the activation of a class or subset of neural functions that work towards a distinct and discernable effect. This would explain why it is so difficult to define exactly what is and is not an emotion. Each emotion would merely be a particular subset of neural functions found by evolution to provide the optimal behaviour for an agent given a certain environmental or bodily state.

For an observer, this by itself may not be sufficient to distinguish an emotion from other evolved responses or behaviours. It may be that an evolved response or behaviour needs to hold sufficient influence upon the actions of an agent for it to warrant being labelled as part of a distinctive class. What we label as being an emotion or otherwise, may rest upon the subjective judgement of how distinctive and how much of an effect the corresponding subset of subsystems has on the behaviour of an agent.

---

[5] The optimum subset for a given environmental state being found using evolution.

5.2.5 The relevance of modulation to (some) emotions

LeDoux [LeD98] pp16 characterises cognitive processing as increasing our range of responses to our environment whereas emotions decrease the range to those that have been selected by evolution.

The non-modulating agents tested in this thesis overfitted their sensory input and were less resistant to the effects of noise. Spiking-activity ceased in the non-modulating agent evolved using fixed length evaluations when tested for an extended period of time. Modulation allowed the agents to be less sensitive to their sensory input. This is the reason why the two-modulator agent evolved using variable length evaluations did not overfit its input signals despite using both spike-dependent leakage and synaptic update for adaptation.

If modulation can make an agent less sensitive to its sensory input then it could perhaps be used to influence the behaviour of an agent regardless of its current environment. For example, it makes sense for prey grazing safely to be sensitive to its sensory input. It needs to sense and taste food and it needs to be alert to the presence of predators. But if it senses a predator nearby then it needs to avoid moving and drawing the attention of the predator, or failing that, it needs to flee.

Predator avoidance is an ideal behaviour to be selected for and optimised by evolution. It is also a behaviour that needs to be maintained in the absence of any further sensing of the predator. Relying upon a continual sensing of the predator to drive the fleeing behaviour until the prey reaches safety will more than likely result in the prey being eaten. Nor will the prey benefit from being distracted by less important sensory input while it is still in danger, such as pain, discomfort or the smell of another food source. Modulation can be used to select for specific output neurons, or parts of the neural network or brain, that are responsible for fleeing behaviour. Exploration of different actions to satisfy different needs would cease and exploitation of learned strategies for a successful escape would be given priority.

Kelso discusses Central Pattern Generators, or CPGs; neural circuits that can produce complex behaviours in an absence of sensory input [Kel95] pp239–243. Neuromodulation can be used by CPGs to switch between different patterns and to create new ones. So even though the experiments in this thesis failed to bring into play different maps from the same neural network, instead seeming to work as a form of selection instead, physiological evidence suggests that this may be the case for real brains. But perhaps the two postulated functions of neuromodulation are not too different. Would CPGs not be reconfigured by selection of different components and neurons anyway?

By hormonally modulating the sensory input, Avila-García and Cañamero have shown that the principle of using modulation to change the behaviour of action-selection architectures is possible [AGC05]. In one experiment, the salience of a prey agents level of well-being is increased using hormonal modulation the moment a predator has been sensed. The prey then behaves accordingly.

The agents used in this thesis are too simple to allow direct comparison with complex natural agents. In reality, prey need to be more aware of their sensory input in order to judge where a predator may be and to react more quickly to their environment. Prey agents also need to be more selective as to what senses they react to. For example, adrenaline heightens sensory awareness in humans, but also leads to a focusing of attention. In such a situation the prey is motivated to perform a behaviour and actively senses the environment as part of that behaviour.

It is plausible to suggest that neuromodulation can be used as a mechanism to activate certain behaviours attributable to emotions. Evolution hard-codes the recognition of certain sensory features to trigger the release of a neuromodulator. This excites parts of the brain responsible for an evolved behaviour to bias the actions of the agent.

So given what we have learnt about the use of modulation when applied to neural networks, can we present a theory as to how emotions might emerge from, or be implemented by the use of neuromodulators? With the example of a natural or artificial prey agent, whilst grazing peacefully it is not in any strong emotional state and therefore can concentrate on exploiting its environment to its fullest. Possibly sensing a predator it enters into a strong emotional state. This comes from the levels of the prey's neuromodulators being increased, agitating its brain and pushing it out of a stable state. This gives other actions a chance to be performed and increases the salience of the prey's somatic and external sensory input. Up to this point the prey might not be sure of the best cause of action or indeed whether it is in danger. Fleeing might draw attention to the prey whereas freezing might mean that it is losing valuable time to escape. Let us say that a predator leaps at the prey. There is not enough time to think but an evolved instinct to jump back kicks in. The prey flees and the levels of other neuromodulators are increased making the prey enter a different emotional state. The evolved emotional behaviour narrows the options of the prey, this time giving precedence to parts of its brain that performs a specific evolved fleeing behaviour.

