Emotional Agents

Ian Paul Wright

A thesis submitted to the
Faculty of Science
of the
University of Birmingham
for the degree of
DOCTOR OF PHILOSOPHY

School of Computer Science
Cognitive Science Research Centre
University of Birmingham
Birmingham, B15 2TT
England

February 1997

Synopsis

The emotions are investigated from the perspective of an Artificial Intelligence engineer attempting to understand the requirements and design options for autonomous resource bound agents able to operate in complex and dynamic worlds. Both natural and artificial intelligences are viewed as more or less complex control systems. The field of agent architecture research is reviewed and Sloman and Beaudoin's design for human-like autonomy introduced. The agent architecture supports an emergent processing state, called *perturbance*, which is a loss of control of thought processes. Perturbances are a characteristic feature of many human emotional states. A broad but shallow implementation of the agent architecture, called MINDER1, is described. MINDER1 can support perturbant states and is an example of a 'protoemotional' agent. Several interrupt theories of the emotions are critically reviewed, including the theories of Simon, Sloman, Oatley and Johnson-Laird and Frijda. Criticisms of the theories are presented, in particular how they fail to account for both learning and the mental pain and pleasure associated with some emotional states. The field of machine reinforcement learning is reviewed and the concept of a scalar quantity form of value introduced. Forms of value occur in control systems that meet a requirement for trial and error learning. A philosophical argument that *a society of mind will require an economy of mind* is presented. The argument draws on adaptive multi-agent system research and basic economic theory. It generalises reinforcement learning to more complex systems with more complex capabilities. A design hypothesis is proposed – *the currency flow hypothesis* – that states that a scalar quantity form of value is a common feature of adaptive systems composed of many interacting parts. A design specification is presented for a motivational subsystem conforming to the currency flow hypothesis and theoretically integrated with Sloman and Beaudoin's agent architecture. An explanation of a subset of mental pain and pleasure is provided in terms of an agent architecture monitoring its own processes of reinforcement, or virtual 'currency flows'. The theory is compared to Freudian metapsychology, in particular how currency flow avoids the vitalism associated with Freud's concept of 'libidinal energy'. The explanatory power of the resulting theory of *valenced perturbances*, that is painful or pleasurable loss of control of attention, is demonstrated by providing an architecturally grounded analysis of grief. It is shown that, amongst other phenomena, intense mental pain and loss of control of thought processes can be readily explained in information processing terms. The thesis concludes with suggestions for further work and prospects for building artificial emotional agents.

To my family and friends.

Acknowledgements

I wish to express warm thanks to my supervisor and teacher, Aaron Sloman, to whom I owe a profound intellectual debt. Without his help this thesis could not have been written.

My thanks to everyone involved in the Cognition and Affect project. In particular thanks to Chris Complin for introducing me to classifier systems, and Luc Beaudoin for his work on NML1.

My warm thanks to Michel Aubé for encouragement.

I would also like to thank the members of the School of Computer Science at Birmingham for providing an environment that supported and encouraged my research.

My doctoral studies were funded by a University of Birmingham studentship.

Contents

List of Figures

List of Tables

Part I

Emotions and agents

Chapter 1

Introduction

In this thesis the emotions are investigated from the perspective of an Artificial Intelligence engineer attempting to understand the requirements and design options for autonomous resource bound agents able to operate in complex and dynamic worlds. Both natural and artificial intelligences are viewed as more or less complex control systems.

Readers patient enough to read to the end will understand how and why certain emotional phenomena do and could occur in humans, animals and machines.

1.1 Overview of thesis

Chapter 2 provides methodological preliminaries. The design-based approach to mental phenomena is described and the importance of the information processing level of explanation emphasised. Reasons why a reified conception of 'consciousness' is irrelevant to a study of the emotions are provided.

Chapter 3 briefly describes Herbert Simon's interrupt theory of the emotions before presenting Aaron Sloman's 'attention filter penetration theory'. The concept of perturbance, a loss of control of one's thought processes, is introduced. The architectural prerequisites for having and losing control of thought processes are discussed.

Chapter 4 reviews the literature on agent architecture research. It classifies agents as either deliberative, reactive, or reactive-deliberative. Three types of control possible in agent architectures are identified. Sloman and Beaudoin's design for human-like autonomy is introduced.

Chapter 5 describes MINDER1, an implementation of Sloman and Beaudoin's design. The agent can handle multiple motives in a simulated complex and dynamic domain. It can potentially support perturbant states, and is an example of a 'protoemotional' agent architecture. MINDER1 is a computational realisation of the 'attention filter penetration' theory of the emotions.

Chapter 6 reviews theories of emotion. The theories of Oatley and Johnson-Laird and Frijda are critically reviewed, followed by an extensive examination and critique of interrupt theories of emotion. Four problems of interrupt theories are identified, which motivates the latter half of the thesis. An important problem is that existing theories cannot account for the valency of some emotional states. Valency is achievement pleasure or failure unpleasure.

Chapter 7 examines a requirement for trial and error learning. Machine reinforcement learning algorithms are reviewed. The importance of a scalar quantity form of value is identified. Forms of value are seen to be ubiquitous features of trial and error learners, deriving from internal 'better than' relations. Possession of value by an internal substate is an ability to buy processing power.

Chapter 8 argues that a society of mind will require an economy of mind, in particular how a scalar quantity form of value will be present in adaptive multi-agent systems. This is a generalisation of machine reinforcement learning to more complex systems. The currency flow hypothesis is presented, stating that 'currency flow' may be common to a large class of adaptive systems.

Chapter 9 presents a specification of a computational libidinal economy, a cognitive subsystem that forms attachments to others. The libidinal economy conforms to the currency flow hypothesis. Valency is explained as the monitoring of credit assignment, or virtual 'currency flows' performing reinforcement learning. The other problems of interrupt theories identified in chapter 6 are partially resolved. The augmented theory is a 'valenced attention filter penetration' theory of the emotions. The new theory greatly improves on Freudian metapsychology, in particular Freud's concept of 'libido'.

Chapter 10 explains the emotional episode of grief in terms of the new theory. It is shown how it can readily account for both the loss of control of thought processes and the intense mental pain often associated with mourning.

Chapter 11 concludes with directions for future work and a discussion of the work presented in the thesis.

Appendix A provides some implementation details of MINDER1.

Appendix B discusses the relationship between labour power and processing power.

Appendix C contains a previously published paper (Wright, 1996b), a requirement of the Faculty of Science submission regulations.

Appendix D lists the abbreviations used in the document.

1.2 Summary of main contributions

The main contribution of the thesis is a theory of mental pain and pleasure and loss of control of thought processes specified in terms sufficiently precise as to be implementable in a computer simulation.

The research presented here:

- Explains why the design-based approach is required for a full understanding of the emotions. (Chapters 2 and 5).
- Argue that 'consciousness' is largely irrelevant to an understanding of the emotions. (Chapter 2).
- Brings together a number of theories from a wide range of research areas relevant to an understanding of the emotions. These theories have never been considered together before. It is shown how they all contribute pieces to the puzzle.
- Provides a classification of different types of control in agent architectures. (Chapter 4).
- Presents an implementation of Sloman and Beaudoin's agent architecture design, called MINDER1. MINDER1 can handle multiple motives in a simulated complex and dynamic domain. It partially supports perturbant states and is an example of a 'protoemotional' artificial agent. (Chapter 5).
- Identifies several limitations of existing interrupt theories of emotion (chapter 6) and presents a theory to overcome those limitations.
- Identifies a particular type of mental pleasure or unpleasure, named valency. (Chapter 6).

- Identifies the importance of a scalar quantity form of value in adaptive trial and error learners, in particular how value functions as an ability to buy processing power. (Chapter 7).

- Generalises reinforcement learning and compares it to currency flow in economic systems. It is shown that the two domains, although different, are related in important ways. (Chapter 8).

- Proposes a currency flow hypothesis for adaptive multi-agent systems, which states that scalar quantity forms of value are likely to be common to a wide class of such systems. (Chapter 8).

- Describes a design specification, based on the hypothesis, that serves as a new model of emotional states that involve painful or pleasurable insistent thoughts. (Chapter 9).

- Explains previously known phenomenological facts, such as achievement pleasure and failure unpleasure, in terms of the monitoring of credit assignment or virtual 'currency flows'. (Chapter 9).

- Explains how the currency flow hypothesis is related to Freudian metapsychology, in particular the relationship between currency flow and Freud's concept of 'libido'. (Chapter 9).

- Explains the human emotional episode of grief in terms of the postulated agent architecture. (Chapter 10).

- Introduces new terminology and concepts important to an understanding of motivation and emotion, such as value, valency, circulation of value, ability to buy processing power and the monitoring of credit assignment. (Chapters 6, 7, 8 and 9).

- Improves upon Oatley and Johnson-Laird's analysis of control versus semantic signalling, and explains the existence of 'simple' control signals in information processing architectures. (Chapter 9).

- Addresses the question, 'Why is the causal power of human metamanagement limited?' (Chapter 6).

- Makes specific proposals for new research that can build on the work presented in this thesis. (Chapter 11).

1.3 A note on joint work

Section 2.3.1.1 on 'ontology and the design-based approach', section 4.4.1 on 'motive processing' and chapter 10 on 'a circulation of value analysis of attachment and loss' contain joint work that first appeared in (Wright, Sloman & Beaudoin, 1996), a paper originally written by me but revised and added to by Aaron Sloman. However, chapter 10 has been revised for this thesis, in particular extending the analysis of grief to mental pain and pleasure.

Chapter 8 first appeared as a joint technical report with Michel Aubé (Wright & Aube, 1997). Aubé provided information and references on multi-agent system research, convinced me of the importance of the concept of 'commitment', and provided comments on a first draft. However, the ideas and text are entirely my own fault.

Chapter 2

Methodological preliminaries: emotions and the design-based approach

This chapter briefly considers what emotions might be, why they are worth studying, and how they may be understood. The design-based approach to understanding mental phenomena is summarised, and the relevance of 'consciousness' to theories of emotion considered.

2.1 What are emotions?

As people use emotion words to describe their own and others internal states and visible behaviours it may appear that we already know what emotions are. For example, observers of a man shouting and kicking a car would describe him as 'angry' or 'frustrated'. The man, if asked how he was feeling, would probably use the same terms to describe his own state. But this is misleading: the ability to know roughly what kinds of conditions generate what kinds of emotional state, to recognise the existence of an emotional state in oneself or in others, and to deduce the range of mental and behavioural dispositions that follow from particular emotional states, constitutes a set of very useful social skills, skills that often take many years to acquire. However, skills are not theories. Anticipating emotions, recognising emotions and projecting the consequences of emotions does not require detailed knowledge of the underlying mechanisms of mind. For example, people can know what televisions, cars, washing machines, and word-processors do, and accordingly plan their interactions with them, without knowing in detail how they work. Folk psychological knowledge of the emotions, although useful, does not explain the mechanisms underlying emotion, although it may contain implicit assumptions about them. To meet the demands of scientific rigour existing concepts of emotion will need to be revised, extended or overthrown. Folk psychology may remain separate and autonomous from a new technical vocabulary of the emotions, in much the same way that naive physics coexists with classical and quantum mechanics.

> Is there a right definition [for emotion]? I suspect we must wait for deeper theories about the underlying mechanisms before we can hope to define precisely what kinds of phenomena we are talking about, just as people had to wait for modern physics and chemistry before they could have good definitions for terms like 'water' and 'salt'. (Sloman, 1993d)

A further problem of providing a definition of emotion is that different emotion researchers (e.g., psychologists, biologists, cognitive scientists etc.) often use different vocabulary for the same phenomena or use the same vocabulary for differing phenomena (Read & Sloman, 1993; Kagan, 1978). There is much terminological confusion in the literature. In addition, Pfeifer (1994) makes the point that no consensus has been reached on what actually constitutes emotion. This is to be expected if no consensus has been reached on the underlying mechanisms of mind.

Theories need to precede definitions. Therefore, 'emotion' is not defined in this thesis. Instead, theories of mental mechanisms are explored and new terminology introduced to refer to the processes those mechanisms generate. The new terminology can then be related to folk psychological concepts of 'emotion'. However, the word 'emotion' is still employed to refer to the assortment of phenomena under consideration.

In summary, we do not really know what emotions are, although there are many detailed theories, some of which are reviewed in chapters 3 and 6. This thesis adds to these theories, and is a partial answer to the question, 'What are emotions?'

2.2 Why study emotions?

Apart from the intrinsic interest of explaining a natural phenomenon, studies of emotion can make practical contributions in a number of areas. For example, a better understanding of emotions could lead to better clinical practice when dealing with emotional disorders, such as depression. Information processing theories of emotion could be mapped onto the neural architecture, providing neuroscientists with a better understanding of the functions of, and relations between, geographical areas of the brain. Developing theories of underlying mechanisms of emotion can feed into the design and implementation of artificial intelligences, particularly if emotional responses are found to be adaptive in various task domains. Theories of emotion can contribute to psychology, philosophy, and artificial intelligence.

2.3 How can the emotions be explained?

The emotions can be explained in different ways depending on the kinds of questions that are asked. For example, one can ask *how* emotions occur, which requires a 'wires and pulleys' causal explanation of the mechanisms that generate emotions. Alternatively, one can ask *why* emotions occur, which requires an evolutionary or social explanation of the reasons why emotional states and behaviours evolved. One can ask questions about the development of emotions in individuals (*when* questions), that is, whether emotions are innate, learned, or both, and what kinds of emotional changes may occur during the life cycle. Or one can ask *what* emotions are for, which requires an explanation of the function of emotions. Different questions lead researchers to concentrate on different aspects of emotional phenomena. This thesis, for example, concentrates mainly on how and what questions, such as how an agent may both react quickly to actual events yet also plan for possible future contingencies, and what niche requirements are satisfied by 'emotional' states and behaviours.

Theories of emotions can be constructed at different levels of abstraction. Neuroscientists study, among other things, the signal processing that occurs in neural circuits, whereas cognitive scientists and AI researchers explore what kinds of information processing may be implemented on those neural circuits (or microprocessors). This thesis is concerned with explanations at what Sloman (1994b) has called the information processing level of abstraction, or 'information level', a level concerned with both agent designs (i.e., part of Dennett's 'design stance' (Dennett, 1991)) and the semantic content of information that is acquired, created, manipulated, stored and used by those agents (Sloman, 1995g). Information level explanations operate at a level of abstraction 'higher' than the physical level but 'lower' than Newell's 'knowledge level' (Newell, 1990) and Dennett's 'intentional stance' (Dennett, 1991). Knowledge level and intentional stance explanations presuppose the rationality of agents (i.e., the agent's actions are reliably determined by a rational relation between its knowledge and goals), whereas the information level does not. Computer programmers normally describe the functioning of their programs at the information level, for example describing a program as manipulating lists of addresses, searching through a database of employees, indexing an array of pixels that represent a picture, or evaluating an arithmetical expression. Such descriptions make no mention of the physical implementation of the program.

In summary, the approach adopted in this thesis is to attempt to explain emotional phenomena at the information level of description.

2.3.1 The design-based approach

The study of emotions is divided into differing 'schools of thought' (Pfeifer, 1994). Approaches to the study of emotions can be very broadly categorised as *semantics*-based, *phenomena*-based and *design*-based (Sloman, 1992; Sloman, 1993d). Semantics-based theories analyse the use of language to uncover implicit assumptions underlying emotion words, for example (Wierzbicka, 1992). Phenomena-based theories assume that emotions are a well-specified category and attempt to correlate contemporaneous and measurable phenomena with the occurrence of an emotion, such as physiological changes or the firing of neural circuits. An early example is William James' peripheric theory (see (Calhoun & Solomon, 1984)); for a comprehensive review of many phenomena-based theories, see (Strongman, 1987).

In contrast, Sloman's design-based approach, a rational reconstruction of the practice of AI, takes the stance of an engineer attempting to build a system that exhibits the phenomena to be explained. Instead of analysing folk psychology in the hope of uncovering an implicit theory, or performing experiments on human and animal subjects, a design-based approach directly explores possible generative mechanisms. The next section briefly outlines the design-based approach.

2.3.1.1 Ontology and the design-based approach

It is assumed that (a) information-processing architectures exist, are implemented on human brains, and mediate both internal and external behaviour; and (b) that the design-based methodology allows a systematic approach towards high level functional congruity between artificial, explicitly designed architectures and certain important aspects of evolved, naturally occurring architectures, despite differences in low level implementation details.

Claim (a) underlies much contemporary Cognitive Science and has been argued for or presupposed by many theorists (e.g., see (Miller, Galanter & Pribram, 1970; Johnson-Laird, 1988; Simon, 1967; Simon, 1981b; Simon, 1981a; Simon, 1995; Newell, 1990) and (Palmer & Kimchi, 1984) for different sub-theses).

Claim (b) is more contentious and depends on finding appropriate levels of abstraction. There are other examples of congruity at high levels despite low level differences. Two physically quite different computing systems may both implement the same virtual machine architecture (e.g., both may be Prolog systems, or both may implement internet utilities, including mail, news, telnet and the World Wide Web). Similarly, it is often taken for granted that general principles of feedback control apply both to natural and artificial systems. What needs to be added to this, following much work in Artificial Intelligence, and the ideas in (Simon, 1967), is a level of explanation that involves richer and more profound forms of control of both external and internal behaviour using richer semantic structures and new sorts of control architectures to support various kinds of motivational processes (e.g., see (Simon, 1967; Sloman & Croucher, 1981; Sloman, 1987; Beaudoin & Sloman, 1993; Sloman, 1993b; Sloman, 1993c; Beaudoin, 1994; Sloman, 1994b; Sloman, Beaudoin & Wright, 1994) .)

Brains appear to support several rich ontologies at different levels of abstraction. In computing systems, ontologies are often 'stacked' in layers of implementation. For instance, a word processor package that manipulates pages, paragraphs, sentences, words, letters, and so forth, may be implemented in a 'virtual machine' corresponding to a high level programming language, which, in turn, is implemented in a lower level machine language, and ultimately by quantum physical states of electronic components, with several machine levels in between. The abstract machines at all levels are compound objects, composed of many different kinds of entities, relations and processes.

Moreover, causal and functional relations may hold between the high level abstract machine structures. (Changes in an abstract data-structure, such as a database of information about employees, can cause changes in what gets printed on pay slips.) These data-structures may have *semantics* in that they refer to individuals and their salaries, and so forth. Sloman has argued that this can include semantics *for the machine* (e.g. (Sloman, 1994b)).

In the case of human brains we do not know what the layers are. Yet causal relations between abstract structures clearly occur when a person's seeing something causes him to get angry, which in turn may cause him to strike out. The fact that ultimately people, like computers, are implemented in (ill-understood) physical mechanisms is not inconsistent with this. Even physical phenomena are normally explained well above the level of fundamental physics: most people who learn how a car engine works are not taught about quantum physics, but about carburettors, chokes, pistons, and so forth.

Though a designer often knows a great deal about how a complex system works, it may be impossible for others who merely observe the system to infer the internal processing. (Sometimes even the designer does not understand all the internal interactions.) This means that any philosophy of science that assumes that theories must be directly or easily testable is ill-conceived: it will fail for complex information processing systems, most of whose behaviour is internal and unobservable. Moreover, even knowing how the system works may not provide a basis for predicting particular behaviours if the behaviour depends not only on the design and current circumstances but also on fine details of enduring changes produced by a long previous history.

When studying systems we have not designed we can, at best, hope for a succession of theories accounting for more and more phenomena, using increasingly powerful explanatory principles, tested in part by implementing the theories in working designs and in part by relating them to the ever growing body of knowledge in neuroscience. There may never be a total ordering of merit among such theories, and the ordering may change over time as new phenomena are discovered. Objections to this approach are often based on a naive philosophy of science, or misplaced 'physics envy'. (For a broader view see (Lakatos, 1970), chapter 2 of (Sloman, 1978), and (Bhaskar, 1978)).

The design-based approach draws its inspiration from software engineering and conceptual analysis in philosophy (see chapter 4 of (Sloman, 1978)). It construes AI as a methodology for exploring an abstract space of possible requirements for functioning agents (*niche space*) and the space of possible designs for such agents (*design space*) and the mappings between them (Sloman, 1994a; Sloman, 1995a). Research strategies vary: they may be top-down, bottom-up or middle-out. All are potentially useful. This thesis is largely top-down, but the design-based approach does not exclude other options, for example the use of genetic algorithms to create designs by simulating evolutionary processes.

Although it is often assumed that AI is concerned only with algorithms (e.g., (Searle, 1980; Penrose, 1989)), *architectures* are more important (see chapter 4 for a definition). There is a need to understand global designs for *complete* systems, including their functional decomposition into coexisting interacting subsystems. Early work, still exploring general principles, need not make any commitment to the implementation details of mechanisms; for example, a neutral stance is taken towards symbolic or connectionist engines. Progress can be made by starting with 'broad but shallow' (Bates, Loyall & Reilly, 1991) architectures that combine many sorts of capabilities (such as perception, planning, goal management, and action). Each capability is initially implemented in a simplified fashion. Subsequent work gradually refines and deepens the implementations.

Sloman (1993b) claims that *architecture dominates mechanism*; that is, global design normally determines global capabilities to a greater extent than implementation details. Of course, ultimately designs must be linked to neural details and research will profit from the 'bottom up' studies of such details, which impose constraints on high level designs. Most of the constraints seem to be quite weak. Exceptions are the high level effects of drugs, which, in this thesis, have not been taken into account.

There is no assumed congruity between the design decisions 'taken' by evolution under environmental and competitive pressures and those taken by a designer when moving from initial requirements (what the system should do) to prototype design (how the system will do it). Rather it is merely claimed that the design-based methodology is a source of potential explanatory theories. Such theories will be improved under pressure of criticism, either because of things they fail to explain, or because they explain too much (e.g., capabilities people don't have), or because the designs could not have evolved naturally, or could not be implemented in brains.

Even an oversimplified or incorrect theory that yields a workable design can help the exploration of design space. Comparing it with other more 'realistic' theories aids understanding of the latter, for a system is not really understood unless it is known how changing it would produce different capabilities.

Designs satisfying the same information processing and control requirements may possess common design features, whether produced by natural selection or human engineering, just as birds and aeroplanes are both constrained by principles of aerodynamics. Over time the design-based approach may gradually approximate natural 'designs'. This could happen by increasingly taking account of empirical constraints and iterating the development cycle to deepen requirements and extend designs. Such designs can also be tested empirically and compared in more and more detail with their natural counterparts. The total research community is effectively engaged in a parallel cooperative search (see chapter 4 for a review of the agent architectures developed by AI researchers).

To summarise: (a) An architecture has causal powers that determine the capabilities of an agent and explain its ability to 'fit' into a part of 'niche' space; and (b) the design-based approach generates candidate architectures that may correspond to naturally occurring high level causal structures implemented upon neural substrates. These candidates can guide empirical investigations to check such claims. The design-based approach takes the stance of an engineer attempting to build, and understand the design options for, a system that exhibits the phenomena to be explained.

2.4 Against the reification of consciousness

From the first person perspective, emotional states are private phenomena. For example, a mourner may inform others of their grief, but those others will not be experiencing pain, or be having thoughts pertaining to the loss, or be unable to concentrate on everyday matters, and so forth. For some, an explanation of the subjectivity of emotions requires an explanation of 'consciousness', and without such an explanation any theory of the emotions is woefully inadequate. But this is not so: the study of emotional states no more requires an explanation of 'consciousness' than does the study of deliberative thought processes (see section 4.2.1). For example, models of cognition, such as GPS (section 5.7 of (Charniak & McDermott, 1985)) and the SOAR architecture (Laird, Newell & Rosenbloom, 1987; Newell, 1990), do not attempt to explain 'consciousness', even though deliberative thought processes are also, unless communicated, private phenomena. If cognition can be modelled and replicated without reference to theories of 'consciousness' then so can the emotions. However, a few points are worth making with regard to possible theories of 'consciousness'.

'Consciousness', like the term 'emotion', is ill defined and used to refer to different phenomena (hence the scare quotes). The study of 'consciousness' is currently very fashionable, and there are many competing theories. An unfortunate number of those theories reject the possibility that 'consciousness' could ever be fully explained in information processing terms, for example (Chalmers, 1996) and (Nagel, 1974). The rejection of the possibility of a mechanical explanation of 'consciousness' has a long philosophical history. Those rejections often rest on what Ryle (1949) has called a 'category mistake'. The following quotation, taken from Leibniz's 'Monadology', provides a good example of the error (for 'perception' read 'consciousness').

> It must be confessed that perception and that which depends on it are inexplicable on mechanical grounds, that is to say, by means of figures and motions. And supposing there were a machine, so constructed as to think, feel, and have perception, it might be conceived as increased in size, while keeping the same proportions, so that one might go into it as into a mill. That being so, we should, on examining its interior, find only parts which work one upon another, and never anything by which to explain a perception. G. W. Leibniz (1646 – 1716) (Liebniz, 1991).

A person may tour the grounds of a university, and see buildings, faculties, schools, offices, lecturers and students, and yet still ask 'Where is the university?' (Ryle, 1949). The visitor's mistake is to think that the term 'university' refers to one particular thing, rather than referring to a collection of things with mutual functional relationships. Leibniz commits this error in the above quotation, for a capability such as 'perception' can be implemented as collections of simpler components with mutual functional relationships (AI vision systems are an existence proof). Similarly, 'consciousness' is not one thing, nor is it an 'essence', but is an assortment of complex capabilities, such as those capabilities pre-theoretically referred to as introspecting, reflecting, deliberating, and self-referencing. The capabilities are implemented on neural components with mutual functional relationships. They may be present in different combinations and different forms in animals, humans and machines (Sloman, 1996c).

'Consciousness' is a collection of things that brains *do*, not a thing that brains do or do not possess. The reification of consciousness normally leads to panpsychism (everything is conscious) or solipsism (only I am conscious). Neither of these 'theories' can help us build artificial, self-conscious machines, nor do they generate any falsifiable propositions or explanations that deepen our understanding of how anything works.

A second, related, error is also made by Leibniz. In the above quotation, the giant, thinking machine, behaviourally equivalent to a mind, when examined by an observer mind, is found to lack precisely what the observer mind so obviously possesses – 'perceptions' (or 'consciousness'); therefore, minds are more than mere mechanical processes. This is a perspectival error, which is a failure to distinguish different types of information generated by different information processing mechanisms. Looking at a brain (or a giant machine) provides information about spatial relationships between things, but does not provide information about causal relationships between information bearing substates of the brain and between those substates and substates of its environment (i.e., causal relations that support semantic states). What that brain is perceiving, what that brain is thinking about, what that brain knows about itself, or what that brain could report about its feelings when asked, are questions that cannot be answered by vision (or hearing, touch and smell) because a human visual information processing system cannot detect and represent the requisite information. To answer such questions one would either have to ask the brain for the information, or actually be that brain introspecting. It should come as no surprise that when brain surgeons open patients' skulls they do not find any conscious experiences, for looking provides different information compared to introspecting (a different kind of information processing mechanism with different forms of representation and in a different access relation to the requisite information). *Just as smells cannot be heard, introspections cannot be seen.* Perspectival errors, in addition to category mistakes, often lead to dualist positions, for example hypothesising that 'consciousness' is separate from 'matter' and reconnecting the two ontological domains with superfluous 'bridging laws' (Chalmers, 1996).

In summary, these very brief comments maintain that the so-called problem of conscious experience does not impact upon theories of emotion, and that arguments against the possibility of information processing theories of 'consciousness' rest on basic philosophical errors. For example, (Dennett, 1978; Dennett, 1991; Sloman, 1990; Sloman, 1996c; Rey, 1997; Hofstadter, 1979; Hofstadter & Dennett, 1981; McCarthy, 1995) discuss alternative ways of viewing the problem of 'consciousness' that avoid these mistakes.

Part II

Perturbances

Chapter 3

The attention filter penetration theory of emotion

Consider jealousy, rage, triumph, excited anticipation, grief, obsessive love, despair, fear, joy, and so forth. These emotional states are all different in important ways. However, what they all have in common is *control precedence* (Frijda, 1986), that is *attention tends to be grabbed*. Both the griever and excited anticipator find it difficult to turn their thoughts to other matters. This is not to imply that people can never learn to control their emotions. However, the fact that people attempt to control emotions reveals that emotions attempt to control. The obverse of the control precedence of emotions is that a person is normally in a *passive* relation to them. For example, unlike actions that are causally and rationally dependent on a person's desires, emotions cannot be so instigated (Green, 1992). Ignoring dissimulation and the abilities of professional actors, one does not normally choose to be angry, sad or disappointed. It is appropriate to command someone to perform an action, but inappropriate to command someone to have an emotion (Green, 1992).

It is these kinds of facts that fit well with 'interrupt of attention' theories of emotion. Two are reviewed here as a preliminary to a partial computational implementation of Sloman's attention filter penetration theory (AFP) in chapter 5. Criticisms are postponed until chapter 6, where other design-based theories of emotion are reviewed, and where the emotions are given a more comprehensive treatment.

3.1 Interrupts: Simon

Simon (1967) proposes a design-based theory of the emotions. He discusses a number of important themes and ideas, primarily: that human behaviour is characterised by having multiple goals, that motivation can be thought of as that which controls attention at any given time, and that human behaviour, in many circumstances, can be interrupted. These ideas form the basis for a proposal that there is a close connection between the operation of an interrupt system and much of what is generally called emotional behaviour.

According to Simon, an event in the environment, a memory or a motivation, can displace current goals by interrupting processing in the 'central nervous system'. For example, the presence of a predator may cause the generation of new goals to deal with the altered situation producing, amongst other things, a flight, fight or fright response. Other effects of an interrupting stimulus could include arousal of the autonomic nervous system and the production of feelings of emotion.

Simon was primarily concerned with the process of goal interruption and consequent change of behaviour, rather than physiological and subjective phenomena. Interruption is required to serve the real-time needs of the organism. Simon distinguishes three types: needs arising from uncertain or unpredicted environmental events (as in the predator example), various physiological needs (internal stimuli, such as hunger), and 'cognitive associations', for example memory associations that may cause anxiety.

Simon also distinguishes adaptive and non-adaptive interruption. The emotional stimulus is normally to be considered as more *interrupting* than *disrupting* serving to help the organism pursue its multiple goals, rather than hindering. However, in certain cases, the emotion-producing stimulus may be persistent and intense causing the invoked goal to be repeatedly interrupted resulting in maladaptive behaviour.

Learning can change the efficacy of certain stimuli to cause interruption, or allow new associations to cause previously non-interrupting stimuli to interrupt. For example, learning to ignore car alarms, or learning to avoid black ice on motorways. In addition, the organism may acquire new or modified response patterns to interrupting stimuli. For example, an experienced car driver has a repertoire of plans for dealing with problem situations. Learning will tend to reduce emotionality of response as situations become more and more familiar.

Simon summarises his own theory as follows:

> The theory explains how a basically serial information processor en-

> dowed with multiple needs behaves adaptively and survives in an environment that presents unpredictable threats and opportunities. The explanation is built on two central mechanisms: 1. A goal-terminating mechanism [goal executor] ... 2. An interruption mechanism, that is, emotion, allows the processor to respond to urgent needs in real time (Simon, 1967).

The theory implies that 'organisms' have two kinds of processing that operate in parallel: a goal executor that generates actions, and vigilational (Beaudoin, 1994) processes that continuously check for contingencies that require urgent attention. The former, being resource limited, is interruptable by the latter.

Jealousy, rage, triumph, excited anticipation, grief, obsessive love, despair, fear, joy, and so forth, all normally involve beliefs about states-of-affairs that are convergent or divergent with important goals. For example, a jealous person has their attention interrupted and held by beliefs about the real or imagined behaviour of a loved one that are divergent to goals of attachment. Similarly, a fearful person has their attention interrupted and held by beliefs about real or imagined threats to goals of self-preservation.

Simon's theory does not account for all aspects of emotional phenomena, but it does explain why mental architectures may need to allocate attentive resources, and how this can be done using 'emotional' interrupts that very quickly divert attention to new, important and urgent matters.

3.2 Loss of control of attentive processing: Sloman

Aaron Sloman's attention filter penetration (AFP) theory (Sloman & Croucher, 1981; Sloman, 1987; Sloman, 1992) is an extension of Simon's theory. It introduces new, architectural detail implicit in Simon's paper. A variable threshold interrupt filter is proposed that controls the ability of new motivators, thoughts or percepts to disturb or divert attention. The need for a filter mechanism is deduced from the assumption that ongoing activities use resources, both cognitive and physical, that are limited; therefore, these activities will need both protecting from and be open to interruption to ensure adaptive behaviour. The variability of the threshold allows the level of protection to be dependent on context.

Sloman's theory focuses on dispositions to interrupt current cognitive processing rather than Simon's focus on interruption of current goals. Emotional states involve a disposition to divert attention without necessarily disturbing ongoing goals or actions. Four dimensions along which motivational states can vary are distinguished: insistence, importance, urgency and intensity[1] The insistence of a motivator is its propensity to interrupt attention, and is a heuristic measure of the importance and urgency of the motivator. It is heuristic as it needs to be computed inexpensively without diverting the very resources the filter mechanism is designed to protect.

The insistence of a motivator, therefore, is a dispositional state (see (Ryle, 1949) that analyses mental terms as disposition words). It has a tendency, or potential, to disturb and divert attention but need not actually surface through the filter or disturb ongoing processing. Sloman describes the strong potential for disturbance and diversion of attention as a characteristic of many of the states called emotional. However, these states can exist without actual diversion of attention; for example, jealousy can persist while other activities occupy attention for some time. One consequence of this theory is that there are only differences of degree between emotional and non-emotional motivational states. The theory also implies that many different emotional states can co-exist.

3.2.1 Perturbances

The AFP theory introduces a technical concept for the 'loss of control' characteristic of emotional states.

The term *perturbant* is reserved for a state in which there is a partial or total loss of control of attention. Perturbances can be occurrent, in which case there is an attempt to control the contents of attention, or dispositional, in which case there is no attempt to control the contents of attention. An occurrent perturbance is due to the continual surfacing of postponed or rejected, or unwanted, motivators (Beaudoin, 1994), or possibly disruptive thoughts, images, and the like (e.g., a catchy tune that won't 'go away'). Such disruption can interfere with the management of other, important goals. It is what the filter normally prevents.

Perturbant states differ in several dimensions: duration; whether the source is internal or external; semantic content (what is referred to); type of disruption (it could be due to a goal, thought, or recollection); effect on attentive processes; frequency of disruption; positive or negative evaluation (compare grieving with being unable to stop thinking about the victory one has recently won – grieving and gloating have much in common); how the state develops; whether and how it decays; how easily it can be controlled, and so forth.

[1]Intensity is the ability of a goal, once management resources have been diverted to it, to remain adopted. See section 4.4 and chapter 5.

Perturbances, like 'thrashing' in an overloaded computer operating system, are emergent effects of mechanisms whose major role is to do something else (just as thrashing arises from the paging and swapping mechanisms in the operating system). Perturbances arise from the interactions between (a) resource-limited attentive processing, (b) a subsystem that generates new candidates for such processing and (c) a heuristic filter mechanism. These design elements arise from the requirements for coping with complex and rapidly changing environments. Perturbances do not initially arise because of some special perturbance generating (or emotion generating) mechanism. Thus it is misguided to ask what the *function* of perturbant states is or to postulate a perturbance mechanism.

For example, a person hears the latest pop single on the radio and soon the tune is constantly 'replayed' in their mind, perhaps diverting attentive resources from other matters. This is not a perturbant state unless the following condition holds: *if* the person were to wish that the tune would 'go away' and tried to put it out of their mind but found it difficult to do so *then* there is a disposition to lose control of attentive resources. For instance, some catchy tunes can become very irritating and annoying. However, the loss of control need not be occurrent for a perturbant state to exist. For example, excited anticipation is a perturbant state even if the person experiencing that state does *not* wish to turn their thoughts to other matters: they may be quite happy continually thinking about the presents they will get for their birthday.

(In a recent clarification of the theory Sloman distinguishes three classes of emotional states associated with three different types of control in agent architectures. These three classes are described in sections 4.3.1, 4.3.2 and 4.3.3.)

3.2.2 Self-control

The definition of perturbance is therefore counterfactual, and requires an architecture sufficiently sophisticated to support goals whose objects are internal states. For example, not all agent architectures can support a distinction between being in control and not being in control of one's thought processes. It is not clear that a rat ever has control of thought processes. In that case it cannot lose control. If that is so, the sorts of processes being discussed cannot occur in a rat (Sloman, 1996c). Having or losing control requires a mental architecture that can monitor, evaluate and modify thought processes.

The ordinary notion of 'self-control' is not a unitary concept admitting of a unique analysis. Like many mental concepts it covers a variety of cases and further study is required to investigate how many of them can be accommodated within a design-based framework.

Sloman emphasises that for a system to support concepts of 'self-control' it must be able to have goals relating to its own thought processes. The system is in control when everything that occurs in it is consistent with all the currently adopted goals. This does not imply that everything that happens is *generated* by those goals, for that would rule out intrusions, such as feeling hungry, or a new desire to help someone in trouble. These can arise without contradicting one's view of what one should be like (Wright, Sloman & Beaudoin, 1996).

There need not be only *one* coherent global set of goals, preferences, and so forth, since some people seem to change personality from one context to another, like the kind father who is an aggressive car driver. A mental architecture might allow a number of *different* sets of mutually consistent high level dispositions that co-exist, though only one set is active at a time (as sketched in chapter 10 of (Sloman, 1978)).

Detection of incongruent states requires some sort of self-monitoring of the global 'picture' and an explicit evaluation of it as fitting or not fitting the agent's ideals, long term objectives, or previous decisions regarding (for example) what to think about, or which desires are unacceptable. If a substantial amount of what is happening at any time is inconsistent with the agent's dominant evaluations and preferences, then the agent is, to that extent, partly out of control, even if all the disturbances and disruptions are generated entirely within the system, for example from lower level automatic, non-attentive, processes, or 'cognitive reflexes'. Addictions are an extreme case.

3.3 Summary

Sloman's AFP theory states that perturbance, a partial or total loss of control of attention, is a characteristic feature of states that are commonly called emotional. Perturbances may be occurrent or dispositional. Perturbant states require mental architectures with vigilational, attentive and self-controlling processes. Vigilational processes can generate new goals for resource limited attentive processing, and self-controlling processes can generate goals that aim to control the contents of attention. For example, grief consumes the mourner. If the mourner wished to turn their thoughts to other matters they would find it extremely difficult to do so.

Chapter 4

Agent architectures

In AI the word 'agent' is used promiscuously to refer to all kinds of entities, such as persistent processes, reactive controllers, programs that control parts of real or simulated worlds, simple adaptive algorithms, and so forth. This author is content to use Franklin's definition of an agent:

> An *autonomous agent* is a system situated within and a part of an environment that senses that environment and acts on it, over time, in pursuit of its own agenda and so as to effect what it senses in the future. (Franklin & Graesser, 1996)

On this definition a simple thermostat, a negative feedback control system, is an agent. However, most AI agents are much more complex than a thermostat.

There are at least two uses of the word 'architecture'; one referring to an abstraction or design that is common to many instances of the architecture; and the other to concrete instances of such designs. In this thesis the word 'architecture' is normally used in the the former sense, in which an architecture is a collection of features common to a class of entities. Each instance of an architecture is composed of coexisting, interacting substructures with various capabilities and functional roles. A substructure may also have an architecture. The architecture of a complex system can explain how its capabilities and behaviour arise out of the capabilities, behaviour, relationships and interactions of the components. An architecture can be specified at different levels of detail, for example at a high level of abstraction the architecture of a house will not include the occurrence of particular bricks, whereas a more detailed architectural specification would (Wright, Sloman & Beaudoin, 1996).

An *agent architecture*, therefore, is a description of the information processing mechanisms of a control system at a particular level of abstraction.

There are many examples of agent architecture design in the literature. This chapter briefly examines requirements for autonomous agency, summarised from a number of sources but mainly from (Beaudoin, 1994), and provides a critical overview of some examples of existing agent architectures. The examples are representative but not exhaustive. A more detailed review can be found in (Wooldridge & Jennings, 1995; Wright, 1994; Beaudoin, 1994). Finally, three forms of control in agent architecture design are identified, followed by a high level description of Sloman and Beaudoin's agent architecture design, which is the chosen design for this thesis.

4.1 Requirements for autonomous agents

The requirements for a system detail what the system should do but not how it will be done. An agent architecture is required to produce coherent, effective and robust behaviour (Beer, Chiel & Sterling, 1990) in a complex and unpredictable domain such that its goals are achieved subject to some evaluatory criteria. These requirements for successful operation in dynamic, unpredictable, real time environments pose certain design problems.

An autonomous agent is capable of producing its own goals but has limited resources with which to satisfy them. The fundamental goals of the agent may have been designed by a human engineer, or evolved by natural selection, and they may be enduring throughout the lifetime of the agent, or subject to modification. Fundamental goals can generate derivative goals. Goals may be generated internally, such as a periodic goal to eat food, or externally, such as a goal to prevent a child touching a hot stove.

The computational resources of the agent will be finite. For example, humans find it difficult to listen to more than one conversation at once. Also, the agent will be physically constrained. It will only be able to move at a certain pace, manipulate a finite number of objects and so forth. Good design solutions will manage an agent's finite resources as efficiently as possible (although efficiency is a difficult notion to define (Sloman, 1995b)).

The agent will need to pursue multiple goals, with perhaps conflicting objectives. It will have many diverse tasks to perform. In addition, goals will have associated temporal constraints, such as a goal to catch a train at a certain time. Therefore, an autonomous agent needs to schedule its goal processing and actions. This requires the ability to select between multiple motives (Sloman, 1985), prioritise goals, decide on a level of commitment towards current intentions, and notice opportunities for actions that satisfy more than one motive.