This would concur with the argument of Fellous [Fel99] that emotion can be seen as a continuous patterns of neuromodulation of certain brain structures. The results of this thesis suggest that emotions would operate in this way by using neuromodulators to regulate the activity of neural circuits according to the needs of the agent. This would probably happen as a smooth transition rather than as a discrete change or flip between behaviours.

5.3 Further work

Most scientific research creates as many new questions as old questions are answered. This thesis is no exception. Now that it has been established that modulation of local learning neural networks can be functionally useful, further work is required to understand the extent of this usefulness.

For the neural network to be of wide-spread practical use it is necessary to increase the chance of a phenotype with good performance levels being produced from a genotype. A stochastic mapping from genotype to phenotype was used for this research in order to improve the odds of evolution finding a system that worked. Now that it has, will deterministic synaptic connectivity reduce the variability of phenotypic performance levels? [6] This by itself will not make the mapping from genotype to phenotype deterministic as it has been shown that a random distribution of synaptic weights is required for adaptation to take place. Stochastic connectivity was originally used to reduce the reliance upon a single random number generator to sufficiently randomise the synaptic weights.

But is it biologically plausible to have each neuron in one layer connect to each neuron in another layer the same number of times? Could the connectivity of the network be learnt as part of adaptation?[7] Adaptation of synaptic weights occurred rapidly and continuously as a form of local modulation during the experiments conducted for this research. It is possible that forming and removing synaptic connections or vesicle release sites throughout the lifetime of an agent could be used as a more permanent form of adaptation that occurs over longer time scales.

Many models of hebbian learning were implemented and tested in order to make the system adapt. It was then discovered that the neural network worked best when used as a minimal disturbance system. Synaptic update is currently implemented using spike timing-dependent plasticity, but this is not the sole means by which adaptation can occur in a neural network; for example, adaptation still occurs when using spike-dependent leakage. How would more traditional forms of hebbian learning such the BCM learning rule [ID03] relate to the functioning of the network as it is currently understood?

What other functions can modulation perform? One way of researching this is to compare more complex modulating and non-modulating agents. For example, agents that can learn effective sequences of actions. In the current models, only the middle layer has receptors for modulators. Could receptors be

---

[6] In other words, each neuron in a source layer connects to each neuron in a target layer a genetically determined number of times.

[7] It is unwise to evolve specific connections between neurons as the chances of evolution producing a solution that cannot be understood, allows for no adaptation throughout the lifetime of the agent, and of evolution becoming stuck on a local maxima, will be increased.

used for the external senses layer or the output layer?

Can an agent be given multiple neural circuits evolved for different purposes such as to breed, with each circuit excited by increasing the strength of a corresponding modulator when needed or relevant to the current environment? This would be especially useful for agent needs which do not correspond directly to an internal resource level. Further work in this area could start by evolving two specialised neural networks, one for cost minimisation and one for reward maximisation. A modulator could then be used to alter the influence or activation of either network.

The evidence suggests that internal sensing agents could benefit from being evolved to use more modulators [8]. Asking how many more modulators would be useful, what extra functionality they would provide and how the evolved modulators are used would increase our understanding of their role.

Because neuromodulators trigger receptors whose effects can vary over a wide range of time scales, future research could focus on the use of modulators in neural networks that temporally adapt.

Ultimately all methods of adaptation have to be applied in order to be useful. As discussed in section 4.6.3, an agent needs to learn the relative value of what it is sensing. Can modulation help with this by making certain senses more or less salient? What benefits can modulation bring when used in a physically situated robot with noisy sensors and actuators, where an action or behaviour is performed over a period of time rather than as a discrete step?

---

[8] As suggested by the number of modulators selected by unconstrained evolution, comparisons between single modulator agents with two-modulator agents and by the huge numbers of neuromodulators found in natural agents.

5.4 Conclusion

5.4.1 Understanding emotions by understanding neuromodulation

The premise behind the thesis was that emotions are phenomena that emerge from the interaction of certain subcomponents within a natural agent. Before one can properly understand what emotions are and recreate them, one must first model these subcomponents and their interactions to increase our understanding of why they exist and how they are useful. The subcomponent emulated in this research was neuromodulation.

What the thesis has achieved is to increase our understanding of how *some* natural emotions could possibly operate using neuromodulation. Confirming whether they do or not requires further study using natural agents and is beyond the scope of this thesis. It has been argued here that modulation should not be seen as an exotic mechanism which can be added to a hebbian learning neural network. It functions by regulating neuronal activity, much like hebbian learning and spike-dependent leakage, but at a different scale of use. Specifically, the research has shown that modulation is a useful mechanism to incorporate into a network of artificial leaky conductance-based integrate-and-fire neurons, regardless of whether the designer intends to emulate certain emotions or not.

5.4.2 A bottom-up approach

At the beginning of this research effort, the author had a fixed idea as to what was required of an agent and how it would work. This overly complex idea was derived from what was understood second-hand from the literature and with limited first-hand experience. Progress was only made when this idea was significantly simplified and an open-mind was adopted to determine how the system 'wanted' to work. It was intuitively envisaged that the agent would act as a dynamical system[D.B95] at the start of the research. But the idea of a minimal-disturbance system was never thought of until it was discovered that this provided the greatest levels of performance. Nor did the final system work in any way similar to what was envisaged at the beginning of the research. It was assumed that the system would work by associating patterns of synaptic weights, as described in [RT98], rather than by using the random distribution of weights to spread the activations out over many neurons and filter spiking activity. Ironically, this is more in-keeping with the idea of an agent as a dynamical system.

It is very easy to evolve an agent whose internal workings are fiendishly difficult to understand. We already have such agents in the form of naturally evolved wetware and there is a limit to how much we can understand from this alone. First-hand experience gained from this research has shown that biologically plausible self-organising neural networks are difficult to design. Evolution allows for alternative and better designs to be found that the designer may never have thought of. But this research has also shown the utility of understanding how those evolved networks provide the functionality that they do.

Whilst it is useful to design agents to be wholly situated in an environment [Bro91b], it is also useful to research and fully understand the underlying mechanisms that can aid in this endeavour. But rather than have independent research teams attempt to build single parts of a system and assume that combining these parts into a whole agent will be a trivial matter, these underlying mechanisms can be useful by themselves. This means that the input, output and functional constraints of the mechanism can be clearly understood through extensive use before being added to the palette of the designer of wholly situated agents.

The experience gained from this research has suggested that a useful approach is to evolve neural networks to provide functions that fulfil well-defined requirements. These networks can then be researched and fully understood, allowing them to be used in many different areas. The more network architectures that a designer has to hand, the more functional requirements can be met when designing an agent to adapt to real environments.

Bibliography

[AG02] Zippora Arzi-Gonczarowski. Ai emotions: Will one know them when one sees them? In *Proceedings of the 16th European Meeting on Cybernetics and Systems Research(EMCSR2002). Symposium Agent Construction and Emotions*, volume 2, 2002.

[AGC05] Orlando Avila-García and Lola Cañamero. Hormonal modulation of perception in motivation-based action selection architectures. In *Proceedings of the Symposium on Agents that Want and Like: Motivational and Emotional roots of Cognition and Action at the AISB-05 conference*, pages 9–16. The society for the study of artificial intelligence and the simulation of behaviour, 2005.

[ANM97] Andrea L. Ames, David R. Nadeau, and John L. Moreland. *The VRML 2.0 sourcebook*. John Wiley & Sons, Inc., 1997.

[Arn60] M.B. Arnold. *Emotion and personality*. New york: Columbia University Press, 1960.

[Ave04] John Avery. *Information Theory and Evolution*. World Scientific Publishing Co., 2004.

[Bar01] Lionel Barnett. Netcrawling - optimal evolutionary search with neutral networks. In *Proceedings of the 2001 congress on Evolutionary Computation CEC2001*, pages 30–37, Korea, 2001. IEEE Press.

[Bed96] Mark A. Bedau. The nature of life. In Margaret Boden, editor, *The Philosophy of Artificial Life, Oxford Readings in Philosophy*, pages 332–357. Oxford University Press, 1996.

[BJT05] T. Bosse, C.M. Jokner, and J. Treur. Simulation and representation of body, emotion and core consciousness. In *Proceedings of the Symposium on Next Generation Approaches to Machine Consciousness at the AISB-05 conference*, pages 95–103. The society for the study of artificial intelligence and the simulation of behaviour, 2005.

[Bod99] Margaret A. Boden. Is metabolism necessary? *British Journal of Philosophy of Science*, 50(2):231 – 248, 1999.

[Bro91a] Rodney A. Brooks. Intelligence without reason. In John Myopoulos and Ray Reiter, editors, *Proceedings of the 12th International*

Joint Conference on Artificial Intelligence (IJCAI-91), pages 569–595, Sydney, Australia, 1991. Morgan Kaufmann publishers Inc.: San Mateo, CA, USA.

[Bro91b] Rodney A. Brooks. Intelligence without representation. *Artif. Intell.*, 47(1-3):139–159, 1991.

[BW02] G. Q. Bi and H. X. Wang. Temporal asymmetry in spike timing-dependent synaptic plasticity. *Physiol Behav.*, 77(4-5):551–555, 2002.