The requirement for timely scheduling of actions and the constraint of limited resources, both computational and physical, require that current processing be interruptable (see (Simon, 1967)). For example, to react to new, motivationally relevant events in the environment the agent will need to interrupt its ongoing processing and switch its 'attention' to new contingencies (Sloman, 1987).

For example, at one moment the agent may have very little to do and have the luxury of deliberation while at the next moment the agent may need to perform many complex tasks very quickly. The unpredictability of the environment renders complete planning prior to action impossible. Instead, opportunities and threats to plans will need to be constantly monitored for. For example, a car driver may be planning his evening while driving, only to rapidly interrupt his current thoughts and switch attention to a child's ball that has bounced into the road.

To detect such events the architecture must be able to generate motivations asynchronously to current processing. A level of coarse-grained parallelism is therefore necessary (Simon, 1967; Sloman, 1978; Maes, 1990) to enable execution of current goals and at the same time check for new information that may entail goal revision.

Limited resources also entail that the agent's knowledge will be incomplete and, in many cases, erroneous. The agent will have limited predictive powers. The total effects of its actions will be unknown. The current situation (both externally in the environment, and internal within the agent) will provide 'too much' information (Hayes-Roth, 1990): the agent, having limited time for processing and selecting, will need to focus its processing and ignore irrelevances. Information will be widely distributed both in time and space, requiring the agent to search for relevant information and remember, recall and integrate past information.

Architectures that model autonomous agency will need to integrate a wide range of behavioural capabilities. Bates, Loyall & Reilly (1991) term such architectures 'broad'. For more requirements see (Boden, 1972; Hayes-Roth, 1990; Oatley, 1992; Simon, 1967; Sloman, 1985; Sloman, 1987).

4.2 Examples of agent architectures

Normally agent designers do not intend to meet all the requirements outlined above but attempt to meet only a subset. The requirements vary depending on research goals. For example, designers of reactive architectures attempt to meet requirements for robust and quick responses to environmental contingencies, designers of deliberative architecture attempt to meet requirements for problem-solving and planning capabilities in novel situations, and designers of reactive-deliberative architecture attempt to integrate both kinds of capabilities within a single control framework. There are many examples of all these kinds of architectures in the AI literature. The following sections critically review a small subset of those designs, concentrating on reactive-deliberative architectures.

4.2.1 Deliberative architectures

Very briefly, deliberative agent architectures are programs that perform something like deductive reasoning in order to solve problems, such as classical planning systems (e.g., STRIPS, NOAH, NONLIN etc., see chapter 9 of (Charniak & McDermott, 1985)) or models of cognition, such as GPS (section 5.7 of (Charniak & McDermott, 1985)), the SOAR architecture (Laird, Newell & Rosenbloom, 1987; Newell, 1990), Anderson's ACT-R (Webmaster, 1996) and blackboard architectures (see section 2.2.2 of (Beaudoin, 1994)). Such systems can be usefully conceived as agents when they attempt to control parts of a real or simulated world. For example, Nilsson's SHAKEY the robot (Nilsson, 1984) used a STRIPS-style planner to plan its behaviour. Deliberative architectures tend to work on a single task at a time and do not interleave deliberation with action. Normally they manipulate declarative representations of possible actions, including representations of states of affairs, such as action preconditions and consequences. Deliberative architectures perform search to find solutions to problems. For example, a planning system can construct a plan to achieve a goal by searching for the ordered list of primitive actions that transforms an initial state into a goal state.

4.2.1.1 Problems with disembodied deliberation

The paradigm domain for early planning systems was 'blocksworld', a world consisting of a number of blocks that the agent may stack. The planning agent is the only active entity in the environment and the configurations of the blocks are precisely known at all times. 'Blocksworld' does not require fast responses and does not change while the planner deliberates.

Problems arise when deliberative architectures are placed in dynamic domains that impose real-time constraints on action. For example, if an agent cannot be certain of the effects of its actions, cannot make the assumption that the world will remain static while it thinks, and cannot assume it knows everything about the world, then a purely deliberative architecture will fail (Lyons & Hendriks, 1992). For example, a planning agent may construct a plan based on out-of-date assumptions about the state of the world. Execution of such a plan is unlikely to achieve the agent's goal.

Much of the criticism directed at deliberative architectures, particularly from designers of reactive systems, is misplaced because deliberative systems do not attempt to meet requirements for operation in dynamic domains. The designers of deliberative systems have rightly abstracted from the problems of real-time behaviour and concentrated their efforts on the problems of intelligent deduction.

4.2.2 Reactive architectures

In reaction to the unrealistic assumptions of classical planning, such as the assumption of an unchanging, static world, a new design paradigm arose in the mid-eighties, the behaviour-based approach (Lyons & Hendriks, 1992), which sought to situate and test agents within unpredictable, changing and preferably real world domains. Brooks (Brooks, 1990; Brooks, 1991b) argued that agents should be composed of reactive modules directly coupled to sensing and acting apparatus, where each module produces a particular behaviour, rather than collections of modules, such as a sensing, acting, and planning modules, that interact to produce overall behaviour. In other words, the agent should be composed of behaviour-producing modules. The subsumption architecture (Brooks, 1991a) is an example of this approach. Each behaviour module is a connected to others via wires that form a layered architecture. There is no explicit representation of goals or other symbolic structures. Some behaviour modules inhibit or prevent others, 'subsuming' the lower level behaviour into a higher level synthesis. For example, the behaviour *chase-light* could be subsumed by a higher level behaviour *avoid-obstacles* to synthesise the composite behaviour of chasing a light and avoiding obstacles.

Agre & Chapman (1987) describe Pengi, an agent that plays a commercial arcade video game called Pengo, a domain that places real-time demands on agent functioning. The design of Pengi is motivated by a theory of activity that emphasises the importance of routines in the dynamics of everyday life. A routine is a frequently used pattern of interaction between an agent and its world. In addition to routines, a simple way of representing the world is introduced, variously called indexical-functional aspects, indexical representation (Chapman, 1989) or deictic representation. A deictic representation represents only what is necessary, immediate and functionally important to the agent in its current situation. For example, instead of representing the location of a block using absolute coordinates, such as **(AT BLOCK-213 427 991)**, Pengi will employ unitary entities such as *the-block-I'm-pushing* that gain semantic significance within a particular action context.

The Sonja system (Chapman, 1990) is another example of a reactive architecture. Sonja is an agent operating in the Amazon domain (a 2-d world of walls, treasures, and mobile opponents). She has various tasks to perform and a human instructor can help and guide Sonja via natural language commands. Sonja has the ability to interleave various courses of action but, like Pengi, uses no explicit representation of plans. The system is implemented as a digital circuit with explicit connections between sub-elements and direct connections to sensory variables. The design motivation is that the world is its own best model and therefore it is more efficient to look at the current, concrete situation for action selection than relying on deliberation and complex, internal models of the real world.

The behaviour-based approach is concerned with building architectures that are robust and successful in real world domains. The approach eschews traditional plan representation and execution. The strength of behaviour-based architectures lies in their ability to use local patterns of activity in their current surroundings as a basis for generating predefined or simple to compute action responses.

4.2.2.1 Problems with the behaviour-based approach

Brooks (1991b) states that intelligent activity does not require representations. Agre & Chapman (1987) state that in practical activity representations mostly 'get in the way'.

Indexical-functional representations are undoubtedly useful. They may be a phylogenetically older form of representation well suited to routine activity and simple stimulus-action responses on the level of insect-like intelligence. However, explicit representation schemes are needed for more complex control tasks and behaviours, such as referring to the past or future.

An implicit assumption hidden in the statement that the 'world is its own best model' is that perception is the sole source of information about the world. A 'concrete' situation for an agent also includes its own internal environment: beliefs, desires, memories and so on. In many instances action selection will be more easily achieved by 'looking' at past experience rather than the immediate, external environment; for example, a rat will press a blue button rather than a red button if it associates pressing the red button with a memory of an electric shock.

Behaviour-based approaches do not solve the problems of action selection. Chapman (1990) claims that 'the problem of action selection is easy and has been overemphasised' as 'Sonja performs actions because they make sense in concrete situations, not because they are the next step in a program'. This is clearly a case of solving a problem by avoiding it: the requirements for Sonja are such that it is not faced with the problem of multiple motives, or even multiple strategies for achieving the same action outcome. For example, Sonja has no long term goals and therefore the problem of such a goal conflicting with the demands of the immediate situation cannot arise. The problem of action selection is effectively factored out. Also, action selection often requires forward planning using hypothetical reasoning to search for solutions. For example, Sloman (1996b) uses the example of crossing a busy road to demonstrate this: if a person cannot cross a busy road they may consider walking to a pedestrian crossing, using the button to stop the traffic, crossing, and then walking back up the road. There may be two pedestrian crossings, one to the left and one to the right. Choosing between them may involve analysing both routes prior to action. The Sonja architecture is incapable of providing this functionality.

The behaviour-based design prescriptions ignore the fact that humans frequently plan their activities, for example planning a holiday, and frequently think about possible, rather than actual, situations. This kind of functionality is necessary for intelligent behaviour. The assumption that non-trivial behaviour can be strictly situationally determined has yet to be demonstrated. It is inconceivable how any behaviour-based architecture could cope with novel global task constraints, such as deadlines. Behaviour-based architectures make use of world regularities at design-time (by pre-compilation of action selection in the connectivity of wires) but make no provision for discovering such regularities at execution-time (Pryor, 1994). (Norman, 1994; Norman & Long, 1995) illustrates how a behaviour-based system will become inefficient when a multiple-goals requirement is introduced and argues that a symbol manipulating mechanism is necessary to overcome this drawback.

Etzioni (1993) describes how simulated domains, such as the UNIX operating system, can satisfy all the real world requirements that Brooks deems necessary for situated activity; and he also makes the important point that designing and implementing 'softbots', or simulated, software robots, is a much speedier process than building real, physical machines.

4.2.3 Reactive–deliberative architectures

Reactive–deliberative architectures are concerned with combining reactivity and deliberation, and are an attempt to integrate the desirable features of both deliberative and reactive architectures.

4.2.3.1 Reactive action packages: Firby

Firby (1987) notes that having to choose actions at execution time is unavoidable in a complex, dynamic domain; that is, reactive execution must occur at some level in any autonomous system to maintain robustness. An agent cannot plan completely in advance: uncertainty prevents correct reasoning, and urgency constrains the time available for such reasoning. Reactive planning of some kind will therefore be needed. A more deliberative, classical planning scheme could then be implemented upon such a reactive ground.

Consequently, Firby (1989) has proposed a model of purely reactive planning based on the concept of Reactive Action Packages, or RAPs. Each RAP can be viewed as an independent entity embodying a goal that competes for processing resources with other RAPs.

A RAP may contain explicit sensory tasks within its plan allowing the same plan execution mechanism to deal with action execution and sensory guidance. Each RAP obeys three principles while running: which action to execute next is based only on the current world state, when a RAP completes execution it is guaranteed to have satisfied its goal, and a RAP will only fail if it does not know of any way to reach its goal from the current state. Each RAP has a pre-defined set of methods for achieving a goal and only need choose between these methods, called the task net, rather than construct new ones. A RAP, therefore, is a structure that links a goal, a success test for achievement of the goal, a collection of methods to achieve the goal applicable in different contexts, and invocation conditions that determine when a particular RAP is appropriate.

The RAP control algorithm is designed to address the problem of execution monitoring and replanning in uncertain domains. A RAP interpreter and execution queue provide a mechanism for coordinating competition between RAPs. Such a scheme allows interleaved RAP execution as, for example, when a running RAP stops and returns to the execution queue to wait for a subgoal to complete; in this situation the interpreter can choose another RAP to run in its place. The problem of goal selection, or choosing which RAP to run next, is based on temporal deadlines and an ordering on RAPs made by task nets.

Two limitations of the RAP system are discussed by Firby. A RAP may fail without preventing the original conditions in the world from re-generating the failed RAP. This could lead to an indefinitely long loop. Also, the RAP system cannot think ahead. Both these limitations point to the need for an extra layer of control that places constraints on RAP behaviour prior to execution; in other words, neither urgency or uncertainty obviate the need for more deliberative decision making. Hence, in (Hanks & Firby, 1990) the RAP system is extended by considering the extra, deliberative layer of control needed in an autonomous agent.

The addition of planning ahead and reasoning abilities generates two new design problems: how to deliberate, and how to coordinate deliberation and reactive execution. This problem is further divided into the representation problem (how to model a complex and dynamic world) and the control problem (how to manage such information so that the agent acts effectively and efficiently). Any solution to the control problem must be able to curtail the deliberation process at any time in order to guarantee reactivity. Hanks and Firby believe that the combination of an execution system based on RAPs and a deliberation system based on a probabilistic world model manager and projector will meet this criteria. Their layered architecture is still in development and they provide few details of the deliberation system.

4.2.3.2 TouringMachines: Ferguson

Another example of a reactive–deliberative architecture is Ferguson's TouringMachine (Ferguson, 1992). The TouringMachine is an integrated software control architecture designed for controlling the actions of autonomous agents operating in complex environments; in particular, the TouringWorld, a multi-agent traffic domain. The design consists of separate activity producing behaviours in a layered control framework, which resembles the behaviour-based approach of Brooks. However, there the similarities end: the TouringMachine uses explicit goal and plan representation, and each activity producing layer is not a simple control system connected via wires, but a more or less sophisticated control algorithm.

The TouringMachine has three different control layers: a reactive layer, a planning layer and a modelling layer. All layers operate concurrently and are connected independently to sensory input and effector output. Each layer is intended to model the agent's environment at a different level of spatio-temporal abstraction.

For example, the reactive layer provides the agent with reactive capabilities to cope with immediate or short term contingencies that higher level layers would have insufficient time to compute responses to. In Ferguson's implementation the reactive layer is hard-wired and domain-specific.

The planning layer generates and executes plans of action to achieve the agent's goals. Use is made of sketchy, procedural plan structures and the system can defer commitment to a specific subplan until run-time.

The modelling layer's function is to attempt to detect and predict potential goal conflict situations. Using knowledge about its own functioning the TouringMachine is able to project this knowledge onto other entities in the environment and model their behaviour. The modelling layer, therefore, provides the agent with limited reflective and predictive capabilities.

The integration of the three layers to produce consistent behaviour is achieved using a control framework. This control framework consists of a set of control rules, called suppressors and censors, that resolve perception and action command conflicts arising from instructions sent from different layers.

The TouringMachine architecture has been implemented and extensively tested in the TouringWorld testbed. However, many of the planning problems in the TouringWorld are problems of navigation: how to get from A to B, how to avoid collisions, how to arrive on time and so forth. The proposed architecture may be robust in such a domain but may not generalise to richer domains. For example, the control framework, consisting of control rules, shunts the control problem into a large collection of exception rules that could prove unwieldy when the architecture is scaled-up (the more the agent does the more control arbitration is needed between active systems). The focus of attention mechanism (i.e., deciding what to concentrate processing resources upon) relies on a 'relatively static focusing rule set'. In other words, there is no disciplined way of controlling attention in this architecture. For example, unlike Sloman and Beaudoin's agent architecture (see section 4.4), no explicit distinction is made between the urgency and importance of goals.

4.2.3.3 PARETO: Pryor

PARETO (Pryor, 1994) is a plan execution system that operates in an uncertain and dynamic environment. The design of PARETO is motivated by the view that information gathering and opportunity taking are essential aspects to planning in an unpredictable domain. In addition, PARETO makes decisions on the fly, filling in the details of sketchy plans at execution time. The system recognises opportunities and threats to its current goals during plan execution by using a heuristic based on reference features, explained below.

Pryor makes a distinction between the types of decision a planning system will need to make: those that are deferred and those that are unforeseen. Deferred decisions arise due to the unavailability of required information during plan formation. Deferred decisions, therefore, will require information gathering at execution time. The Cassandra system (Pryor & Collins, 1993) was designed to solve this problem in a traditional planning system by including explicit decision and information gathering steps within plan representations. PARETO, however, was designed to address the problem of unforeseen decisions, which are due to the unpredictability of the environment. Such decisions require opportunity taking, which involves detecting when an unforeseen situation is helpful for achieving a goal (an opportunity) or harmful to a goal (a threat).

PARETO recognises opportunities (and threats) by using a filtering process based on reference features (Pryor & Collins, 1992). Reference features label functional stability, that is, they mark the functionally important aspects, relative to the agent's goals, of the elements that comprise a situation. For example, an agent may have an interview for a job and notice that a thread is showing on their jacket. Before entering the interview room the agent spots a pair of scissors on a desk; immediately, the agent uses the scissors to cut the thread. Thus an opportunity is detected and taken. The reference feature used in this situation would have been the functional label *sharp*. The goal to cut the thread would have been labelled with *sharp* as one of its reference features, therefore something sharp would be connected to the goal of cutting the thread. Importantly, the perceptual recognition of scissors would have generated the reference feature *sharp*. A simple matching process would then detect the opportunity. Such reference features are argued to be cheap to infer and readily available to the agent as they are already computed in the normal course of perception and situated activity.

A hypothesis – *the critical factor hypothesis* – underlies the use of reference features. It states that the presence of a single factor is often crucial for the existence of an opportunity in a given situation (the presence of *sharp* in the above example). The critical factor hypothesis maintains that many situation elements remain constant across many different situations. The use of reference features or similar would provide a mediating representational scheme between, for example, Agre and Chapman's diectic representations and traditional goal representations. Pryor (1994) anticipates such a possibility:

> Reference features form the basis of an effective filter for opportuni-

ties because they constitute an intermediate level of conceptualising the world between the physical vocabulary provided by perception and the functional vocabulary required to reason about goals.

The PARETO plan execution system is based on Firby's RAP system, extending it via a layered control architecture consisting of a robot control, plan execution and reasoning layer. It operates in the TRUCKWORLD domain consisting of a road system, various storage buildings, building sites in need of materials and a delivery truck.

No methodology is provided for choosing the reference features of situation elements, and it appears that they may have been chosen in an *ad hoc* fashion. However, addressing this problem is an instance of the more general problem of integrating learning and the acquisition of reference features, an area Pryor highlights for future research. A central claim for the opportunity taking filter process based on reference features is that it is computationally inexpensive; however, no mathematical argument is provided to support this claim. Such an analysis could show, for instance, that there is no combinatorial explosion of reference feature matching.

A reference feature mechanism for opportunity taking will need to be augmented by more sophisticated, deliberative reasoning. PARETO's reasoning layer, although not implemented, is designed to address this issue. The use of reference features is an elegant and simple solution to the problem of opportunity taking.

4.2.3.4 TRP architectures: Benson and Nilsson

Benson & Nilsson (1995) describe a sophisticated agent architecture that can act reactively, select between multiple competing goals, plan novel action routines, and observe the effects of its own actions in order to produce more robust plans. It operates in a simulated domain called Botworld, which consists of robots, obstacles, and moveable bars.

Benson and Nilsson identify four main requirements for their agent. First, the agent needs to react rapidly to commonly occurring situations that require stereotypical responses. Nilsson's *teleo-reactive program (TRP) formalism* (Nilsson, 1994) addresses this requirement by implementing the agent's plan steps as small reactive 'packages' that direct the agent toward a subgoal in a manner that continuously takes into account changing environmental circumstances. The formalism is also similar to Firby's RAPs and is described in more detail in section 5.3.2.

Second, the agent needs to switch attention between multiple goals with different time constraints. At any one time a number of goals may be active and may suggest incompatible actions. The agent must select which goal is to control behaviour. Benson and Nilsson, drawing on work in reinforcement learning, associate a *reward* with the achievement of a goal condition. Rewards may vary between goals and between contexts. The attention mechanism selects a goal that promises the highest reward that can be achieved with little effort as soon as possible; that is, the m active goals are ordered such that the sum $\sum_{i=1}^{m} R_i T_i$, where R_i is the reward for goal i and T_i is the earliest time at which i is expected to be achieved, is minimised (this is an approximation to maximising discounted future reward in delayed reinforcement learning). However, a problem with this approach is that the agent will exhibit inflexible behaviour: it will prefer a single high reward goal to a set of medium reward goals that may be easier to achieve sooner rather than later. Neither will the agent exploit opportunities:

> ... if one of the agent's lesser goals is to deliver a certain library book to a far-away office, it will not realize that it can save time by picking up the book while it happens to be in the library for some other reason (Benson & Nilsson, 1995).

These problems are avoided by introducing the concept of a *stable node*. A stable node is a condition that continues to hold relative to a set of actions (TR programs). For example, the condition that an agent holds a book is stable to many TR programs, assuming that the agent has unlimited carrying capacity. But the condition that the agent is in the library is not stable to any TR program that involves leaving the library. (The authors associate STRIPS-style add and delete lists with each TR program. Therefore, a node is stable relative to a set of TR programs if the node condition does not appear in any of the delete lists of the TR program actions.)

Stable nodes function as *milestones* towards achieving a goal condition. The attention mechanism, instead of minimising $\sum_{i=1}^{m} R_i T_i$, minimises $\sum_{i=1}^{m} R_i T_i^{sn}$, where T_i^{sn} is the estimated time required to achieve the closest stable node in goal i's TR program. Therefore, the agent will work towards achievable milestone conditions that are unlikely to be negated during subsequent action. For example, if the agent is in the library and has two goals, one which is very urgent and involves leaving the library, and one which involves delivering a book to a far-away office, then a close stable node will be the condition that states that the agent is holding the book. The delivery task may have a high value of R_i/T_i^{sn} compared to other goals, resulting in the agent deciding to pick up the book before leaving the library. Benson and Nilsson report that this mechanism chooses intuitively reasonable actions in a variety of Botworld situations.

Third, the agent needs to plan for novel contingencies and augment or modify its set of teleo-reactive programs. The TRP architecture can generate new hierarchical plans and extend existing plans, expressed as TR programs, to handle unexpected circumstances. For example, the agent may encounter situations that are not covered by the constituent conditions of a TR program. The planner can extend the TR program to deal with the new situation by repeatedly performing backward chaining reasoning from the current TR program condition until the novel condition is reached. The result of this operation is a new TR program segment that specifies actions to perform when the novel condition holds. This is similar to chunking in SOAR: when a goal impasse is reached, SOAR searches for the combination of existing rules that solve the impasse. When a combination is found this solution is 'chunked' into a new, higher level rule that can be used in similar situations in the future.

Fourth, the agent needs to be adaptive, able to change itself to meet new demands or improve upon its current behaviour. A major advantage of the TR architecture over the other architectures reviewed in this section is its ability to learn. The architecture can observe the environmental effects of its own actions and use this new information to construct new plans. The agent can explore its environment trying out different combinations of actions in different situations. Exploration can lead to the construction of new TR programs. The technical details of the learning algorithm are not reproduced here (but see (Benson & Nilsson, 1995)).

Benson and Nilsson have successfully managed to integrate a number of desirable capabilities within a single agent architecture. Of particular importance is the TR program formalism (see section 5.3.2) that combines features of reactivity (continual monitoring of environment and predefined responses to contingencies) with features of classical planning (hierarchical and recursive plans, free variables, add and delete lists etc.), and the attention mechanism that selects goals based on reward, expected completion time and stability of milestones. In addition, the architecture can support a form of learning.

However, as the number of active goals increases the computation required for selecting goals is likely to become more expensive (e.g., the more active goals, the more computation is required to discover stable nodes, and the more computation is required to order goals). For timely responses the architecture would need to concentrate attentive resources on a subset of the goals. Currently, the TRP architecture has no mechanism to do this. Also, it is not clear how the TRP architecture could meet deadlines, such as a goal to catch a train at a specified time. The selection of goals is based on a heuristic of high reward sooner rather than later; therefore, situations could occur where the agent pursues a rewarding goal at the expense of a less rewarding deadline goal, resulting in, for example, a missed train.

4.2.3.5 Procedural reasoning systems: Georgeff

Georgeff's Procedural Reasoning System (PRS) (Georgeff & Ingrand, 1989) aims to achieve a balance between acting and decision making. It has many features in common with Firby's RAP system but is designed to provide more powerful mechanisms for balancing decision making requirements against the constraints on time and information that are typical of complex domains.

PRS consists of a database containing the current beliefs or facts about the world, a set of current goals to be realised, a set of plans (called *knowledge areas*) describing processes for achieving goals, and an intention structure containing plans chosen for execution. An interpreter manipulates these components, selecting appropriate plans based on the system's beliefs or goals, and places these plans on the intention structure for execution.

PRS is an example of a BDI-architecture (Belief–Desire–Intention architecture) (Rao & Georgeff, 1991b; Rao & Georgeff, 1991a), which is a formalisation of an autonomous agent based on a branching-time, possible-worlds model of behaviour. Although this formalisation restricts the flexibility of the agent by requiring the agent's intentions to be consistent, the authors argue it is useful for characterising the major components of an autonomous agent. Informally, Beliefs are all the statements the agent believes are true in the world, Desires are all the states-of-affairs the agent would like to bring about in the world (including its own processing) and Intentions are commitments to action formed from the interaction of Beliefs and Desires. Any architecture that has explicit representations of Beliefs, Desires and Intentions can be viewed as an instantiation of a BDI-architecture.

Goals in PRS are specified using procedural logic. The use of a procedural logic is motivated by the belief that much commonsense knowledge about the world is in the form of procedures or sequences of actions for achieving goals. In brief, the procedural logic is concerned with processes, making use of such temporal operators as !p (make proposition p true), ?p (test p) and #p (preserve p), which can be combined with other conditions in an arbitrarily complex way. The operators of many standard planning systems, such as NOAH, SIPE and so forth, are restricted forms of these process descriptions.

Each element of procedural knowledge, or knowledge area (KA), is a belief of the system about the utility of performing certain action sequences in particular contexts. The KA consists of an invocation condition and a body. The interpreter matches system beliefs to KA invocation conditions. If unification occurs the KA can be instantiated and chosen for execution (i.e., become an intention), its variables bound to the current context and its body run. At any one moment, the intention structure can contain a number of intentions, some of which may be suspended, conditionally suspended or deferred. Subgoals of executing intentions are posted as new goals of the system; otherwise, primitive actions are directly executed.

Goal descriptions are not restricted to specifying desired behaviours in an external world but can also apply to the internal behaviour of the system. These descriptions are called metalevel goal specifications and have corresponding metalevel KAs – that is, information about the manipulation of the beliefs, desires and intentions of the PRS itself. For example, metalevel KAs could include various methods for choosing amongst multiple applicable KAs, modifying and manipulating intentions, and computing the amount of reasoning that can be undertaken given the real-time constraints of the problem domain (i.e., decision-theoretic control and goal selection knowledge can be encoded within KAs).

The interpreter is relatively inflexible yet can be overridden whenever the system can bring more powerful decision-making knowledge to bear. Such knowledge is encoded in metalevel KAs and can be invoked when needed; however, these processes are themselves interruptible; therefore, reactivity is maintained.

The PRS architecture has been implemented within a physical robot concerned with navigation and emergency tasks (Georgeff & Lansky, 1989) and has also been used for fault isolation and diagnosis in the Space Shuttle.

The features of PRS that Georgeff believes have contributed to its success are its partial planning strategy (not examined here), its reactivity, its use of procedural knowledge and its metalevel, or reflective, capabilities.

One problem with the PRS system as it stands is the metalevel KA invocation problem. The interpreter checks for invocation conditions once every cycle and problems can arise when invocation conditions for a particular KA are asserted during the cycle. The KA, therefore, may be invoked late or – if the conditions cease to hold before the next unification process – not at all.

An attractive feature of the PRS architecture is the simplicity of its interpreter and the addition of extra control knowledge in metalevel KAs.

4.3 Three forms of control

By abstracting from the design details of the agent architectures reviewed and including design considerations from the natural world it is possible to identify three major types of control in information processing control systems: reactive, deliberative and meta-deliberative (Sloman, 1996f). This taxonomy provides a high level common framework for the agents reviewed in this chapter.

The three different types of control represent different degrees of causal coupling between agent information states and environmental states. Reactive control, when compared to deliberative control, is highly coupled to the environment. For example, a negative feedback control loop will immediately respond to *actual* changes to an input signal, whereas a deliberative planning system may produce plans in response to *possible* changes to inputs. In addition, a planning system may operate at a level of generality that is invariant to most environmental contingencies. For example, a plan to go shopping may abstract from the problems of walking, avoiding obstacles, maintaining balance, and so forth: deliberative control primitives may be implemented as complex sequences of reactive controllers. Meta-deliberative control, which is control of internal information states, is highly decoupled from the environment.

Kiss (1992) discusses variable coupling between agent and environment states. His analysis is very similar to the one given here.

4.3.1 Reactive control

Reactive forms of control, unlike deliberative control, cannot explicitly construct representations of alternative possible actions, and evaluate and choose between them prior to performance. Reactive architectures are characterised by relatively inflexible links from perception to action that immediately activate on the satisfaction of predefined conditions. Normally there is continual vigilance for the satisfaction of those conditions (Lyons & Hendriks, 1992). Different behaviours are mediated by different dedicated coexisting circuits normally operating in parallel. Such dedicated circuits do not store much enduring state as they are required to react very quickly to new environmental contingencies. At this level of control 'it is better to use the world as its own model' (Brooks, 1991b). Control conflicts between reactive systems are resolved via predefined hierarchical relations, such as subsump-

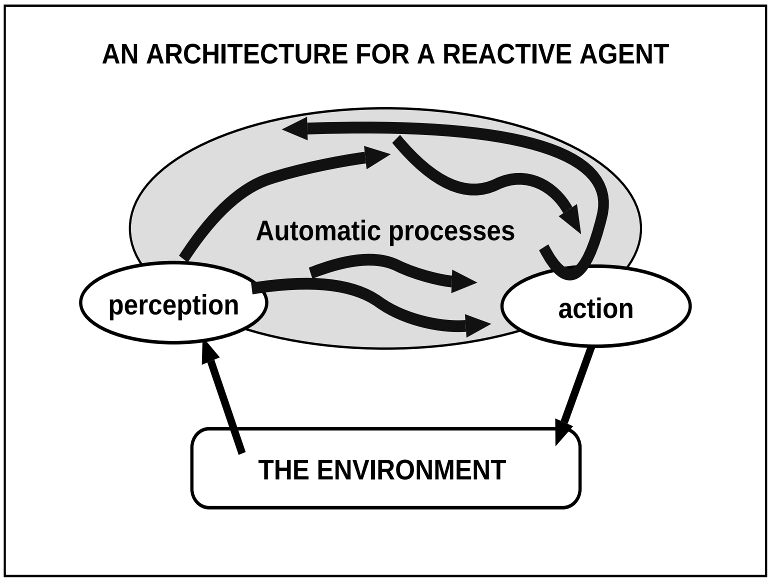


Figure 4.1: **An architecture for a reactive agent**

tion (Brooks, 1991b), summed weights, simple 'voting', gating, or inhibitory and excitatory links. Reactive control evolved to respond to aspects of niches that are invariant over long time scales, such as the requirements for locomotion, including aspects that require speedy responses, such as the requirements for avoidance of looming objects.

Examples of reactive control are Brooksian behaviour-based architectures, Maes' spreading activation networks (Maes, 1989; Maes, 1991), the routines of Agre and Chapman, insect flight, and automatic and reflex behaviours in humans, such as low level motor control processes. Figure 4.1 (taken from (Sloman, 1996c)) is a rough sketch of a reactive architecture with dedicated links from perception to action. Engineering control system theory is primarily concerned with reactive forms of control, such as open and closed loop systems describable by differential equations.

Sloman claims that there are some aspects of emotion that we share with many other animals that depend on reactive subsystems of the brain. Examples include being startled, terrified, thrilled, distressed by extreme pain, disgusted by horrible tasting food, sexually aroused, and so forth (Sloman, 1996e).

4.3.2 Deliberative control

Deliberative control, unlike reactive control, can explicitly construct representations of alternative possible actions, and evaluate and choose between them prior to performance, allowing much more flexible behaviour (Sloman, 1996f). Such rep-

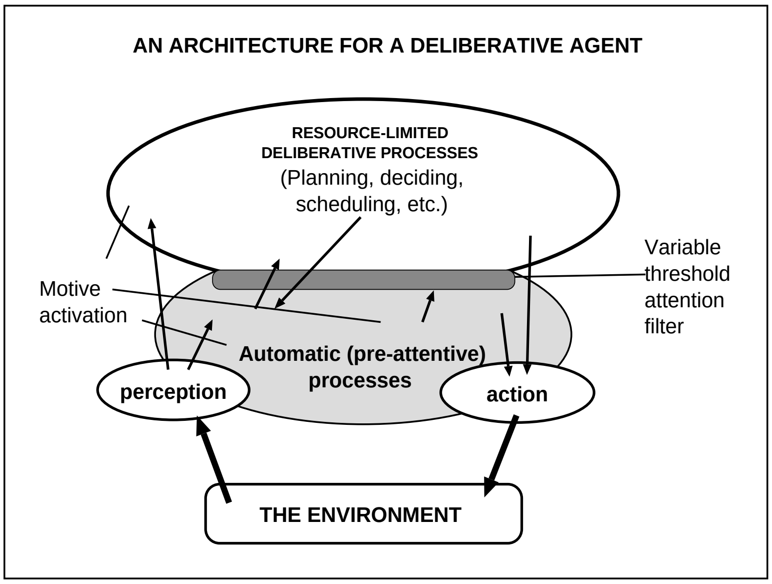


Figure 4.2: **An architecture for a deliberative agent**

resentations may have complex syntactic forms and refer to things not present to the senses. Deliberative control can respond to novel environmental contingencies; that is, construct responses to events that may not activate or match predefined conditions or may not have predefined responses. Deliberative control requires a more or less abstract model of the world in which proposed actions can be tried out in advance prior to actual performance, enabling the agent's hypotheses to die in its stead[1]. Trial and error within a world model requires the construction and storage of qualitatively different temporary information structures, such as proposed goals and plans, and possible future consequences of actions. Therefore, unlike reactive control that has dedicated circuitry for classes of events and corresponding reactions, deliberative control cannot preallocate hardware resources but instead must use a general purpose information store that may be written and overwritten. Therefore, if the agent is to meet a requirement for reactivity in addition to deliberation, problems of handling limited resources arise: there may be too many demands on the information store requiring mechanisms for selecting what processing tasks are more pressing or important.

Examples of deliberative control are the planning and problem-solving capabilities of humans, and the kinds of deductive reasoning exhibited by artificial intelligence 'classical' planning systems, such as STRIPS, or the problem solving

[1]This sentence paraphrases Karl Popper's assertion that trial and error learning within a world model 'permits our hypotheses to die in our stead' (quoted in (Mook, 1987)).

capabilities of the SOAR architecture. Examples of architectures that begin to combine deliberative control and reactive control are Firby's RAP system, Ferguson's TouringMachines, Pryor's PARETO, the Oz Project's HAP (Loyall & Bates, 1991; Reilly, 1993), Hayes-Roth's Intelligent Adaptive Systems (Hayes-Roth, 1990; Hayes-Roth, 1991a; Hayes-Roth, 1991b; Hayes-Roth, 1993b; Hayes-Roth, 1993a), Fehling's heuristic control virtual machine (Fehling, Altman & Wilber, 1989), Georgeff's PRS and Benson and Nilsson's TRP architecture.

Figure 4.2 (taken from (Sloman, 1996c)) is a rough sketch of a deliberative architecture that subsumes a lower level reactive architecture. An attention filter protects deliberative resources from demands that may not be very urgent or important. Engineering control system theory, which is normally concerned with quantitative measurements of environment variables, does not usefully apply to deliberative control systems that can create and manipulate symbolic representations of complex actual and possible states of affairs.

Emotional states, such as being anxious, apprehensive, relieved, and pleasantly surprised depend on deliberative capabilities in which plans can be created, inspected and executed.

4.3.3 Meta-deliberative control

Meta-deliberative control involves the monitoring and control of deliberative control; that is, unlike deliberative control that is essentially concerned with control of the environment mediated by a world model, the objects of meta-deliberative control are *internal* information processing states and processes. Meta-deliberative control may involve such generic tasks as ensuring that deliberation does not fail in too many tasks, ensuring that a particular goal does not interfere with other goals, avoiding wasting too much time on unsolvable problems, detecting processing conflicts between deliberative processes and arbitrating, detecting problematic emergent states, and so forth.

Examples of meta-deliberative control are the kinds of self-monitoring and self-evaluating capabilities of humans, for example realising that one is unjustifiably angry and attempting to stop being angry or ensuring that the anger is not acted upon. Sloman (1996f) states that:

> Although such a meta-management [meta-deliberative] system may have a lot in common with a deliberative sub-system, the point of making the distinction is that the deliberative mechanisms could exist without

[these] kinds of self-monitoring and self-assessing capabilities.

There are no convincing examples of artificial architectures exhibiting meta-deliberative control, though Beaudoin (1994) analyses several of the requirements for such an architecture. Figure 4.3 (taken from (Sloman, 1996c)) is a rough sketch of a deliberative architecture with a meta-deliberative controller (labelled 'meta-management'). The diagram is fully explained in the next section.

Emotional states such as feeling humiliated, infatuated, guilty, or full of excited anticipation, in which attempts to focus attention on urgent or important tasks can be difficult or impossible, require an architecture with meta-deliberative control.

4.4 Sloman and Beaudoin's design for human-like autonomy

This section describes the high level features of the Cognition and Affect project's (C&AP) design for human-like autonomy, a design that informs all our work. It is based on many facts about human capabilities, considerations regarding the evolution of intelligence, engineering design considerations inspired by reflection on the limitations of current AI systems (Sloman, 1996f), and a requirements analysis for an intelligent system able to handle multiple motives in a complex and continually changing environment. It exhibits all three forms of control identified in the previous section. Much of the specification is still provisional in that work on implementations may reveal serious problems. Also, the design is intended to be a model of mind: the high level functional decomposition of the design is a theory of the high level functional decomposition of the human mind. However, the theory is speculative in that empirical checking has not yet been attempted and may be very difficult.

The majority of the design work is due to Aaron Sloman and Luc Beaudoin. It has been described elsewhere, notably in (Beaudoin, 1994), summarised in (Beaudoin & Sloman, 1993; Sloman, Beaudoin & Wright, 1994; Wright, Sloman & Beaudoin, 1996), and first elaborated in (Sloman, 1978; Sloman, 1985). It is partly inspired by Georgeff's Procedural Reasoning System (Georgeff & Ingrand, 1989; Georgeff & Lansky, 1989; Rao & Georgeff, 1991b; Rao & Georgeff, 1991a), but allows a richer mental ontology, including asynchronous goal generation, more coexisting concurrent sub-mechanisms, a richer set of representations relating to motivators, an attention filtering mechanism, and reactive, deliberative and meta-deliberative forms of control. A full account of the design would be too long for this thesis, so the account concentrates on the processing of motivators.

4.4.1 Motive processing

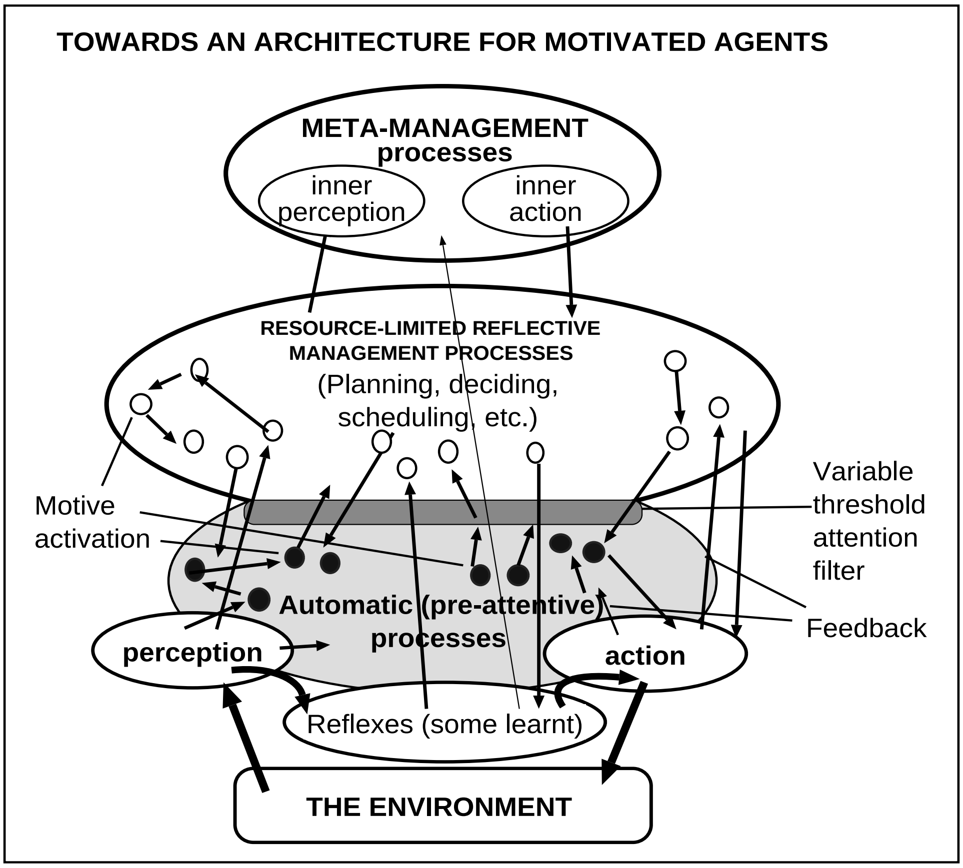


Figure 4.3: **Towards an architecture for a motivated agent**

The summary of the agent architecture is taken from (Wright, Sloman & Beaudoin, 1996).

The term 'motivator' is used to refer to a subclass of information structures with dispositional powers to determine action (both internal and external). This subsumes desires, goals, intentions and wishes. The precise definitions of these structures and their powers can be given only in terms of the architecture, which is roughly sketched in figure 4.3: an impressionistic diagram. Within this architecture, motivators can be generated or re-activated asynchronously as a result of internal or external events, and can generate processes of varying complexity, including evaluation, prioritisation, selection, planning, plan execution, plan suspension and more.