[Can27] W.B. Cannon. The james-lange theory of emotion: A critical examination and an alternative theory. *American Journal of Psychology*, 39:106–124, 1927.

[Cañ97] Dolores Cañamero. A hormonal model of emotions for behavior control. In *Proceedings of the Fourth European Conference on Artificial Life*, 1997.

[Cha01] Eric J. Chaisson. *The rise of complexity in nature.* Harvard University Press., 2001.

[Cha03] E.J. Chaisson. A unifying concept for astrobiology. *International Journal of Astrobiology*, 2(2):91–101, 2003.

[Cha05] E.J. Chaisson. Non-equilibrium thermodynamics in an energy-rich universe. In *Non-equilibrium thermodynamics and the production of entropy*, pages 21–31. Springer-Verlag, 2005.

[Cha06] Eric J. Chaisson. *Epic of evolution: seven ages of the cosmos.* Columbia University Press., 2006.

[Che99] Chaomei Chen. *Information Visualization and Virtual Environments.* Springer-Verlag London Limited., 1999.

[CM95] Dave Cliff and Geoffrey F. Miller. Tracking the red queen: Measurements of adaptive progress in co-evolutionary simulations. In *European Conference on Artificial Life*, pages 200–218, 1995.

[CP93] Robert Carter and Kihong Park. How good are genetic algorithms at finding large cliques: an experimental study. Technical Report 1993-015, Computer Science Department, Boston University, 1993.

[Dam94] A.R. Damasio. *Descartes' Error: Emotion, Reason, and the Human Brain.* Quill, 1994.

[Dar72] Charles Darwin. *The expression of the emotions in man and animals.* UCP, Chicago, 1872.

[Das98] D. Dasgupta, editor. *Artificial Immune Systems and Their Applications.* Springer-Verlag, 1998.

[Daw82] R. Dawkins. *The Extended Phenotype.* Oxford University Press., 1982.

[D.B95] Randy D.Beer. A dynamical systems perspective on autonomous agents. *Artificial Intelligence*, 72:173 – 215, 1995.

[Doy00] Kenji Doya. Metalearning, neuromodulation, and emotion. In G. Hatano, N. Okada, and H. Tanabe, editors, *Affective Minds*, pages 101–104. Elsevier Science Ltd., 2000.

[Doy02] Kenji Doya. Metalearning and neuromodulation. *Neural Networks*, 15(4):495–506, 2002.

[DP03] E. Di Paolo. Evolving spike-timing-dependent plasticity for single-trial learning in robots. *Philosophical Transactions of the Royal Society of London, Series A: Mathematical, Physical and Engineering Sciences*, 361(1811):2299 – 2319, 2003.

[DW95] D.J. Depew and B.H. Weber. *Darwinism evolving. Systems dynamics and the genealogy of natural selection.* Cambridge Mass. MIT Press, 1995.

[EG72] Niles Eldredge and Stephen Jay Gould. Punctuated equilibria: An alternative to phyletic gradualism. In T.L.M Schopf, editor, *Models in paleobiology*, chapter 5, pages 82–115. San Francisco. Freeman, Cooper, 1972.

[Elm99] Jeffrey L. Elman. Connectionism, artificial life, and dynamical systems: New approaches to old questions. In W. Bechtel and G. Graham, editors, *A companion to cognitive systems (Blackwell companion to philosophy).* Blackwell publishers, 1999.

[ERKJ00] James H. Schwartz Eric R. Kandel and Thomas M. Jessell. *Principles of Neural Science.* McGraw-Hill Companies Inc., 2000.

[Etz93] Oren Etzioni. Intelligence without robots: A reply to brooks. *AI Magazine*, 14(4):7–13, 1993.

[Eva02] D. Evans. The search hypothesis of emotion. *British Journal for the Philosophy of Science*, 53(4):497–509, 2002.

[Fel99] Jean-Marc Fellous. The neuromodulatory basis of emotion. *The neuroscientist*, 5(5):283–294, 1999.

[Fel04] Jean-Marc Fellous. From human emotions to robot emotions. In *Architectures for Modeling Emotions: Cross-Disciplinary Foundations. Papers from the 2004 AAAI Spring Symposium*, pages 37–47. AAAI Press, 2004.

[Gad99] S. Gadanho. *Reinforcement Learning in Autonomous Robots: An Empirical Investigation of the Role of Emotions.* PhD thesis, University of Edinburgh, 1999.

[GCM97] Stephen Grand, Dave Cliff, and Anil Malhotra. Creatures: Artificial life autonomous software agents for home entertainment. In W. Lewis Johnson, editor, *The First International Conference on Autonomous Agents (Agents '97)*, pages 22–29, Marina del Rey, California, USA, 5-8 1997. ACM Press.

[GD05] P. Goldie and S.A. Döring. Emotions as evaluations. In *Proceedings of the Symposium on Agents that Want and Like: Motivational and Emotional roots of Cognition and Action at the AISB-05 conference*, pages 45–50. The society for the study of artificial intelligence and the simulation of behaviour, 2005.

[GH98] Sandra Clara Gadanho and John Hallam. Emotion-driven learning for animat control. In *Proceedings of the fifth international conference on simulation of adaptive behavior on From animals to animats 5*, pages 354–359. MIT Press, 1998.

[GH01] Sandra Clara Gadanho and John Hallam. Robot learning driven by emotions. *Adapt. Behav.*, 9(1):42–64, 2001.

[GM01] Jonathan Gratch and Stacy Marsella. Tears and fears: modeling emotions and emotional behaviors in synthetic agents. In *Agents*, pages 278–285, 2001.

[GNK00] David G. Green, David Newth, and Michael G. Kirley. Connectivity and catastrophe - towards a general theory of evolution. In Mark A. Bedau, John S. McCaskill, Norman H. Packard, and Steen Rasmussen, editors, *Artificial Life VII. Proceedings of the seventh internation conference on Artificial Life*, pages 153–161. MIT Press, 2000.

[GPG$^+$00] Fernando Montes Gonzalez, Tony Prescott, Kevin Gurney, Mark Humphries, and Peter Redgrave. An embodied model of action selection mechanisms in the vertebrate brain. In J.-A. Meyer et al., editor, *Proceedings of the sixth international conference on the simulation of adaptive behaviour*, 2000.

[Gro92] Stephen Grossberg. *Neural Networks and Natural Intelligence.* MIT Press, 1992.

[Har94] I. Harvey. Evolutionary robotics and saga: The case for hill crawling and tournament selection. In C. G. Langton, editor, *Artificial Life III, Proceedings Volume XVI*, pages 299–326. Santa Fe Institute Studies in the Sciences of Complexity, Addison-Wesley, 1994.

[Has99] Michael E. Hasselmo. Neuromodulation: acetylcholine and memory consolidation. *Trends in Cognitive Sciences*, 3(9):351–359, 1999.

[Hey00] Francis Heylighen. The science of self-organization and adaptivity. In *The encyclopedia of life support systems*, pages 253–280. EOLSS Publishers, 2000.

[HHC93] Inman Harvey, Philip Husbands, and Dave Cliff. Issues in evolutionary robotics. In *Proceedings of the second international conference on From animals to animats 2 : simulation of adaptive behavior*, pages 364–373, Cambridge, MA, USA, 1993. MIT Press.

[HKP91] John Hertz, Anders Krogh, and Richard G. Palmer. *Introduction to the theory of neural computation.* Addison-Wesley Longman Publishing Co., Inc., 1991.

[Hoo96] J. Hooker. Testing heuristics: We have it all wrong. *Journal of Heuristics*, 1:33–42, 1996.

[HSJO98] P. Husbands, T.M.C Smith, N. Jakobi, and M. O'Shea. Better living through chemistry: Evolving gasnets for robot control. *Connection Science*, 10(3-4):185–210, Dec 1998.

[HSO$^+$98] P. Husbands, T.M.C. Smith, M. O'Shea, N. Jakobi, J. Anderson, and A. Philippides. Brains, gases and robots. In L. Niklasson, M. Boden, and T. Ziemke, editors, *Proceedings of the 8th International Conference on Artificial Neural Networks: ICANN98*, Perspectives in Neural Computing, pages 51–64. Springer-Verlag, 1998.

[HT97] Inman Harvey and Adrian Thompson. Through the labyrinth evolution finds a way: A silicon ridge. In T. Higuchi, M Iwata, and L. Weixin, editors, *Proceedings of the 1st International conference on evolvable systems (ICES)'96*, volume 1259 of *Lecture notes in computing series*, pages 406 – 422. Springer-Verlag, 1997.

[Hus98] P. Husbands. Evolving robot behaviours with diffusing gas networks. In P. Husbands and J. A. Meyer, editors, *Evolutionary Robotics: First European Workshop, EvoRobot98*, pages 71–86. Springer-Verlag, April 1998.

[ID03] Eugene M. Izhikevich and Niraj S. Desai. Relating stdp to bcm. *Neural Comput.*, 15(7):1511–1523, 2003.

[IML05] Aleksander I., Lahnstein M, and Rabinder L. Will and emotions: A machine model that shuns illusions. In *Proceedings of the Symposium on Next Generation Approaches to Machine Consciousness at the AISB-05 conference*, pages 110–116. The society for the study of artificial intelligence and the simulation of behaviour, 2005.

[Jak98a] N. Jakobi. Evolutionary robotics and the radical envelope of noise hypothesis. *Adaptive Behaviour*, 6:325–368, 1998.