The large shaded area in 4.3 represents 'automatic' and reactive processes (associative memory, low level sensory analysis, low level motor control processes, innate and trained reflexes) all implemented in highly parallel dedicated (but trainable) 'hardware'. The management processes involve deliberative forms of control, and meta-management processes involve meta-deliberative forms of control.

The architecture has the following components (among others), which coexist and operate concurrently. Many of these components require a depth and complexity that cannot be described here.

Perceptual mechanisms. These are extremely complex systems, which detect potentially relevant sensory episodes and analyse and interpret them in terms of states and processes in the environment, creating or modifying internal representations. A perceptual control mechanism directs sensing operations. (Figure 4.3 oversimplifies drastically.)

Database of 'beliefs'. Information derived from perceptual representations, internal monitoring and reasoning processes is stored in a database or world model, which acts as a store of information. This will include both specific and general information, including a generic ontology for objects, processes, actions, and so forth. It need not look anything like current computer databases, for instance if implemented in a neural net. Planning may use temporary 'what-if' extensions to this store.

A changing collection of motivators. A motivator is a semantically rich information structure that tends to produce, modify or select between actions. It typically expresses a motivational attitude ('make false', 'keep true', etc.) towards a possible state of affairs ('short of food', 'warm', 'in danger', etc.), which may be expressed in propositional or non-propositional from. Motivators have various associated information items, including urgency, importance and an insistence value (Sloman, 1987)[2].

A new motivator's *insistence level* determines its ability to penetrate a (variable threshold) filter in order to be considered by management processes. *Importance* helps to determine whether it is adopted as something to be achieved, if it is considered. In order not to divert scarce resources, mechanisms assigning insistence values must work using simple 'heuristic' measures of importance and urgency. Computing accurate measures of urgency and importance could be too slow and computationally expensive, possibly diverting management processes which the in-

[2] There are many other motivator components not detailed here; see (Beaudoin, 1994; Sloman & Poli, 1995).

sistence mechanism is ‘designed’ to protect. However, insistence measures based on fallible heuristics can sometimes cause ‘bad’ decisions about what should and should not divert attention.

Pre-attentive and attentive motive generactivators (Beaudoin, 1994). These express agent ‘concerns’ (Frijda, 1986; Moffat & Frijda, 1995). They operate asynchronously in parallel, triggered by internal and external events. They generate and activate or reactivate motivators, and set or reset their insistence level. For example, the concern to maintain fluid levels may activate a drink seeking motivator. A simple type of generactivator could scour a world model for its ‘firing’ conditions, and when they are met a motivator is constructed and its insistence level set. (However, this is inefficient. Indexing methods can avoid the need for searching through a complete world model.) Other generactivators may be built into the physiological control system or into perceptual mechanisms. Some may be very abstract and general, for example reacting to states where another agent is in difficulty and creating a desire to help. Some are innate, many are learnt and culturally determined.

Variable threshold attention filter. An attention filter protects management processes by allowing only items with insistence values ‘higher’ than the current filter threshold to divert management processes. When first learning to drive, many find it difficult to hold a conversation simultaneously, as all high level processes are required for driving and the filter threshold is set high. However, attention can still be diverted by, say, the passenger screaming (loud noises can trigger high insistence levels). Later, when the driver is an expert, lower level mechanisms derived from earlier management processes control most of the driving. This reduces the management load, allowing a lower filter threshold and easier diversion of attention. Filtering is content sensitive, with different thresholds for different contents (Beaudoin, 1994): A baby’s cry can divert attention even when other loud noises do not. Motivators that fail to surface may remain available, so as to take advantage later of a lower filter threshold, or they may die unless continually reactivated by the relevant generactivators. Motivators survive until they are satisfied or decay.

Motive management. Once a motivator has ‘surfaced’ it can cause many diverse and complex processes, including: assessing (evaluating its importance, urgency, costs and benefits etc.), deciding (whether to adopt the motive, i.e. form an intention), scheduling (when and under which conditions to act), expansion (how to do it, i.e. planning), prediction (projecting the effects of hypothetical decisions), detecting motive conflicts, detecting opportunities, abandoning a motive and changing filter thresholds. Information about current motivators (including intention structures discussed below) is stored in a 'motivator database'. Only a limited amount of parallelism is available for management processes.

Plans and other 'databases'. The system will have many short term and long term memories containing information about current, future, or possible activities. In particular, certain goals require plans, and some re-usable plans will be stored as well as information about how to create new plans. Other information stores will include collections of particular skills, e.g. linguistic skills, mathematical skills, skills relating to social activities, games, one's job, and so forth. Some of this will be stored in an opaque form among the pre-attentive mechanisms. Some will be accessible to management processes. Some will be innate, others learnt, some fixed and others modifiable.

Meta-management. A meta-management process is any goal-directed process whose goal refers either to a management or to a meta-management process (Sloman, 1993a). Deciding whether to decide whether to adopt a goal, deciding which management process to run now, deciding whether the agent needs to change its current processing, determining whether all relevant issues have been considered, deciding whether management processes are taking too long to arrive at decisions, and detecting problematic processing states or episodes, such as detecting that too much motive-swapping is occurring, are all examples of meta-management processes, where motive management itself is the object of control. Meta-management requires some degree of ability to monitor and change the management processes. Conflicts can lead to 'loss of self control'. Again, resources are limited. Meta-management includes 'self-monitoring' functions.

Plan execution and effectors. Selected motivators are *intentions.* Executing them may occur with or without planning, with or without high level management, with or without monitoring. External actions are dispatched to an effector driver that controls agent actions within the environment. Internal actions may produce a succession of management states.

Global control mechanisms. An agent sometimes needs to modulate global features of its processing, for instance speeding everything up in times of extreme danger, slowing things down when losing energy too fast, generally being cautious when the environment is unfriendly, not wasting time on caution when the environment is generally friendly. Other more subtle global changes may be required when the social context changes, for instance when an individual of higher status is present. These global changes of state seem to be related to the colloquial concept of ‘mood’. In humans it seems that some of these global mood states are implemented in part at the chemical level (as shown by effects of drugs and hormones). Some control mechanisms may be ‘global’ only relative to a subset of a system: for example, global changes in the ease with which unsuccessful attempts are abandoned.

The architecture is ‘broad’ in that it combines many different capabilities. It is ‘shallow’ in that the components are themselves not specified in any great depth.

Chapter 5

MINDER1: an implementation of motive processing

> 'It is easy to build a philosophy – it doesn't have to run.'
> Charles F. Kettering
>
> 'The moment of truth is a running program.'
> Herbert Simon (1995).

This chapter describes a toolkit that allows rapid prototyping of agent designs, a simulated microworld domain that serves as a demanding testbed for the agent architecture, and the implementation itself, including the capabilities and behaviour of the agent, and its three processing layers (reactive, management and metamanagement). The final sections of the chapter explain the relevance of the implementation to the AFP theory of emotion.

5.1 Sloman's SIM_AGENT toolkit

For the C&AP's work exploring architectural design requirements for intelligent human-like agents, and other kinds, a facility is needed for rapidly implementing and testing out different agent architectures. The SIM_AGENT toolkit (Sloman & Poli, 1995; Sloman, 1995f; Sloman, 1995e), designed and implemented by Aaron Sloman, and freely available on the internet, is designed to make it relatively easy to implement a collection of interacting objects and agents, where each object (or agent) has internal complexity represented as sets of concurrent interacting condition-action rules that can invoke additional mechanisms of any kind, including ordinary procedures, neural nets, theorem provers, databases, genetic algorithms, and so forth.

The toolkit has two main components: Poprulebase (Sloman, 1995c) and the SIM_AGENT library (Sloman, 1995e; Sloman, 1995f). Poprulebase is a sophisticated and very general interpreter for condition-action rules, written in Pop-11. It is a forward-chaining production system interpreter, but provides a collection of unusual facilities, including a smooth interface to neural net or other 'sub-symbolic' mechanisms, mechanisms for control to be transferred between collections of rules as the context changes (allowing SOAR-like (Laird, Newell & Rosenbloom, 1987) pushing and popping of contexts), and facilities for altering the allocation of processing resources to different rulesets (Sloman, 1995d).

The SIM_AGENT library provides a set of base classes and scheduling mechanisms for running simulations involving a number of objects and agents whose internal mechanisms are implemented using Poprulebase. The scheduler simulates parallelism between agents and between subcomponents of agents. SIM_AGENT makes use of the object oriented facilities provided in the Pop-11 Objectclass package, a CLOS-like extension to Pop-11 providing multiple-inheritance, generic functions and so forth. Objects allow the production of re-usable, extendable software modules and shareable libraries. The agent and domain about to be described have been implemented using the toolkit[1].

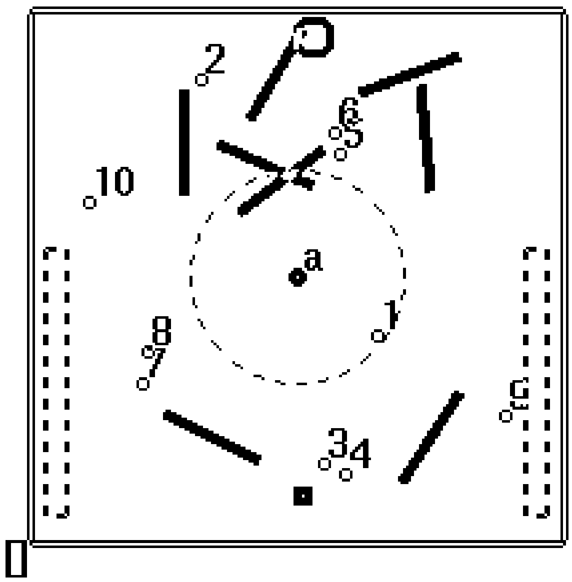
5.2 The minder domain

The nursery or minder domain is used in much of the C&AP's work because it imposes many tasks on the agent requiring the management of multiple motives and the control of attention. The domain is analogous to a nursemaid or minder in charge of a collection of babies (see (Beaudoin & Sloman, 1993))[2], but highly simplified in order to avoid the need to solve all the problems of AI, including 3-D vision, motor control, and naive physics. Simplification allows us to address motive processing while avoiding problems best left to others (Beaudoin & Sloman, 1993). The 'nursery' can take various forms. Here it is a two-dimensional room

[1] Other computational experiments with the toolkit are described in (Davis, Sloman & Poli, 1995; Davis, 1996). See http://www.cs.bham.ac.uk/~axs/cog_affect/sim_agent.html for mpeg movies of some of the experiments, including the minder domain.

[2] The idea for movable and rotatable bars is borrowed from Nilsson's botworld domain (Nilsson, 1994; Benson & Nilsson, 1995).

that contains a collection of simple reactive agents, called 'minibots' or dependents, which wander around getting into various difficulties, such as running out of 'charge' or falling into ditches. The reactive agents are dependents as they require the assistance of a minder agent whose task is to 'care' for them. For example, the minder agent can pick up dependents and carry them to a safe distance from a ditch. The wandering of the dependents ensures that new minder motivators are continually generated. Hence, the minder needs to arbitrate between many motives while maintaining reactivity to new, possibly urgent and important events. The domain could easily be extended in several directions. Figure 5.1 is a screenshot of a simple graphical representation of the nursery that is used for viewing the agent's external behaviour and debugging purposes.



Object	**Capabilities**	**Graphic**
Minder agent	See section 5.3	Dark circle labelled "a"
Dependent agents	Erratic roaming	Numbered light circles
Fences	Movable	Dark bars
Recharge point	Refuels dependents	Small square
Dismissal point	Removes dependents	Large rounded square
Ditches	Damages dependents	Dashed rectangles
Walls	Encloses 2-d surface	Bounding box

Figure 5.1: **The microworld domain**

5.3 The minder agent

For ease of reference this particular implementation of the design is named MINDER1. The name is not an acronym.

5.3.1 Behaviour and capabilities

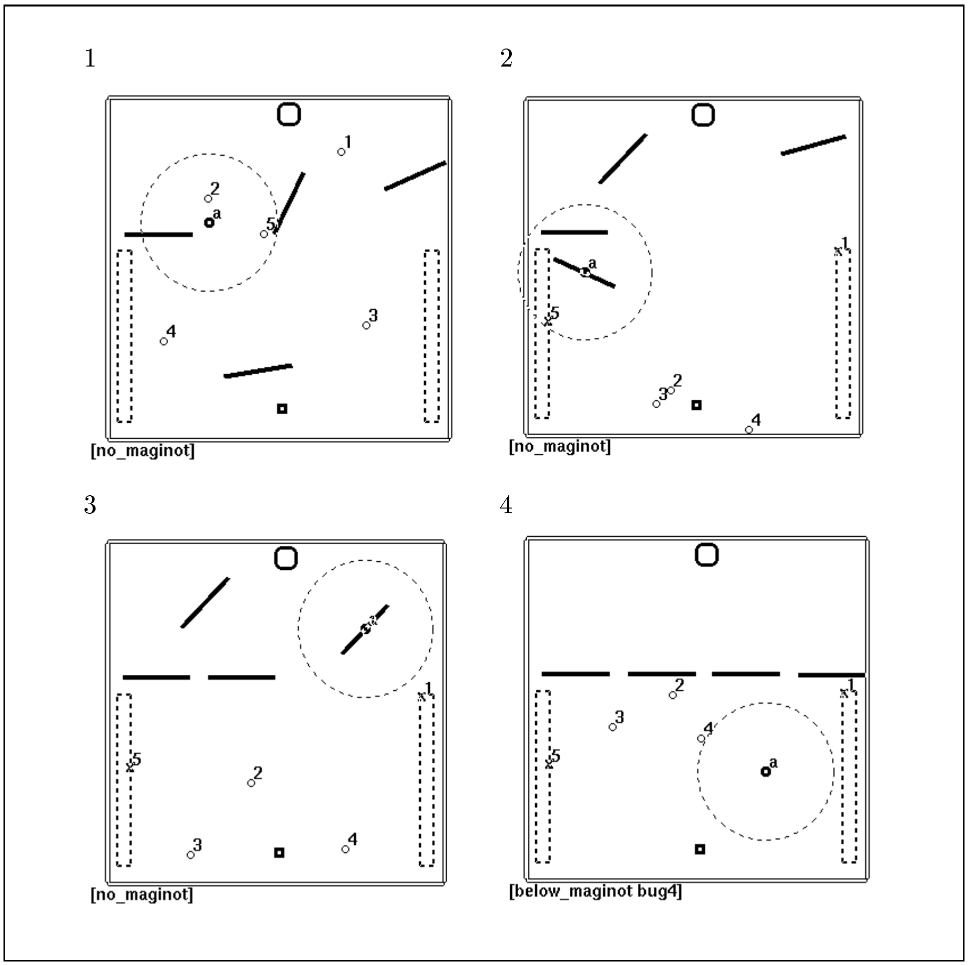


Figure 5.2: **Stages in the construction of an enclosure**. Note: text under window is the descriptor of the currently active motive.

Figure 5.2 shows MINDER1 constructing an enclosure used to keep minibots at the north end of the nursery to prevent them from wandering into ditches. MINDER1's behaviour can be observed in real-time but with occasional brief halts as pop-11 collects accumulated 'garbage' (old data that can be destroyed to free up memory). A separate window contains the textual trace output of the simulation. During a normal run the trace monitors the state transitions of all motives. However, a collection of debug flags can be altered so that MINDER1's current beliefs, sense data, motives, other knowledge stores, and internal processing may be examined at runtime.

MINDER1 'scurries' around the nursery pursuing its various motives. For example, it may sense that a dependent needs recharging and move towards it to pick it up. Also, MINDER1 often replaces its current motive with a new motive. For example, MINDER1 may be taking a dependent to the recharge point but 'notices' that another dependent is about to fall into a ditch. MINDER1 drops the minibot it is holding, saves the other minibot from falling into the ditch, and then returns to the recharge task. If the original minibot has moved MINDER1 may then search for it in the immediate vicinity. Minibots frequently fall into ditches and become damaged because MINDER1's perceptual field and speed of movement is limited. If there are no other pressing motives then MINDER1 may take the damaged minibot to the dismissal point. There are many other kinds of motive and corresponding behaviours that could be described (see appendix A.1 for a full list).

Informally, MINDER1 seems to prioritise its tasks in an reasonably intelligent manner: it successfully acts in the nursery domain to achieve its motives. However, as the number of minibots increases its performance deteriorates. MINDER1 can manage multiple motives, dynamically rescheduling them where necessary. Its resource limited management processes are protected from too high a processing load and interruption by a filter threshold that can vary according to the number and insistence level of motives. However, there are limitations, discussed in section 5.4.2. The architecture is evaluated in section 5.4.1.

The following three subsections describe how this functionality is achieved.

5.3.2 The reactive layer

Purely reactive processes cannot explicitly construct representations of alternative possible options and select between them: they have reflex-like or 'ballistic' causality (section 4.3.1). If conflicting actions are generated simple weight combinations or winner-take-all mechanisms determine the outcome. The reactive subsystems in MINDER1 are perception, belief maintenance, reactive plan execution, and preattentive motive generation. Each is described in turn.

5.3.2.1 Shallow perception

MINDER1's perceptual subsystem is implemented in a very shallow manner: in each time slice the required, externally visible features of domain objects are converted into a representation that is stored in an internal database, which, at any time, will have only partial and possibly incorrect information about the environment. The sampling range is limited to a fixed radius (see dashed circle in figure 5.1). Any object within range can be sensed but, unlike vision, the occlusion of objects is not supported. The perception rulesets are provided with sufficient processing cycles to perform all necessary domain sampling within each time slice. An example *new_sense_datum* is the following:

```
[new_sense_datum
    time 64 name minibot4 type minibot status alive distance 5.2 x 7.43782
    y 12.4632 id 4 charge 73 held false]
```

The data structure describes a dependent *minibot4* that was sensed at time 64, which is alive, has charge 73, is not held by another agent, and is at distance 5.2 units from MINDER1. The sensing of other objects is essentially similar, except the sensing of fences, which includes information about orientation and size.

5.3.2.2 Shallow belief maintenance

Beliefs, compared to sense data in this system, are more complex structures. They represent states-of-affairs that, due to the dynamism of the domain and the limits of perception, may or may not hold. There are two kinds of belief: sensory-based beliefs, which contain information about objects in the environment and are constructed from *new_sense_datum* items, and agent-based beliefs, which are generated by agent actions, such as beliefs about fences serving as components of an enclosure (see figure 5.2).

At any particular moment many of MINDER1's beliefs will be wrong. For example, it may believe that *minibot4* is in the absolute coordinate location (80,90). However, location (80,90) may be currently out of sensor range and *minibot4* may have moved. A design problem arises when MINDER1 returns to a location within sensor range of location (80,90). If *minibot4* is sensed the corresponding belief is updated; however, if *minibot4* is absent then no *new_sense_datum* will update the belief: it is not possible to sense an absence. Without rudimentary belief maintenance MINDER1 will continue to hold a false belief despite having sufficient information to infer its falsity. The problem is solved by storing defeaters with each belief. Defeaters are conditions that must remain false in order for the belief to remain true. If the defeating conditions for a particular belief evaluate to true then the belief is removed. An example *belief* is the following (note that double equals matches any sequence of items in a list):

```
[belief time 20 name minibot8 type minibot status alive distance 17.2196
    x 82.2426 y 61.2426 id 8 charge 88 held false
    [defeater
        [[belief == name minibot8 == x ?Xb y ?Yb ==]
        [WHERE distance(myself.location, Xb,Yb) < sensor_range]
        [NOT new_sense_datum == name minibot8 ==]]]]
```

The defeater is composed of poprulebase conditions, and says, 'IF I have a belief about minibot8 AND I have no new sense data about minibot 8 AND I am in a position that, according to my belief, I should have new sense data about minibot 8 THEN my belief is false'. The defeater mechanism allows arbitrary size hierarchies of defeats to be formed. For example, a 'second order' belief may be justified by a 'first order' belief. If the defeater of the first order belief evaluates to true then the second order belief is also removed. In this way beliefs are automatically maintained, although it should be noted that belief maintenance is a difficult problem to solve in general.

5.3.2.3 Reactive plan execution

MINDER1 needs to be able to act in the domain. As the nursery is changing continually it is fruitless for MINDER1 to attempt to construct and execute precise but inflexible plans that depend on beliefs that can rapidly become obsolete. Instead, MINDER1 needs a level of plan execution that is robust and therefore able to recover from unexpected failures, and reactive, that is, can immediately react to new contingencies in the course of plan execution without the need to engage higher level, resource limited systems. For example, MINDER1 may have a plan to move to location (50,50). The execution of this plan will involve many steps, including moving forward, rotating, sensing the route ahead, planning routes around obstacles and so forth. It is possible that obstacles, such as fences, can be moved by other agents; therefore, a purely classical planning approach, in which a complete plan is constructed prior to action and then blindly followed in detail, is likely to fail. Plans that can alter themselves to achieve their goals via continuous feedback from the current situation avoid this difficulty. An approach that partially meets this requirement is Nilsson's *teleo-reactive (TR) program* formalism (Nilsson, 1994).

A TR program is an ordered set of production rules that directs the agent toward a goal in a manner that takes into account changing environmental circumstances (Nilsson, 1994):

$$K_1 \longrightarrow a_1$$

$$K_2 \longrightarrow a_2$$

$$\ldots$$

$$K_n \longrightarrow a_n$$

K_i are conditions on agent knowledge (including information items such as *beliefs*, *new_sense_datum*, representations of current motives etc.) and a_i are actions on the world or on beliefs. On *every* cycle the conditions of all active TR programs are evaluated from top to bottom. The a_i of the first K_i that evaluates to true is executed. If, on the next cycle, the same K_i evaluates to true then the same action is executed, and so on until the conditions change. A TR program must satisfy a regression property, which, informally, states that an action, a_i, will eventually achieve a condition, K_j, which is higher in the list ($j < i$). TR programs can call other TR programs or themselves, that is they can be hierarchic and recursive. Figure 5.3 provides an example TR program implemented in MINDER1.

K_1: ~(held(obj)) $\wedge$ charged(obj) $\longrightarrow$ a_1: null

K_2: held(obj) $\wedge$ charged(obj) $\longrightarrow$ a_2: DROP(obj)]

K_3: held(obj) $\wedge$ close_enough(obj, recharge_point) $\longrightarrow$ a_3: CHARGE(obj)

K_4: T $\longrightarrow$ a_4: TAKE_OBJECT(obj, recharge_point)

Predicate	**Semantics**
held(obj)	T if agent holds object obj
charged(obj)	T if agent believes obj has sufficient charge
close_enough(obj1, obj2)	T if agent believes obj1 and obj2 are adjacent
Imperative	**Semantics**
DROP(obj)	Agent attempts to drop obj, makes hold(obj) false
CHARGE(obj)	Agent charges obj at recharge_point, makes charged(obj) true
TAKE_OBJECT(obj1, obj2)	Agent moves obj1 to obj2 by invoking another TR program

Figure 5.3: **TR program charge_object**

The TR program is called by a higher level executor (see section 5.3.3) that unifies a parameter value with *obj*, such as the value *minibot4*. In this example, the TAKE_OBJECT action is a call to further, more complex, TR programs that can search for objects, home in on their location, pick them up, move while avoiding obstacles, and so forth. Even this simple program can deal with unexpected failures: for example, if, for whatever reason, the obj is no longer held the TR program will reactively 'drop down' to condition K_4 and attempt to relocate and hold the obj.

MINDER1 currently has thirteen TR programs (see appendix, section A.4) that serve as reactive behavioural building blocks. Each TR program is implemented as a set of SIM_AGENT production rules satisfying the regression property. As TR programs are fully evaluated each time step it is helpful to think of their semantics in terms of dedicated circuits that continuously evaluate feedback from actions (Nilsson, 1994) (and section 4.2.3.4).

5.3.2.4 Generactivation of motivators

For MINDER1 to use its TR programs it requires goals to achieve. The source of motives in MINDER1 is a suite of generactivators that express the agent's concerns (Frijda, 1986). For example, a particular generactivator *G_low_charge* searches the internal database for beliefs about dependents with very low charge; if such a belief is found then the generactivator constructs a declarative representation of a motive and places it in a motive database:

```
[MOTIVE motive [recharge minibot4] insistence 0.322 status sub]
```

The motive contains a motivational attitude towards a state of affairs in the domain ('make it true that minibot4 is recharged')[3], an insistence value, which, in MINDER1, is a heuristic, quantitative representation of the urgency and importance of the motive, and a current status, *sub*, which is a flag that states that the motive has not surfaced through the variable threshold attention filter (see appendix, section A.1 for a full list of possible motives). Currently, the insistence heuristics have been built by hand. This is a simplification: the need to design mechanisms that can construct insistence heuristics from domain interaction has been avoided. The heuristics are functions that map conditions on agent knowledge to the reals in the interval [0,1]. For example, the generactivator that constructs motives when a dependent is too near a ditch uses the simple function *f(distance_from_ditch)* $\longrightarrow$ *insistence* to calculate insistence. The higher the number the higher the insistence and, consequently, the greater the motivator's dispositional powers to surface and grab management resources.

MINDER1 has eight generactivators expressing various concerns. A selection, described informally, are 'dependents must be fully charged', 'damaged dependents need to be removed from the nursery', 'the ditches need to be patrolled', 'an enclosure needs to be built to protect the dependents', 'ensure dependents are at safe

[3]Note, however, that the motivational attitude is not explicitly represented. The motivational attitude is implicit in the procedures that use the declarative representations of motives. In this case MINDER1 does not follow the specification for motives provided in (Beaudoin & Sloman, 1993; Beaudoin, 1994).

distances from ditches' and so forth. In addition to constructing motives, generactivators also remove motives when the original rationale for activation no longer holds, and dynamically alter insistence values, thereby reactivating motives to be candidates for surfacing.

The next section describes the resource limited management layer that takes explicit representations of multiple motives and translates them into intentions for action.

5.3.3 The management layer

The processes in the management layer are resource limited but can operate on explicit representations of alternative motives and select between them (see section 4.3.2). Management processes pose difficult design problems, and this is reflected in the shallowness of the management implementation. Discussion of the many limitations of the implementation is postponed to section 5.4.2 where the urgent need for a comprehensive theory of motive management is identified. The main management subsystems are the motive filter mechanism, shallow motive management, and shallow plan execution. Each is described in turn. Figure 5.4 is a schematic representation of the architecture of and processes within MINDER1. The diagram is fully explained in this section.

5.3.3.1 Shallow motive filter mechanism

MINDER1's filter threshold is a real number in the interval [0,1]. Beaudoin (1994) considers more complex filter mechanisms and Norman (1996) considers a similar, but more developed, filter-based selective attention mechanism implemented in a factory domain. Newly generated motives are initially of state *sub*. Call the set of motives of state *sub*, M_{sub}. All members of M_{sub} are candidates for surfacing: if their insistence values are equal to or higher than the current filter threshold they surface and their state changes to *surfacing*; otherwise, their state remains the same. Figure 5.5 lists the variety of motive states represented in MINDER1 and conditions for the surfacing and 'diving' (a surfaced motive returning to state *sub*) of motives. Motives that remain members of the set M_{sub} for more than one cycle are not sufficiently insistent to gain management resources. However, this can change by generactivators recomputing insistence levels or by the lowering of the filter threshold (see section 5.3.4 on metamanagement). The filter acts to limit the number of motives that management processes need to consider, that is it functions

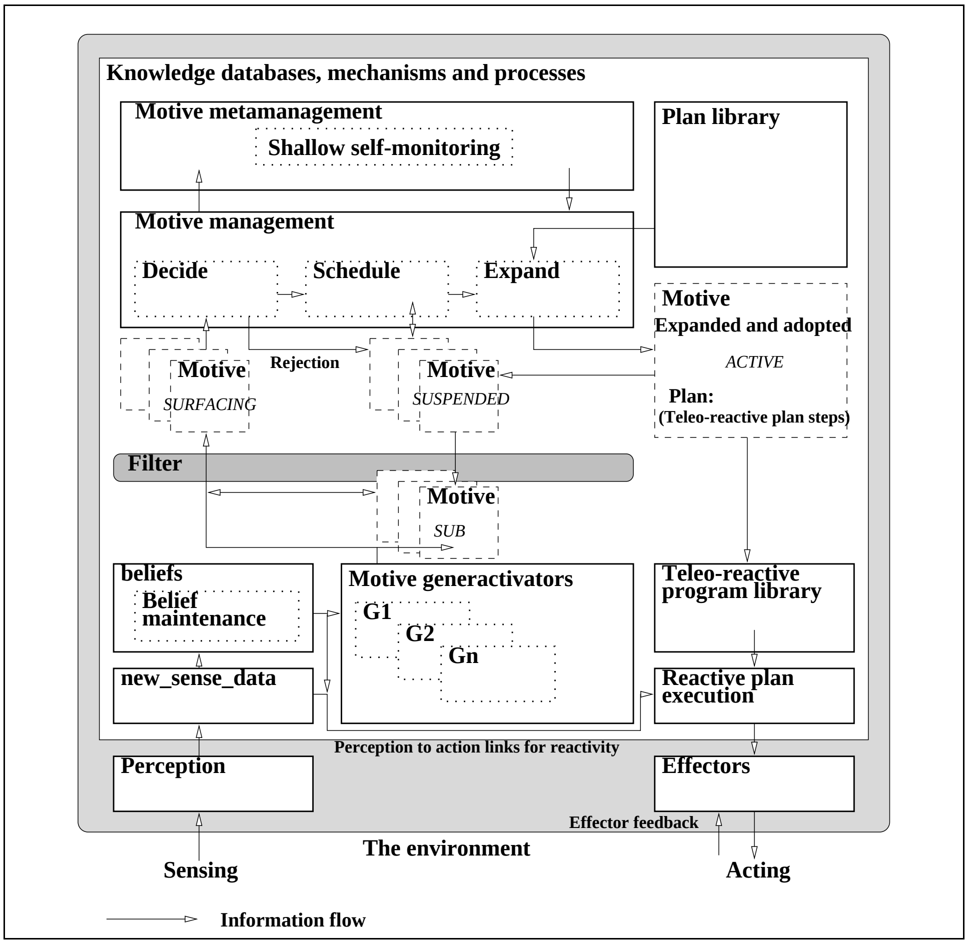


Figure 5.4: **The route from perception to action in MINDER1** Solid lines represent mechanisms and databases, dotted lines represent processes that may occur within a mechanism, and dashed lines represent declarative data structures, such as motives.

as an 'attention' filter.

5.3.3.2 Shallow motive management

Before describing motive management processes a short note explaining figure 5.5 is required. The set M represents all MINDER1's current existing motives. There are various subsets of M, including the sets unsurfaced, M_{sub}, and surfaced, $M_{surfaced}$. The subsets of $M_{surfaced}$ represent motive state transitions that are possible within management. These state transitions are now explained.

(a) Deciding. All $m_i \in M_{surfacing}$ are *decided* by management processes. Deciding involves determining whether the motive should remain surfaced and continue to have management resources devoted to it. A full implementation of deciding

Motive sets	Explanation of state
M	Current agent motives
M_{sub}	Unsurfaced motives
$M_{surfacing}$	Surfacing motives
$M_{surfaced}$	Surfaced motives
$M_{suspended}$	Surfaced but suspended motives
$M_{suspended,meta}$	Surfaced and suspended during meta-planning
$M_{suspended,execute}$	Surfaced and suspended during execution
M_{active}	Surfaced and adopted motives
$M_{active,meta}$	Surfaced and adopted for meta-planning
$M_{active,execute}$	Surfaced and adopted for execution

Where,

$$M_{sub} \cup M_{surfaced} = M$$
$$M_{surfacing} \cup M_{suspended} \cup M_{active} = M_{surfaced}$$
$$M_{suspended,meta} \cup M_{suspended,execute} = M_{suspended}$$
$$M_{active,meta} \cup M_{active,execute} = M_{active}$$
$$length(M_{active}) = 1$$

And,

if insistence($m_i \in M_{sub}$) $\geq$ filter_threshold then $m_i \in M_{surfacing}$;

if insistence($m_i \in M_{suspended}$) $<$ filter_threshold then $m_i \in M_{sub}$.

Figure 5.5: **Motive states and the transition between attentive and preattentive processing**

would need to include processes of motive assessment, such as developing sophisticated qualitative measurements of urgency and importance, determining the risks and benefits of adopting the motivator, and comparing these measurements with similar measurements of other motives. Often a motivator cannot be assessed until it has been partially expanded, or cannot be decided until it has been assessed, or cannot be assessed until partially executed, and so forth (Beaudoin, 1994). In other words, motive management systems that have purely predefined and inflexible motive state transitions are unlikely to meet the full requirements (compare (Sloman, 1978)).

MINDER1 has an extremely simple and shallow deciding process yet exhibits complicated interactions between deciding, scheduling, expanding and motivator states. The relations between these management processes are complex and have been distinguished for the sake of exposition. For example, if a motive has already been scheduled, and is therefore active, it may be partially expanded in preparation for deciding. A motive such as:

```
[MOTIVE motive [save ditch1 minibot5] insistence 0.646361 status active]
```

is partially expanded to:

```
[MOTIVE motive [save ditch1 minibot5] insistence 0.646361 status active
    plan [[decide] [get_plan]]
    trp [stop]
    importance undef]
```

The partially expanded motivator, $m_i \in M_{active,meta}$, contains an initial metaplan with plan steps *decide* and *get_plan*. These plan steps are not external actions but calls to management processes. A metaplan is executed by the management system, whereas a normal plan is executed by the plan executor (see later). If the motive remains scheduled (i.e., another motivator has not displaced it as the active motive) the plan step *decide* is executed, invoking a decide routine stored in the plan library (see node 3 of figure 5.6, which is a graphical representation of management processes that occur on surfaced motives). Currently, MINDER1 has a single, shallow decide routine for all motivators, of whatever type. The decide routine determines the importance of the motivator, which for most motivators is the designator 'normal', meaning that the motivator's heuristic insistence level is held to be a good approximation of a developed measure of the importance of the motivator. However, no developed measure of importance ever occurs, and developed measures of urgency are not supported, which is a significant limitation of the implementation when compared to the complete design[4]. An important exception (node 4 in figure 5.6) to this is discussed in section 5.5. If the motivator still remains scheduled on subsequent cycles the next plan step, *get_plan*, is invoked. It has the effect of retrieving a stored plan from the plan library (see node 6). However, due to the parallelism of the MINDER1 architecture, new motives may surface at any time, be scheduled for immediate processing and replace the currently active motive. The replaced motive can be suspended either during a metaplan phase, or during an execution phase, that is, be either $m_i \in M_{suspended,meta}$ or $m_i \in M_{suspended,execute}$ respectively. In both cases, the motive remains partially expanded to allow readoption at a later time.

(b) Scheduling. Scheduling involves determining when a motive should become active, be executed and control current internal or external actions. (Contrast

[4]However, for many situations, heuristic urgency as represented by insistence is sufficient to order motives, such as choosing which of two minibots to rescue from falling into a ditch.

meta-scheduling, a metamanagement function, which schedules scheduling, that is, determines *when* to consider a motive). A full implementation of scheduling processes would include developed measures of urgency that could answer questions such as: when will it be too late to satisfy the motivator? need it be satisfied at a particular time? can it be postponed? is it too early to do this? and so forth. MINDER1 bases its scheduling decisions on the insistence and importance of motivators. Therefore, urgency measures are only implicitly and heuristically represented. Nevertheless, the scheduling mechanism dynamically orders the set $\mathrm{M}_{surfaced}$ such that a single motive, $m_i \in M_{active}$, is chosen to be activated for metaplanning, $m_i \in M_{active,meta}$, or execution, $m_i \in M_{active,execute}$, depending on its current expansion status. Beaudoin (1994) considers limited management parallelism that allows the adoption of more than one active motive, but MINDER1 does not support this desirable feature. Scheduling operates every management cycle; however, the filter mechanism ensures that the number of motives that needs to be considered is always low.

(c) Expanding. As stated, the metaplan step *get_plan* retrieves a stored plan from the plan library suited to the particular motive. Currently, MINDER1 has seven plans in its plan library (see appendix, section A.3). In general, information contained in the motive is unified with plan variables. An example expanded motive to build an enclosure of fences is the following:

```
[MOTIVE motive [no_maginot] insistence 0.05 status suspended
   plan
       [[make_wall 40 60 0 second]
       [make_wall 65 60 0 third]
       [make_wall 90 60 0 fourth]]
   trp [stop]
   importance normal]
```

In this example the partially executed plan consists of three plan steps of the same type, *make_wall*. The plan step *make_wall* is itself a TR program that can be executed by the reactive plan executor. A more complete implementation of MINDER1 would include a planning mechanism that could construct new plans for new situations based on the agent's available action primitives. Planning capabilities would require storing STRIPS-style add and delete lists with both plans and primitives to allow reasoning about chains of behaviours. Benson & Nilsson (1995) (and section 4.2.3.4) describes an agent architecture that can learn pre and postconditions for TR programs from observations of the effects of its own behaviour and then dynamically construct novel reactive plans. Such flexibility would be a desir-

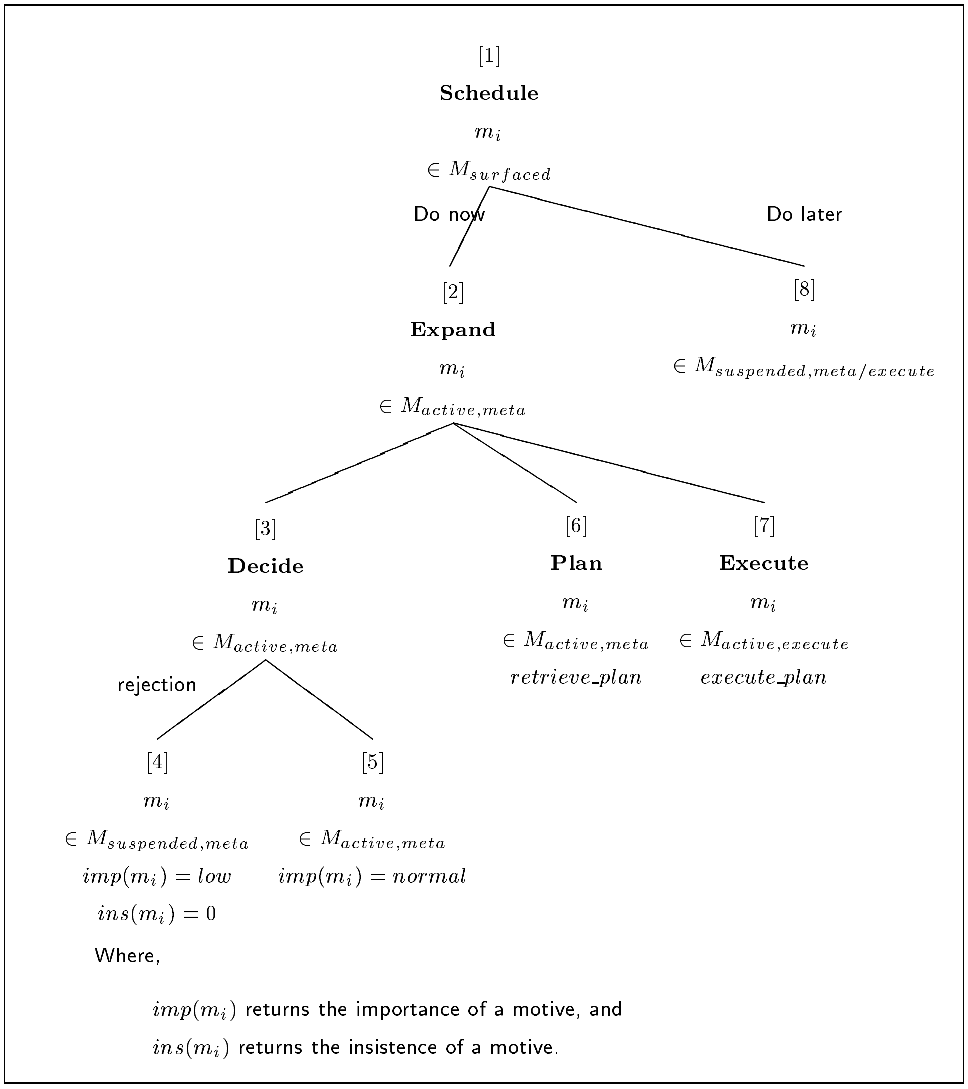


Figure 5.6: **Management processes on surfaced motives in MINDER1**

able extension to MINDER1's management processing. There are many planning algorithms in the AI literature, and incorporating a planner would not be difficult. However, this was not a major goal of the project.

5.3.3.3 Shallow plan execution

The plan steps of the currently adopted motive are executed as TR programs (see right-hand-side of figure 5.4). For example, the *make_wall* plan involves calls to further TR programs, such as *grab_wall* and *place_bar*. These TR programs themselves call other TR programs; some examples are *search*, which makes the agent search the nursery for a specified object if the agent has no beliefs concerning the object's location, *grab_object*, which makes the agent approach an object and pick it up, and *amble*, which moves the agent to a specified location while avoiding obstacles (see appendix, section A.4). The leaves of TR program trees are atomic actions, such as *MOVE*, *ROTATE*, *SETSPEED*, *GRAB*, *ROTATE_BAR*, and so forth. Note however that the TR program tree is constructed as a complete circuit on every cycle. In other words, the links in the tree can dynamically change to allow unexpected contingencies to be catered for, such as an object being moved by another agent. Currently, MINDER1 has nine action primitives (see appendix, section A.2). Building real robots requires much work to develop robust action primitives, for example Marjanovic, Scassellati & Williamson (1996) discuss designing primitives for robot arm pointing. Simulation work allows us to abstract from these engineering problems and concentrate on motive management.

This completes the description of the route from perception to action in MINDER1.

5.3.4 The metamanagement layer

Metamanagement involves control of management functions (see section 4.3.3). MINDER1 has two metamanagement functions: changing the filter threshold level and detecting perturbant states. The discussion of perturbance detection is postponed until section 5.5. Metamanagement implementation is extremely shallow when compared to the C&AP's design. A full implementation would include sophisticated 'self-monitoring' processes that detect, evaluate and control management processes (Wright, Sloman & Beaudoin, 1996).

Management processing is resource limited. To reflect this a maximum limit of three surfaced motives together with a maximum cycle limit for management rules was chosen arbitrarily. (A cycle limit defines how much processing resources are devoted to a ruleset in each time slice.) A metamanagement process monitors the number of surfaced motives. If the number of surfaced motives is more than three then the filter threshold level is incremented. The threshold level continues to increase every cycle until there are three or less surfaced motives. For example, if a $m_i \in M_{suspended}$ has an insistence level lower than the threshold it will 'dive' and return to status *sub*. Similarly, if there are fewer than three surfaced motives then the filter threshold is decremented. The process continues until the threshold reaches zero or a sufficient number of *sub* motives surface into management processing.

The joint operation of the dynamic filter and generactivators recomputing insistence values ensures there can be a continual movement of motives from preattentive to attentive processing and back again. For example, MINDER1 may have ten motives in total, four of which have surfaced, and one active. In this case the number of surfaced motives exceeds the maximum and the filter threshold is raised on each time step. The filter rises until the least insistent surfaced motive 'dives' and the processing load returns to a manageable level (see figure 5.7).

The original agent design (Beaudoin, 1994) did not specify a state transition of motive diving. Instead, suspended motives not attended to for some time decayed and were eventually removed. However, this is not entirely satisfactory as suspended motives, whether they be postponed for execution or deciding, will impose extra computational load on management processes. For example, to meet a requirement of mutual compatibility between motives there will need to be processes that consider the relations between all motives in the set $M_{active} \cup M_{suspended,execute}$. For example, motives that have been decided for execution, but are currently suspended, might be incompatible with newly surfaced, active motives (e.g., a person may intend to resume job hunting later in the day, but receives news that a friend is in hospital). Management processes would need to detect and resolve such incompatibilities. Detecting incompatibilities requires considering both M_{active} and $M_{suspended,execute}$. Therefore, if a subset of suspended motives can be removed from management the amount of computation required can be reduced. Allowing motives to dive, in addition to surface, based on heuristic measures of insistence, is a way to achieve this. However, the disadvantage is that the agent may fail to detect a serious conflict between a new action and a non-urgent but very important suspended motive. Therefore, allowing motives to dive could be disastrous. Hence, the implemented solution is not entirely satisfactory. A better solution would allow surfaced motives to be always accessible but use indexing mechanisms to overcome computational expense. But indexing may not be perfect, and relevance could still be missed.

There are many reasons why the filter level changes. For example, an active motive may be removed due to successful completion of its plan, or removed due to the loss of its rationale. The removal of a surfaced motive may stop the filter threshold being raised further and begin lowering it. The interactions between all these kinds of processes can be complicated. A simple illustration is provided below. In this short trace a new motive surfaces that causes metamanagement processes to detect the presence of too many surfaced motives, resulting in the filter being raised, an existing, surfaced motive to dive, and the activation and adoption of the new motive.

```
====================== end of cycle 82 =================
```

```
======================= end of cycle 83 ===================
 ** [[Surfacing --
               [MOTIVE motive
                       [recharge minibot5]
                       insistence 0.21 status sub]]]
 ** [[RAISING filter threshold to 0.02]]
======================= end of cycle 84 ===================
 ** [[Diving --
            [MOTIVE motive
                    [default]
                    insistence 0.02 status suspended plan
                    [[decide] [get_plan]]
                    trp
                    [stop]
                    importance undef]]]
======================= end of cycle 85 ===================
 ** [[Activated --
               [MOTIVE motive
                       [recharge minibot5]
                       insistence 0.215 status active]]]
======================= end of cycle 86 ===================
```

A filter mechanism of this sort is connected to the folk-psychological concept of focus: a very high filter level would correspond to a high level of focus, during which attentive resources are concentrated on a single, very important and urgent motive while being protected from unnecessary interruption. The situation of fleeing a battleground might engender this state, in addition to causing physiological changes to increase action readiness. A low filter level, however, would correspond to a low level of focus, during which attentive resources are can be readily interrupted, easily shifting from one concern to another, a situation that might occur while talking among friends.

MINDER1's filter level is incrementally altered. However, there are no *a priori* reasons not to use different mechanisms. One possibility is to store the current lowest insistence level of surfaced motives. The filter level would then be made slightly higher than this value forcing the least insistent motive to dive. The problem of unmotivated design decisions is discussed in the next section.

5.4 Evaluation of the architecture

This section briefly describes how the architecture is to be evaluated, and the limitations of the implementation, including how it should and could be extended.

5.4.1 Evaluation

The main aim of building MINDER1 was to show (i) that the C&AP's high level design could be implemented, albeit in a simplified fashion, (ii) that the design could, in principle, meet the requirements, and (iii) that this kind of motive processing architecture would lead to perturbant states. (i) has been shown, but (ii) is more problematic, and (iii) is discussed in section 5.5.

A general problem of software engineering is to show or prove that a design, and its corresponding implementation, meets or satisfies a set of requirements. Design validation is difficult. MINDER1 appears to cope in its domain but a full evaluation of the architecture would require tests in a variety of domains, including more complex variants of the nursery domain, and the collection of performance statistics that could be compared with the performance of other possible designs. However, the aim was not to explore design-space searching for the 'best' motive management system (in any case, the best design would vary over niche-space). It is sufficient for current purposes that the prototype implementation demonstrates that it is possible that the original design meets its requirements.

A related point is that many of the detailed design decisions taken during implementation were arbitrary. In (Sloman, Shing, Read & Beaudoin, 1992) six types of design decision were identified: (i) design decisions linked to initial requirements, (ii) decisions linked to empirical data, (iii) decisions linked indirectly to requirements via higher level design decisions, (iv) decisions made in order to test a theory, (v) arbitrary decisions where previous requirements and design decisions do not prescribe a unique decision, and (vi) decisions due to hardware or software limitations[5]. Some of the high level design decisions were motivated. For example, three layers of processing is suggested by empirical evidence, in particular evolutionary neuroscience. Empirical observations of many kinds, such as the difficulty of attending to two conversations at once, justifies the design decision of management resource limits. In addition, there are theoretical reasons for such limits, such as limited physical

[5]Existing work in software engineering attempts to formalise the relationships between requirements and design decisions, for example (MacLean, Young, Bellotti & Morgan, 1991). Work of this kind can help us think about niche-space and design-space and the mappings between them.

resources imposing a bottleneck on cognition, limited memory resources imposing a limit on the creation and storage of temporary structures, and the need for mutual compatibility of adopted motives may limit the number that can be considered and adopted at any time. Insistence heuristics can be similarly justified, in particular from the requirement for reactivity in dangerous domains; for example, there is strong evidence of 'quick and dirty, emotional' processing pathways in the brain (LeDoux, 1994)). Also, existing psychologial theories of motivation describe mechanisms that assess the importance of goals before adoption (Heckhausen & Kuhl, 1985) and self-regulatory processes that control motive adoption (Kuhl, 1992; Kuhl & Kraska, 1989) (this is discussed more thoroughly in (Beaudoin, 1994)). However, deciding on the precise form of representation of beliefs and other intentional structures was largely a matter of convenience. Accordingly, the implementation should be viewed as an exemplary illustration of the C&AP's design theory, but the details of the implementation are of secondary importance.

However, implementation remains an essential part of the design-based approach: not implementing MINDER1 would be like an engineer producing designs for bridges without ever building one to see if it stays up. It was always a possibility that the implementation would *fail* to manage multiple motives in the nursery domain. If so, reasons why the implementation was inadequate would have been discovered, which may have motivated a revision of the theory. A full analysis, however, would include a study of the surrounding design-space. It must be admitted that it is possible that a fundamentally different design might also meet the requirements, for example if computers of the future have speeds many orders of magnitude faster than now.

MINDER1 is intended to have implications for an understanding of human minds because human minds have evolved to satisfy similar requirements: humans need to manage multiple motives with resource limited attentive processes. The gross mechanisms of the design – reactive motive generation, motive filtering, and motive management and metamanagement – are held to exist in human brains. Note however that there need be no invariant neuronal correlates of these mechanisms. The invention of the computer has demonstrated that the mapping between information processing mechanisms and physical implementation is not straightforward.

MINDER1 is an engineering solution to a control problem, and could be used as a command and control system or put to use in computer games, but this was not the main reason for building it; rather, it is intended as a 'toy designed to stretch our minds' (Sloman, 1978), in particular to help us think about the full complexity of emotional states. The relevance of MINDER1 to theories of emotion constitutes its primary scientific content and is discussed in section 5.6. MINDER1, like many computer simulations of mental phenomena, is both an artificial agent in its own right and a model of motive processing in natural agents (Morris, 1991).

5.4.2 Some limitations

MINDER1 could be improved in many ways, both from the standpoint of developing agents that handle multiple motives in more intelligent ways, and from the standpoint of developing a richer cognitive model of motive processing.

All the mechanisms described could be 'deepened'. The management layer should include a planning module, and much more sophisticated scheduling mechanisms, including developing sophisticated measures of the urgency and importance of motives. The plan executor should be extended to notice opportunities or threats to current plans ((Pryor & Collins, 1992; Pryor & Collins, 1993; Pryor, 1994) and section 4.2.3.3), or possibilities for satisfying more than one motive with a single plan ('killing two birds with one stone'). Also, a more intelligent agent should construct its own insistence heuristics. Currently, they are hand-coded. Humphreys (1996) describes a reinforcement learning mechanism that learns the relative priorities of various goals in different contexts. The metamanagement layer should include mechanisms for 'self control', which are ways for motive management to be controlled, in particular to handle problematic emergent processing states (see next section). The filter mechanism could be extended to include a facility for 'exception handlers' (discussed in (Beaudoin, 1994)), which would allow management processes to selectively prevent classes of motives surfacing regardless of their insistence.

Finally, one of the difficulties of developing a motive processing architecture is the lack of a comprehensive theory of motive management. AI researchers have developed sophisticated theories of planning, including the extension of such work to cope with complex, uncertain and dynamic domains (e.g., (Nilsson, 1994; Firby, 1989; Pryor, 1994)). However, the problem of managing multiple motives, which includes synthesising such tasks as deciding whether to process a motive, at what time to expand and execute the motive, how to compare the benefits and costs of pursuing a motive compared to other motives, and so forth, has not been sufficiently addressed. A key problem in this context is how to effectively manage limited resources, particularly computational resources, in real-time environments. The mechanisms of reactive, heuristic motive generation, a filter mechanism, and deliberative motive management is the C&AP's preliminary solution to this problem

((Bratman, Israel & Pollack, 1988) also propose a filter mechanism to meet similar requirements). Other work in decision-theoretic control (rational deliberation under resource limitations), such as anytime algorithms (Boddy & Dean, 1989), amended utility theory (Horvitz, Gregory & Heckerman, 1989; Miranda, 1996), algorithms that trade quality of solution with speed of response (Lesser, Pavlin & Durfee, 1989), or rational self-government (Doyle, 1989), is also of relevance for developing a theory of motive management. Armed with such a theory MINDER1's primitive motive management mechanisms could be improved.

5.5 Emergent processing states

There is much philosophical wringing of hands over the meaning of the term 'emergence' and what it might really mean. Here the term is used in two ways. First, to refer to unexpected consequences of a design. Normally it is too difficult to deduce all the consequences of a design, which is one reason for the necessity of implementation. Both filter and decision oscillation, discussed below, are emergent in this sense. The second use of the term is to refer to expected processing states that arise from interactions between submechanisms. Perturbant states were hoped for consequences of the design but there is no 'perturbance-producing' mechanism or module. Emergence of this kind occurs because relations are real; that is, the interactions between processes are just as real as the processes themselves (compare thrashing in operating systems, or the laws of supply and demand in economic systems). 'Emergent' is normally reserved for this kind of occurrence.

5.5.1 Filter oscillation

If there are fewer than three surfaced motives the filter level is gradually lowered to allow any $m_i \in M_{sub}$ to surface into management. However, there are occasions when many $m_i \in M_{sub}$ have identical insistence values. Consequently, when the filter level is lowered a number of motives may surface at the same time. If this occurs the filter level needs to be raised because there are now *more* than three surfaced motives. However, when the filter is raised *all* the recently surfaced motives dive. The filter will then be re-lowered and the cycle repeats (see dense oscillation regions of the filter level in figure 5.7). The threshold level continues to oscillate until insistence levels change, or more insistent motives are generated, or the active motive is completed, or a motive is removed, and so forth.

This author had not considered the possibility of filter oscillation but in retrospect it seems unavoidable when the implementation of the filter relies on a real number that is incremented or decremented in discrete amounts. Moreover, the granularity of some insistence heuristics is not sufficient to assign unique insistence values to different motives. However, filter oscillation is a feature of this implementation, but not the design. Oscillation could be avoided by using a neural network or 'fuzzy' implementation of the filter mechanism, and Beaudoin (1994), anticipating such a problem, discusses a 'filter refractory period' that briefly increases the resistance of the filter after a motive surfaces.

5.5.2 Decision oscillation

Occasionally, MINDER1 will 'see' two minibots that are close to a ditch. The new sense datum generates new beliefs that generate new motives to rescue the minibots from falling into the ditch and damaging themselves. Normally, MINDER1 will, all other things being equal, adopt the motive with the higher insistence[6], grab the particular minibot, and remove it to a safe distance. However, if in the course of executing the plan for this motive, the other minibot moves even closer to the ditch, MINDER1 may drop what it is doing and attempt to rescue it. Occasionally, however, both minibots are a similar distance from the ditch, which results in a similar magnitude of insistence for each motive. Such a situation results in 'dithering' or 'indecision', both internally, in terms of the repeated adoption and replacement of each motive by the other, and externally, in terms of the agent repeatedly moving first to one dependent then stopping and moving towards the other. Occasionally, such 'indecision' results in neither motive being completed and both minibots falling into the ditch.

There are at least three ways to avoid this problematic motive processing state. Abilities to construct a single plan to satisfy more than one motive would enable MINDER1 to save both minibots. Alternatively, an implementation supporting developed measures of urgency and importance could impose an ordering on motives when insistence heuristics do not. Finally, metamanagement processes could detect states of decision oscillation and arbitrate. However, there are examples from human and animal behaviour of decision oscillation, which suggests that, in general, the state cannot be avoided. For instance, humans are extremely indecisive when confronted with hard ethical problems.

[6]Note that, in the C&AP design, insistence is not a criterion for adoption but for consideration. However, because MINDER1's management processes do not develop measures of urgency and importance, insistence normally has this role in the implementation.

5.5.3 Perturbant states

Perturbances are the type of information processing state that the AFP theory posits as a characteristic feature of many human emotional states ((Simon, 1967; Sloman & Croucher, 1981; Sloman, 1987; Sloman, 1992) and chapter 3). For example, both intense grief and joy involve perturbant states: the mourner and lottery winner both find it difficult to direct their attention to other concerns – there is a notion of 'loss of control' common to both states.

This section describes how MINDER1 can *potentially* support perturbant states but lacks the necessary architectural features to lose control of its management processes.

5.5.3.1 Perturbant scenario

Consider the following scenario that could occur in the nursery domain.

> Robby the robot minder notices that a dependent has fallen into a ditch and has been damaged. Robby decides not to retrieve the damaged minibot because it has other pressing things to do. It has various active motives that are more urgent and important, such as recharging other active minibots, building a protective enclosure of fences, and ensuring that more minibots do not fall into ditches. However, the thought of the damaged minibot lying there continues to enter Robby's thoughts, diverting attentive processing resources from the current set of active tasks. Robby can't seem to get the thought out of his mind and finds it difficult to concentrate on the task at hand ... Sometimes, however, Robby is so very busy looking after the minibots that he temporarily forgets that there is a damaged dependent waiting to be repaired ... When things eventually calm down, Robby retrieves the damaged minibot and places it in the dismissal point for repair.

MINDER1 is not as sophisticated as Robby the robot because, unlike Robby, MINDER1 has no mechanisms that can support a notion of 'loss of control' of management processing. There are no metamanagement processes that express goals about what *should* be occurring in management. Without these specific kinds of normative, goal-directed processes there can be no notion of MINDER1 controlling or failing to control its own management processing. Perturbance requires an architecture sufficiently sophisticated to support goals whose object is to control management (or 'attentive') processing. MINDER1 does not have this architectural complexity; consequently, it can only partially or potentially support perturbant states.

However, it *does* possess mechanisms that support the continual surfacing of motives that are repeatedly rejected by management processes. The repeated interruption of management processing is precisely the kind of state that has the potential to conflict with higher level metamanagement decisions if the mechanisms for such decisions were added to the architecture. It is these potential or *proto-perturbances* that are now described.

5.5.3.2 Proto-perturbances in MINDER1

To show that MINDER1 could potentially support perturbant states, motives pertaining to damaged minibots were assigned relatively high insistence values that management processes would subsequently 'disagree' with. That is, management processes would reject these motives by assigning them *low* importance, contradicting their heuristically assigned insistence values (see node 4 in figure 5.6). The design decision was made in order to test a theory (see section 5.4.1) and is therefore slightly contrived. However, if MINDER1 possessed more sophisticated motive management processing then contradictions between insistence and developed measures of urgency and importance would be commonplace, some of which may be correctable by learning.

In order to detect proto-perturbances a metamanagement process was devised that measured the *rate of rejection* of motives. If the rate exceeded a threshold then an occurrent proto-perturbant state was detected. If the rate subsequently dropped below the threshold then the state had ended. MINDER1 maintains internal records of proto-perturbant episodes. However, it must be stressed that, at present, the information about the occurrence of a proto-perturbant state is not used by metamanagement to control the state. For example, such information could be used to change the insistence heuristics for such motives, or place exception handlers in the filter mechanism. But this kind of 'self-control' of motive processing was not implemented. (It would have been a trivial matter but the AFP theory hits difficulties explaining the powers and limits of meta-deliberative control. See the 'control precedence problem' discussed in chapter 6.) If it had been then MINDER1 would be an agent architecture that had goals directed towards controlling its own management processes, and the beginnings of a simulation of a perturbant state proper.

Figure 5.7 is a graphical representation of some statistics collected during a sin-

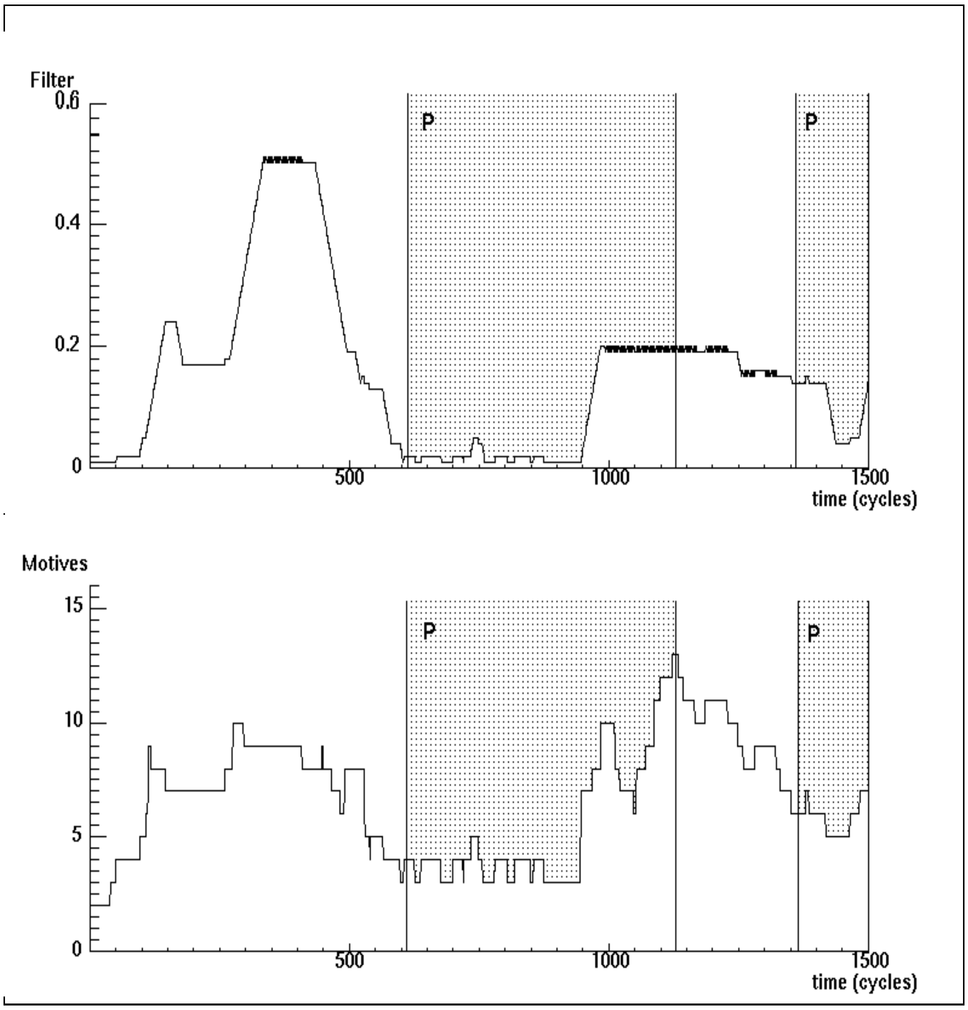


Figure 5.7: **Filter level, number of motives and proto-perturbant (P) episodes**

gle run of MINDER1 in the nursery domain. It shows that MINDER1 supports proto-perturbant states, correlatating their detection (shaded areas labelled 'P') with the total number of agent motives (motive axis) and the filter threshold level (filter axis) over time (cycles axis)[7]. A proto-perturbant state occurs when MINDER1 knows that a minibot has fallen into a ditch. The resultant belief satisfies the preconditions of a generactivator that generates a motive to dismiss it. Management processes decide that the importance of this motive is low compared to other motives and rejects it. However, the insistence level is such that the motive resurfaces into management processing. If this event happens with sufficient frequency, metamanagement processes detect the proto-perturbant state. Yet if there are many other

[7]The run included 5 minibots and 4 fences. These parameters can be varied.

highly insistent motives the filter may be raised to such a level that the dismissal motive cannot surface. When this occurs the proto-perturbant episode temporarily ceases (see cycles 1137 to 1355 in figure 5.7). When the highly insistent motives have been dealt with, the filter may be lowered allowing the dismissal motive to once again grab management resources, demonstrating that a proto-perturbance may be either occurrent or dispositional.

MINDER1's proto-perturbances are a pale shadow of human perturbances. For instance, MINDER1's proto-perturbances are all of one type. They arise from the difference between the insistence value of the motive and the importance assigned to it by management processes. There are no 'catchy tunes' in MINDER1, or vivid episodic memories, or unsatisfiable motives, and so forth. Compared to the flora and fauna of the human mind, and the corresponding variety of perturbant states, MINDER1 is a simple automaton. But although MINDER1 does not support 'loss of control' it may, from the outside, be judged as dysfunctional when in a proto-perturbant state by an observer who knows what is in the best long term interests of the agent, even if the agent cannot make that assessment (compare parents watching their young children).

Perturbances (and proto-perturbances) do not arise due to a special perturbance generating mechanism but the mechanisms that generate perturbant states may themselves be functional and have evolved, in natural minds at least, for specific, adaptive purposes.

5.5.3.3 An aside: perturbance and 'rumination'

The repeated surfacing and rejection of a disrupting motive is costly in terms of management processing resources. Sometimes a proto-perturbant state has noticeable behavioural effects, such as when MINDER1 is slightly slower to adopt new motives due to the effects of repeated management interruption. Disruption of behaviour becomes even more pronounced if there are two 'perturbing' motives because the execution of an adopted motive is continually suspended resulting in MINDER1 moving more slowly through the nursery. However, if there are three or more 'perturbing' motives MINDER1 stops moving altogether and enters an internal loop or state of 'rumination'. 'Rumination' occurs because there are no other motives with higher insistence than the three 'perturbing' motives, and the whole of management processing time is spent continually rejecting them. For example, on each cycle the following occurs: (i) management processes will decide to *reject* one of the surfaced dismissal motives, (ii) another motive will *dive* due to being rejected on the previous cycle, and (iii) another motive will *surface* into management after diving from the previous cycle. The process then repeats (see appendix, section A.5 for trace output of this cycle). Note that there are never more than three surfaced motives. Consequently, the filter is never raised (which would prevent the proto-perturbant state). The fruitless merry-go-round of 'perturbing' motives will continue until an event in the nursery generates a highly insistent motive that is adopted for execution. However, the proto-perturbant state may continue if the filter level is not raised. Introducing an 'intensity' measure to motives would avoid this kind of state, where intensity is a dispositional ability for a motive to hold onto management resources once adopted.

5.6 Perturbance and other design-based implementations

MINDER1 is the only, albeit simple, example of a fully-functioning implementation of an 'interrupt' theory of emotionality.

There are other design-based simulations of emotional phenomena. However, they tend to emphasise *representation* over emergent processing states and concentrate on the semantics of emotional appraisals or emotion words. For example, Dyer's BORIS, OpEd and DAYDREAMER systems (Dyer, 1987; Meuller, 1990), influenced by an evaluative theory of the emotions, appraise story or 'daydreaming' scenarios with respect to built-in goals. The appraisals may then generate a prediction of what emotional state is appropriate in the given scenario. These systems, therefore, reason about emotional labels and their semantics. Scherer (1993) has developed an expert system that requests information from a user about emotional scenarios and then predicts what emotion the user felt in that scenario. The expert system models the kinds of appraisals that are important in the elicitation of an emotion, and is therefore of relevance for developing generactivators that more closely model human-like motive generation; however, it is not a model of a complete agent and does not support emergent processing states. Frijda and Swagerman's ACRES system (Frijda & Swagerman, 1987) is a computer program that stores facts about emotions and reasons about those facts, but, in addition, has various goals or concerns (see section 6.3.1) that it attempts to meet, such as a concern to have correctly typed input. The satisfaction or dissatisfaction of the program's concerns may cause it to interrupt current processing and generate new responses, such as a request to the user for correctly typed input. ACRES can be asked for information regarding its internal state, which is a measure of how well its concerns have been met. However, ACRES is not an implementation of an architecture that can support a distinction between attentive and pre-attentive processing, and does not exhibit emergent processing states. Pfeifer's FEELER system, reviewed in (Pfeifer, 1994), also predicts appropriate emotional states given story scenarios. The OZ project has investigated the role of emotion in artificial, believable agents (Bates, 1994; Reilly, 1993), which are agents that make it easy for an audience to suspend their disbelief and accept the 'reality' of the agents before them, much as an audience accepts that an actor is King Lear. However, the mechanisms driving the emotional agents are designed to express emotional states in animated movement, to time and accentuate those expressions for maximum effect, and to perform prescripted emotional scenarios. There can be no loss of control of attentive processing in these agents, though they may be able to 'smile' in appropriate scenarios.

Such implementations tend to 'program in' representations of emotional intentionality. The approach adopted here is quite different. The process moves from requirements for complete functioning agents, to designs that meet those requirements, and the testing of implementations. Perturbant states arise from other mechanisms designed to meet those requirements. In the abstract, the requirements for MINDER1 are the same as those for human-like autonomy; hence, perturbant states have greater claim to model actual aspects of human information processing. Other models have difficulties making such claims.

An exception is Moffat and Frijda's WILL architecture (Moffat & Frijda, 1995), partly similar to MINDER1, and a design for a concern realisation system (see section 6.3.1). However, WILL does not exhibit protoemotional states. Also, Botelho & Coelho (1996) describe an agent architecture that can interrupt its attention in response to new environmental contingencies. However, it does not support reactive, deliberative and meta-deliberative forms of control.

MINDER1 demonstrates that phenomena-based approaches to the study of the emotions are necessarily limited: the causal relations between perturbant states and observable behaviour[8] are indirect. Any highly complex information processing mechanism will have this property. However, this is not to say that phenomena-

[8]That is, the causal relations between internal states and behaviour observable by other agents in the nursery domain. If tracing facilities were available for human minds that enabled us to examine their internal processes then the problems of psychology would be largely solved. Unfortunately, this is not the case.

based approaches are of no use; on the contrary, they have generated empirical data and driven the development of theories. But to understand fully the complexity of emotional states, which can involve complex internal states not directly linked to observable phenomena, a design-based approach is also required.

5.7 Conclusion

Implementing agent architectures, even with good tools, is a time consuming exercise and presents difficult software engineering and artificial intelligence problems. However, it is a necessary stage of the design-based approach, an approach that the C&AP believes is currently the only way to fully explore the complexity of mental phenomena. Building MINDER1 has shown that the C&AP's paper design can meet the discipline of computational realisation, or at least that part of the design that has been built given the available resources; and that it probably meets its requirements, or at least this particular implementation appears to manage multiple motives in the nursery domain. Human minds also have to manage multiple motives in complex and dynamic domains while maintaining reactivity to current events. These results at least support the possibility that the gross functional decomposition of our design corresponds to information processing mechanisms that exist in human minds, for human minds also have to manage multiple motives in complex and dynamic domains while maintaining reactivity to current events.

In addition to having relevance to agent architecture research, MINDER1 has the architectural prerequisites to support perturbant states that involve a loss of control of attentive resources. Perturbant states are characteristic features of emotional states. Therefore, MINDER1 can be described as a protoemotional architecture, as long as this is understood not to be a claim about the first-person phenomenology of MINDER1 nor a claim about what constitutes emotionality in general. MINDER1 can enter states in which a motive continually surfaces through a variable threshold attention filter despite being continually rejected by resource limited management processes. This is precisely what is meant by 'protoemotional', nothing more and nothing less.

Chapter 6

Theories of emotion

The existing literature on the emotions is large, spanning disciplines and centuries. This chapter does not attempt a comprehensive review of emotion theories; instead, a taxonomy of emotion theories is presented. The classification shows what kinds of high level theoretical commitments the thesis makes and how those commitments differ from other theories.

A number of design-based theories of emotion are reviewed, followed by extensive critical comments. The criticisms of existing design-based theories motivate part III of the thesis.

6.1 Green's classification of emotion theories

The philosopher O. H. Green provides a useful classification of emotion theories (Green, 1992). Green classifies emotion theories according to the account they give of emotional intentionality and the relation of emotions to non-intentional phenomena. In philosophical language, 'intentionality' means that a mental content, such as a belief, is 'about' something or represents something, such as a belief about today's sunny weather. Non-intentional phenomena are mental contents that do not represent in this way, such as occurrent hedonic states of mental pleasure or unpleasure, which, in themselves, do not refer to other states, objects, or events. Colloquially, non-intentional phenomena are often referred to as 'feelings', but this word will be avoided because of its vagueness. Table 6.1 classifies emotion theories into three broad categories: component, evaluative and belief-desire theories.

Component theories of emotion hold that the intentionality of emotions derives from beliefs about states of affairs, and that the contemporaneous physiological

Table 6.1: **Green's classification of emotion theories**

Theory type	**Intentional element**	**Non-intentional phenomena**
Component	Belief	Constituitive
Evaluative	Belief	Non-constituitive
Belief-desire	Belief + Desire	Non-constituitive

and 'affective' phenomena are necessary constituents of an emotion. Component theories view emotions as combinations of beliefs and non-intentional phenomena. For example, component theories would normally conceptualise an emotion such as anger as intentional due to a belief about a perceived slight, and emotional due to an increased arousal of the autonomic nervous system. Green names the psychologists Arnold, Izard, Averill, Lazarus, Plutchik, Ortony et al., James, and others, as proponents of component theories of emotion.

Evaluative theories of emotion hold that the intentionality of emotions derives from beliefs that express evaluations about states of affairs, but that contemporaneous physiological phenomena are unnecessary constituents of an emotion. For example, evaluative theories would normally conceptualise anger as intentional due to a belief about a perceived slight but would not agree that other non-intentional phenomena are also necessary for anger. Green names the philosophers Augustine, Hobbes, Spinoza, Brentano, Sartre, and others, as proponents of evaluative theories of emotion.

Belief-desire theories of emotion hold that the intentionality of emotions derives from beliefs *and* desires about states of affairs, and that related non-intentional phenomena are unnecessary for an emotion proper. Belief-desire theories view emotions as relations between beliefs and desires. For example, belief-desire theories would normally conceptualise anger as intentional due to a belief about a perceived slight and a desire to be shown respect, but would not agree that other non-intentional phenomena, such as a reddening of the face, are necessary for anger. Green himself proposes a belief-desire theory of emotion, reviewed in section 6.2.

A point to emphasise is that design-based theories of emotion are belief-desire theories. The distinction made by philosophers and psychologists between beliefs (cognition) and desires (conation), where 'the success of cognitive representations depends on the way the world is, that of conative representations on the way the world is made to be' (Green, 1992), is directly mirrored in control system architecture from the very simplest negative feedback loop to the more complex agent architectures developed by AI researchers. A feedback loop is the simplest example of a belief-desire system, as long as it is understood that the terms 'belief' and 'desire' have been freely abstracted from folk psychology and applied to control systems in general. The input signal of the feedback loop is a belief, the reference signal a desire, and the output signal an action that attempts to make the world conform to desire (McCarthy, 1979; Sloman, 1993c; Dennett, 1996; Powers, 1988; Braitenburg, 1984). Similarly, in MINDER1 there is a functional distinction between beliefs deriving from sense data that represent states of affairs in the world, and motives or goals deriving from generactivators that represent states of affairs to be achieved or maintained. Component and evaluative theories neglect *the* important aspect of agency: motivation.

Green raises more technical objections to component and evaluative theories, but his full analysis is not repeated here. Very briefly, however, Green argues that component theories of emotion cannot account for dispositional emotions and evaluative theories cannot explain the difference between emotional and non-emotional evaluations. For example, it is possible to have an emotion without it being manifest; hence, non-intentional phenomena cannot be constituitive of emotions: a mourner may function normally at work and show no signs of crying or preoccupation, only to break down in the evening. At work the grief was dispositional, in the evening occurrent. Emotion words can be used as disposition terms because grievers, aggressors, lovers, and depressives are prone or tend to manifest certain behaviours that are recognisably emotional; and an occurrent failure to manifest does not imply the absence of a disposition.

This thesis, therefore, rules component and evaluative theories out of court for two main reasons: emotions can be dispositional, and all forms of control system support a distinction between beliefs and desires. Phenomena-based and semantic-based theories of emotion are ignored in the literature review for the reasons given in section 2.3.1 and section 5.6.

6.2 A philosophical theory: Green

Green's belief-desire theory views emotions as relations between beliefs and desires: they are 'intentional structures of beliefs and desires' (Green, 1992). For an emotion to occur a set of beliefs and a set of desires must be about a common topic and semantically interrelated. Basic emotions arise from the basic configurations of beliefs and desires.

Believes that p, Desires that p $\rightarrow$ convergent emotion;

Believes that p, Desires that not p $\rightarrow$ divergent emotion;

where p is a proposition.

Convergent emotions are all types of 'positive' emotion, such as happiness, joy, triumph, gratitude etc., whereas divergent emotions are all types of 'negative' emotion, such as fear, sadness, despair, anger, etc. In other words, emotional states can arise from matches or mismatches between desires and the way the world is perceived to be. In Green's terms emotions have 'hedonic state' components, which for convergent emotions are forms of pleasure, and for divergent emotions forms of displeasure. Whereas beliefs can be true or false and desires can be satisfied or unsatisfied, emotions are neither true or false, satisfied or unsatisfied but are constituted by relations between beliefs and desires and forms of pleasure or displeasure. The intensity of emotions is a function of the 'strength' of the beliefs and desires that constitute them. So, for example, if a desire to be shown respect is 'weak' the corresponding emotion occurrent on a perceived slight will also be 'weak'. Green subdivides his classification of convergent and divergent emotions into concrete emotions by consideration of the content of the constituitive beliefs and desires.

For Green emotions are motivating.

Convergent emotion $\rightarrow$ not motivated to act further;

Divergent emotion $\rightarrow$ motivated to act further.

For example, a person may believe that he has won the lottery and desire to be rich. The belief and desire are about the common topic of money and are convergent; hence, the person will experience a convergent emotion that he and his friends will call 'joy' or 'happiness'. As the emotion is convergent, and a desire has been satisfied, the person need not act further, for the time being at least. Contrast a person who believes he has been sacked and who desires to have a job. The divergent emotion encourages further action. According to Green, negative emotions have greater explanatory power in the understanding of behaviour. Knowing that someone has a divergent emotion can help to predict their behavioural dispositions (sometimes referred to as 'action tendencies' in the emotions literature), whereas knowing someone has a convergent emotion is less informative of behavioural dispositions. According to Green, this is reflected in folk psychology, which has more concepts for divergent than convergent emotions.

The belief-desire theory emphasises the passivity of emotions: 'emotions are not causally or rationally dependent on desires that one have them, as actions are dependent on desires that one perform them, and are not identified by reference to such desires.' In other words, one cannot normally choose to have an emotion like one can normally choose to act.

6.2.1 Criticism

A problem with the belief-desire theory is that it does not consider how beliefs, desires and forms of pleasure and displeasure could be implemented in an information processing architecture. The theory is expressed in the terminology of folk psychology, augmented by the technical language of philosophers, and does not delve below the knowledge level to the information processing level of description. This lack results in redescriptions of phenomena at the same level masquerading as explanations. For example, Green's explanation of the passivity of emotions is a comparison of the relation between desires and acts and desires and emotions. There is no explanation of why or how emotions can control or be controlled.

The motivational consequences of emotions are overly simplified in Green's theory. Convergent emotions may also generate second-order motives, for example gratitude may engender a desire to repay a good turn, and triumph may lead to a desire to share the good news.

Also, Green's definition of an emotion is too broad. Many structures of beliefs and desires are not emotional. For example, an attitude, such as loyalty to a friend, consists of relations between beliefs and desires, but is not emotional. Similarly, many hedonic states or forms of pleasure or displeasure are not emotional, such as enjoying a sunset, or enjoying fishing. So hedonic states, in addition to beliefs and desires, are not constitutive of emotions. Also, the functional role of hedonic states, what they are and what they do, is not analysed in sufficient detail.

However, the belief-desire theory's high level of abstraction, while resulting in oversimplification in many instances, does place a number of issues in sharp relief: the relation between beliefs and desires, the passivity of emotions, the importance of forms of pleasure and displeasure, and the convergent-divergent, or match and mismatch, dimensions of emotional states.

6.3 Design-based interrupt theories

This section summarises two design-based theories, Frijda's 'concern realisation' (CR) theory and Oatley and Johnson-Laird's communicative (COM) theory. They both share a family resemblance to Simon and Sloman's interrupt theories.

6.3.1 Concerns: Frijda

Frijda (1986) holds that emotions arise in systems that realise multiple 'concerns' in an uncertain environment. Frijda defines concerns as dispositions to desire the occurrence or nonoccurrence of a situation, for example the concern to achieve sexual satisfaction. Therefore, concerns can be thought of as the enduring goals of an agent. For example, MINDER1 has a set of concerns implemented as generactivators, such as ensuring that a minibot does not fall into a ditch. According to Frijda the major phenomena that constitute emotions follow from considerations of the design of a multiple concern realisation system. Frijda's theory begins with the very same requirements as Sloman's AFP theory and is similarly interested in possible designs that could satisfy those requirements.

For Frijda, the major phenomena that constitute emotions are the existence of feelings of pleasure and pain, processes of appraisal based on concerns, the presence of innate, pre-programmed behaviours in addition to complex constructed plans for achieving emotion goals, the occurrence of behavioural interruption and disturbance, and the control precedence of emotion goals (Frijda & Swagerman, 1987). An emotion can be generated by a process of appraisal that analyses a situation to detect whether there is a match or mismatch between events and concerns.

> Conditions involving the possibilities of obtaining or losing ... attractive or aversive objects give off the signals of pleasure and pain. Pleasure and pain can be understood as the signals for match and mismatch. ... They imply action: 'Pleasurable' indicates that the thing is welcomed, painful that the event is not as events should be. They are penetrant; they persist; they are compelling with respect to action control (Frijda & Swagerman, 1987).

The appraisal process, therefore, generates internal signals, such as pleasure and pain relevance signals, and also a control precedence signal. The control precedence signal may cause an action system to interrupt its current activities and adopt a new plan for action to deal with the situation, or a new mode of activation, such as a high state of arousal. For example, a perceived slight may mismatch a concern to be shown respect. The appraisal generates a relevance signal of 'pain' or 'displeasure'. Further appraisal generates a control precedence signal that interrupts current processing and instigates a new mode of action readiness, such as preparedness for a heated exchange and accompanying physiological arousal. Frijda's control precedence signal performs a function similar to the insistence control signal of the AFP theory. However, Frijda does not consider a filter mechanism that protects resource limited attentive processes from too much interruption. Neither is the CR theory specified in as much design detail as the AFP theory.

The CR and AFP theories both share the view that the word 'emotion' does not refer to a 'natural class' (Frijda, 1995), that it fails to refer to a well-defined class of phenomena clearly distinguishable from other mental and behavioural events. Instead, Frijda defines a natural class that interests him: the process of emotion elicitation – event appraisal, generation of control signals, entailing control precedence, of new types of action readiness – *is an emotion*: 'these are the phenomena I am concerned with, whatever name you give them. I assert that they form a major part of what the word "emotion" refers to, and this of course is an object (the only object) of possible dispute' (Frijda, 1995).

There are three main differences between the CR and AFP theories. First, the CR theory is a belief-desire theory but equivocates over whether occurrent processes of event appraisal and changes in action readiness are necessary for emotion, or whether the disposition to change action readiness is sufficient for an emotion. The CR theory appears not to support dispositional emotions. For example, the following quotation suggests that new emotion goals must manifest in attentive processing for an emotion to be present.