[Jak98b] N. Jakobi. The minimal simulation approach to evolutionary robotics. In T. Gomi, editor, *Evolutionary Robotics. Vol II. From Intelligent Robots to Artificial Life (ER'98)*, pages 133–190. AAAI Books, 1998.

[Jam84] W. James. What is an emotion? *Mind*, 9:188–205, 1884.

[JJ99] C. G. Johnson and G. J. F. Jones. Effecting affective communication in virtual environments. In *Proceedings of the Workshop on Intelligent Virtual Agents (Virtual Agents99)*, pages 135–138, 1999.

[Joh04] C.G. Johnson. Do somatic markers need to be somatic? analogies from evolution and from hardware interlocks. In *Proceedings of the Symposium on Emotion, Cognition and Affective Computing at the AISB-04 conference*, pages 33–35. The society for the study of artificial intelligence and the simulation of behaviour, 2004.

[Kak96] S. Kak. Can we define levels of artificial intelligence. *Journal of Intelligent Systems*, 6:133 – 144, 1996.

[Kau93] Stuart Kauffman. *The Origins of Order: Self-Organization and Selection in Evolution.* Oxford University Pres, 1993.

[KB02] U. R. Karmarkar and D. V. Buonomano. A model of spike-timing dependent plasticity: One or two coincidence detectors. *Journal of Neurophysiology*, 88:507–513, 2002.

[Kei98] F.A. Keijzer. Some armchair worries about wheeled behavior. In *Proceedings of the fifth international conference on simulation of adaptive behavior on From animals to animats 5*, pages 13–21. MIT Press, 1998.

[Kel95] J. A. Scott Kelso. *Dynamic patterns: The self-organization of brain and behavior.* A Bradford book. The MIT Press., 1995.

[Kel05] Ann E. Kelley. *Who needs emotions? The brain meets the robot.*, chapter 3, pages 29–77. Oxford University Press., 2005.

[KL87] Leonard K. Kaczmarek and Irwin B. Levitan. *Neuromodulation: The Biochemical control of neuronal excitability.* Oxford University Press., 1987.

[KLM96] Leslie Pack Kaelbling, Michael L. Littman, and Andrew W. Moore. Reinforcement learning: A survey. *Journal of Artificial Intelligence*, 4:237 – 285, 1996.

[Klo88] A. H. Klopf. A neuronal model of classical conditioning. *Psychobiol*, 16(2):85–123, 1988.

[KNB02] U. R. Karmarkar, M. T. Najariana, and D. V. Buonomano. Mechanisms and significance of spike-timing dependent synaptic plasticity. *Biological Cybernetics*, 87:373–382, 2002.

[Koc99] Christof Koch. *Biophysics of Computation.* Oxford University Press., 1999.

[Lan85] C. Lange. The emotions. In E. Dunlap, editor, *The emotions.* Williams and Wilkins, Baltimore, 1922 edition, 1885.

[Lan96] Christopher G. Langton. Artificial life. In Margaret Boden, editor, *The Philosophy of Artificial Life, Oxford Readings in Philosophy*, pages 39–94. Oxford University Press, 1996.

[LeD98] Joseph E. LeDoux. *The Emotional Brain.* Simon & Schuster, 1998.

[LGX97] Yee Leung, Yong Gao, and Zongben Xu. Degree of population diversity - a perspective on premature convergence in genetic algorithms and its markov chain analysis. *IEEE Transactions on Neural Networks*, 8(5):1165 – 1176, 1997.

[Lis89] J. Lisman. A mechanism for the hebb and the anti-hebb processes underlying learning and memory. *Proceedings of the National Academy of Sciences of the USA*, 86:9574–9578, 1989.

[MB00] J. Morén and C. Balkenius. A computational model of emotional learning in the amygdala. In J.-A. Meyer et al., editor, *Proceedings of the sixth international conference on the simulation of adaptive behaviour*, pages 383–391, 2000.

[McD81] D. McDermott. Artificial intelligence meets natural stupidity. In J. Haugeland, editor, *Mind Design: Philosophy, Psychology, Artificial Intelligence*, pages 143–160. MIT Press, Cambridge, MA, 1981.

[MCWvR00] G. G. Turrigiano M. C. W. van Rossum, G. Q. Bi. Stable hebbian learning from spike timing-dependent plasticity. *The Journal of Neuroscience*, 20:8812–8821, 2000.

[MFB$^+$88] J. S. Morris, K. J. Friston, C. Büchel, C. D. Frith, A. W. Young, A. J. Calder, and R. J. Dolan. A neuromodulatory role for the human amygdala in processing emotional facial expressions. *Brain*, 121(1):47–57, 1988.