> Control precedent plans, if not receiving top priority, should remain activated, ready to be triggered and to assume that priority at a suitable moment. [T]his means that ... top priority input channels and information processing capacity are shared between the current goal and the 'precedent' one. 'Nervousness', the beating heart, manifests advance energy mobilisation; being distracted manifests the sharing of resources. The general action system can ward off being dominated by the concern realisation system, but necessarily at a cost (Frijda & Swagerman, 1987).

It is not clear whether Frijda would agree that an emotion could be present without itself manifesting in attention or behaviour, or whether some degree of distraction or 'nervousness' is required. In contrast, the AFP theory clearly states that perturbances may be both dispositional and manifest, and describes an attention filter mechanism and motive management system that can support this distinction.

The second difference is that Frijda's definition of an emotion encompasses more phenomena than Sloman's concept of perturbance. The AFP theory requires loss of control of attentive processing, whereas the CR theory does not. For example, playing sport involves appraisal of events, repeated interruption of current goals, and changes in action readiness. A tennis player may notice an opponent is about to smash the ball and appraise the event as one that mismatches a concern to win the point. The mismatch may interrupt the player's current plan and instigate a new mode of action readiness, such as preparedness to return the smash. These events conform to the CR definition of emotion. However, the casual player practicing with friends and the professional player competing at Wimbledon are both repeatedly appraising, interrupting and instigating new plans, but both need not get angry at double faults, elated at won points, and triumphant when victorious. The CR theory can distinguish the cases by distinguishing the strength of the relevant concerns: the amateur's concern to win the game is weak compared to the professional's. The AFP theory can distinguish the cases by distinguishing the control of attentive resources: the casual player may laugh off his anger at a double fault, while the professional cannot (i.e., the casual player has meta-deliberative control over the contents of attention). There is no contradiction between the CR and AFP theory on this point: a 'strong' concern, according to the AFP theory, is one that generates goals that tend to grab attentive resources. However, the AFP theory states that the disposition to lose control of attentive resources is a necessary, though not sufficient, condition for what is commonly called an emotion, whereas the CR theory does not state that strong concerns are necessary for emotions. The need for 'high strength' concerns is missing from Frijda's definition of emotion, although implicit in his total theory.

Finally, Frijda emphasises the pleasure and pain dimensions of emotional states, whereas the AFP theory does not. This point will be discussed in more detail later.

To date there are two partial computational realisations of the CR theory: the ACRES system (Frijda & Swagerman, 1987) and the agent architecture Will (Moffat & Frijda, 1995). These systems were discussed in section 5.6.

6.3.2 Cognitive broadcasts: Oatley and Johnson-Laird

(Oatley & Johnson-Laird, 1985; Oatley, 1992; Johnson-Laird, 1988) presents a communicative theory of emotions (COM). They emphasise the communicative role of emotion signals within a cognitive architecture, and the communicative role of emotional expression within a social community. This review will concentrate on the former aspect of their theory.

Oatley states that the central postulate of their theory is:

> Each goal and plan has a monitoring mechanism that evaluates events relevant to it. When a substantial change of probability occurs of achieving an important goal or subgoal, the monitoring mechanism broadcasts to the whole cognitive system a signal that can set it into readiness to respond to this change. Humans experience these signals and the states of readiness they induce as emotions (Oatley, 1992)

They assume that mammalian cognitive architecture is modular, exhibiting a high degree of asynchrony, and is hierarchically organised with a top level processor that functions like the scheduling component of an 'operating system', capable of invoking lower level processes in serial sequences. The basic tenets of their theory are that emotions are a design solution to problems of the transition between plans in systems with multiple goals, serving to coordinate quasi-autonomous processes in the nervous system by communicating significant junctures of current plans.

For example, if events are going according to plan an agent will be happy. Happiness corresponds to a situation in which subgoals are being achieved without any problems, or if any problems do occur then they can be locally solved, patched, or modified without recourse to global problem-solving mechanisms. The emotion signal of 'happiness' ensures the architecture continues with its current plan. Conversely, if a major plan or an important subgoal fails, and there are no local 'fixes', an emotion signal of 'sadness' is broadcast that causes a transition from following the plan to either doing nothing or searching for a new, replacement plan. According to Oatley and Johnson-Laird if an active plan is frustrated or prevented from coming to fruition by an obstacle, a signal, corresponding to anger, can cause the architecture to aggress, that is, devote more resources to the achievement of the goal.

The example of computer programming is used to illustrate their ideas. An expert programmer can find programming 'euphorically fascinating' (Oatley & Johnson-Laird, 1985) because their level of skill is such that the majority of coding problems

Table 6.2: **Plan junctures and emotions: Oatley & Johnson-Laird's communicative theory**

Emotion	Plan juncture	State transition
EUPHORIC		
Happiness	Subgoals being achieved	Continue with plan
DYSPHORIC		
Sadness	Failure of major plan	Do nothing and/or search for new plan
Anxiety	Self-preservation goal violated	Stop, attend to world, and/or escape
Anger	Active plan frustrated	Try harder and/or aggress

can be solved locally and relatively easily. Novice programmers, however, are often faced with situations in which they cannot cope with coding problems: they lack the skill and knowledge to overcome the obstacles. It is well known that learning to program can cause feelings of anxiety, anger, frustration and hopelessness.

By analysing such plan junctures, thought to be common between mammals and humans, Oatley and Johnson-Laird draw correspondences between plan junctures and five basic emotion categories, four of which are listed in table 6.2. More complex emotions are the result of basic emotions plus additional phenomena produced as a result of the emotion signal disturbance.

An important distinction is made between *semantic* and *control* signalling.[1] Semantic signalling is the propagation of information that has meaning, representations that generally refer to something, be it objects, events, processes and so forth. For example, a semantic signal can call lower level submodules, or procedures, or represent different aspects of the world, or pass arguments to or return results from functions. In constrast, control signalling does not refer or have semantic content but performs a control function, such as changing the control flow of the system, or putting it into a distinct kind of processing state[2]. Control signals are thought to be generated by phylogenetically older machinery and, due to their simplicity, need not be parsed or interpreted. Emotion control signals propagate globally among

[1]Oatley & Johnson-Laird originally drew a distinction between propositional and control signalling; however, in *Best Laid Schemes* (Oatley, 1992) Oatley states that Sloman's suggestion of 'semantic' as opposed to 'propositional' (Sloman, 1992) better captures their intentions.

[2]The following analogy may help capture the distinction: Imagine trains travelling on a complex network of tracks. Postal trains contain mail (semantic content) with destination addresses on the envelopes. These trains travel to the destinations and deposit the mail (the information). However, a different kind of train, a 'control signal' train, can travel through the network altering the points of the tracks. This has the effect of changing the topology of the network; that is, trains will continue to deposit their mail but will use different routes.

the processors to set them into specific modes at particular junctures of planning sequences. An emotion has both control and semantic aspects. For example, sadness has a negative phenomenological tone corresponding to the 'sadness' emotion control signal, but can also be about something, that is has semantic content, such as being sad about poor results in an exam.

The COM theory is an interrupt theory as emotion signals cause current goal processing to be altered. However, the COM theory states that an emotion signal ensures that attention remains focused on the plan juncture until the situation is resolved. This differs from the AFP theory: the insistence signal is a disposition to be *initially* considered by attentive processing. The COM theory is also a belief-desire theory as emotion signals occur at significant junctures of plans, which is when beliefs match desires (euphoric emotion) or when beliefs mismatch desires (dysphoric emotion).

A major drawback of the COM theory from a design-based perspective is that an architecture that supports both semantic and control signals is not specified in any great detail; also, there are no computational realisations of the theory. Sloman (1992) points out that the COM theory stresses episodic phenomena as constituting emotions, that is, states in which signals are actually generated and disturb or modify behaviour, as opposed to dispositional phenomena; and also that control signal disturbance is ubiquitous, propagating through a society of quasi-autonomous processes, which is vague compared to the AFP theory in which emotional states have the potential to disturb attention. Oatley, however, suggests that the AFP theory has difficulty accounting for moods and longer term emotional states (Oatley, 1992). The COM theory accounts for such phenomena by proposing a dissociation between semantic and control signals. For example, dissociative effects can occur when an emotion signal is transmitted but some other goal prohibits the expression of the semantic content to ourselves or others. Oatley links such an effect to Freud's concept of repression (or suppression): the suppression of the implications of an emotion may disconnect its nonsemantic parts from its semantic parts. Hence, 'sadness' control signals could persist without an accompanying semantics, corresponding to a general and unattributable feeling of sadness.

The COM and AFP theory offer different accounts of emotion. The COM theory holds that an emotion occurs when an interrupting signal is broadcast, but, unlike the AFP theory, has no notion of loss of control of attention, that is, an architecture that can support perturbant states.

6.4 Criticism of interrupt theories

The theories of Simon, Sloman, Frijda and Oatley and Johnson-Laird are design-based interrupt theories. The proposed architectures, mechanisms of interruption, and emotion definitions differ yet they all share a set of basic assumptions: the requirement for multiple goals in dynamic and uncertain environments, a resource limited 'top level' deliberative system, interruption of that system by processes that generate new goals or check for achievement and failure of, and threats and opportunities to, current goals, followed by new plans of action, or changes in action readiness, perhaps with accompanying states of arousal.

This section outlines a number of related criticisms of interrupt theories, providing examples from the reviewed theories where appropriate.

6.4.1 Disruption, interruption and control precedence

Emotions are sometimes irrational (Frijda, 1995): they are unpremeditated, the agent has a passive relation to their occurrence or non-occurrence, unlike thoughts and actions that can be rationally dependent on a desire to have them; and often there are better alternatives than reacting in an emotional manner, from both a moral and a selfish standpoint, that may do less harm to the agent's other goals or have less harmful consequences (e.g., hurling invectives at a lover in a fit of jealousy, or kicking a stone in anger). Both these related types of irrationality are examples of the control precedence of emotions, in which mental life may be 'out of control', externally determined. 'Emotions do not act upon us, but involve something *else* which acts upon us' (R. M. Gordon, paraphrased in (Beaudoin, 1994)).

Grief is an extreme example of this. Normal cases of mourning are characterised by highly insistent perturbant states that disrupt attention, adversely affecting day-to-day functioning, perhaps for many months and years. Attention is partially and episodically out of control. It is impossible that a recently widowed young mother could decide to postpone her mourning until retirement in order to pursue her career now. Grievers normally do not have this kind of control over their thought processes.

Simon (1967) mentions the difference between 'interrupting' and 'disrupting' motivators, implying that the former are adaptive while the latter not. Longer term perturbance during grieving certainly appears 'disruptive' and not 'interrupting': it seems to serve no adaptive purpose. A designer of an intelligent agent would naturally be interested in avoiding disrupting perturbances. In the AFP architecture there are at least two ways to do this. Metamanagement could set exception fields in the filter mechanism to prevent perturbing motivators from surfacing, whatever their insistence level, freeing management resources from disruption. Alternatively, management processes could alter the insistence heuristics of generactivators that produce perturbant motivators, or remove such generactivators entirely. Altering insistence heuristics is necessary if the agent is to learn, but processes like mourning suggest that the human mind does not possess this ability, or, more accurately, altering local insistence functions and removing generactivators takes time and occupies attention.

However, conceiving of possible mechanisms for avoiding perturbances is weaker than designing and implementing those mechanisms, for there may be good design reasons why such causal powers are hard to engineer; for example, recognising matches between exception fields and motives may be computationally expensive, or constructing efficacious exception fields may require knowledge of all the potential perturbing motives, which may be difficult to predict in advance (Beaudoin, 1995).

Other attempts have been made to explain longer term, disruptive perturbances. For example, Sloman has argued that disrupting perturbance is due to insistence being a heuristic, and therefore fallible, measure of the urgency and importance of motivators: bad decisions can be made as to what does and does not surface through the filter to grab attentive processing resources. This is a partial answer as it may account for the initial disrupting perturbance but it does not argue against the possibility that local insistence functions could be subsequently altered, or certain classes of motivators selectively ignored through the use of an exception field.

(Wright, Sloman & Beaudoin, 1996) provides other possible explanations, for example the possibility that information processing attachment structures are both distributed within the architecture and highly differentiated, such that reorganisation after loss may require extensive cognitive work that, by its very nature, takes time (cf. updating a large and complex belief maintenance system). However, this explanation can explain why it may take time, or why it may be difficult to deconstruct an attachment structure, but it cannot account for persistent, disruptive perturbances, for cognitive reorganisation could either take place without management resources, or metamanagement could *decide* when to reorganise (such as a young widow postponing her grief).

The possibility of agent architectures that pursue multiple goals in complex, dynamic and uncertain domains, possessing both reactive, deliberative and meta-deliberative forms of control, which *do not exhibit perturbant states*, qualifies Sloman's position that perturbances are unavoidable consequences of resource limited agency (e.g., see (Sloman & Croucher, 1981)). This thesis agrees that interruption is unavoidable but that disruption (i.e., loss of control) may not be. The AFP theory explains *how* perturbances can occur in an information processing architecture but does not explain *why* they occur.

It is possible that perturbances are contingent features of our evolutionary history. Sloman considers this:

> Why can't my tongue reach my left ear? It's just too short. I can't say that evolutionary and survival considerations explain why my tongue isn't much longer. Similarly if an architecture happened to evolve with certain limitations, that need not be because it would have no value to overcome those limitations. I think some things have limited access to higher level information simply because they evolved much earlier, and originally needed only access to particular sub-mechanisms. E.g. detecting shortage of fluid and sending a signal to the brain may be done by a primitive mechanism that simply can't find out if the corresponding goal has previously been considered and rejected or adopted. (A. Sloman, quoted in (Beaudoin, 1994)).

However, a person who desperately needs to urinate, or needs to eat, or needs warmth, is in a perturbant state. Needing to remove waste products, needing to get energy and needing to maintain body temperature are, from an evolutionary perspective, highly important concerns that *should* have control precedence. Agent designs that could and did rationally alter the insistence heuristics of such concerns or removed those concerns or suppressed the motivational states arising from those concerns would be unfit individuals. It is therefore equally possible that perturbances are adaptive and not contingent. If so, to explain why other types of perturbance occur, such as those during grief and loss, evolutionary explanations, in addition to design reasons and contingency, need to be invoked. Evolutionary requirements, such as the need to construct and maintain attachments in successful, cooperating multi-agent societies, may impose constraints on the causal powers of deliberative and meta-deliberative forms of control. As many ascetics have discovered, one can choose not to engage in sexual acts, but it takes many years of discipline and mental exercise to stop thinking about the possibility of engaging in sexual acts. Interrupt theories tend to concentrate on domain requirements while abstracting from other requirements, such as requirements for agents able to reproduce, for agents able to learn during their lifetimes, and for agents able to operate in multi-agent societies (an exception to the latter is the COM theory). In addition to abstract *domain* requirements, *phylogenetic*, *ontogenetic*, and *social* requirements need to be considered for complete explanations of emotional phenomena. Such additional requirements will be partially explored in part III.

In summary, significant junctures of plans, or strong concerns, or highly perturbing motivators may be those that have evolutionary significance. The limited causal power of meta-deliberation or self-control is the obverse of adaptive control precedence. In order to explain disruptive, long term perturbances, such as grief, the AFP theory, and related interrupt theories, need to be augmented by new mechanisms that show how, and new requirements that explain why, meta-deliberative control is constrained.

This criticism is more developed than but in accordance with Oatley's view (Oatley, 1992) that the AFP theory has difficulty accounting for longer term emotional states, states that do not simply interrupt but disrupt and coexist with other processing states over time. However, the COM theory fares little better: the dissociation of semantic and control signals does not constitute a theoretical explanation; rather, it is a restatement of the phenomena in terms of the theory. The question of why 'sadness' control signals persist and are hard to control remains.

A full answer to these questions is well beyond the scope of this thesis. However, section 9.4.4 will provide some suggestions.

6.4.2 Learning

Simon's original paper (Simon, 1967) distinguished two types of learning pertinent to attention interruption: modification of the efficacy of certain 'stimuli' to interrupt a central processing system, and the acquisition of new response programs to interrupting 'stimuli'.

Beaudoin has highlighted the absence of learning in his agent design (Beaudoin, 1994), although the C&AP has understood the need for learning (for example, theses written by Tim Read (Read, 1995) and Edmund Shing (Shing, 1996) examined the role of learning in agent architectures). The lack is particularly acute when longer-term emotional episodes are analysed: without a theory of useful change it is difficult to account for both the persistence and decay of emotional episodes over time. Similarly, both the COM and CR theories do not explicitly address forms of learning and how they may relate to the function of emotional states. For not only are emotional states motivating but they are also moments for learning, such as the child who avoids the broom cupboard after a fearful episode accidentally locked inside. Convergent emotions are opportunities for learning why desires have been satisfied and divergent emotions are opportunities for learning why desires have been unsatisfied. For example, Frijda discusses how reinforcement learning can alter the tendency to express emotional reactions instrumental in satisfying or failing to satify desires.

> Learning affects behaviour not merely in the appearance of new modes of behaviour, but also in the readiness with which behaviour modes are displayed. Behaviour frequency, given conditions for its occurrence, is a function of expected outcome of that behaviour; and that, in turn, is a function of previous reinforcements. If ... running during fear is rewarded, it tends to increase. The same holds for aggression. Success in fighting, in rats, produces animals that are more aggressive, whereas a history of defeats rather tends to induce subsequent submissiveness (Frijda, 1986).

Despite these remarks the CR theory does not integrate designs for learning mechanisms with the concern realisation system. The same is true of the other interrupt theories. A related point is the correlation between learning and the intensity of an emotion. For example, a child locked in a broom cupboard may still fear and avoid enclosed spaces in adult life if the fear experienced at the time was intense. Bowlby (1991a) discusses this link in the context of attachment behaviour:

> Regular monitoring both of behavioural progress and of consequences is of course necessary if the organism is to learn. That is a large and controversial field and one not for discussion here. It may, however, be noted that the more strongly an appraised process is felt, and the more keenly, therefore, the consequences of some behaviour are experienced as pleasurable or painful, the quicker and more persistent is the ensuing learning likely to be. Since the formation of affectional bonds is commonly experienced as intensely pleasurable, it comes as no surprise that bonds often develop rapidly and, once made, are apt to be long-lasting ... 'they are easy to learn and hard to forget'.

The link between 'intensity' of emotion and 'intensity' of learning is a correlation not addressed by interrupt theories.

6.4.3 Pleasure, pain and intensity

This section describes how interrupt theories explain the pleasure and pain components of some emotional states, and why those explanations leave a number of important questions unanswered.

6.4.3.1 Private pleasure and pain

Terms and phrases such as 'pleasurable', 'unpleasurable', 'feelings of enjoyment', 'painful thoughts', and so forth, are notoriously vague and unwieldy when used to theorise about the workings of minds. Ryle has argued (Ryle, 1949) that it does not make sense to assert that a man taking pleasure in fishing is performing two acts, one of fishing in a certain manner and another of taking pleasure. 'Taking pleasure' need not refer to an occurrent, private mental episode, but rather serve as shorthand for a set of dispositions, such as a disposition to regularly and frequently fish, or to continue fishing despite having engagements elsewhere, or to talk about fishing more than other subjects, and so forth.

However, agents have unique perspectives on their internal states. A person suffering from a headache may be disposed to do various things, such as avoiding company and saying little, but the painfulness of the headache is an occurrent private phenomenon, and none the less real for that. It is taken as given that a full understanding of emotional states will include an understanding of internal, mental states that may not be predictably related to behavioural phenomena or exhausted by an analysis of behavioural dispositions. Unlike Ryle, this author is interested in 'wires and pulleys' explanations of private phenomena that are known to exist from introspection and the self reports of others.

In this context, it is useful to analyse the phenomenology of mental states. Phenomenological evidence is prima facie evidence derived from an examination of the structure and function of mental states via introspection. An assumption is that introspection can provide information that is useful for theory construction, and that the phenomenological structure of mental states matches aspects of their functional implementation; in other words, it is possible that functional and phenomenological analyses can partly converge. However, it is admitted that introspection depends on self-monitoring capabilities that may be limited to very superficial aspects of internal processes, and like external perception may often be wrong. In the absence of detailed theories of the underlying mechanisms, care is required when employing phenomenological concepts.

Some emotional episodes, such as grief and loss, characteristically involve states of intense 'mental pain', a form of unpleasure not linked to a part of the body in organic distress. Subjectively, the 'mental pain' of such emotional episodes is arguably their most prominent feature. Mourners often use analogies with physical distress when describing their mental states (Wright, Sloman & Beaudoin, 1996), employing words such as 'hurt', 'pain' and 'intensity'. Some emotional states may be painful and some may be pleasurable.

6.4.3.2 Thinking and feeling

In everyday conversation people often make a distinction between thinking and feeling. There are grounds for this distinction.

Cognitive representations can denote or refer to states of affairs. For example, an agent within a simulated domain may possess information about other agents in the environment, including their type, location, or speed. Such information control sub-states[3] of the agent are causally linked to their referents: B's representation of the speed of agent A will alter if it is perceived that agent A has altered its speed. This is a simple example: referential links can be very indirect in more complex information processing systems. The relation between a representation and what it represents forms one basis for information processing theories of mind: *thinking refers.*

Emotions, desires, and pleasures and pains, differ from 'cold' cognition: they can be 'hot', often involving feelings of pleasure or unpleasure with associated intensities. Unlike 'straightforward' representational thinking, an emotional state sometimes has both a representational content, for example, a state of happiness about passing exams, and a hedonic component, such as the form of intense pleasure occurrent on knowing that the exams are passed. The hedonic component does not represent a state of affairs: *feelings just 'are'.* This is not to say that 'feelings' do not play a functional role, or are not causally linked to other processes. Neither is it to deny that many everyday feeling terms are also used as disposition terms in Ryle's sense. The intensity dimension of 'affects' has also been noted by Simon (Simon, 1982): 'Affect is susceptible to continuous gradation in degree, like something that can be scaled by real numbers or modeled by an analogue device. Cognition is digital in

[3]A control sub-state is a general term for a subcomponent of an architecture bearing information (Sloman, 1996d) that has relations of control to other subcomponents or parts of an environment. For example, the curvature of a thermostat's bi-metallic strip is a substate of the thermostat architecture that contains information about the ambient temperature of the room. The cumbersome 'control sub-state' is simplified to 'substate' in this thesis.

character; symbol structures can be discriminated by yes-no tests from other symbol structures.'

There are two reasons why explanations of hedonic components are often absent from information processing theories of mind: first, hedonic components appear not to conform to the representational model that supports cognition; and second, unlike, say, deductive reasoning, their possible functional role is unclear. Faced with these difficulties theorists either opt for *physiological reduction*, asserting that hedonic components arise 'from the body', or opt for *intentional emergence*, asserting that they do not really exist, or if they do, they arise from intentional evaluations about the 'goodness' or 'badness' of states of affairs. A third possibility is to analyse their causal functional role, which is the approach adopted by Frijda, Oatley and Johnson-Laird, and Sloman.

6.4.3.3 Interrupt theorists on hedonic components

The extent to which interrupt theories can account for the hedonic components of some emotional states is now examined. In what follows, the term 'hedonic' or 'hedonic tone' is used as a synonym for 'pleasure', 'displeasure', or 'pain'. Later, a more precise definition of a particular hedonic component will be given.

Simon opts for physiological reduction. Feelings are brushed under the physiological carpet by assuming that all hedonic states arise from perceptions of bodily states. For example, Simon (1967) outlines a view that closely resembles William James' peripheric theory of the emotions: '... sudden intense stimuli often produce large effects on the autonomic nervous system, commonly of an "arousal" and "energy marshaling" nature. It is to these effects that the label "emotion" is generally attached'; and '... the feelings reported are produced, in turn, by internal stimuli resulting from the arousal of the autonomic system'. Although Simon, as a functionalist, clearly understands the need for control in addition to representation, it is difficult to conceive how his view of hedonics could account for the mental pain associated with, for example, grief, which does not necessarily require bodily arousal or disturbance.

Our other interrupt theorists, Frijda, Oatley and Johnson-Laird and Sloman, consider hedonic components as simple, phylogenetically older control signals (see section 6.3.2). This approach has the advantage that it avoids physiological reduction by describing signals with a functional role at the information processing level of description (which, of course, may be implemented in lower level processes, such as neurotransmitters) and avoids intentional emergence by distinguishing control signalling from intentional representations.

Frijda stresses the importance of the hedonic tone of emotional states, stating that 'it is pure silliness to essay explanations of such experiences as mere organic sensations or cognitive assessments or some combination of both' (Frijda, 1986). He agrees that both physiological reduction and intentional emergence provide inadequate explanations. The CR theory posits relevance signals of pleasure, pain, wonder or desire that occur when an event is compared to the satisfaction conditions of various concerns. For example, a concern to achieve sexual gratification and the event of orgasm produces a relevance signal of pleasure.

Oatley and Johnson-Laird make extensive use of the control signal concept to explain the hedonic components of basic emotional states. For example, the COM theory explains the hedonic tone of happiness and sadness states by positing basic and irreducible hedonic control signals. Control signals have different hedonics (i.e. are either pleasurable or unpleasurable) because they differ in their functional roles. For example, the 'sadness' control signal has a terminate or change plan function, whereas 'happiness' has a preserve or maintain plan function.

The AFP theory defines the insistence of a motivator as its dispositional ability to surface and obtain attentive processing resources; however, insistence is not linked to hedonic components, nor is it clear why it should be as, for example, many 'attention grabbers' are neither pleasurable or unpleasurable (consider a loud bang). However, the intensity of the hedonic component of some emotional states can be correlated with the allocation of attention. The intense mental pain of grief is associated with a drastic loss of control of attention.

Sloman has discussed hedonic components in terms of negative and positive evaluations that have a terminating or preserving motivational role.

> States in which something is enjoyed, relished, found pleasant, admired etc., involve explicit positive evaluations and therefore tend to produce motivations to preserve or extend or reproduce the current state, or something that causes it, while states in which something is suffered, disliked, found painful, found unpleasant, despised, etc., involve negative evaluations and therefore tend to produce motives to terminate, shorten, or prevent the state, or something that causes it. The preceding uses of the word 'therefore' presuppose a mechanism linking positive and negative evaluations with specific classes of dispositions to produce motivational states. (Having such dispositions may, in certain simple organisms, be all that such an evaluation amounts to.)(Sloman, 1992)

For example, enjoying fishing involves an explicit positive evaluation of the act of fishing which tends to produce motivations to keep on fishing, to fish again at the earliest possible opportunity, to inform people of reasons why fishing is enjoyed when asked to do so, and so forth. However, explicit evaluations are not the whole story. Intense grief involves both explicit negative evaluations *and* an implicit negative evaluation that is like the pain component of a headache, or like a throbbing ache in the left leg, but linked to the loss of a loved one rather than a bodily location. The implicit evaluation is often referred to as the 'feeling' of the emotional state.

An important difference between an explicit and implicit evaluation is that the former is intentional while the latter is not. For example, a jilted lover may form an explicit negative evaluation derived from a belief that a loved one has left her for another, and a desire that the loved one remain true. The evaluation inherits intentionality from the constituent beliefs and desires; it may therefore refer to a state-of-affairs that is false (the jilted lover may be mistaken in her belief). In contrast, the jilted lover may also form an occurrent implicit evaluation consisting of intense mental pain due to the loss of the loved one, a pain which is like a headache as it can be more or less intense and, also like a headache, is neither true or false, in the way that beliefs can be, nor satisfied or unsatisfied, in the way that desires can be. Pain is never true or false or satisfied or unsatisfied. Pain can have location or be caused by events but the painfulness does not refer and is not intentional. It is these kinds of mental contents that lead some to believe that the emotions are ineffable, irreducible, irredeemably subjective, and opaque to mechanistic analysis and explanation.

In (Wright, Sloman & Beaudoin, 1996) the concept of perturbance was used to provide a design-based analysis of loss or grief. However, we were unable to provide a satisfactory account of perturbant states that possess a pleasure or unpleasure dimension (compare grieving with glee). It was also unclear whether an information processing architecture could account for types of mental pleasure and pain. In that paper Sloman wrote:

> *Pleasure and pain mechanisms.* Pleasure and pain require an extension to the architecture sketched so far, probably involving phylogenetically old structures in the nervous system that are deeply implicated in mechanisms of learning and online control: i.e. whether current activities should be maintained or terminated. Whether an information processing architecture can fully account for such subjective states is an open question. For the time being we assume that there are mechanisms pro-

> ducing motivational control states that are pleasurable in the sense of involving a disposition to preserve or extend something that is currently happening, whilst others are painful in the sense of involving dispositions to terminate or reduce some state or process. Some of the control states in the architecture may have a positive or negative [hedonic tone] that arises out of genetically programmed drives (e.g., seeking after novelty, urge to procreate and form attachments) and states of body arousal. The experience of pain and pleasure will require self-monitoring mechanisms that detect these 'preserve/terminate' control states. The mechanisms may be linked in that termination or reduction of pleasure inducing states may function as a pain inducing state and *vice versa*. In some cases these links will need to be built up through learnt associations. In others they may be 'hard-wired' in the architecture (Wright, Sloman & Beaudoin, 1996).

In this quotation the phrase 'hedonic tone' is used to refer to the implicit evaluations. Implicit evaluations are, in agreement with the COM theory, assumed to be phylogenetically older than explicit, semantic evaluations. Also of note is that the causal role of the hedonic signals is similar in both theories: pleasure has a preserve role (cf. 'continue with current plan') and pain or unpleasure has a terminate role (cf. 'stop current plan'). This is not a new idea: 'Pain cries "Be gone!", but pleasure craves eternity ...' (Nietzsche, 1896). However, the above quotation from (Wright, Sloman & Beaudoin, 1996) does not stipulate the presence of plans: pleasure and pain mechanisms may be present in both reactive and deliberative architectures.

To distinguish between the two types of evaluation, the term *evaluation* will be reserved for explicit, 'cognitive' evaluations, such as explicit descriptions, for example 'this is good' or 'this is bad' (note, however, that descriptions need not be represented in natural language), whereas the term *hedonic tone* is reserved for implicit evaluations. Hedonic tone can be either positive or negative corresponding to its 'preserve/terminate' causal role.

A theory of pleasure and pain is required independently of any theory of emotions as events may be pleasurable or painful without involving emotions (Sloman, 1996a), for example enjoying one's food or sitting uncomfortably. Pleasure and pain in emotional states are a special case of a more general phenomenon. However, for more complete explanations of some emotional states the hedonic tone component needs to be addressed.

6.4.3.4 Preliminary definition of valency

Given the great variety of hedonic states it would help matters if a subset could be identified for analysis.

'Feeling' is an ill-defined word, for it can cover such diverse sensations as one's cheeks burning with embarrassment, an itch on the left ear, or the happiness associated with triumph. Also, the word 'feeling' can be used in many other contexts, such as feeling like having a cup of coffee, or feeling in the mood to dance and so forth. 'Hedonic tone' is similarly general: it can be used to refer to the enjoyable sensation of a full stomach after a large and hearty meal, the pleasure of sexual anticipation, or the pain of failure. To attempt to be more precise a new term, valency, will be introduced. First, some distinctions are required.

A division can be made between *physiological* forms of pleasure and unpleasure, and *cognitive* forms. For example, the (self-monitored) 'itchiness' on one's left ear is a form of unpleasure linked to information concerning bodily location. In contrast, the mental pain of intense grief is a form of unpleasure linked to information about a loved one's death. There can be no 'pain receptors' for this kind of unpleasure, unlike the nerves that detect a pin pricking one's finger. For example, an athlete may be experiencing the occurrent emotional state of triumph while standing on the winner's podium. The intentional component of her state includes thoughts pertaining to the achievement of long-term goals and, for example, information about a rapid heart-rate and the warm sun beating on her brow; the non-intentional component *includes* the pleasurable sensation linked to the warm sun on her brow, *and* a state of cognitive pleasure not linked to the body but linked to her athletic success. Unlike the intentional component of an emotional state, the form of cognitive pleasure or unpleasure does not represent a state of affairs but allows quantitative degrees of intensity. The magnitude of intensity may be reported to oneself or others in terms of qualitatively different concepts, but the intensity itself admits of only quantitative differences.

During many states often called emotional there is an indissociable connection between hedonic tone and intensity. One cannot occur without the other. The hedonic tone always has a quantitative component, admitting of degrees of 'positiveness' or 'negativeness', a magnitude of intensity. However, subjects use the term 'intensity' in different ways to describe different aspects of an emotional episode. For example, Sonnemans & Frijda (1994) distinguish six aspects of emotional intensity: the duration of the emotion, perceived bodily changes and the strength of felt passivity (i.e., loss of control of attention), recollection and re-experience of the emotion, strength and drasticness of the action tendency (e.g., the strength of the desire to hit someone when angry) and the drasticness of actual behaviour (e.g., actually hitting someone), belief changes and influence on long-term behaviour, and an overall felt intensity. Hedonic tone intensity does not fit any of these categories, although may be related to them. For example, mourners cannot choose to ignore the mental pain of occurrent grief, and the more intense that pain the harder it is to ignore. Therefore, hedonic tone intensity is related to Sonnemans and Frijda's category of 'strength of felt passivity', for both intense pleasure and intense unpleasure are hard to deliberately control.

Given these considerations a preliminary definition of valency can be provided. It is preliminary because the definition is based on introspection and folk psychological concepts.

> *Preliminary definition of valency.* Valency is a form of cognitive pleasure or unpleasure not linked to information concerning bodily locations, and is a quantitatively varying, non-intentional component of occurrent convergent or divergent emotions. Valenced states are contingent on the success or failure of subjectively important goals.

This definition of valency is more specific than hedonic tone and is a special case of a more general notion. For example, valency as defined here is not to be confused with other forms of pleasure or unpleasure that are shorter term control states involved in whether current activities are preserved or terminated; neither should it be confused with 'values' in general, such as evaluative, qualitative, or symbolic, affective dispositions towards states of affairs. Instead, valency is 'achievement pleasure' or 'failure unpleasure' that occurs when certain highly important concerns are met or violated, for example the concerns that develop during the formation of personal attachments.

6.4.3.5 Lacks in control signal explanations of hedonic tone

For Simon valency arises from bodily disturbances, for Frijda and Oatley and Johnson-Laird valency arises from control signals, and similarly for Sloman valency is deeply implicated in both on-line control, such as preserving or terminating current activities, and longer term learning, such as repeating events associated with occurrent pleasure, or avoiding events associated with occurrent pain or displeasure in the future. Both Frijda and Sloman distinguish valency and control precedence:

for Frijda there is a relevance (valency) signal and control precedence signal, and for Sloman there are pleasure and pain mechanisms and the insistence of motivators. Oatley and Johnson-Laird, however, conflate valency and control precedence: the 'happiness' and 'sadness' control signals are both valenced and grab attentional resources (section 9.4.1 argues that this is a mistake).

All these formulations leave a number of important questions unanswered. First, interrupt theorists do not describe control signal mechanisms. For example, the control signals of the COM theory have a well-defined function, but there is no detailed design specification of mechanisms that could perform those functions. It would be desirable to specify control signal mechanisms in detail similar to the specification and implementation of the AFP theory's insistence signal.

Second, there are no explanations as to why control signals are 'simple' or may vary in intensity, other than assuming their simplicity is due to primitive or older origins. More precisely, the deduction from requirements for agents to designs for agents that include valency control signals, which can vary in intensity and can be negative or positive, is missing. Neither are there explanations as to why non-intentional control signals are present in natural agent architectures in addition to semantic representations.

The AFP theory has yet to integrate valency, and more generally hedonic tone, with its existing theoretical concepts. The insistence of a motivator has no causal link to valency. This lack makes it difficult to account for valenced perturbant states, that is, perturbances that include a valenced component. For example, grief and triumph, while both perturbant states, differ in valency. As currently constituted the AFP theory could account for this difference only in terms of further explicit evaluations. Currently, it does not include the pleasure of success or the pain of loss.

6.5 Summary: what needs explaining

This section has reviewed two belief-desire, design-based, interrupt theories of emotion, Frijda's concern realisation theory and Oatley and Johnson-Laird's communicative theory. A number of limitations of these theories, and Simon and Sloman's interrupt theories, were identified.

The first limitation is that interrupt theories encounter difficulties explaining the control precedence of longer term emotional episodes, such as grief (section 6.4.1). This lack is the *control precedence problem.*

The second limitation of interrupt theories is that they do not integrate learning and emotions (section 6.4.2). This lack is the *emotional learning problem*.

The third limitation of interrupt theories is that they do not provide mechanisms for hedonic control signals, nor do they explain why such signals are simple, why they differ from semantic representations, and why they are either positive or negative (section 6.4.3). This lack is the *hedonic tone problem*.

This problem is closely linked to the AFP theory's inability to explain valenced perturbant states. This will be called the *valenced perturbant states problem*.

Part III of this thesis describes an extension to the AFP theory that resolves the emotional learning, hedonic tone and valenced perturbant states problems.

Part III

Valenced perturbances

Chapter 7

Forms of value in adaptive agent architectures

> 'The training of the human child depends largely on a system of rewards and punishments, and this suggests that it ought to be possible to carry through the organising with only two interfering inputs, one for "pleasure" or "reward" (R) and the other for "pain" or "punishment" (P).'
> Alan Turing, quoted in (Muggleton, 1994).
>
> 'Rewards and punishments are the lowest form of education.'
> Chuang-Tzu

This chapter examines adaptive agent architectures and how they support concepts of 'value'. Understanding the importance of forms of value in agency is a preliminary to addressing the problems of interrupt theories of emotion identified in chapter 6.

7.1 The requirement for trial and error learning

In many niches it is impractical to anticipate at design-time all the possible contingencies an agent may meet. For example, a designer of a foraging robot may not know in advance what kind of energy sources will be available in the environment. Therefore, the designer should construct an agent that tests possible energy sources, evaluates the consequences on energy levels, and recalls which sources are beneficial and which are neutral or harmful. An agent with this ability can learn new facts about the environment and, in future situations, use this knowledge to guide and improve its behaviour. An agent that can adapt itself at run-time is more flexible than an agent with fixed, predefined responses. Many natural agents learn during their lifetimes, people being the best known examples.

There are many possible mechanisms of run-time adaptation, some more complex than others. For example, MINDER1 'learns' in a very limited way [1]: it may discover where the recharge point is, and henceforth know where to go when a minibot needs recharging. MINDER1 can be described as having learned because its behaviour has improved: if MINDER1 did not know the location of the recharge point, and had a motive to recharge a minibot, it would need to search for the recharger; but, due to the new knowledge, MINDER1 can go straight to it without wasting time. The relatively straightforward mechanism of acquiring new knowledge about a domain, such as perceiving the presence of the recharge point, can improve MINDER1's behaviour. Other forms of learning may be performed by more complex mechanisms.

The requirement for run-time adaptation is very general and inspecific, and can be notionally satisfied by an agent, such as MINDER1, that acquires new information. However, MINDER1 does not try out new actions, the results of which are unknown or uncertain, evaluate the results of those actions with respect to its goals, and learn to repeat useful actions and not repeat useless actions, that is, MINDER1 does not perform trial and error learning. It is this specific kind of run-time adaptation that the requirements in section 4.1 ignored. This chapter investigates some of the design consequences of meeting a requirement for trial and error learning.

7.2 Selective systems are inductive reasoners

This section describes the generic selective system that has the minimal design elements necessary to meet a requirement for trial and error learning.

7.2.1 Induction

Trial and error learning under uncertainty is a type of inductive reasoning. Inductive reasoning, unlike formal deductive reasoning, involves the continual revision and augmentation of a knowledge base through interaction with an environment. For example, borrowing the terminology of formal systems, if a reasoner A has a consistent, but not necessarily complete, formal theory T of a domain D, then a process of deductive reasoning involves proving a formula F from the set of axioms and

[1] Stan Franklin first made this observation when watching a demonstration run of MINDER1.

inference rules that constitute T. Induction, however, involves (i) assuming a formula F, undecidable in T, (ii) checking the truth value of F in D by interaction with D, and finally (iii) producing a new consistent and sound theory T', constructed from T, in which F is decidable. Some inductive learning, however, produces probabilistic conclusions; for example, T' may be only probabilistically sound. Deductive reasoners are closed systems, whereas inductive reasoners are open systems, continually generating deductively unjustifiable hypotheses or guesses that can then be tested for consistency with current knowledge or tested for truth or falsity through practical interaction with a domain. Often guesses are generated by probabilistic reasoning over the current knowledge base. In other words, induction is a process that increases knowledge, unlike deduction that calculates the logical consequences of current knowledge:

> Whenever reasoning occurs, one can ask: does the conclusion contain more semantic information than the premises? More precisely, does it rule out some additional state of affairs over and above those ruled out by the premises? If not, then the inference is a valid deduction: its conclusion is true in any situation in which the premises are true. But, if the conclusion does rule out additional state of affairs, it is not valid. *Induction* can be defined as a systematic way of reasoning that increases the given information.
>
> P. Johnson-Laird, *The Computer and the Mind*, (Johnson-Laird, 1988), p. 219.

However, an inductive step can be validated, not by logical deduction, but by empirical checking. Validation may show that the inductive step is of notional utility, but validation is partial and always open to refutation (e.g., see (Popper, 1963)).

Trial and error learning, therefore, can provide new knowledge about an environment, allowing run-time adaptation to actual conditions rather than design-time preadaption to predicted conditions. However, trial and error learners, unlike the reasoner A, normally have goals that they pursue in the given domain or environment (i.e., they are control systems of more or less complexity). Therefore, the hypotheses or guesses that they generate can be control hypotheses or propositions concerning the efficacy of actions in particular contexts to satisfy goals.

7.2.2 The relation *is_better_than*

Consider a simple negative feedback control system. It has substates with different functional roles, in particular 'belief-like' and 'desire-like' substates. For example, the belief-like substate of a thermostat is the curvature of its bi-metallic strip, which alters in accordance with the ambient temperature of a room; the desire-like substate is the setting of the control knob. Negative feedback ensures that the temperature of the room stabilises around the control knob setting: the thermostat 'acts' in the world to achieve its 'desire'. Of course, a thermostat does not have sufficient architectural complexity required to support human beliefs and desires, but it is an illustrative 'limiting case' (Sloman, 1993b) of a system that controls its environment with reference to an internal norm.

A new feature and a new requirement can be imposed on the simple thermostat. The new feature is the effect of the thermostat's 'actions' are now *uncertain*; that is, from the perspective of the thermostat designer the output signal has (initially) unknown effects on the temperature of the room. The new requirement is trial and error learning, that is, the thermostat should learn the most efficacious output signals for controlling temperature. A change in requirements is a movement in niche-space. The new type of thermostat will need to try out actions, monitor the effects of those actions, evaluate the results, and then select and retain those actions that work best. This is the corresponding movement in design-space.

The design elements of an adaptive thermostat performing trial and error learning correspond to a *selective system* (Pepper, 1958), which is the abstract schema for an adaptive system (Cziko, 1995). A selective system has three components: (i) a *trial generator*, which is any mechanism that produces and uses a variety of functions to generate outputs for particular inputs, (ii) an *evaluator*, which is a mechanism that evaluates the results of using particular functions to generate trials, where evaluation occurs through comparison to a norm, such as success in bringing the temperature closer to the desired setting, and (iii) a process of *selection*, which selects those functions associated with 'good' evaluations for further use, while deselecting others. Selective systems implement the well-known generate, test, and select cycle. Specific examples of selective system improve their behaviour over time (cf. natural selection, genetic algorithms, classifier systems (described in section 7.4.3), neural networks, and so forth). Note that the three components of the selective system directly implement the three steps of inductive reasoning described in the previous section (section 7.2.1). Trial generation is the production of new, undecidable formulae, evaluation is interaction with a domain to determine the truth or falsity of formulae, and selection is the integration of new formulae and their associated truth values with existing theory. In other words selective systems are inductive reasoners.

Many possible designs could meet the specification of a selective system, yet all such designs evaluate their trials. In the adaptive thermostat example, if the effector signal $trial_1$ brings the room temperature closer to the norm, whereas the signal $trial_2$ does not, then the function that produced $trial_1$ *is_better_than* the function that produced $trial_2$ with respect to the norm of controlling temperature. Selective systems impose an ordering over the internal functions that produce behavioural trials. An example at a different time scale and physical extension is natural selection. The evolutionary process imposes an ordering over individuals: if individual i_1 produces more offspring than individual i_2 then i_1 *is_better_than* i_2 with respect to the norm of reproduction (i.e., i_1's genes will be used to produce further individuals, or trials, in the future). All selective cycles induce knowledge to order trials in terms of their efficacy for satisfying the system's internal norms.

7.3 Natural reinforcement learners

Animals and humans learn through trial and error. Reinforcement theory – the theory that those behaviours that lead to reinforcing consequences are maintained – is a narrow theory of learning arising from laboratory experimentation with animals, particularly rats (for a full review of its methods, theories and results see (Mackintosh, 1983)). It is a behaviouristic theory; that is, it avoids assuming internal mechanisms or states that mediate behaviour. However, simplified models, based on operationally defined variables, have been proposed. A number of its concepts, first formulated by B. F. Skinner in the thirties, are relevant to trial and error learning, particularly the concept of a *reinforcer*.

Animal responses that are elicited by a stimulus are respondents. This category includes such things as built-in reflexes, classically conditioned responses (e.g., the salivation of Pavlov's dogs) and action patterns evoked by releasing stimuli (e.g., a cat pouncing on a mouse or substitute). Operants, however, are responses not elicited by stimuli but by the animal itself, voluntary rather than reflexive.

Skinner's *law of conditioning* states 'if the occurrence of an operant is followed by a presentation of a [positively] reinforcing stimulus, the strength of the operant is increased' (Mook, 1987). The *law of extinction* states 'if the occurrence of an operant already strengthened through conditioning is not followed by the reinforcing stimulus, the strength is decreased.' A reinforcing stimulus is a stimulus that strengthens the response that precedes it. [2] Strength is correlated to such empirical quantities as frequency of operant response, intensity of response, and so forth. Operants, although not elicited by external stimuli, can become to be controlled by them: for example, lever-pressing may be reinforced with food when a light is on, but not reinforced with food when the light is off. A rat will learn to press the lever only when the light is on. When this occurs the light has acquired stimulus control over the emission of the operant.

If a rat is provided with food after pressing a bar, and bar-pressing increases as a result, then the food is a positive reinforcer (i.e., it is something the animal will work to obtain). If bar-pressing serves to turn off a painful shock, then shock is a negative reinforcer (i.e., it is something the animal will work to be rid of). Punishment is the application of a negative reinforcer dependent on the occurrence of a response, and therefore differs from negative reinforcement.

These are (apparently) simple definitions. No commitment is made to the information processing capabilities and cognitive architecture of animals. All the factors are empirically observable and measurable. Of course, there are many problems with the behaviourist approach and concepts, and there is an extensive literature on this. As an example, it is difficult to sustain a distinction between voluntary and reflexive behaviour by observation alone.

Experiments with rats in the early fifties demonstrated the existence of reinforcement 'centres' in the brain (including such areas as the lateral hypothalamus, brainstem and forebrain) (Mook, 1987). An electrode can be directly inserted into a rat's brain. When the rat presses a lever a small current is passed through the electrode. If the electrode is in the correct place the rat will press the lever again and again, for hours at a time. Conversely, aversive 'centres' have been discovered too: a rat will press a lever to prevent the stimulation of such areas. Similar effects occur in humans. Electrical stimulation of the brain of conscious patients can evoke feelings of pleasure. This and other experiments has led researchers to posit reward and aversion systems within animal brains that affect hedonic responses, that is, responses of pleasure and displeasure, or approach and withdrawal, to external stimuli. If the eating of food, the drinking of sweet solutions, and ejaculation are causally linked to rewarding 'centres' of the brain then it is possible to move from the behaviourist notion of a reinforcer as an external stimulus to the cognitivist no-

[2] Consequently, the law of conditioning is not a law but a definition of a reinforcing stimulus.

tion of the existence of internal rewarding mechanisms that, when their conditions are met, send 'signals' that tend to 'strengthen' antecedent acts, all other things being equal.

This brief description has ignored important complications such as satiation, habituation, and the relationship between reinforcement learning and other types of learning.

7.3.1 Reinforcement as a selective system

Staddon and Simmelhag (1971; discussed in (Mook, 1987)) postulate *principles of variation*, which determine what subset of action responses from an animal's repertoire may occur in certain contexts. The initial repertoire of actions can be genetically determined and/or acquired through previous learning experiences. When one of these possible responses is required to, say, produce food (such as a pigeon pecking a coloured light), reinforcement will operate as a *principle of selection*, selecting the correct response from the subset of responses performed in the situation. For example, a pigeon might flap its wings, turn in circles, peck, or preen within the experimental situation. A reinforcement learning schedule can select one of these responses by associating it with the presentation and consumption of food.

There is a strong analogy to natural selection: forms of behaviour undergo extinction if they are poorly adapted to the environment, leaving well-adapted, successful forms of behaviour. The animal adapts over its lifetime and a species adapts over many generations by the same process – trying various things (generating trials), and discarding what doesn't work and keeping what does (selection).

7.4 Artificial reinforcement learners

This section examines reinforcement learning architectures developed by AI researchers to meet a requirement for trial and error learning. They are all examples of selective systems, and all exhibit *is_better_than* relations that order internal components with respect to norms.

7.4.1 Reinforcement learning algorithms

The study of reinforcement learning (RL) algorithms is an active area of research (Kaelbling, Littman & Moore, 1995). RL algorithms are selective systems as defined above. RL is a type of trial and error learning, and holds out the promise of programming control programs for agents by reward and punishment without the need to specify how a task is to be achieved.

> Reinforcement Learning (RL) is a class of problems in which an autonomous agent acting in a given environment improves its behaviour by progressively maximising a function calculated just on the basis of a succession of scalar responses (rewards or punishments) received from the environment. No complementary guidance is provided for helping the exploration/exploitation of the problem space, and therefore the agent can rely only on a trial-and-error strategy. (Dorigo & Bersini, 1994)

Kaelbling, Littman & Moore (1995) provide a standard description of the RL situation.

> ... an agent is connected to its environment via perception and action ... On each step of interaction the agent receives as input, i, some indication of the current state, s, of the environment; the agent then chooses an action, a, to generate as output. The action changes the state of the environment, and the value of this state transition is communicated to the agent through a scalar *reinforcement signal*, r. The agent's behaviour, B, should choose actions that tend to increase the long-run sum of values of the reinforcement signal. It can learn to do this over time by systematic trial and error, guided by a wide variety of algorithms ... (Kaelbling, Littman & Moore, 1995)

Figure 7.1 is an example dialogue taken from (Kaelbling, Littman & Moore, 1995) that provides an intuitive way to understand the relation between the agent and its environment.

More precisely, RL involves discovering functions defined on the state and action space of a task, driven by a real-valued reinforcement signal, that maximise some long term measure of reinforcement. The details of how this is achieved depends on the particular function representation used. In the abstract, RL involves constructing a function, called the policy function, $f : X \longrightarrow A$, where X is the set of possible learner-environment states and A is the set of possible actions the learner can perform, such that the cumulative reinforcement r over time t, $\sum_{t=1}^{\infty} r_t$, is maximised. Reinforcement is derived from conditions that define the utility of particular states, and can be thought of as the abstract requirements of the task.

Environment:	You are in state 65.
	You have 4 possible actions.
Agent:	I'll take action 2.
Environment:	You received a reinforcement of 7 units.
	You are now in state 15.
	You have 2 possible actions.
Agent:	I'll take action 1.
Environment:	You received a reinforcement of 4 units.
	You are now in state 65.
	You have 4 possible actions.
Agent:	I'll take action 2.
Environment:	You received a reinforcement of 7 units.
	You are now in state 44.
	You have 5 possible actions.
...	
...	

Figure 7.1: **The reinforcement learning situation**

Conditions of reinforcement are the norms of the RL selective system. For example, taking action $a_i \in A$ in state $x_i \in X$ may have a reinforcement of r. The set of all conditions of reinforcement is normally called the 'payoff' function. In many implementations f is implemented as a set of candidate functions (e.g., $f(x_i) \longrightarrow a_j$, where $x_i \in X$ and $a_j \in A$), each of which has an associated value that determines the likelihood of it being selected for use and determining behaviour. The associated value implements an *is_better_than* relation over the candidate functions; for example, if $value(f_i(x_i)) > value(f_j(x_i))$ then f_i *is_better_than* f_j for controlling actions in state $x_i \in X$. In many RL problems function values change dynamically until they stabilise (if the payoff function is unchanging). When this has occurred the *is_better_than* relation has become an *is_best* relation: the RL learner *is* maximising cumulative reward.

The main design problem to be solved in RL is the *credit assignment* problem, which is the problem of 'properly assigning credit or blame for overall outcomes to each of the learning system's internal decisions that contributed to those outcomes' (R. S. Sutton, quoted in (Cichosz, 1994)). For example, if the RL agent decided to take action 2 in state 65 and received a large reward on the next time step, then credit assignment would involve recording this fact, such that the same action would be taken again in the same situation in the future, all other things being equal. Often there may be long and uncertain delays between a decision and its outcome.

7.4.2 Q-learning: Watkins

Q-learning (Watkins & Dayan, 1992) is a much studied reinforcement learning algorithm. A Q-learner maintains a reward prediction P for each state-action combination of the policy function. In Q-learning work, P is referred to as the quality of the state-action pair and is written $Q(x, a)$. Quality values are stored in a lookup table. The table stores all the possible state to action pairs and associated quality values for the particular task. The Q-learning algorithm selects that action a with the highest Q value for the particular current state x. The following update rule is then used to alter the quality value of the chosen action:

$$Q(s,a) := Q(s,a) + \alpha(r + \gamma maxQ(s^{'},a^{'}) - Q(s,a))$$

where,

(s, a, r, s') is a single transition in the environment, such that
s is the agent's state before the transition,
a is its choice of action,
r the instantaneous reward it receives, and
$s^{'}$ its resulting state.
$maxQ(s^{'},a^{'})$ is the maximum Q value from the state-action pairs
that specify an action in state $s^{'}$.
α is the learning rate, such that $(0 < \alpha < 1)$.
Normally α is slowly decreased during learning.
γ is a discount factor, such that $(0 < \gamma \leq 1)$.
For example, if $\gamma = 1$ then actions are taken that
tend to maximise delayed reward,
whereas if $\gamma = 0$ then actions are taken that
tend to maximise immediate reward.

For example, if $Q(65, 2) = 5$, the new state is 15, the reward received is 7 and $maxQ(15, a) = 6$ (see figure 7.1), then the updated Q value is $Q(65, 2) := 5+\alpha(7+\gamma 6-5)$. So, if the learning rate $\alpha = 0.5$ and immediate reward is maximised, $\gamma = 0$, then $Q(65, 2) = 6$. In other words, the Q value is increased, making it more likely that action 2 will be executed in state 65 in the future, assuming that other $Q(65, a)$ do not also increase.

Mathematical proofs show that, in certain environments, Q-Learning converges to an optimal policy (Watkins & Dayan, 1992), although it may take a long time. However, the inability of the basic Q-Learning to generalise over the state and action space of the task is a serious weakness. For large state and action spaces the number of table entries and associated trial and error search can easily become unmanageable.

Q values specify an *is_better_than* relation over the state-action pairs of the policy function. For example, if $Q(s_i, a_j) > Q(s_i, a_k)$ then action a_j *is_better_than* action a_k in circumstances s_i for maximising cumulative reward.

7.4.3 Adaptive rule-based systems

Rule-based systems have a readable and writeable global workspace or blackboard that contains information accessible to a set of rules. A rule is composed of conditions of activation that may match information in the workspace and actions that may change the workspace by adding or deleting information. A rule may fire if its conditions hold in the workspace. Information processing occurs through the continual firing of applicable rules. Rule-based systems normally do not alter their rules at run-time. J. Holland, dissatisfied with the 'brittleness' of existing rule-based systems (Holland, 1986), developed the classifier system in the early 1970's. It specifies a class of rule-based systems that adaptively alter their rules at run-time through interaction with a domain. Classifier systems are reinforcement learners.

7.4.3.1 The classifier system: Holland

Holland's classifier system (Holland, 1986; Holland, 1995; Holland, 1975; Holland, Holyoak, Nisbett & Thagard, 1986; Riolo, 1988) is a RL architecture. It consists of a *performance system*, and *credit-assignment* and *rule discovery* algorithms.

The performance system consists of a *classifier list* that consists of a set of condition-action rules called *classifiers*, a *message list* that holds current messages, an *input interface* that provides the classifier system with information about its environment in the required form, and an *output interface* that translates action messages into events in the environment. The *basic cycle* of the performance system matches messages in the message list (including sensory messages) with classifiers, which then post their actions 'back' to the message list. Many classifiers may become active and fire in parallel. Any current action messages are sent to the output interface.

Each classifier has the form

$$\text{cond}_1, \text{cond}_2, ..., \text{cond}_n \ / \ \text{action}$$

where each cond_n and action is a string built from the alphabet {0, 1, #}. A cond_n can be optionally prefixed by a '~' (a 'not-sign' character). The # is a wildcard symbol that facilitates pattern matching as follows: condition '10##' matches message '1010' and '1001' but not message '1110'. A condition string prefixed by the 'not-sign' is matched when no message matches the condition. Classifiers match messages in the message list. If a match occurs then the action part of a classifier is posted to the message list. The # symbol in an action has the semantics 'follow through', i.e. the # is replaced by the corresponding symbol from the matching message.

Credit assignment (RL) is achieved via a *bucket-brigade* (*bb*) algorithm. This algorithm introduces competition between classifiers based on a quantitative 'strength'. Each classifier that has its condition activated by a message *bids* to post its action part to the message list. Only the top subset of highest bidders on each cycle are allowed to post their actions. The bid of a classifier depends on its strength, which is a measure of the classifier's 'usefulness' to the system. The higher the strength of a classifier the more likely it will win the competitive bidding round and post a message.

The strength of classifiers specifies a probabilistic partial ordering on the classifier list. The ordering is partial because only some classifiers will bid for the same message. The ordering is probabilistic because the classifier selection mechanism is stochastic. For example, both classifiers c_i and c_j match message m; c_i has strength s_i, and c_j has strength s_j, with $s_i > s_j$. In competitive bidding for message m, c_i will out bid c_j more often than not: there is a probabilistic total ordering on the set $C = \{c_i, c_j\}$ (another example of the *is_better_than* relation). Other classifiers in the classifier list, such as c_k, never compete against any $c \in C$; consequently, no ordering holds between them.

The behaviour of the classifier system, therefore, can be modified by changing the strengths associated with individual classifiers. If the strength of the classifiers that tend to lead to 'useful' behaviour can be increased, and the strength of the classifiers that tend to lead to 'useless' behaviour can be decreased, the system will learn to produce more useful behaviour. The *bb* is designed to bring about these types of changes in strength.

The basis for the *bb* is information from *reinforcement mechanisms* about whether the classifier system as a whole is behaving correctly (as defined by the designer, or, if complete classifier systems are evolved in a simulated domain, by a niche fitness function). This information derives from *rewards*; that is, the system will receive positive reward when it behaves correctly and negative reward when it behaves incorrectly. When a reward is received the *bb* adds the reward value to the strength of all classifiers currently active, thereby changing the strength of classifiers directly associated in time with useful behaviour. Also, when a classifier is activated it 'pays' the amount it bid to the antecedent classifier that produced the message it matched. The strength of the active classifier is decreased by its bid amount. In this way, the *bb* acts to increase the strength of classifiers indirectly involved in the production of useful behaviour (e.g., a recently rewarded classifier will pay the same proportion but a higher amount of its strength to antecedent classifiers in subsequent cycles). The *bb* allows reward to 'circulate back' through the system. Chains of high strength classifiers, performing useful computation, can emerge from such a scheme.

The *bb* can lead to improved system behaviour through the selection of some classifiers over others; however, it cannot create entirely new classifiers. The system, therefore, needs an additional learning algorithm that can *discover* new classifiers. One can envisage many possible types of discovery algorithm. Holland chose the genetic algorithm (Holland, 1975), which is based on Darwinian natural selection. Genetic algorithms are widely used and well-known, so their operation will be only briefly summarised here.

First, a subset of the classifier list is selected to act as parents. Selection is based on classifier strength, which is an estimate of the 'usefulness' or 'fitness' of the classifier. Second, copies are made of the selected parent classifiers and *genetic operators* applied to those copies to modify them. The *crossover* genetic operator exchanges one portion of one classifier with a portion of another (as per chromosome crossover in sexual reproduction). The *mutation* operator probabilistically alters a portion (usually a single character) of an offspring classifier (as per random mutation in nature).

New classifiers generated by the GA do not usually replace their parents, since parents tend to be high-strength and classifiers that are replaced tend to be low-strength. Hence, new classifiers can be tried without losing the expertise accumulated in their parents. Normally, the GA is applied less frequently than the *bb* otherwise new classifiers will not have had sufficient time to be evaluated.

Classifier systems have been extensively used in the Artificial Life (ALife) community (Steels, 1994), for example in developing control programs for robots through supervised learning (Dorigo & Colombetti, 1993) and learning paths through mazes (Donnart & Meyer, 1994). However, a classifier system is limited in many ways. It does not have an explicit memory store. It tends to be an entirely reactive system with no representation of goals. It does not anticipate, or perform prior search within a world model before acting. In real-world applications it can be difficult for a classifier system to learn appropriate behaviours (Wilson & Goldberg, 1989).

7.4.3.2 XCS: Wilson

XCS (Wilson, 1995) is a development of Wilson's earlier ZCS classifier system (Wilson, 1994), which is a 'minimalist' version of Holland's classifier system. In contrast to traditional classifier systems, XCS bases the fitness of a classifier on the accuracy of its predicted payoff. Fitness is the tendency of a classifier to be selected as a parent for the genetic algorithm during rule discovery. For example, a classifier that accurately predicts that it will receive low reward is a fit individual for reproduction, whereas a classifier that inaccurately predicts that it will receive a high reward is unfit for reproduction. Traditional classifier systems base classifier fitness on strength, which is used as a predictor of the expected payoff when selected. The change from using strength to using accuracy was motivated by a number of reasons and leads to a number of improvements in classifier system performance (see (Kovacs, 1996) for an extended review and analysis of the performance of XCS classifier systems).

The first reason for using accuracy based fitness is that differing, yet useful, behaviours may have different payoff levels, both in terms of absolute magnitude and frequency of reward. For example, avoiding a predator may earn a payoff of 10 and eating food may earn a payoff of 5. Both are useful behaviours and a classifier system should learn, through reinforcement, to perform both. However, basing classifier fitness on strength skews rule discovery in favour of exploring predator avoidance rules rather than eating rules, as the former behaviour yields rules of higher strength. Accuracy of payoff, in contrast, is invariant over payoff level. For example, a predator avoidance rule may predict a payoff of 10 with an accuracy of 0.8, and an eating food rule may predict a payoff of 5 with an accuracy of 0.8; hence, both rules have an equal chance of being used in rule discovery. This avoids classifiers associated with high payoffs 'taking over' the classifier list and preventing rule discovery in those areas of the problem space associated with lower payoffs.

Second, traditional classifier systems only attempt to find the most rewarding classifiers. Other RL algorithms, such as Q-learning, have complete maps of the payoff environment. A complete map of the payoff function is beneficial for action selection as a greater range of alternative predictions can be made in a particular situation. Basing fitness on accuracy allows XCS to construct complete payoff maps because rule discovery occurs over the whole state and action space of the task (i.e., rules that accurately predict low or no reward are also fit for rule discovery).

Third, traditional classifier systems do not discover accurate generalisations in classifier conditions. Accuracy allows XCS to evolve maximally general classifiers, those that could not have any more #'s without becoming inaccurate.

A number of other researchers have incorporated accuracy information in their classifier systems (for an overview see (Wilson, 1995)). Holland, in the original paper on classifier systems (Holland, 1986), suggested that accuracy information be incorporated in calculating classifier fitness, but did not develop the idea.

In XCS the traditional bucket-brigade has been replaced with a learning algorithm similar to Q-learning. The Q-learning rule, in combination with accuracy based fitness, allows the construction of more complete mappings of the problem space compared with the mappings produced by traditional classifier systems using the bucket-brigade. In addition to using accuracy and a Q-learning update rule, there are more subtle differences between traditional classifier systems and XCS systems that are not discussed here.

Traditional classifier systems probabilistically select classifiers based on their bid, which is proportional to the classifier's strength. In XCS the situation is more complex. A *match set* [M] is formed from those classifiers in the general population [P] which match the system's current input. Next, a *system prediction* $\mathrm{P}(a_i)$ is computed for each action a_i in [M] using a fitness-weighted average of the predicted payoffs of classifiers advocating a_i. The system prediction for each advocated action is placed in a *prediction array* in preparation for action selection. The system next selects an action from the prediction array and forms an *action set* [A] of classifiers in [M] advocating the selected action. Action selection may then occur in any of a number of ways. The chosen action is then sent to the output interface and an environmental reward returned. Therefore, in XCS the *is_better_than* relation is a composite of accuracy and predicted pay-off. XCS selects those actions that have a higher certainty of maximising payoff.

7.4.4 Dyna: Sutton

The Dyna architecture (Sutton, 1991) is an agent architecture that attempts to integrate trial and error learning, planning and reactive execution. Dyna uses reinforcement learning to induce an optimal reactive policy (the mapping from states to actions). It also constructs an action model of the domain that takes as input a domain state and an agent action and outputs a prediction of the resulting domain state (i.e., Dyna models the results of its own actions). Dyna has planning capabilities: it can use its action model to try out actions 'in its head'; therefore, Dyna attempts to discover an optimal policy via trial and error within a world model rather than the environment. The planning capability does not intervene between perceiving an environment state and producing an action; in other words, responses are performed reactively. Finally, all these processes occur incrementally and in parallel: Dyna can react, be reinforced, build an action model, and plan[3] all at the same time. Figure 7.2 (taken from (Sutton, 1991)) is pseudo-code for the generic algorithm driving Dyna.

Dyna makes no commitments to the implementation details of the action model. Neither does it specify how hypothetical world states and actions are to be chosen for model-based, rather than action-based, reinforcement learning. A number of exploration strategies could be devised. In order to maintain an accurate action model Dyna must sometimes try actions that are inferior with regard to maximising cumulative reward (e.g., if Dyna chose only the best actions then it could not discover changes to the world, such as a previously unrewarding response now having highly rewarding consequences). Normally, Dyna's RL component is a Q-learner. Typically, it requires more computation than Q-learning per cycle (depending on the value of K), but requires less interaction with the environment than Q-learning (Kaelbling, Littman & Moore, 1995). Dyna incrementally deduces an optimal policy through hypothetical exploration of a world model *while* actually exploring the 'real' world.

[3]Normally planning means the production of an ordered list of primitive actions that transform an initial state into a goal state. Planning in Dyna consists of performing hypothetical actions in the action model to modify the policy function such that it maximises cumulative reward. However, the action model could be used to produce an ordered list of actions that move from an initial state to a goal state.

The reward received during planning is based on rewards received during action execution. In early stages of learning, Dyna's action model is likely to be inaccurate. Hence, hypothetical RL will be similarly inaccurate. Over time, however, the world model will increase in accuracy until hypothetical RL closely approximates actual RL.

REPEAT FOREVER

1. Observe the world's state and reactively choose an action based on it;
2. Observe resultant reward and new state;
3. Apply reinforcement learning to this experience;
4. Update action model based on this experience;
5. Repeat K times
   (a) Choose a hypothetical world state and action;
   (b) Predict resultant reward and new state using action model;
   (c) Apply reinforcement learning to this hypothetical experience.

Figure 7.2: **The Dyna algorithm**

If the RL component of Dyna is a Q-learner the *is_better_than* relation is specified by the Q values of the policy function's state-action pairs.

7.4.5 Limitations of reinforcement learning algorithms

The artificial trial and error learners reviewed in this section are relatively simple systems. For example, they normally rely on the Markov assumption, which states that the environment state is fully observable and environment state transitions are independent of any previous states or agent actions (Kaelbling, Littman & Moore, 1995). For example, in a non-stochastic environment, if two actions, a_1 and a_2, are possible in state s_1, and performing a_1 results in state s_2 and a_2 results in state s_3, then if the agent revisits state s_1 and the Markov assumption holds, the state transitions will be identical. This is a strong assumption that does not hold in most domains of interest. For example, if pressing a button five states previously alters the state transitions of s_1 such that $a_1 \rightarrow s_3$ and $a_2 \rightarrow s_2$ then the Markov assumption is violated. An equivalent problem is the problem of 'hidden state', which occurs when environment states 'hidden' from the RL algorithm have causal effects on current actions (pressing a button five states previously is not represented in the current input from the environment, yet changes the effect of current actions). Solutions to the problem of hidden state involve memory-based methods, for example (McCallum, 1996), which describes a RL algorithm that can reason over an interaction history to determine which previous states affect possible future states (imagine deducing the connection between a prior button press and changed state transitions in state s_1).

RL algorithms do not scale well to larger problems. For practical applications, RL algorithms are given prior domain knowledge or bias to reduce the trial and error search required. For example, the technique of shaping, used for training animals, can be used. A teacher presents the RL algorithm with initially simple problems, then gradually exposes it to more complex problems. Another option is to provide local reinforcement signals which provide the learner with a gradient of reward towards a final goal. This technique can significantly speed up the learning process when compared to providing a single, high reward when the goal is accomplished. Other techniques are imitation (watching another agent perform tasks), and 'reflexes' (built-in action responses to particular states that force the learner to explore 'correct' parts of the problem space). There are also problems of controlling explore and exploit strategies, but this is not discussed here.

7.5 The concept of value

A requirement for trial and error learning can be met by a selective system. Both natural and artificial reinforcement learning implement selective systems. The definition of a selective system is sufficiently abstract to include a wide class of systems. However, all such systems support an *is_better_than* relation that orders the internal functions that produce trials. For example, Q-learning, classifier systems, and Dyna implement the *is_better_than* relation as a scalar quantity directly associated with internal functions.

This section argues that concepts of 'value' derive from relations of 'better' and that trial and error learners, in addition to supporting concepts of 'belief' and 'desire' also support concepts of 'value'.

7.5.1 'Better' and 'value'

Stating that P is better than Q involves evaluating their qualities with respect to a basis of comparison, B (Sloman, 1969). For example, a chainsaw is better than an axe for cutting down trees. In other words, 'better' is logically related to a goal or norm. Comparison of things with respect to a norm can be qualified. For example, cars are better than buses for getting to work quickly except when the roads are very busy. Similarly, one classifier rule may be better than another for maximising cumulative reward in certain situations (see section 7.4.3). It would require extensive conceptual analysis to uncover the logical relations between concepts like ‘better’, ‘good’, ‘ought’, and ‘value’, and such an endeavour is outside this author’s competence; instead, the relation between ‘better’ and ‘value’ will be merely sketched.

If a woodcutter has a goal of felling trees then a chainsaw has more *value* to him than an axe because the chainsaw is better with respect to felling trees. Similarly, if a reinforcement learner has a goal of maximising cumulative reward then the state-action function $f_1(s_i, a_j)$ has more *value* to it than the function $f_2(s_i, a_k)$ because f_1 is better with respect to maximising reward. One difference between the two cases is that axes and chainsaws are external tools whereas behaviour producing functions are internal substates. However, in both cases the concept of ‘value’ serves to denote a relation between a goal-directed system and a possible means to an end.

There are at least two different uses of the word ‘value’, one of *prizing* and one of *appraisal*:

> ... in ordinary speech the words ‘valuing’ and ‘valuation’ are verbally employed to designate both *prizing*, in the sense of holding precious, dear (and various other nearly equivalent activities, like honoring, regarding highly), and *appraising* in the sense of *putting* a value upon, *assigning* value to. This is an activity of rating, an act that involves comparison, as is explicit, for example, in appraisals in money terms of goods and services. The double meaning is significant because there is implicit in it one of the basic issues regarding valuation. (...) Valuation as *appraisal* (...) is primarily concerned with a relational property of objects so that an intellectual aspect is uppermost of the same general sort that is found in *‘estimate’* as distinguished from the personal-emotional word *‘esteem’*.
>
> J. Dewey, 1939, *Theory of Valuation*. Chicago: The University of Chicago Press. Quoted in (Miceli & Castelfranchi, 1989).

The woodcutter prizes the chainsaw because it is a means to felling trees, and, by comparing it to the axe, he appraises or rates it more highly than the axe. The double meaning of ‘valuing’ can be readily understood by viewing ‘value’ as a derivative of the concept ‘better’. If P is better than Q with respect to B then P is rated or appraised more highly than Q; also, as P is more conducive to satisfying B than Q, and assuming that P and Q are the only available means, then P is prized as the means to satisfying B.

Developers of RL algorithms already incorporate folk psychological concepts of 'value' or 'goodness' in the descriptions of their goal-directed systems. For example, the Q in Q-learning stands for the 'quality' of a state-action function and represents its 'goodness' for maximising cumulative reward; similarly, the 'strength' of a classifier represents its 'fitness' or 'usefulness'. This is a legitimate use of language as the everyday concept of 'value' logically depends on a prior concept of 'better'. Something that is valued is useful as a means to an end, and appraised as better than other possible means. RL algorithms support concepts of 'value' because they implement *is_better_than* relations. An RL algorithm 'values' behaviour producing functions as means to ends, and 'appraises' them more or less highly compared to other possible functions. Using the term 'value' in descriptions of both natural and artificial situations avoids the need to create a new technical word with the same descriptive function as an extant word. Much as the terms 'belief' and 'desire' can describe functional aspects of very simple control systems, the term 'value' can describe functional aspects of very simple adaptive, trial and error controllers.

To distinguish between an agent valuing an external object that is a means to an end (such as the woodcutter prizing the chainsaw), and a trial and error learner valuing internal substates as means to an end (such as the reinforcement learner constructing quality values for state-action functions) the terms 'prizing' and 'appraising' will be reserved for the former case, and the term 'value' and 'evaluating' reserved for the latter case. In this thesis, 'value' is a relation between a goal-directed system and its own internal components.

Just as there is a great variety of belief-like and desire-like representations in control systems, *there will also be a great variety of value-like representations.* It is a highly abstract concept, and is as important as the information processing definitions of 'beliefs' and 'desires'. Unlike beliefs, which have a 'world-to-mind' direction of causality, and desires, which have a 'mind-to-world' direction of causality, value is internally relational, having 'mind-to-mind' causal interactions, and *refers*, in an impoverished sense, *to the utility of internal substates.* Value appears in design-space when a requirement for adaptivity is imposed. It orders internal components with reference to normative criteria within a selective system. [4]

[4]The analysis of 'value' presented here is influenced by S. C. Pepper's *The Sources of Value* (1958) (Pepper, 1958). Pepper illustrates how all kinds of selective system support the concept ((Wright, 1995) contains a brief summary). He defines a selective system as follows:

> A selective system is a structural process by which a unitary dynamic agency is channeled in such a way that it generates particular acts, dispositions, or objects (to be called 'trials'), and also activates a specific selective agency (to be called the

7.5.2 Scalar quantity representations of value

The *form* of value in RL algorithms is a scalar quantity. A scalar quantity, unlike a vector, is not decomposable into components with differing semantics. For example, the strength of a classifier rule is a number that is compared with other numbers. The strength quantity is never decomposed or decoded to a more complex structure, unlike, for example, binary representations of networks, Gödel numbering of statements in elementary number theory (Nagel & Newman, 1958), or numerical representations of n-dimensional vectors. Strength is only ever added to, subtracted from, or compared with.

Scalar representations of value do not represent anything within or external to the RL system; rather, value specifies an *internal is_better_than relation* between substates. It is this property of value that helps make RL algorithms *domain-independent* learning algorithms: their representation of 'better' need not alter from domain to domain.

There may be deeper mathematical reasons why a scalar quantity of value is useful in inductive learning systems, but a full analysis is beyond this author's competence. Clues can be found in (Goldfarb & Nigam, 1994), which argues that real number metrics, in addition to symbolic representations, are required in general for inductive learning. Also, connectionist learning systems normally use numerical representations and approximated continuous functions. Continuous functions have a derivative and, due to this property, calculus can be used to determine rules that manipulate weights to perform a gradient descent in an error space (Dawson & Shamanski, 1994). A deeper analysis would provide reasons why real numbers are important for inductive learning algorithms.

RL algorithms in general, and Q-learning, classifier systems, and Dyna in particular, are existence proofs of trial and error learners that use a scalar quantity form of value to meet a requirement for adaptivity. Unfortunately, the form or forms of value in natural reinforcement learners are unknown.

> 'norm') by which some of the trials are rejected and others are incorporated into the dynamic operation of the system.
>
> A natural norm generates values in the sense that it evaluates something by its selective action ... A natural norm is also a fact – another sort of value fact, a normative fact – in that it performs a normative function ...
>
> S. C. Pepper, 1958, *The Sources of Value*, p667 – 668 (Pepper, 1958).

7.5.3 Value as a control signal

In sections 6.3.2 and 6.4.3.5 Oatley and Johnson-Laird's distinction between semantic and control signalling was introduced. Semantic signals are intentional and refer, whereas control signals are non-intentional and perform a control function, such as changing the 'control flow' of the system.

Value is a control signal, although a detailed analysis of its function reveals the inadequacy of the concepts 'semantic', 'control' and 'intentional'. First, unlike 'belief-like' and 'desire-like' substates, value is non-intentional and does not refer (although this will be qualified in a moment). Consider a classifier system: an internal message may represent a state of the environment, and is an example of a belief, and a reinforcement mechanism may define conditions for reward, and is an example of a goal or desire. However, the strength of a classifier, which is a form of value, does not represent a state of the environment to be believed or achieved, but specifies an *is_better_than* relation over the classifier list (section 7.4.3). However, in another sense, value *is* intentional and represents the 'goodness' or 'badness' of a classifier for meeting conditions for reward and maximising cumulative reward (i.e., it is a measure of utility). Value, therefore, confounds naive formulations of intentionality. However, for shorthand, value will be described as 'non-intentional' to denote the fact that it does not refer in the way belief-like and desire-like substates do, but specifies a relation over internal system components.

Value performs a control function. For example, in Q-learning, if $Q(s_i, a_j) > Q(s_i, a_k)$ then action a_j will be chosen for execution in state s_i. In other words, control will pass to the state-action pair that advocates a_j. If values change through further reinforcement learning and $Q(s_i, a_j) < Q(s_i, a_k)$ then action a_k will now be chosen. The change in value has altered the control flow of the system. Note, however, that beliefs and desires also have control functions. For example, MINDER1 may acquire a new belief that may generate a motive that interrupts and changes current processing. To distinguish these cases the term 'control signalling' will be reserved for information that is both non-intentional (in the qualified sense defined above) and has a control function. The scalar quantity form of value in RL algorithms conforms to Oatley and Johnson-Laird's definition of 'control signalling'.

7.5.4 Value as an ability to buy processing power

There are two kinds of finite computational[5] resource in the classifier system: (a) the total *information* capacity of the performance system, which consists of the set of 'if-then' rules, a (usually fixed) number of classifiers that perform simple computations, and (b) *processing* limits, which is the amount of parallel computation allowed per time step; for example, a maximum of ten classifiers may be allowed to fire during each basic cycle. For current purposes, information limits will be placed to one side.

Whether the implementation of the classifier system is truly parallel (with perhaps separate processors for each classifier), or only simulates parallelism, *the value of a classifier is an ability to buy processing power*. A high value classifier will be more likely to win bidding rounds, be processed, and post its action part. For example, an internal sensory message may match the first in a chain of high value classifiers that instigate an action sequence. The high value of such a processing chain will make it unlikely that other rules will out bid and switch processing to other ends. The allocation and reallocation of value to classifier rules adaptively alters their ability to buy resource limited processing power and dispositionally determine the behaviour of the system. The same holds, *mutatis mutandis*, for Q-learning, XCS and Dyna. If a substate *is_better_than* another for satisfying a goal then it is more likely to be used. More precisely, the quantitative representation of this relation in RL algorithms functions as an ability to buy the system's available processing power.

7.6 Summary

An *is_better_than* relation appears in design-space when types of selective systems attempt to meet a requirement for adaptation by trial and error learning. For example, a behaviour producing substate that is selected over and above other substates has greater value or utility to the system. Belief-like and desire-like substates are relations between environment and system, and are to be contrasted with forms of value that constitute internal relations between parts of the system. There are many possible forms of value. Some may be implicit and not explicitly represented.

Existing designs for reinforcement learning algorithms employ an explicit quantitative, scalar representation of value (credit, strength, quality or predicted payoff) that is stored with behaviour-producing substates. A concrete example of such a

[5] A classifier system animat (a jargon term for a simulated animal) embedded in an environment will also have 'physical' resource constraints, such as the number and kind of effectors and detectors.

system is Holland's classifier system, in which value plays a role as an ability to buy processing resources and dispositionally determine the behaviour of the system.

Chapter 8

The society of mind requires an economy of mind

This chapter argues that a society of mind will require an economy of mind; that is, multi-agent systems (MAS) that meet a requirement for the adaptive allocation and reallocation of scarce resources will use a universal, scalar quantity representation of value that mirrors the flow of agent products, much as money is used in simple commodity economies. The chapter provides reasons why the scalar quantity form of value present in existing reinforcement learning (RL) algorithms can *scale up* to RL systems with more sophisticated capabilities. The hypothesis, described in section 8.4.7, is that a scalar quantity form of value is common to a large class of RL algorithms. The conclusion follows from postulating a partial identity between cognitive and economic systems at the information processing level of abstraction.

It is shown that, in human economic systems, the money-commodity is an emergent exchange convention that serves both to constrain and allow the formation of commitments by functioning as an ability to buy processing power. MAS with both currency flow and minimally economic agents can adaptively allocate and reallocate control relations and scarce resources, in particular labour or processing power.

8.1 The society of mind: Minsky

> ... a group of agencies inside the brain could exploit some "amount" to keep account of their transactions with one another. Indeed agencies need such techniques even more than people do, because they are less able to appreciate each other's concerns. But if agents had to "pay their

> way," what might they use for currency? One family of agents might evolve ways to exploit their access to some chemical that is available in limited quantities; another family of agents might contrive to use a quantity that doesn't actually exist at all, but whose amount is simply "computed".
>
> M. Minsky, *The Society of Mind*, 'magnitude and marketplace', page 284.

Marvin Minksy's *The Society of Mind* (Minsky, 1987) is the best example of the social metaphor applied to the understanding and design of minds. It outlines a computational society of heterogenous agents that compete and cooperate to produce mental capabilities. The approach of decomposing a computational mind into a society of less intelligent agents is compelling because social systems and large, parallel computing systems share design features. Simon (1981b) discusses how generic design solutions in one type of system can be usefully transferred to the design or analysis of another. For example, both kinds of system consist of a set of mutually connected, interacting subcomponents or agents that are able to perform work, for example computational units that process or people that labour. Such agents can function as both producers and consumers, for example the input and output of information or the consumption and production of commodities. The agents operate within a social division of labour (i.e., a functional specialisation of the subcomponents of the system), which affords fine-grained parallelism, that is, the subcomponents function relatively autonomously and concurrently perhaps pursuing their own local goals. Both types of system need to be coordinated by mechanisms for the production, distribution and consumption of agent products, such as globally accessible databases or free market mechanisms. In addition, such systems must adaptively allocate scarce resources, be they limited labour resources or processing time or commodities or information in restricted supply.