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

[MM99] David E. Moriarty and Risto Miikkulainen. Efficient reinforcement learning through symbiotic evolution. Technical Report AI94-224, Department of Computer Sciences, The University of Texas at Austin, September 1999.

[Mor70] Masahiro Mori. The uncanny valley. *Energy*, 7(4):33–35, 1970.

[MS96] Brad L. Miller and Michael J. Shaw. Genetic algorithms with dynamic niche sharing for multimodal function optimization. In *International Conference on Evolutionary Computation*, pages 786–791, 1996.

[MV80] Humbert R. Maturana and Francisco J. Varela. *Autopoiesis and Cognition: The realization of the living.* D Reidel Publishing Company, 1980.

[Nes90] R.M. Nesse. Evolutionary explanations of emotion. *Human Nature*, 1(30):261–289, 1990.

[Nes00] D.W. Nesse. *Introduction to Mineralology.* Oxford University Press, 2000.

[Nob97] J. Noble. The scientific status of artificial life. In *Proceedings of the Fourth European Conference on Artificial Life*, 1997.

[OJL87] K. Oatley and P. Johnson-Laird. Towards a cognitive theory of emotions. *Cognition and Emotion*, 1:29–50, 1987.

[Pat96a] H. H. Pattee. Learning from functionalism - prospects for strong artificial life. In Margaret Boden, editor, *The Philosophy of Artificial Life, Oxford Readings in Philosophy*, pages 361–377. Oxford University Press, 1996.

[Pat96b] H. H. Pattee. Simulations, realizations, and theories of life. In Margaret Boden, editor, *The Philosophy of Artificial Life, Oxford Readings in Philosophy*, pages 379–393. Oxford University Press, 1996.

[Pav27] I. Pavlov. *Conditional Reflexes.* Oxford University Press, 1927.

[PGG$^+$02a] Tony Prescott, Kevin Gurney, Benoit Girard, Vincent Cuzin, and Agnes Guillot. Comparing a brain-inspired robot action selection mechanism with 'winner-takes-all'. In *From animals to animats 7: Proceedings of the seventh international conference on simulation of adaptive behavior.* MIT Press, 2002.

[PGG$^+$02b] Tony Prescott, Kevin Gurney, Fernando Montes Gonzalez, Mark Humphries, and Peter Redgrave. The robot basal ganglia: Action selection by an embedded model of the basal ganglia. *Advances in Behavioral Biology*, 52, 2002.

[PHO98] A. Philippides, P. Husbands, and M. O'Shea. Neural signalling: It's a gas! In L. Niklasson, M. Boden, and T. Ziemke, editors, *Proceedings of the 8th International Conference on Artificial Neural Networks: ICANN98*, Perspectives in Neural Computing, pages 979–984. Springer-Verlag, 1998.

[Pre00] Steve Prestwich. An informal tutorial on search techniques in constraint programming. In *Second International Workshop on Information Integration and Web-based Applications and Services*, pages 105–121, 2000.

[PSHO99] A. Philippides, T.M.C Smith, P. Husbands, and M. O'Shea. Diffusible neuromodulation in real and artificial neural networks. In A. Ochoa, M.R. Soto, and R. Santana, editors, *AI Symposium, Second International Conference on Cybernetics, Applied Mathematics and Physics: CIMAF99*, pages 1–9. Editorial Academia, Havana, Cuba, 1999.

[PW02] Bernd Porr and Florentin Wörgötter. Isotropic sequence order learning using a novel linear algorithm in a closed loop behavioural system. *Biosystems*, 67(1-3):195–202, 2002.

[PW03] Bernd Porr and Florentin Wörgötter. Isotropic sequence order learning. *Neural Comput.*, 15(4):831–864, 2003.

[Rey94] C. W. Reynolds. Evolution of corridor following behaviour in a noisy world. In *From Animals to Animats 3: Proceedings of the Third International Conference on Simulation of Adaptive Behaviour*, pages 402 – 410. MIT Press, 1994.

[RN95] Stuart J. Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach.* Prentice-Hall Inc., 1995.

[Rol99] Edmund T. Rolls. *The Brain and Emotion.* Oxford University Press., 1999.

[RT98] Edmund T. Rolls and Alessandro Treves. *Neural Networks and Brain Function.* Oxford University Press, 1998.

[SB98] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction.* MIT Press, Cambridge, MA, 1998. A Bradford Book.

[SC81] A. Sloman and M. Croucher. Why robots will have emotions. In *Proc. of the 7th IJCAI*, pages 197–202, Vancouver, Canada, 1981.

[SC02] Wei-Min Shen and Cheng-Ming Chuong. The digital hormone model for self-organization. In *From animals to animats 7: Proceedings of the seventh international conference on simulation of adaptive behavior*, pages 242–243. MIT Press, 2002.