It is these kinds of considerations that suggest that the social 'metaphor' is no metaphor at all, but is an accurate description of a class of complex systems at the information processing level of abstraction, just as the rules that govern genetic evolution have parallels in memetic evolution (Dawkins, 1990; Dawkins, 1986). However, as with all compelling parallels it is important to identify differences as well as similarities. There are many ways in which the evolution of ideas is different from the evolution of genes: for example, ideas are often subjected to rational criticism along dimensions such as probable truth and consequences for others and self.

Human societies and computational societies will also differ in important ways and it would be a mistake to construct computational theories dominated by current ideas about social organisation without exploring the full range of possible designs for social organisation. Despite these warnings, the aim of this chapter is to argue that *a society of mind will require an economy of mind*, and that economic theories, concepts and methods will have new applications in multi-agent systems (MAS) and the understanding of cognition within a single agent, in particular motivational and emotional phenomena. The chapter, therefore, emphasises similarities not differences, and is primarily speculative, bearing on the epistemological foundations of MAS. The key idea is that the management of computational *resources* and computational *processes* needs to be combined. Such combination can occur through MAS that utilise a quantitative universal representation of value that mirrors the flow of agent products, much as money mirrors the flow of commodities in human social systems. Minsky anticipated such an idea, but did not develop it.

8.1.1 The coordination problem in multi-agent systems

A MAS can be thought of as a system that is composed of a collection of agents that normally have their own beliefs and goals, sharing a domain that allows actions to be performed, including communicative actions, such that the system meets some global requirements. The global requirements normally specify goals that can be met by agents acting cooperatively or competitively to discover solutions. Jennings (1996) discusses the *coordination problem* in MAS, which is the problem of ensuring that a society of agents interact in such a manner to achieve global goals given available resources. Coordination is required because 'there are dependencies between agent actions', 'there is a need to meet global constraints' and 'no one individual has sufficient competence, resources or information to solve the entire problem'. Without coordination the MAS would fail to produce useful global results. In this context, Jennings introduces the 'centrality of commitments and conventions hypothesis' which states that: *all coordination mechanisms can ultimately be reduced to commitments and their associated conventions.* However, commitments need not be deliberately generated: attachment structures in most bird and mammal species, for instance, involve some kind of built-in commitments already 'installed' between certain individuals (selective mating, caring and protection of the young, territorial defense and so forth), without necessarily relying upon an explicit contractual basis. Deliberative commitment occurs in agents able to know about their own commitments, formulate them, compare them, and choose to reject them, and so forth, whereas implicit commitment may occur in agents without such capabilities.

A commitment is essentially a goal: an agent can make a commitment to itself (e.g., 'I will tidy my desk today') or to others, in which case it can be thought of as a pledge or promise (e.g., 'I will meet you at ten tomorrow'). As goals, commitments could result from many goal generactivators (Beaudoin, 1994), some very primitive, and some more deliberative. Joint commitments are possible (e.g., 'We will both move house') and are preconditions for cooperative action. Conventions, however, are rules that determine how an agent's commitments are to be formed, reconsidered, or rejected, and social conventions are rules that determine how agents should behave towards each other, for example if they change mutual commitments. For example, agent A may commit to meet agent B at ten because it is conventional for A to obey B because B has greater authority. Subsequently, however, A acquires a more pressing commitment and does not have sufficient time resources to honour the commitment to B. Hence, A informs B of the difficulty because it is a social convention to do so, allowing B to replan and ask another agent C, who can do the work of A, to meet at ten. This is an example of cooperation, communication of failure and replanning. Designing conventions and social conventions is difficult. It is likely that in natural systems, powerful mechanisms have evolved to generate, protect, manage and regulate conventions (Aube & Senteni, 1996a; Aube & Senteni, 1996b). Designing conventions amounts to designing a set of rules that can interact to produce coherent and useful emergent behaviour. Even simple rules, such as those that define Langton's ant, can produce emergent behaviours that are very difficult to deduce from the rules themselves (Cohen & Stewart, 1994). All coordination mechanisms may be reducible to, or expressible as, collections of commitments and conventions, but discovering useful coordination mechanisms is no easy task. Gasser (1991) has also written that commitments should not be seen as mainly initiated by individuals, but might rather be understood as emerging from the web of social interactions, which operate upon individuals as 'field forces' that constrain individual courses of action into joint behavior of cooperation or conflict.

Different requirements entail different designs that meet those requirements. The relations between *niche-space* and *design-space* (Sloman, 1995a) are complex, involving many trade-offs and compromises, for example sacrificing speed of processing for depth of reasoning. The relations between requirements and design are important for understanding all types of systems, from software systems, agent architectures (Sloman, Beaudoin & Wright, 1994), robots, and organisms in the natural world. MAS are no different in this respect: there will be many types of

MAS that satisfy different sets of requirements. Despite the potential variety, some aspects of designs may be common over niche-space; for example, it appears that visual sensing is a good design solution to the requirements imposed by many diverse niches, or, alternatively, fast but inflexible, hard-wired reflexes are a ubiquitous solution to surviving in niches that undergo rapid and perhaps dangerous changes. If there are general principles for the design of coordination mechanisms for MAS waiting to be found they will be precisely those aspects of designs that are invariant or common over large 'areas' of niche-space.

8.1.2 Adaptive multi-agent systems

Adaptive MAS (AMAS) are a subclass of MAS that can continually reconfigure their activity to produce solutions that meet changing global constraints. For example, Schaerf, Shoham & Tennenholtz (1995) describe an AMAS that adaptively allocates jobs to different processing units under variable loads. The class of AMAS is sufficiently general to include many diverse kinds of system and mechanism, in much the same way as the class of adaptive agent architectures can include such mechanisms as reinforcement learning algorithms, artificial neural networks, genetic algorithms, and so forth. In the abstract, there are three distinct ways in which an AMAS can modify its global behaviour. It can (i) alter the behaviours of individual agents or (ii) alter the control relations between agents, for example dynamically defining groups of leader and follower agents. The former is a change of commitments, the latter a change in conventions and social conventions. Yet electing a leader or respecting some authority do involve commitments from the individuals concerned, and altering the kinds of control relations between agents may, in turn, alter the kinds of cooperative groups that can form. Alternatively, (iii) existing agents may be removed or qualitatively new agents may be introduced into the system. AMAS require coordination mechanisms that can cope with this kind of complexity. Such mechanisms need to allocate and reallocate agents to different tasks, alter social hierarchies, change individual agent behaviours to fit new circumstances, and provide means by which global constraints can direct local processing without the need for high bandwidth communication. In addition, there need to be natural ways in which global constraints can be defined within the system.

This section has very briefly outlined the kinds of problems that coordination mechanisms for adaptive MAS need to solve and has suggested that there may be general principles for the design of such coordination mechanisms.

8.2 Money and exchange-value

Human designers of robots often turn to the natural world for design ideas. Similarly, human designers of coordination mechanisms for MAS can also turn to the natural world. The study of ant colonies, primate groups and human social interaction are all potential sources of inspiration. For example, Aube & Senteni (1996a; Aube & Senteni (1996b) propose that certain classes of emotions (such as gratitude, guilt and anger) and emotional communication arose to coordinate animal groups and therefore can serve as a foundation for coordination in MAS. They view commitments as a special kind of resource that ensures access to basic commodities of survival value, and emotional structures as the control mechanism that manages these special resources. This section develops the contention that human economic activity provides an example of another important coordination mechanism – *currency flow* – that may be common to a certain class of *adaptive* MAS. Chapter 9 will show how such a view can begin to uncover the inner mechanics of motivations: that is, why and how some mental processes within the society of mind come to take precedence (be 'preferred') over others.

8.2.1 Basic requirements for the development of money

All human societies are in commerce with nature, extracting raw materials from the environment and returning human waste to the earth. Social organisation implies a division of labour amongst the individuals of the society, that is, individuals perform different, socially useful functions. The total labour of society is shared amongst the different functions, and the products of this labour distributed according to some, usually implicit, scheme and through some collection of mechanisms. One very obvious requirement for a successful social system is that it reproduce its conditions of existence; that is, it must create conditions such that individuals survive and produce offspring so that the available labour of society is continually replenished. This requirement entails that what is produced, distributed and consumed should be so organised to satisfy those needs. This is one of the important coordination problems that social organisations are required to solve: *labour must be divided* and its *products distributed* so that at least a sufficient number of individuals' *basic needs are met*. It is the satisfaction of at least the basic needs of consumers that defines one of the major global constraints for successful human social systems. Money – the representation of the, as yet to be defined, value of a thing – arose at a certain point in human history to solve certain problems of production, consumption and exchange. Pure gold was first coined as money in 625 BC in Greece (Boardman, Griffin & Murray, 1993). In a matter of fifty years trade had burgeoned, and there were banks, merchants, and money-lenders. A numerical representation of value had a revolutionising effect on the capabilities of human society. Subsequently, currency flow has been a common feature of human social organisation, surviving and developing through classical society, feudal arrangements, and industrial and modern finance capitalism. To understand the nature of money it is necessary to examine how and why it arose. The following account of the development of money is based on the opening analysis in Marx's *Capital* (Marx, 1887). It is a rough historical sketch of the *emergence of a social convention* in a human society. The account abstracts from the real historical development of money and uses simple stages and examples for the purpose of exposition. In addition, the emergence of money is examined in an idealised *simple commodity economy*, without complications such as price-fixing, cartels, monopolies, taxation, trade tariffs, travel costs, power relations, trade unions, and the legislative power of the state.

Stage one – simple exchange or swapping. Individual and relatively self-sufficient producers with a small surplus product, for example the peasant farmer whose chickens have lain too many eggs, exchange their goods for other goods. For example, 24 eggs may be exchanged for 2 loaves of bread. In this isolated act of exchange the equality relation (24 eggs = 2 loaves) is determined by the producers' respective opinions of the use-value of the other's goods. The term 'use-value' simply means that the good satisfies some desire or need. In other words, the respective values of the goods are determined locally and subjectively. The exchange of products has a precondition: each producer must have a surplus-product that the other desires. All exchange is performed with a view to obtaining another's surplus-product for the purposes of consumption. Money does not as yet exist.

Stage two – extended exchange or organised swapping. The development of better production techniques and increase in population size creates a greater surplus-product available for exchange. Instead of isolated acts of exchange there may be a definite geographical locale where trading takes place, which is the market. The peasant's 24 eggs now enter into potential relations with all the other commodities available. For example, the 24 eggs may now be exchanged for 2 loaves, or a pair of socks, or five candles, or a pound of butter and so forth. Importantly, an element of *competition* appears that was not present in stage one. Instead of a single peasant and consumer there is a social community of interconnected producers and consumers, for example peasants, bakers, and candlestick makers. Given the choice a baker will tend to exchange his bread for as many eggs as he can get from the community of peasants; conversely, a peasant will tend to exchange his eggs for as many loaves as he can get from the community of bakers. This systemic dynamic – colloquially, the notion of 'shopping around' – will, all other things being equal, have a tendency (which may not be fully realised) to force the equivalence relation between eggs and bread towards a particular ratio that holds for *all* such transactions. This equivalence relation will thus be *determined by the joint action* of the peasants and bakers in mutual competition. In other words, the respective values of the commodities are determined globally and socially (as opposed to locally and subjectively in stage one): local utility functions give way to global utility functions. An individual's local calculation increasingly becomes ineffective in the determination of the equivalence relation, which now tends to be fixed by the community as a whole.

Stage three – ubiquitous exchange. A community in which a good deal of exchange occurs soon finds it convenient to select a particular commodity to serve as the general form of value. A widely valued article would be the commodity to choose. For example, in a cattle community, cows would be a natural choice as medium of exchange (hence comes the derivation of the word 'pecuniary'). This special commodity then serves as a unit of comparison of value and is *directly* exchangeable with all other commodities. This overcomes the limitations of organised swapping, as all producers will now be willing to swap their goods for the general form of value. There need be no local coincidence of wants.

Stage four – money. As soon as a particular commodity is socially agreed upon to serve as the general form of value it becomes the money-commodity, that is, it serves as a universal means of exchange. In most societies this commodity has been gold or silver, and not cows. For example, if 24 eggs = 1 measure of gold, and 1 measure of gold is coined as 10 pence, then 24 eggs have the price 10p. Gold serves as *the* embodiment of value, and can be exchanged for any other commodity. 'Although gold and silver are not by Nature money, money is by Nature gold and silver' (Marx, 1887). Precious metals were chosen because they exhibit uniform qualities but can be repeatedly divided and reunited at will to represent fine-grained differences in the numerical values of things. Also, they have a high value to weight ratio, which is useful if wealth is to be transported in pockets. Paper money normally represents an amount of gold, although the link may be weak between the nominal value of a note and its real value in gold. However, in times of currency crises investors buy gold and silver, trusting the enduring value of gold as opposed to the nominal value of a note.

There has been little computational, as opposed to historical, work on the development of universal means of exchange in MAS: Marimon, McGrattan & Sargent (1990) describe investigations of the conditions in which money emerges in an artificial economy of adaptive, classifier system (Holland, 1986; Holland, Holyoak, Nisbett & Thagard, 1986) agents, although the chosen domain ontology bears only a superficial resemblance to real economies.

8.2.2 A technical aside: economic theories of value

Two theoretical traditions concerning the nature of economic value can be distinguished within the history of economic thought. Both traditions derive from an earlier, classical tradition that first introduced labour-time as a measure of value. The classical tradition is associated with the economists Adam Smith and David Ricardo. The neoclassical tradition, which emerged from such thinkers as Jeremy Bentham and Herman Gossen, diverged from the classical tradition and held that, ultimately, value was a *subjective, psychological* measure of the utility of a commodity that is determined in the minds of individual economic agents. In contrast, the Marxist and to some extent the Keynesian tradition holds that value is an *objective, social* measure linked to the share of social labour expended in the production of the commodity (Lichtenstein, 1983). The highly technical debates between these traditions continue to this day and have yet to be decisively resolved. To confuse matters, the neoclassical tradition is essentially concerned with the determination of price, whereas the Marxist tradition views value as a precondition for price; that is, an underlying 'law of value' generates the surface appearance of prices within a system of commodity production (see *Wages, Price and Profit* in (Marx & Engels, 1968)). This is pursued further in a short appendix to the chapter.

8.3 The currency flow design solution

Money, therefore, is just like any other commodity except for a social convention that ensures it is the means of exchange in all transactions. The particular form of value, be it gold, silver, bronze, paper or virtual currency flows, is a secondary matter: it is function that counts. The function and properties of money will now be examined in greater detail.

Universal use-value. Stage two serves as the starting point. This stage presupposes a social division of labour that allows individual producers to specialise in the production of particular commodities. Specialisation leads to greater productivity of labour and a greater surplus-product. Such producers are therefore no longer self-sufficient and have a greater need to exchange. Exchange may only occur when there is a local coincidence of wants, which is a limiting factor. When this is the case, chains of exchange, or mediated exchanges involving "middle-men", will tend to occur. In fact, coincidence of wants can be only overcome through the use of "middle-men", and the probability of chains of coincidences of wants occurring decreases with the length of the chain; hence, the flow of commodities will be constrained. Exchange will tend to occur at definite locations of high connectivity, that is market places, which provide a higher probability of coincidences of wants. Also, due to the perishable nature of some commodities the chains of exchange may have to occur within definite time periods – for example, an agent will have difficulty accumulating enough apples in order to exchange them for a new boat. Although this stage of a social development is largely hypothetical it does highlight the limitations of bartering, and demonstrates that the eradication of the requirement for local coincidence of wants and commodities is one function of money; that is, it serves as a *universal use-value*: it is a commodity that all agree, or have no choice but to agree, to find useful. Producers become willing to exchange for a representation of value which has the functional property of being able to buy the products of others' labour. One effect of the introduction of money, therefore, is to free up the flow of commodities.

Multiple instrumentality. In a developed money economy everything has a price. Money may be exchanged for any product of any labour.

Semantic determinacy. The exchange-value of commodities as represented by the money-commodity is expressed *quantitatively* and is compared to other quantities of value. Consequently, the meaning of money is globally determined in a society of numerate agents.

Constraints on possible exchanges. A loaf of bread may cost 50p but will not normally be exchanged for 49.5p because of the prevailing social convention. An agent with money can enter into many possible exchanges, whereas an agent without money has those possibilities severely constrained. The globally determined value of commodities defines what is and what is not a legal exchange, and serves as a kind of economic 'all-or-nothing' law.

Low communication costs. Consider the following thought experiment: instead of money there are a host of 'middle-men' exchanging lengthy notes listing individuals with their surplus-products and needs in an attempt to coordinate great chains of exchange mediated by coincidences of wants – a kind of global 'swap shop'. Such notes will entail high communication costs, due to the high information content of the notes, and high administration costs, such as matching up lists with lists. In direct contrast, money, being a number, is easily represented and removes the need for middle-men and their costly communications. (However, in some real markets, such as the housing market, chains of exchange occur frequently.)

Low storage costs. The quality of money does not change. It can be stored by adding up all the quantities into a bigger quantity – a larger denomination of note, for example. There need be no storage of many qualitatively different things, such as filing cabinets of 'coordination notes' in the above example.

Simple operators. Money requires only the very simplest operators: addition, subtraction and numerical comparison. No sophisticated local machinery is required to mediate the transaction. Money is quickly and easily parsed.

Accumulatability. Money, if it is metal, such as gold, does not perish. It can be stored indefinitely.

Distal connectivity. The coincidence of geographical location, time and wants for exchange to occur in a barter economy is overcome with the introduction of the money-commodity. Money can mediate wants, be easily transported from place to place, and be stored for future use, unlike perishables.

Domain independent representation. In exchange, value is compared with value. The value of a commodity does not represent anything external to the economy, nor does it represent any thing within the economy: it is internally relational, specifying an ordering over the set of commodities. Consequently, it would not make any qualitative difference to the functional role of money if the specific kind of labours within society changed, or if the external environment changed.

Coordination mechanism. Importantly, money introduces *supply and demand* dynamics that implement a distributed solution to a global coordination problem. The coordination problem is how private labour can be coordinated on a social scale so that individuals' basic and higher needs are met. Without a coordinating mechanism the social system would break apart; for example, basic goods might not be produced in sufficient quantities, or non-use-values (commodities that are not in demand) might be produced indefinitely.

Consider the following simplified scenario. An increase in productivity in one branch of production, say egg production, means that more eggs can now be produced by the same share of the total labour of society. As the rate of appearance of eggs on the market is higher then before, their value will decrease relative to other commodities, all other things being equal. Hence, egg producers will receive less money for their goods. This may force some egg producers 'out of business' as the money they get for their eggs cannot sustain their needs. Labour will then be freed to be employed elsewhere in other branches of the economy: some egg producers may turn to producing milk or bread, or any commodities that have sufficient social value to sustain their needs. In this way, the value of commodities and the market mechanism regulate the economy as a whole. The total labour of society is dynamically allocated and reallocated in definite proportions to reflect changes in production techniques and demand for products. 'It is only through the "value" of commodities that the working activity of separate, independent producers leads to the productive unity which is called a social economy, to the interconnections and mutual conditioning of the labour of individual members of society. Value is the transmission belt which transfers the working processes from one part of society to another, making that society a functioning whole' (Rubin, 1988). Currency flow *reinforces* social cooperation: for example, a particular agent will not be able to acquire a commodity without first expending labour that has sufficient value to other agents. The market mechanism of exchange-value, the social convention of money, and the local reasoning of autonomous economic agents serves to meet the basic requirements of economic organisation outlined at the beginning of section three.

8.4 Currency flow in multi-agent systems

Our analysis of the role of money in a commodity economy has implications for the design of adaptive multi-agent systems. In this section the particular form of value in economic systems is examined and compared to existing RL algorithms, followed by a sketch of how currency flow could solve the problems of coordination in AMAS. Finally, a design hypothesis for AMAS coordination is proposed.

8.4.1 A universal, scalar quantity form of value

As described in section 7.2.2, all adaptive systems conform to the abstract schema of a selective system, and all selective systems support concepts of value. In the abstract, economic systems are selective systems: the trials are the various concrete labours that produce commodities, the evaluatory mechanisms are the various needs and demands of individual consumers, and selection occurs through the buying and selling of commodities. Over time what is produced matches what is required given available resources.

Economic systems have employed, from the beginnings of commodity production to the present day, a *quantitative representation of exchange-value* that is associated with every commodity and universally represented by a money-commodity. *Money mirrors the flow of commodities*, reinforcing those productive activities that meet the demands of consumers. Human economic systems are an existence proof that complex processing systems can be regulated by exchanging numerical quantities. Information-theoretic analogues of some of the properties of currency flow that were identified in section four may be useful for coordinating adaptive, largely parallel information processing systems composed of autonomous agents (e.g., multiple instrumentality, semantic determinacy, locally constraining, low communication and storage costs, simple operators, domain-independence and the ability to form a basis for coordination). In fact, recent work in artificial intelligence uses economic ideas for resource allocation problems (Wellman, 1995), including allocation of processing time, and reasoning about plans (Doyle, 1994).

Notably, the form of exchange-value is a scalar quantity (see section 7.5.2). Money is only ever added to, subtracted from, or compared with. The form of value in human economic is the same as that found in RL algorithms.

8.4.2 Generalised reinforcement learning

Reinforcement learning (RL) was discussed in section 7.4.1. Examples of RL algorithms are Q-learning ((Watkins & Dayan, 1992) and section 7.4.2), classifier systems ((Holland, 1975; Holland, Holyoak, Nisbett & Thagard, 1986; Wilson, 1995) and section 7.4.3), and W-learning (Humphreys, 1996). Marvin Minsky's *Snarc* machine was an early reinforcement learner that encountered the credit-assignment problem (see section 7.6 of (Minsky, 1987)).

RL algorithms use a scalar quantity representation of value, the reinforcement signal, to select those behaviour-producing components that satisfy conditions of reward over and above those components that do not. Behaviour-producing components with high reward will be more likely to dispositionally determine the behaviour of the system in the future than those components with low reward. For example, the bucket-brigade algorithm used in early classifier systems was inspired by an economic metaphor (Holland, Holyoak, Nisbett & Thagard, 1986), in which system rules are agents consuming and producing internal messages (commodities) who each possess a certain amount of strength (money) which they exchange for messages at a global blackboard (the market). Much as money mirrors the flow of commodities in a simple commodity economy, *value mirrors the flow of messages* in the learning classifier system.

Most RL algorithms are composed of rules. Shoham & Tennenholtz (1994) discuss a generalisation of RL to MAS called co-learning. Co-learning involves individual agents learning in a social environment that includes other agents. Co-learning agents must adapt to each other. Kittock (1995) describes some computational experiments on the emergence of social conventions through co-learning. Work of this kind is beginning to explore how MAS can adapt by reinforcement signals. The use of a universally recognised, scalar quantity form of value is common to both RL algorithms, co-learning, and economic adaptation via currency flow. However, the latter may require MAS with substantially more sophisticated agents than those used currently. The theoretical relations at the information processing level of abstraction between reinforcement and payment for goods is an issue that can be fruitfully investigated by MAS research.

8.4.3 The ability to buy processing power

In economic systems and reinforcement learners, *possession of 'money' by an 'agent' is a dispositional ability to buy processing power* ((Wright, 1996b; Wright, 1996a) and section 7.5.4). For example, a producer who makes a profit will have more money to employ more people (to buy processing power directly) and more raw materials (to buy the results of prior processing). Whether a thing is purchased or a person is purchased for a certain period of the day, an amount of labour power has been assigned to the purchaser. That the labour power has already been expended and is in the form of a commodity, or will be expended and is in the form of a commodity-maker, is a secondary matter. In both cases, processing resources have been bought. Individual profits and losses regulate this ability to commandeer and allocate social resources. Similarly, a rule in a classifier system uses its accumulated strength to bid against other rules for messages in the 'marketplace'. Rules with high strength are more likely to outbid rules of low strength, process the message, and dispositionally determine the behaviour of the system. The bucket-brigade adaptively alters the ability of rules to buy processing power. The same holds for the weights of policy functions in Q-learning. One of the most important scarce resources in MAS are the agents themselves. The total processing power of the MAS is limited, where processing power is ability to do work. Similarly, Marx, drawing on the classical tradition in economics, emphasised labour-power as a finite resource in economic systems, developing the labour theory of value based on this conception. Labour-power is also the ability to do work. Whether it is computational agents performing abstract operations, or real people performing concrete operations, a transformation is taking place that can be called work.

Adaptive MAS must search for solutions to, perhaps continuously changing, global constraints. Therefore, there needs to be an ordering over the various agents of the adaptive system: some agents will perform more useful work than others with respect to certain constraints. The computational resources of the system should be concentrated on useful agents, be it in terms of giving them greater social power or allowing them access to more social products. In other words, useful work within a society (or useful processing within a mind) should be reinforced. The design principle of a quantitative representation of value that functions as an ability to buy processing power can integrate processing (useful computational work) and resources (limited computational power) with relatively low communication costs. Agents with more money can employ other agents, buy the products of other agents' work, and have greater control over system behaviour. Given these abstract and general considerations it is possible to sketch how currency flow could serve as a basis for coordination in adaptive MAS.

8.4.4 Specifying global constraints

Economic systems suggest a natural way to specify the global constraints of an AMAS. In simple commodity economies it is the wants of consumers that determine what is and what is not a use-value. In just so happens that in real economies consumers are normally also producers, but in artificial AMAS the functions can be separated and assigned to different agents. A set of consumer agents that function as the *sole sources of payment* can define the goals of the system. Producer agents must satisfy consumers' wants if they are to receive value for their work. It is feedback from consumers to antecedent producers in the form of payment that selects those productive behaviours that satisfy the global goals of the system, much as conditions for reward select adaptive policies in RL algorithms. For example, an AMAS may be designed to find plans for successful operation in a microworld domain, such as blocks-world. A set of consumer agents can be defined whose various needs are information items declaring that the system has achieved certain objectives, such as stacking a tower of blocks or building certain shapes and so forth. These information items are analogous to desired commodities in economic systems: they are the use-values of the system. A set of producer agents may then attempt to produce the required information items by performing work in the domain, that is, produce information items interpretable as actions by a scheduler. Only those agents or groups of agents that produce the correct set of actions and corresponding results receive money from the set of consumers. Partial solutions may receive partial payment allowing hill-climbing and iterative trial and error search. Baum (1996) describes the 'Hayek machine' that learns to solve blocks world planning problems using a free market of interacting agents and a simplified price mechanism. Weiss (1995) describes the 'Dissolution and Formation of Groups' algorithm that solves block world problems using a collection of agents that learn through reinforcement and form into cooperative groups with 'leaders'. The Contract Net Protocol (d'Inverno & Luck, 1996; Smith, 1980; Smith & Davis, 1981) has, for many years in the field of DAI, also embodied some of these economics-flavored ideas. In a contract net, a manager agent broadcasts a task announcement message, and receives bids from contractor agents. The manager evaluates the bids, selects among them, and allocates the task, or part of it to the most interesting bidder. One innovative aspect of this proposal however, is to place emphasis on the currency flow itself as embodying global constraints, as opposed to examining the mechanics of a local announcement-bidding-allocation process.

8.4.5 Dynamic control relations

As stated, an AMAS may need to alter the control relations between agents in order to meet global goals. A relation of control exists between agent A and agent B if A can determine, or dispositionally determine, B's processing. For example, A may be able to command B to perform a particular task, or A may be able only to request that B perform a task in particular circumstances, and so forth. In human societies there is a wide variety of relations of control, some more benign than others. *Autonomous* agents will often have objectives that conflict with other autonomous agents. One way for agents to overcome conflicts of interest is through negotiation, a process by which a group of agents communicate with one another to arrive at a mutually acceptable course of action. For example, when a conflict is encountered the agents involved may generate proposals for joint commitments with associated explanations. The mooted proposals may then be evaluated, and various counter-proposals or compromises suggested. The Socratic dialogue continues until agreement is reached (Parsons & Jennings, 1996).

In order that local negotiations can meet global requirements there is need for local information, referring to those requirements, that can form a basis for controlling the negotiations. Without such information agents could negotiate commitments that led to globally incoherent behaviour or that required too many resources (i.e., the construction of unrealisable social plans). In human societies many negotiations occur within the context of financial costs. For example, much institutional behaviour consists of negotiating compromises constrained by available funding. The local possession of value limits the formation of commitments, which are essentially about resources (Bond, 1990; Gerson, 1976). Commitments provide access to additional resources and thus become valuable resources in themselves (Aube & Senteni, 1996a; Aube & Senteni, 1996b). However, local possession of value can allow in turn the formation of new commitments. For example, a new injection of funding can release prior constraints on planning: planners may now have sufficient power to employ other agents to do their bidding (that is buy commitments from agents to behave or not to behave in certain ways) or buy the resources needed to complete their plans. Money, the ability to buy processing power, is therefore an ability to form control relations; and *the flow of money adaptively allocates and reallocates constraints on local commitment formation*[1]. Again, one reason for this rests on the fact that commitments themselves constitute a special kind of resource, and that money embodies the value that is computed for these resources through social transactions. It is the requirements for global problem solving that necessitate the imposition of limits on local problem solvers: *Hobbes chairs the Socratic dialogue*. 'Participation in any situation, therefore, is simultaneously *constraining*, in that people must make contributions to it, and be bound by its limitations, and yet enriching, in that participation provides resources and opportunities otherwise unavailable' (Gerson, 1976). Social agents commit to a social convention of money that simultaneously constrains and enriches possible local outcomes.

8.4.6 Dynamic reallocation of labour

As stated, an AMAS may need to frequently reallocate agents to different tasks in order to meet global goals and maintain coherent behaviour. One possible solution is a global controller that has a wider picture of the whole system and directs the activities of others; however, keeping the agent informed could entail high communication costs, create a communication bottleneck, and render the other agents unusable if the controller failed (Jennings, 1996). The alternative is to distribute data and control, and economic systems suggest at least two possible mechanisms. On one side, a system composed of adaptive agents that attempt to maximise personal utility will exhibit distributed reorganisation of labour. Adaptive utility maximisers

[1] Compare (Bond, 1990; Gerson, 1976) where money is viewed as just another kind of resource.

will search for rewarding tasks, allocating and reallocating themselves to different parts of the developing solution. For example, if a system constraint changes, such as a consumer agent requesting a qualitatively different result, then the agents that previously serviced the consumer will search for new forms of cooperation in order to produce the new result and regain gainful employment (c.f. rule discovery of rewarding areas of the pay-off landscape in classifier systems). On the other side, a system that allows agents to sell their processing power to employer agents will exhibit organisational control, that is 'centralised' reorganisation of labour. For example, sufficiently wealthy employers may direct and redirect the processing of large groups of agents, perhaps at the expense of relatively high communication costs within the organisation. In both cases, however, it is money that forms the basis of the allocation of labour, either as a universal want or an ability to buy processing power. Note also that areas of the search space may be redundantly assigned to multiple agents, much as competition occurs within branches of production in real economies.

8.4.7 The currency flow hypothesis

Given these considerations the following design hypothesis is proposed:

> **The currency flow hypothesis (CFH) for adaptive multi-agent systems:** Currency flow, or *circulation of value*[2], is a common feature of adaptive multi-agent systems. Value serves as a basis for coordination; it integrates computational resources and processing by constraining the formation of local commitments. Circulation of value involves (i) altering the dispositional ability of agents to gain access to limited processing resources, via (ii) exchanges of an explicitly represented, domain-independent, scalar quantity form of value that mirrors the flow of agent products. The possession of value by an agent is an ability to buy processing power.

The design hypothesis is a hypothesis because it is a statement about designs that can be falsified. It states something about the *functional* organisation of AMAS at the level of information processing. If the MAS research community discovers

[2]The term 'circulation' is intended to convey the continual adaptive movement of value through the society of agents, just as money circulates within an economy. It is not meant to imply that a fixed amount of value is conserved within the system, or that value repeatedly traverses a 'circular' path.

designs that meet the requirements for AMAS but do not use a currency flow mechanism then the hypothesis is falsified: the design feature is not common to that set of requirements. It is more likely, however, that the hypothesis in its current form is too general and imprecise. Future research may show that currency flow cannot meet all possible requirements for adaptive MAS behaviour, or that currency flow is necessary but not sufficient, or it is simply one of a range of possible alternatives, or it works for only certain types of constituent agents, and so forth. Therefore, the hypothesis serves as a guide, pointing towards perhaps fruitful areas of AMAS design-space based on an analysis of an existing AMAS. Such an exploration will also help elucidate the conditions under which the CFH is a good design solution. Conditions partially identified in this chapter are as follows: (i) the various agents of the system are relatively autonomous, that is they have their own goals and cannot be directly coerced into making choices that contradict their own goals, a condition that does not hold for many hierarchical multi-agent systems with strong top-down control (Sloman, 1996a) or most commercial, procedure-based computer programs; (ii) it is possible for agents to take conflicting and contradictory decisions and act on them, a condition that cannot hold for systems that need to be synchronously controlled such as, for example, motor systems; (iii) there is redundancy such that many agents may compete for the same payoff niche, a condition that, for example, normally holds for reinforcement learning algorithms; (iv) the control relations between agents are dynamic and induced rather than static and predefined; (v) communication within the system is restricted such that no one agent has a global view of the whole system; instead, agents base their decision making on local information; and (vi) the agents are required to meet a set of global conditions that serve as normative criteria for the functioning of the system as a whole.

8.4.7.1 Minimally economic agents

For a MAS to use currency flow mechanisms the constituent agents will require a minimal set of capabilities. A first pass requirements analysis suggests that minimally economic agents will need to be able to form mutual plans with other agents, possess planning capabilities to construct and choose between alternative possible options, handle money, reason about costs, negotiate, and take and give requests and commands. Without these capabilities the economic system may fail to use currency properly or fail to find solutions to global requirements and so forth. Minimally economic agents also require motivations to cooperate. A number of theories of the evolution of cooperative, altruistic behaviour in the natural world have been proposed, notably how selfish behaviour leads to widespread TIT-FOR-TAT strategies (Axelrod & Hamilton, 1981), or how genuinely altruistic behaviour can be inculcated in docile (those that are receptive to social influence) agents with bounded rationality (those unable to fully evaluate how acquired behaviours affect personal fitness) (Simon, 1990). The Baldwin Effect (Baldwin, 1896) predicts that adaptive, acquired traits tend to be genetically assimilated over time. Assimilated TIT-FOR-TAT may form the foundation for certain social emotions, such as gratitude and guilt, which facilitate cooperative behaviour in natural MAS. The existence of a universal 'norm of reciprocity' (Gouldner, 1960) in social behaviour has been known for some time in sociological theory. Shoham & Tennenholtz (1994) report experimental results that seem to show that MAS consisting of agents that use a 'highest cumulative reward' rule (i.e., agents that choose actions likely to yield the highest pay-off) are inefficient in developing social cooperation. Societies of pure personal utility maximisers may not cooperate effectively. As Aube & Senteni (1996a; Aube & Senteni (1996b) have argued, MAS agents may need powerful 'emotional' control structures to support cooperative behaviour, perhaps even as a precondition of the social transactions from which the money-commodity 'device' and the commitments upon which it has to rely could emerge at all. Sloman (1992) conjectures that social cooperation requires some involuntary revelation of some true intentions. Thus, whether minimally economic agents need such control structures is an open question. MAS design can explore what kinds of cooperative behaviours are required for what kinds of global behaviour.

8.5 Conclusion: the circulation of value

This chapter has argued that a society of mind will require an economy of mind, in particular hypothesising that adaptive multi-agent system design will use currency flow as a coordination mechanism. A hypothesis was proposed stating that currency flow mechanisms are likely to be a common feature of AMAS. The useful design properties of a money-commodity were analysed. One important feature is that currency flow adaptively allocates and reallocates the ability of agents to form local commitments. The social convention of money integrates both resources and processing by functioning as an ability to buy processing power.

Chapter 9

A computational libidinal economy

This chapter introduces reinforcement learning to the attention filter penetration (AFP) theory. The currency flow hypothesis is used to build a specification for a motivational subsystem that forms attachments to other agents. The specification constitutes a theory of valenced perturbant states, thereby resolving some of the problems of interrupt theories identified in section 6.4.

9.1 The currency flow hypothesis (CFH) for natural reinforcement learners (CFHN)

The currency flow hypothesis (CFH) implies that natural reinforcement learners employ currency flow mechanisms. This is an empirical claim, and needs to be distinguished from the CFH that makes a claim about designs, whether they have evolved or not. The new, empirical hypothesis is as follows:

> **The currency flow hypothesis for natural reinforcement learners (CFHN):** The currency flow hypothesis holds for the reinforcement learning mechanisms of individual, natural agents that meet a requirement for trial and error learning.

The CFHN, compared to the CFH, is more difficult to falsify. Human designers of artificial systems have immediate, although partial, understanding of the mechanisms they have built. For example, chapter 7 described the internal workings of artificial RL algorithms. It is clear that those RL algorithms conform to the CFH because the internal mechanisms have been created and described by their designers. Similarly, it is clear that the CFH is true for economic markets because we see the circulation of coins and notes, are informed of the circulation of electronic money, and read the theories of professional economists who explain what all this means.[1] Our own products are relatively transparent to us.

The situation is different with regard to naturally evolved systems, such as natural reinforcement learners. We do not know the internal information processing mechanisms. Imagine attempting to deduce the internal mechanisms of a classifier system from its behaviour alone, and then imagine attempting to deduce the internal mechanisms of a rat from behaviour alone. Complex information processing systems possess internal complexity that confounds a purely behaviourist approach. Going beyond external behaviour and *looking* at internal system components does not avoid the difficulty, whether that looking is through a microscope, at a monitor connected to a non-invasive brain imaging system, or at a CPU circuit diagram. Looking provides information about spatial, not causal, relations (see section 2.4). A child ignorant of motor vehicles could not look under the bonnet of a car and *see* how it works. A similar problem confronts those who study the brain: the mental processes are real but unobservable, just as the computational processes occurring in a CPU are real but unobservable.

Whether natural reinforcement learners actually use a scalar quantity form of value to perform credit assignment will be difficult to ascertain for precisely these kinds of reasons. However, evidence from introspection (a 'looking' of sorts), discussed in section 9.2.5, provides further support for the CFHN.

9.2 Specification of a computational libidinal economy

Four problems with existing interrupt theories of emotion were identified in chapter 6. This section provides a sketch of an information processing mechanism held to exist in natural reinforcement learners and conforming to the CFHN.

[1]This is not meant to imply that the circulation of money is the only adaptive mechanism operating within these systems. A system may conform to the CFH yet still exhibit other forms of coordination and control.

9.2.1 Specification

What follows is a specification of the high level features of a *computational libidinal economy*. It is not of sufficient detail to be called a 'design'. For example, the simulation of a computational libidinal economy would require advances in reinforcement learning architectures that include more complex motive management processes. The specification constitutes an abstract theory for a computational libidinal economy (CLE) that instigates and controls the formation of attachments between agents. It is integrated with the agent architecture that was described in section 4.4 and partially implemented in chapter 5. The libidinal architecture is not a general theory of motivational systems, but is currently restricted to attachment behaviour. This is for two reasons: first, attachment is well studied so the consequences of the CFHN can be compared to existing theories; second, attachment and loss are characterised by valenced perturbant states (Wright, Sloman & Beaudoin, 1996), involving just the kind of mental states and processes that existing interrupt theories encounter difficulty explaining. The adjective 'libidinal' is intended to reflect this current restriction of the theory.

The specification fails to stipulate the kinds of substates that constitute the libidinal economy: details of the mechanisms that produce information with semantic content, what that information represents, how it is transformed, and what relations these mechanisms have with themselves and other cognitive systems are missing. The specification describes a CLE composed of black boxes, or economic agents without content. This is a deficiency from the perspective of building a working system, but is less of a problem from the perspective of explaining valency, valenced perturbances, allocation of attention and relations between perturbances and reinforcement learning. A more comprehensive theory would be sufficiently specified as to be implementable. The problem of specifying the information processing 'agents' within the libidinal system will depend on advances in artificial intelligence systems, in particular adaptive multi-agent systems. However, the previous two chapters, in particular section 8.4.7.1 on 'minimally economic agents', give pointers.

Finally, the specification oversimplifies drastically, and hides much unknown complexity. This is currently unavoidable when first attempting to understand a highly complex information processing system, especially when the computational processes are unobservable.

The specification of a CLE is as follows:

9.2.2 A libidinal selective system

The libidinal selective system is a cognitive subsystem. Its main purpose is to develop social attachments to others, such as caregivers or partners.

(a) Untaught conditions of satisfaction. Untaught reinforcement mechanisms define the fundamental attachment goals by specifying various conditions of satisfaction that have been selected by evolution. Example conditions of satisfaction are orgasm, proximity of mate, positive emotional signals from opposite sex (e.g., facial expression, laughter, interest etc.) and so forth. It must be stressed that attachment in humans involves much more than reinforcement learning. For example, it appears to include, at the very least, imprinting-like mechanisms and the construction of predictive models of the attachment figure (Bowlby, 1991a). However, the libidinal selective system is specifically concerned with reinforcement learning. Its relations to other kinds of learning is an open question.

Evolution, therefore, is the ultimate source of attachment motivation. Natural agents include genetically programmed dispositions to evaluate some occurrences as inherently 'good' and some as inherently 'bad', in the sense that the former leads to survival and reproduction whereas the latter leads to the opposite. Evolution has 'hard-wired' domain knowledge or bias (see section 7.4.5) into natural agents that reduces the trial and error learning required during development. However, developmental processes, in humans at least, can produce reinforcers that replace or dominate the evolved ones (e.g., suicide pacts, dieting for fashionable looks, etc.).