[Sch02] Matthias Scheutz. Agents with or without emotions? In *Proceedings of the Fifteenth International Florida Artificial Intelligence Research Society Conference*, pages 89–93. AAAI Press, 2002.

[Sch04] Matthias Scheutz. An artificial life approach to the study of basic emotions. In *Proceedings of Cognitive Science*, pages 1203 – 1208, 2004.

[Sha05] M. Shanahan. Consciousness, emotion, and imagination a brain-inspired architecture for cognitive robotics. In *Proceedings of the Symposium on Next Generation Approaches to Machine Consciousness at the AISB-05 conference*, pages 26–35. The society for the study of artificial intelligence and the simulation of behaviour, 2005.

[She98] Gordon M. Shepherd, editor. *The synaptic organization of the brain.* Oxford University Press., 1998.

[She04] Matthias Sheutz. Useful roles of emotions in artificial agents: a case study from artificial life. In *Proceedings of the nineteenth national conference on artificial intelligence, sixteenth conference on innovative applications of artificial intelligence*, pages 42–48. AAAI Press / The MIT Press, 2004.

[Sim67] H. A. Simon. Motivational and emotional controls of cognition. *Psychological Review*, 74(1):29–39, 1967.

[SMA00] S. Song, K. D. Miller, and L. F. Abbott. Competitive hebbian learning through spike-timing-dependent plasticity. *Nature neuroscience*, 3(9), 2000.

[SS62] S. Schachter and J.E. Singer. Cognitive, social and physiological determinants of emotional state. *Psychological Review*, 69:379–399, 1962.

[SSH01] Catherine Soanes, Alan Spooner, and Sara Hawker, editors. *Oxford paperback dictionary thesaurus and wordpower guide.* Oxford University Press., 2001.

[SSW00] W. Shen, B. Salemi, and P. Will. Hormones for self-reconfigurable robots. In *Intelligent Autonomous Systems. 2000. Venice, Italy.* IOS Press, 2000.

[Ste00] Robert J. Sternberg, editor. *Handbook of Intelligence.* Cambridge University Press., 2000.

[Tay98] Tim Taylor. Using bottom-up models to investigate the evolution of life: Steps towards an improved methodology. In C.L. Nehaniv and G.P. Wagner, editors, *The Right Stuff: Appropriate Mathematics for Evolutionary and Development Biology, number 315 in Computer Science Technical Report*, pages 23–26. 1998.

[THH96] A. Thompson, I. Harvey, and P. Husbands. Unconstrained evolution and hard consequences. In E. Sanchez and M. Tomassini, editors, *Towards Evolvable Hardware: The evolutionary engineering approach*, volume 1062 of *Lecture Notes in Computing Series*, pages 136–165. Springer-Verlag, 1996.

[Tho11] E.L. Thorndike. *Animal Intelligence.* Hafner, Darien, CT, 1911.

[TT94] Nadia M. Thalman and Daniel Thalmann, editors. *Artificial Life and Virtual Reality.* John Wiley & Sons, Inc., 1994.

[VCPF99] Rodrigo Ventura, Lus Custdio, and Carlos Pinto-Ferreira. Artificial emotions - good bye mr. spock! In *Proceedings of the 2nd International Conference on Cognitive Science*, pages 938 – 941, 1999.

[Wan95] Pei Wang. On the working definition of intelligence. Technical report, The Center for Research on Concepts and Cognition, Indiana University, 1995.

[WBDH95] Andrew Wood, Russell Beale, Nick Drew, and Bob Hendley. HyperSpace: A World-Wide Web visualiser and its implications for collaborative browsing and software agents, 1995.

[WDKvE89] Udo Wehmeier, Dawei Dong, Christof Koch, and David van Essen. Modeling the mammalian visual system. In Christof Koch and Idan Segev, editors, *Methods in Neuronal Modeling: From synapses to networks*, chapter 10, pages 335–360. The MIT Press, 1989.

[Wit58] L. Wittgenstein. *Philosophical Investigations.* Basil Blackwell & Mott, Ltd., 1958.

[WP04] F. Wörgötter and B. Porr. Temporal sequence learning, prediction and control - a review of different models and their relation to biological mechanisms. *Neural Computation*, 17:1–75, 2004.

[Yae94] L. Yaeger. Computational genetics, physiology, metabolism, neural systems, learning, vision and behavior or polyworld: Life in a new context. In C. G. Langton, editor, *Artificial Life III, Proceedings Volume XVII*, pages 263–298. Santa Fe Institute Studies in the Sciences of Complexity, Addison-Wesley, 1994.

[YI02] Masataka Yoshimura and Kazuhiro Izui. Smart optimization of machine systems using hierarchical genotype representations. *Journal of Mechanical Design*, 124:375 – 384, 2002.