(b) Means of satisfaction. To develop an attachment the libidinal system is required to construct a superstructure of motivational substates (or 'agents') that constitute the means of satisfaction of various attachment goals. The motivational substates can generate motivators to higher level systems, such as management processing. The libidinal system is an automatic, largely preattentive process. composed of a 'society' of competing and cooperating substates (i.e., it is an adaptive multi-agent system). All substates have conditions of activation, which are patterns that can match semantic products (much like the condition part of a production rule). Substates, when activated, output semantic products based on internal state and their current input. For example, some substates may have conditions of activation that match sets of beliefs (which, in the current agent design are assumed to be stored in a globally accessible world model, see section 4.4 and (Beaudoin, 1994; Sloman, Beaudoin & Wright, 1994)).

(c) Learnt conditions of satisfaction. A subset of the means of satisfaction are learnt conditions of satisfaction, or taught reinforcers. These substates reinforce subgoals, which are useful 'landmarks' towards the eventual achievement of untaught conditions of satisfaction. Learnt reinforcers inherit their reinforcing properties from untaught reinforcers and may come to dominate them.

(d) A selective cycle. The libidinal system is a selective system and is therefore required to perform three functions: generate substates that are candidate means of satisfaction, evaluate those substates with reference to internal norms, such as utility in satisfying untaught and learnt conditions of satisfaction, and select better substates for future use, while deselecting others. This is achieved by reinforcement and processes of selection on value. For example, a substate involved in the production of behaviour that satisfies a reinforcer will gain in value and be more likely to dispositionally determine the behaviour of the system in the future. However, actual reinforcement is not the only form of evaluation in human-like architectures: depending on domain knowledge, evaluation could occur with reference to a world model sufficiently detailed to predict the consequences of actions (compare the discussion of Dyna in section 7.4.4). For example, a libidinal substate may generate a motivator to a higher level system that is able to predict the likely consequences of adopting the motivator as an intention. A motivator with high expected reward may be more likely to be adopted by management processes than a motivator with low expected reward. These kinds of relations between reinforcement and more complex forms of motive management will only be resolved through the further exploration of agent architectures.

(e) Substate discovery. The generate substate stage of the selective cycle requires both inductive and deductive discovery. Inductive discovery involves 'guesses' to generate new candidate means of satisfaction; for example, the genetic algorithm of a classifier system performs inductive discovery of new classifier rules, which are then used to generate actions and the results evaluated. Deductive discovery, in contrast, involves the application of, hopefully sound, rules of inference to current information to produce new candidate means of satisfaction. The RL learners reviewed in chapter 7 do not employ deductive reasoning, although Dyna's planning phase, in which the action model is used to hypothetically update the policy function (section 7.4.4), can be considered an example of deductive discovery: current information is used to deduce probable quality values for state-action pairs.

The discovery of 'substates' involves the production of new agents, new rules that agents use, new representations, new generalisations over state, and so forth.

(f) Varieties of control substates. The complexity of the substates of the system will vary. There will be hierarchical and dynamic relations of control between substates (see section 8.4.5).

Examples of control states referred to in folk psychology are beliefs, images, suppositions, desires, preferences, intentions, moods, learnt associations, innate or trained reflexes, personality traits, and emotional states. The dispositions may be of a high order (i.e. dispositions to invoke dispositions to invoke dispositions... (Ryle, 1949).) Example substates discussed so far are untaught and learnt conditions of satisfaction and means of satisfaction.

Every control state within the agent architecture has structure, aetiology, powers, transformation capabilities, liabilities, and in some cases semantics (Wright, Sloman & Beaudoin, 1996). These ideas include analogues of the linguistic notions of syntax, semantics, pragmatics, and inference (Sloman, 1994b). For example, a motive in the MINDER1 implementation may have a complex internal structure (syntax), a content (based on that structure) referring to certain states of affairs (semantics), and a functional role, that is, dispositional powers to determine internal and external actions (pragmatics). It may also enter into processes that derive new motivators or plans (inference), it may be brought about or triggered in various ways (aetiology), and may be modified, suppressed or terminated by other processes (liabilities), such as the attention filter. Some control states are short-lived (e.g., a motivator which is immediately rejected, or whose goal is quickly achieved.) Others endure. A control state may exist but not express its causal powers, or may express its causal powers but not determine action. Observed behaviour will generally fail to indicate the full richness of the underlying control states, especially the dormant states.

Libidinal control states may gradually change their status over time, via the operation of the selective cycle and reinforcement. Useful substates may 'percolate' up hierarchies, gaining in abstractness and resistance to change, and increasing their field of influence. Defunct long-term substates may gradually lose influence through lack of use or utility, leaving only a few relatively specific active instances. For example, the selective cycle and credit assignment (see below) within the CLE change control substates through circulation: substates may be formed, removed or reinforced. Reinforcement increases the field of influence of substates through an increased ability to commandeer processing resources. Control states may be qualitatively transformed during circulation, for instance acquiring more general conditions of applicability. Internal connections between control states will set up suppressive or supportive relationships, dependencies, mutual dependencies and, occasionally, dead-locks.

The net effect of all this is a process of 'diffusion' in which the effects of a major control substate are gradually distributed in myriad enduring other substates, such as motive generactivators, plan schemata, preferences, and predictive strategies. In some cases the effects will be *irreversibly* embedded in a host of reflexes and automatic responses. (This is loosely analogous to the process of compiling a high level language.)

The totality of libidinal control substates is a dynamic structure with complex internal relations and many levels of control. High level control states amenable to some degree of self-monitoring are among the requirements for any architecture underlying human-like capabilities. Where there is appropriate self-monitoring the system may also acquire self-knowledge that triggers negative evaluations and new high-level motivators attempting to change some aspects of the system itself, whether previously learnt cognitive reflexes or high level attitudes.

1. **Libidinal generactivators.** An important subset of libidinal substates are the libidinal generactivators, whose semantic products are candidate *motivators* (see section 4.4 and chapter 5) for management ('attentive') processing. These motives may attempt to meet conditions of satisfaction of reinforcers, or may detect threats or opportunities relevant to current attachment plans. They are examples of Frijda's *concerns* (Frijda, 1986). For example, particular libidinal generactivators may detect threats to attachment plans, such as interest from other sexually active males or females towards the loved one.

9.2.3 A conative universal equivalent

(g) A scalar quantity form of value, the *conative universal equivalent* (CUE). The universal, scalar quantity representation of value within the libidinal selective system is the conative ('motivational') universal equivalent (CUE). It functions as a universal means of exchange between the substates of the system. It can be exchanged by a substate for the semantic products of a producer substate, as money is exchanged for commodities in an economic system. CUE is stored with each individual substate. It is assumed that the initial allocations of CUE to untaught substates are genetically specified.

(Note: CUE is a universal means of exchange *within* the libidinal system. It is an open question whether circulation of value can be generalised to other motivational systems. For example, homeostatic and simpler feedback motivators, such as maintaining temperature, removing waste, altering body posture, hunger, thirst and so forth (for example, the motivational systems described in (Toates, 1986)), do not seem to involve the kind of valenced perturbant states characteristic of processes of attachment.)

(h) Possession of CUE is an ability to buy processing power. The possession of CUE by a substate is a dispositional ability to buy processing power. This can take a number of forms.

1. **Possession of CUE is a dispositional ability to commandeer preattentive processing resources.** CUE can be used to grab processing power within preattentive, automatic processing. A substate can exchange CUE for the semantic product of another substate. If many agents compete for limited processing resources in the same context then a conflict resolution mechanism is required. The specification simply states that the tendency to win all kinds of computational resources (e.g., processing time, the semantic products of prior processing, the ability to direct the processing of other substates etc.) is correlated with the possession of CUE. There are many possible conflict resolution mechanisms – a very simple example is the bidding mechanism of classifier systems, and a more complex mechanism is computational negotiation between agents in a MAS (section 8.4.5). Note that, according to the CFH, an adaptive MAS will use currency flow, in addition to negotiation, to allocate and reallocate constraints on local commitment formation, such that local agents meet requirements for global problem solving. The

global requirements are defined by untaught and taught conditions of satisfaction, which are the sole sources of CUE (see next section).

2. **Possession of CUE is a dispositional ability to construct motivators for management processing.** Libidinal generactivators exchange CUE for semantic products that satisfy their conditions of activation. Consequently, if there is competition for computational resources, generactivators with high CUE will tend to win the competition and construct motivators. These become candidates for surfacing.

   The production of a motivator by a libidinal generactivator can *cost*. For example, a generactivator needs to 'pay its suppliers', that is, exchange CUE for the semantic products of other substates, an example of local credit assignment (c.f. the bucket-brigade of classifier systems). This is explained further in the next section. There are 'prices of production' associated with the construction of a motivator and an attempt to enter management.

3. **Possession of CUE is a dispositional ability to make motivators surface and commandeer management resources.** This can be understood by considering the relations between CUE, importance and insistence.

   Insistence is a locally computed heuristic measure of the urgency and *importance* of a motivator (see section 4.4 and (Beaudoin & Sloman, 1993)). Motivators that lead to highly rewarding consequences are important. For example, libidinal generactivators high in CUE, that is those generactivators that led to rewarding consequences in the past, will compute relatively high insistence values for the motivators they produce. The importance component of insistence is proportional to the expected reward extrapolated from past results. In other words, libidinal generactivators high in CUE will have high dispositional powers to produce motivators that surface and determine attentive processing, all other things being equal. The *ceteris paribus* clause is needed because highly urgent but relatively unimportant motives may have control precedence over highly important but non-urgent motives. For example, mourners can ignore their grief when running to catch a bus.

   (The action selection mechanism used in Benson and Nilsson's TRP agent architecture, described in section 4.2.3.4, is somewhat similar: both architectures select goals based on expected reward. However, the CLE produces motives that are candidates for adoption, whereas the TRP architecture has no management processing, immediately adopting goals based on reward and expected time to achieve that reward.)

Once a motivator has surfaced, management processes decide whether to adopt (but not schedule) a motivator based largely on more sophisticated calculations of the importance of a motivator. Major differences between (i) the local, preattentively computed importance component of a motivator's insistence, and (ii) attentively computed importance measures, will lead to perturbant states. For example, a libidinal generactivator may continue to produce insistent motivators even when management processes decide they are unsatisfiable, such as when the object of the motivator has died.

Therefore, CUE is causally related to both the asynchronous interrupt capability of motivators and the capability of those motivators to commandeer attentive resources once surfaced and compared with alternatives. The amount of CUE held by a generactivator affects the importance component of the insistence heuristic (asynchronous interrupt capablility) and the importance calculated by deliberative processing, which may be partially based on expected reward.

Management and metamanagement processes, as opposed to processes in the CLE, can use rule-based knowledge and global access to information to reason about the relative importance, urgency, cost, and so forth, of surfaced motives rather than primarily relying on past and projected quantitative rewards. Deliberative processes can consider many alternative motives together, as opposed to CLE agents that may possess only local knowledge. Scalar quantity forms of value should 'give way' to more sophisticated, symbolic representations of value, which allow reasoning about *why* motive A is more important than motive B, rather than representing *that* A has led to higher reward than B in certain contexts.

9.2.4 Credit assignment

(i) Credit assignment. The exchange of CUE mirrors the flow of semantic products to perform credit assignment. For a substate to enter circulation it must pay the substate that produced the semantic product it matches. Consequently, a single, local exchange of semantic information also involves a local exchange of CUE. The antecedent substate, because it produced 'useful' information that was 'bought', receives a local 'reward', gaining an amount of CUE from the buyer. The buyer, therefore, partially selects the producer. If the antecedent substate receives more CUE than it paid to *its* producer it will gain in social power: the chain of production is profitable. There may be many different credit assignment strategies in much the same way that there are many different payment strategies in economic systems (e.g., direct payment, credit, renting).

(j) CUE derives from reinforcers. The ultimate sources of value are the normative criteria of the libidinal selective system, which are untaught reinforcers. Secondary sources of value are learnt reinforcers. When the conditions of satisfaction of a reinforcer are met the substates involved in producing those conditions receive CUE. Detailing this process further requires solving the temporal credit assignment problem.

(k) Gain of CUE. A substate gains value by receiving more CUE for the information it produced than it paid out to an antecedent substate for its preconditions. This case subsumes the situation in which the preconditions for a reinforcer (learnt or untaught) are met and the antecedent substates involved in the production of the preconditions are rewarded accordingly.

(l) Loss of CUE. A substate loses value by entering circulation and receiving less CUE for the information it produced than it paid out to an antecedent substate for its preconditions. This case subsumes the case where preconditions for a negative reinforcer (learnt or untaught) are met and the antecedent substates involved in the production of the (aversive) preconditions are negatively rewarded accordingly. There need be no 'sink' for lost amounts of CUE: it is assumed that destructive computational operations can be applied, for example rewriting the value of a variable to zero.

(m) Accumulation is 'reinforcement'. The accumulation of CUE by a substate is 'reinforcement' learning, in the sense that the substate will have increased dispositional ability to determine (internal or external) behaviour in similar informational contexts in the future.

(n) Loss is deselection. The loss of CUE by a substate is the partial deselection of that substate. It will have less dispositional ability to determine (internal or external) behaviour in similar contexts in the future.

The exchange of CUE is an important part of the reinforcement learning that occurs within the libidinal selective system. Substates that are adapted, in the sense that they are involved in the achievement of conditions of satisfaction, gain in value; those that are not, lose value.

(o) CUE is internal economy and has control semantics. CUE is a form of value within a selective system. It is internally relational specifying an ordering of utility over substates. It has the control function of an ability to buy processing power. Unlike beliefs or desires it does not refer to any thing within the system, nor does it refer to anything external to the system. It is a domain-independent control signal (see section 7.5.3).

9.2.5 A circulation of value theory of achievement pleasure and failure unpleasure

A computational libidinal economy that meets the specification outlined above has at least two distinguishable types of internal state, intentional and non-intentional states.

The intentional component of the CLE is the set of substate products, in particular the motivators produced by libidinal generactivators. In contrast to CUE, substate products have representational content: they are 'about' other things, be they states-of-affairs in the environment or within the system itself.

A concrete, if simple, example can be provided by the classifier system (for a fuller account see (Wright, 1996b)). The substate products in this system are messages. Imagine an artificial frog embedded in an environment of real or simulated flies[2]. The control program for *simfrog* is a classifier system with an adapted set of classifiers. The *simfrog* has an eye sensor, which forms part of the classifier system's input interface. The eye can detect a number of attributes of any fly within range. Attributes could include whether the fly is moving, what colour it is, its size and proximity. If a fly is detected the eye sensor posts a message to the message list that encodes this information. This sensory message is the result of a *mapping* between a state of the environment and a sub-state of *simfrog*, providing the message with representational content. Internal messages, less directly linked to sensing or acting, will have more complex representational roles within the system. The *semantics* of messages depends on the dynamic relationship between message

[2]The artificial frog is taken from (Holland, 1995).

and environment. For example, the sensory message may match a classifier that posts an action message that results in *simfrog* throwing its sticky tongue in the direction of the detected fly. The meaning of the message, therefore, would be an impoverished version of the imperative, 'eat that fly!'.

The non-intentional component of the CLE is the circulation of value. The circulation of value is a pattern of flow of control, as opposed to semantic, signals. Such signals have no semantic content (although see caveats in section 7.5.3) and propagate around the system altering control flow.

Consider that self-monitoring mechanisms (see section 4.4 and (Wright, Sloman & Beaudoin, 1996)) are required to monitor the circulation of the CUE occurring within the libidinal economy. The reasons why this may be necessary are not discussed here. The circulation of value from one moment to the next will involve a *net* exchange of value, which can be written as V_t, from matching substates to antecedent substates. The self-monitoring mechanism records each V_t over a specified time period, say $t = 1 \ldots n$, and displays the *change* in value, denoted δV_t, which is exchanged from one time step to the next, where $\delta V_t = V_{t+1} - V_t$ (although the monitoring may be analogue rather than discrete). Again, a concrete example is provided by the classifier system. For example, *simfrog* may be in the process of learning how to catch and eat a fly. During this process δV_t can be either: positive, implying (a) a net increase in the utility of antecedent classifiers (substates), and (b) currently active classifiers are likely to lead to positively rewarding consequences; or negative, implying (a) a net decrease in the utility of antecedent classifiers, and (b) currently active classifiers are likely to lead to negatively rewarding consequences; or zero, implying no net change in the utility of antecedent substates. Therefore, the self-monitoring of circulation of value will display a rate of change of value with both sign and magnitude.

Consider connecting the output of this kind of self-monitoring to *simfrog*'s skin, which can change colour. If δV_t is zero *simfrog* remains *green*, if δV_t is positive he displays *yellow* with an intensity $|\delta V_t|$, and if δV_t is negative he displays *blue* with intensity $|\delta V_t|$. When *simfrog* catches and eats a fly he will blush bright yellow as innate reinforcement mechanisms strongly positively reward antecedent classifiers. If *simfrog* possessed more sophisticated reflective capabilities and the self-monitoring of credit assignment played a functional role within the system itself, he might wonder why he has beliefs that refer *and* an odd quantitative intensity that is either positive or negative but doesn't seem to be 'about' anything or serve any readily identifiable purpose. Depending on philosophical prejudice, one might be tempted to say that *simfrog feels* happy, sad or indifferent depending on circumstance.

Obviously, the example is a major simplification. But it shows that the monitoring of circulation of value in the CLE can generate non-intentional control states, which can be either positive or negative, and vary in intensity. In addition, when a goal is achieved, such as the achievement of the conditions of satisfaction of a reinforcer, there will be a monitored increase in value. Similarly, if a goal of avoiding the conditions of satisfaction of a negative reinforcer fails, there will be a monitored decrease in value. Achievement or failure of certain fundamental goals as defined by reinforcers is linked to positive and negative exchanges of value respectively.

Therefore, another design feature can be added to the libidinal economy:

(p) Valency is the monitoring of a process of credit assignment. The monitoring of circulation of value, or virtual currency flows, is the process that gives rise to valenced states, which is a form of cognitive achievement pleasure and failure unpleasure (see definition of valency in section 6.4.3.4).

1. **Negative valency is a loss of CUE.** A monitored circulatory process that involves a loss of value corresponds to negative valency. (Note: this implies that movements of value need not always be monitored.)
2. **Positive valency is a gain of CUE.** A monitored circulatory process that involves a gain in value corresponds to positive valency.
3. **Intensity is rate of exchange of CUE.** The rate of exchange of CUE between substates corresponds to the quantitative intensity of the valenced state.
4. **Gain in CUE is contingent on the achievement of goals.** A gain of CUE can occur when the achievement of a goal is equivalent to the conditions of satisfaction of a reinforcer.
5. **Loss of CUE is contingent on the failure of goals.** A loss of CUE can occur when the failure of a goal is equivalent to the conditions of satisfaction of a negative reinforcer. (The concept of expected reward may be of use here – it would need to be addressed by a more detailed specification and requirements analysis.)

In other words, certain types of 'feelings' are the self-monitoring of adaptations; that is, the pleasure and unpleasure component of goal achievement and goal failure states is the monitoring of a movement of internal value that functions to alter the dispositional ability of substates to buy processing power and determine behaviour.

Such a process is self-monitored as a 'brute' feeling because value does not refer, unlike beliefs that can be true or false, or goals that can be achieved or not. Or put more simply – you are experiencing yourself changing, but that which is changing is not a belief or an explicit evaluation that refers; rather, it is a movement of internal value that alters the dispositional ability of your own internal subcomponents to determine your behaviour. The process is normally associated with other semantic states, such as knowing that you are pleased or displeased, knowing why you are pleased or displeased, and knowing what you are pleased or displeased about.

Valenced states are present in agent architectures that attempt to meet a requirement for trial and error learning. To meet such a requirement it is possible to evolve or design credit-assignment mechanisms that use a domain-independent representation of utility or value, a kind of internal 'common currency'. Such a representation does not refer (in the way that belief and desire-like substates refer) but is internally relational, and it can be gained or lost depending on whether actions are successful or unsuccessful in leading to rewarding consequences.

To be precise, specification (p) is both a design feature and empirical claim. It requires that the process of credit assignment be monitored by another system. It also makes the empirical claim that in human architectures the monitoring is such that, in circumstances of attachment and loss, we have introspective, direct knowledge of the process of credit assignment, although this information is just one component of the overall mental state. Valency, defined in section 6.4.3.4, is the name for this knowledge, and the monitoring of credit assignment is the architectural process that gives rise to it: phenomenological and functional analyses have converged. Introducing reified conceptions of 'consciousness' at this point would only serve to mystify the convergence. However, artificial architectures that monitor their credit assignment processes may not have the required higher level design features that would justify the description of the architecture being self-conscious of its valenced states; for example, without mechanisms to map a monitored state to a concept, and the use of that concept in the production of natural language, the architecture would be unable to inform us what kind of valenced state it was in. Similarly, without categorisation capabilities an agent could not have a goal of achieving that state or make plans to achieve it.

The CFHN opens up the possibility of architectures that generate valenced states far removed from physiology. High level cognitive processes may operate with value, allowing the production of semantic messages (not linked to bodily location, or physiological arousal) coupled with losses or gains in a scalar quantity form of value.

9.3 Prospects for building computational libidinal economies

Designing and implementing a computational libidinal economy conforming to the outlined specification will require extensive work. However, prototype implementations are well within the reach of current AI technology. For example, chapters 7 and 8 reviewed a number of AI systems that already possess some of the required features. RL algorithms, including Q-learning, classifier systems, and Dyna, exhibit simple forms of conditions of satisfaction, means of satisfaction, the selective cycle, CUE, and circulation of value to perform credit assignment. The work of Wellman (1995) and Doyle (1994) use economic markets for resource allocation problems. Baum's Hayek machine, Weiss' Dissolution and Formation of Groups algorithm, the Contract Net protocol (section 8.4.4), and Humphrey's W-learning (Humphreys, 1996) use computational 'agents', price mechanisms and currency flow to learn intelligent behaviour in various task domains. The multi-agent system research community is actively exploring negotiation algorithms for competing and cooperating agents. In addition, agent architecture research has progressed to the stage where more complex forms of motive, or goal, management, such as that exhibited in the design and implementation of MINDER1, can be integrated with reinforcement learning techniques. Benson and Nilsson's TRP architecture and the CLE specification are preliminary steps towards this. However, more work is required to fully integrate the CLE and MINDER1.

Any future work on design and implementation will reveal the inadequacies of the CLE specification, and necessitate revisions of the theoretical structure. The CLE specification is only a first step.

9.4 Resolution of some problems with interrupt theories of emotion

This section examines how the CLE and CFHN resolve three of the four problems of interrupt theories of emotion identified in chapter 6.

9.4.1 The hedonic tone problem

As stated, interrupt theories do not provide mechanisms for hedonic control signals, nor do they explain why such signals are 'simple', why they differ from semantic representations, and why, in the case of forms of pleasure and unpleasure, they are either positive or negative. The interrupt theories did not satisfactorily address valency and intensity. The AFP theory pointed towards the difficulty when it called for the inclusion of 'pleasure and pain' mechanisms within the postulated architecture. This was called the hedonic tone problem (section 6.5).

Circulation of value can answer these questions with regard to valency. The circulation of value is a pattern of flow of control signals. The signal is 'simple', as it is a non-decomposable scalar quantity. The useful design properties of a scalar quantity form of value were discussed in chapter 8, providing new reasons for the existence of simple signals.

Oatley and Johnson-Laird's communicative theory explains the hedonic tone of happiness and sadness states by positing basic and irreducible valenced control signals. Control signals differ in valency because they differ in their functional roles (e.g., 'sadness' has a 'terminate or change plan' function, whereas 'happiness' has a 'preserve plan' function). Control signals are held to explain the subjective experience of 'hedonic tone', whereas a semantic signal referred to the 'object' of the emotion.

In contrast, the monitoring of circulation of value can generate the negative (sadness) and positive (happiness) control signals of the communicative theory: instead of two signals, there are now two qualitatively different outputs from the monitoring of one unifying mechanism (see requirement (p)). This is a more parsimonious state of affairs. More importantly, there is a new, previously unidentified functional role for the control signal: the circulation of value implements a type of adaptation. This is not inductive learning of new hypotheses about a domain, but an ordering and reordering of the utility of motivational substates to dispositionally determine behaviour. The single control 'signal' is a scalar quantity representation of value that can be used to allocate processing resources. It can be both stored and exchanged ('communicated'), and used to coordinate a society of relatively autonomous substates. On this view, the communication of significant junctures of attachment plans is a secondary or derived functional role of the happiness and sadness control signals: monitoring of circulation can provide this information but the primary purpose of the control signal is to perform credit assignment. However, the monitoring of circulation of value, although able to occupy attention, is not an interrupting control signal. Therefore, it differs from the control signals of the communicative theory (COM) theory that are 'broadcast' to cause changes in action readiness. Simon has also noted that 'happiness' and 'sadness' do not normally involve interruption of attention: 'Arousal of the autonomic and endocrine systems can occur at a more diffuse and subtle level, without obvious interruption of attention. Again, ordinary language includes these phenomena in the orbit of affect and emotion, although when the state is not acute and interruptive, the terms 'mood' and 'feeling' will often be applied instead of 'emotion'. Typical examples are happinnes and sadness' (Simon, 1982). The theory presented here concurs with Simon: valency, in itself, is not interrupting, although it may occur contemporaneously with interruption of attention.

Additionally, the analysis of value as deriving from an *is_better_than* relation, explains why 'happiness' and 'sadness' control signals differ from semantic representations (i.e., there is now a deduction from requirements to valenced control signals – see section 6.4.3.5). Unlike belief-like and desire-like substates, value is internally relational specifying an ordering of utility over internal substates. Value refers to utility and functions as an ability to buy processing power. Also, as scalar quantity forms of value are non-decomposable and domain-independent, the monitoring of such signals provides information that is 'brute', 'ineffable', more or less intense, positive or negative, and invariant over task domains, properties that exactly mirror the phenomenological properties of valency. This explanation avoids both physiological reduction (not all forms of pleasure and pain need to be reduced to the 'body' but can arise from higher level virtual machines implemented in the brain) and intentional emergence (value is a 'non-intentional' representation, differing from intentional beliefs and desires). However, the communicative theory accounts for other types of control signals, whereas the CLE only accounts for valency, that is, forms of achievement pleasure and failure unpleasure during processes of attachment (and, if generalised, all types of goal-directed activity involving reinforcement learning). The CLE in its current form does not account for shorter-term control signals involved in initiating, preserving or terminating action tendencies (section 6.3.1 and (Frijda, 1986)), for example the pleasure of the warm sun on one's brow, the pain of a headache, or the sting of tired muscles.

9.4.2 The emotional learning problem

As stated, interrupt theories do not provide mechanisms that integrate emotional states and learning. This was called the emotional learning problem. The introduction of a CUE regulating a CLE introduces one type of learning, trial and error reinforcement learning, to the AFP theory.

Reinforcement learning can alter the dispositional ability of libidinal generactivators to construct motives that grab management resources (see requirement (h3)). This is an example of the modification of the efficacy of certain 'stimuli' to interrupt a central processing system, one type of learning that Simon identified as important in his original interrupt theory (Simon, 1967). For example, a generactivator may gain CUE by constructing motives that, by their adoption by management and satisfaction by planning and execution, meet conditions of satisfaction. A gain in CUE is an increase in causal power: the generactivator, for example, increases its ability to commandeer management resources. Obsessive love is an extreme example of a set of libidinal generactivators continually grabbing attentive resources (although the meeting of conditions of satisfaction can occur in fantasy rather than reality, a case of hypothetical reinforcement based on an inaccurate world model). Similarly, generactivators may lose their ability to commandeer management resources by failing to meet conditions of satisfaction. In summary, *reinforcement learning can alter the ability of generactivators to interrupt and commandeer management resources.*

In section 6.4.2 the correlation between the intensity of an emotion and the 'intensity' of any associated learning was noted: '... the more strongly an appraised process is felt, and the more keenly, therefore, the consequences of some behaviour are experienced as pleasurable or painful, the quicker and more persistent is the ensuing learning likely to be' (Bowlby, 1991a). Specification (p), stating that valency is the monitoring of credit assignment, partially explains this correlation. The greater the assignment of credit the greater the change in causal powers of substates and the greater the monitored intensity of the valenced state. That is, there is a link between the 'intensity' of learning and felt intensity. Note however that 'intensity', although a precise description of an assignment of credit, is a misleading description of the multifarious effects an increase or decrease of CUE may have on libidinal substates.

9.4.3 The valenced perturbant states problem

Chapter 6 identified the following lack in the AFP theory:

> The AFP theory has yet to integrate valency, and more generally hedonic tone, with its existing theoretical concepts. The insistence of a motivator has no causal link to valency. This lack makes it difficult to account for valenced perturbant states, which are perturbances that include a valenced component. For example, grief and triumph, while both perturbant states, differ in valency. As currently constituted the AFP theory could only account for this difference in terms of further explicit evaluations. Currently, it does not include the pleasure of success or the pain of loss.

Specifications (h2) and (h3) provide the causal link between the insistence of a motivator and valency.

The interruption of attention by an insistent motivator may occur without the occurrence of valency (e.g., loud bangs), and valency may occur without interruption of attention (e.g., achievement pleasure) because generactivation of motives and the monitoring of credit assignment are separate processes. However, the previous reinforcement received by a libidinal generactivator at least partially determines both the insistence level and importance of any motivators it may produce. Important motives are those that have led or a likely to lead to highly rewarding consequences (h3). Libidinal generactivators high in accumulated CUE produce highly interrupting and intense[3] motives. (Note, however, that this is a dispositional propensity as other motives may be produced at any time and grab management resources, or the filter threshold may be raised, and so forth. See section 5.5.3.2, which describes how MINDER1 supports dispositional interruption of attention.) The link between previous reinforcement (CUE) and importance provides an explanation of valenced perturbant states, states that involve both loss of control of attention and valency.

Consider a negatively valenced perturbant state, such as can occur during loss. It involves both a loss of control of attention (due to important motives pertaining to the loved one surfacing into management) and the monitoring of a loss of CUE (libidinal generactivators lose credit because they now fail to meet conditions of

[3]'Intensity' was briefly mentioned in section 5.5.3.3 and is discussed in (Beaudoin, 1994). Intensity, a technical term used in our group, is the dispositional ability of a motive to hold onto management resources once adopted, and should not be confused with the intensity of valency, or non-technical uses of 'intensity'.

satisfaction due to the loss the of object of attachment). The more important the motives, the higher the associated loss of CUE, and the more intense the associated valency (see (p)). Similarly, a positively valenced perturbant state, such as can occur during triumph or glee, involves both a loss of control of attention (due to thoughts pertaining to a major achievement surfacing into management) and the monitoring of a gain in CUE (generactivators gain credit because they were responsible for meeting conditions of satisfaction). A subset of perturbant states are valenced perturbances that involve the production of attention disrupting motivators coupled with the self-monitoring of circulation of value performing credit assignment; that is, *occurrent reinforcement learning together with the monitoring of credit assignment plus loss of control of attention is experienced as a valenced perturbant state.*

This explanation links the 'strength' of a motivational concern to the intensity of the valency experienced when the concern is met or violated (see (Frijda, 1986), pages 340–342). A concern, implemented as a libidinal generactivator (see section 5.3.2.4), has motivational 'strength' to the extent that it can buy processing power and dispositionally determine behaviour. Its causal power depends on its accumulated CUE. The intensity of mental pain or pleasure can be correlated with the amount of value invested in the goal object: people are happy when they get what they really want, and sad when they can't get what they really want. A deduction from the CLE specification has produced a conclusion in accordance with the commonsense knowledge of everyday life.

However, the theory of valenced perturbant states presented here needs to be extended, both in terms of specifying in detail the processes at work in terms of a design and working implementation, and stating conditions for the activation of the monitoring of credit assignment and reasons why such monitoring is needed. *Prima facie*, monitoring of credit assignment provides a link between reinforcement learning and types of deliberative learning, such as when management resources are employed to discover why conditions of satisfaction have been met and to deduce any consequences for future motive adoption and planning. In addition, the monitoring of credit assignment 'broadcasts' significant junctures of plans to higher level systems, a function first identified by the communicative theory.

9.4.4 The control precedence problem

The control precedence problem is the problem of explaining the limited causal powers of meta-deliberative control, particularly in the context of longer term emotional episodes that include persistent perturbant states, such as loss. Mourners cannot decide to postpone their grieving. Section 6.4.1 argued that the limited powers of meta-deliberation are adaptive rather than contingent features of our evolutionary development or the result of the intrinsic limitations of information processing architectures. The limited causal powers of meta-deliberation or self-control is the obverse of adaptive control precedence. Section 6.4.1 concluded that 'the AFP theory, and related interrupt theories, need to be augmented with new mechanisms that show how, and new requirements that explain why, meta-deliberative control is constrained'. As stated, a full answer to this question is well beyond the scope of this thesis. However, the AFP theory augmented by the currency flow hypothesis (CFH) suggests possible mechanisms that can restrict the causal powers of meta-deliberation, whether those restrictions are adaptive or not. This lays the groundwork for possible future research.

Empirically, human meta-management is unable to immediately and directly alter the insistence heuristics of libidinal generactivators (otherwise we could postpone and even avoid grief). The allocation and reallocation of credit in the CLE is not under metamanagement direction, although management and metamanagement can choose to adopt or reject motives from that system, and also select what plans to follow. In summary, *credit assignment may be monitored but not directly controlled.*

Therefore, self-control can fail because the value accumulated by libidinal generactivators cannot be altered by management or metamanagement systems. A libidinal generactivator's dispositional ability to buy processing power remains until it is deselected by the libidinal selective system itself. For example, in the case of grief, libidinal generactivators may have persistent causal powers to interrupt attention due to previous reinforcement. It is only through generactivators constructing motives, and hence expending some of their accumulated CUE (see requirement (h2)), that they lose their causal powers. Hence, paradoxically, libidinal generactivators need to construct candidates for attention in order to stop constructing candidates for attention (and eventually be deselected). Value must be expended in order to be lost. The motives that are produced will be unsatisfiable, as the loved one is no more, and therefore fail to meet conditions of satisfaction. The resultant loss of CUE is self-monitored as 'painful' or 'unpleasurable' (this explanation of grief is expanded in chapter 10.) Note that if the rejection or suppression of a motive by metamanagement also prevents the discovery of its failure to meet conditions of satisfaction, then there will be no loss of value, no mental pain, and the causal powers of the libidinal generactivator will persist.

The circulation of value is opaque to management and metamanagement control, but partially determines the ability of motives to commandeer management resources; therefore, the libidinal economy *constrains* the allocation of attentive resources. Assuming that this generalises to other reinforcement learning subsystems then it serves as a candidate mechanism to account for the limited causal powers of meta-deliberation. However, more exploration of designs and implementations of agent architectures is needed before these notions can be made precise.

This relation between libidinal generactivators and management and metamanagement systems is surprisingly Freudian. Therefore, in section 9.5 Freud's metapsychology is briefly compared to the computational libidinal economy.

9.4.5 Missing elements

Much work is still required to deal with many difficulties and gaps in the specification and the explanations deduced from it. Problematic areas include mechanisms for: self monitoring, self-controlling abilities, the kinds of global control indicated by mood changes, and the processes which assemble and disassemble attachment structures (i.e., the selective cycle). Also, designs for minimally economic agents are required.

Terms such as 'attention' have been used without precise definitions. When fully specified the architecture will be used as the basis for a host of new definitions of classes of mental states and processes (like basing the descriptions of types of physical stuff on a theory of the architecture of matter (Sloman, 1993d)).

It is hard to think about the multifarious states and processes that can occur in such a complex paper design. A working implementation can aid analytical thinking, by exposing consequences of the design.

The full articulation between substates that possess CUE within the libidinal system and other cognitive processes, such as the management layer, has yet to be described. Understanding such causal relations would constitute a theory of 'interaction between so called cognitive belief systems and the phylogenetically older reward systems' (Read, 1995).

Also, there is no theory of the full variety of substates that could mediate behaviour within the CLE, nor is there a theory of the types of semantic signalling that can occur (i.e., the forms of representation that substates use and exchange).

Finally, no mention has been made as to how the CLE is implemented in the brain. That would require an investigation of the results and theories of neuroscience. For example, it appears that negative credit assignment in the CLE may be based on older pain circuitry: 'It is likely that this system [the separation-distress circuit] got built so as to manage the interdependence relationships that are necessary for survival, especially between mother and young. It seems that it developed across cerebral evolution from the pain circuitry already in place. This would explain the fact that the loss of a close being remains a painful experience and also that the substances which are active in relieving pain, such as morphine, have a surprisingly powerful effect in reducing psychological distress due to social isolation' (Panskepp, 1980).

9.5 A reappraisal of libido theory: computational psychodynamics

Freud was concerned with motivational and dynamic aspects of cognition. In this section, the theory outlined here is compared to aspects of Freudian metapsychology, in particular, his much criticised concept of libidinal energy. Obviously, there is not the space for a full and detailed comparison of the two theories here.

9.5.1 Freudian libidinal economy

Freud held that the instincts are the source of *psychical energy*, which he calls *libido*, *interest*, or *cathectic energy*. Libidinal energy derives from the sexual instincts and is a particular type of the more general cathectic energy.

The *Id* is a subsystem of the mind, consisting of unconscious processes. It is where instincts can attach libido to various *objects*. Object is a technical term, but can be best understood by translating it into modern terminology: it is a cognitive representation of some sort. For example, an instinct may attach libidinal energy to the representation (or object) of another human being, such as a caregiver in early life.

Mental processes, in particular those in the unconscious Id, are regulated by the *pleasure principle*. This is the seemingly simple postulate that psychic processes strive towards gaining pleasure and avoiding unpleasure. The instincts bestow or withdraw libidinal energy to and from various mental objects according to the pleasure principle. For example, a child will discover objects or events in the environment that are associated with pleasurable occurrences. Such objects will have libidinal energy transferred to them, a process sometimes called *cathexis*. It is the investing of energy in the object of desire – an assignment of positive value to it.

Freud held that there is no negation, or contradiction, in the Id. This means that the various unconscious processes are unaware of each other, are entirely selfish, and strive for individual satisfaction. In modern terminology we might say that processes within the Id are relatively autonomous, operate in parallel, and act with mainly local knowledge, unaware of the possible contradictory demands they make on higher level systems.

When discussing libido Freud uses a number of analogies. An 'energy' metaphor is often used, implying that a mental object contains libidinal energy in a latent state, ready to be utilised at any time. A 'hydraulic' metaphor is used, particularly when discussing the dynamic flows of libido within the Id. When discussing the 'striving' nature of instinctual forces, in particular their efforts to circumvent conscious repression, he favours an 'amoeba' analogy. The pseudopodia of the amoeba are the instinctual flows of libido testing out mental pathways in a continual search for satisfaction.

The major functional role of libido is motivational: it is the carrier of instinctual demands. Libido is motivational energy in the sense that it is a 'force' or 'interest' that can direct thought and behaviour. Objects with high libidinal energy *tend* to occupy attention (the Id, for example, places instinctual demands on conscious processes by cathecting libido to various objects). Freud writes –

> We have defined the concept of 'libido' as a quantitatively variable force which could serve as a measure of processes and transformations occurring in the field of sexual excitation.
>
> S. Freud, *Three Essays on the Theory of Sexuality* (1905), p. 138 of (Freud, 1987).

Therefore, libido is also quantitative; it becomes attached to objects in definite amounts. In *Repression* (1915) Freud writes –

> Clinical observation now obliges us to divide up what we have hitherto regarded as a single entity; for it shows us that besides the idea, some other element representing the instinct has to be taken into account, and that this other element undergoes vicissitudes of repression that may be quite different from those undergone by the idea. For this other element of the psychical representative the term *quota of affect* has been generally adopted. It corresponds to the instinct in so far as the latter has become detached from the idea and finds expression, proportionate in its quantity, in processes which are sensed as affects.

S. Freud, *Repression* (1915), p. 152 of (Freud, 1991).

It is unclear what is the precise meaning of affect is in this context. However, if it is taken to mean 'feelings' of whatever kind, as opposed to objects or ideas that have explicit representational content, then libido can be related to non-intentional phenomena (but it should be noted that the conceptual relations between affect, emotions, 'discharge' and libido in Freudian theory is complex and arguably confused). Therefore, the link to affect can be considered a 'weak' property of libido.

There are a number of problems with libido theory. Two major difficulties are the dynamism of libido and its relation to affect. Freud writes –

> ... the mechanisms of repression [conscious or ego-based suppression of motives] have at least this one thing in common: *a withdrawal of the cathexis of energy* (or of *libido*, where we are dealing with sexual instincts).
>
> S. Freud, *Repression* (1915), p. 154-5 of (Freud, 1991).

This quotation (and others) are often ambiguous as to what causes change. Is it libido that is dynamic, withdrawing from an object when it encounters repression, or do the mechanisms of repression direct the withdrawal of libido? Libido itself could be dynamic or the processes that operate upon it. The metaphors chosen by Freud (hydraulics, energy and pseudopodia) imply that it is libido itself that is the agency of change.

Also, Freud did not develop a comprehensive theory of affect or emotion. In *Beyond the Pleasure Principle* (1920) Freud writes -

> Here might be the starting point for fresh investigations. Our consciousness communicates to us feelings from within not only of pleasure and unpleasure but also of a peculiar tension which in its turn can be either pleasurable or unpleasurable. Should the difference between these feelings enable us to distinguish between bound [static, cathected to an object] and unbound [dynamic, flowing between objects] processes of energy? or is the feeling of tension to be related to the absolute magnitude, or perhaps to the level, of the cathexis, while the pleasure and unpleasure series indicates a change in the magnitude of the cathexis *within a given unit of time*?
>
> S. Freud, *Beyond the Pleasure Principle* (1920), p. 337 of (Freud, 1991).

This quotation conflates a number of phenomenological issues. The huge diversity of phenomenological phenomena that can be classified as either 'pleasurable' or 'unpleasurable' are treated together, without an attempt to distinguish cases. Also, the feelings involved in desiring ('tension'), which can be either pleasurable or unpleasurable (compare a strong desire to urinate with the pleasure of desire during the sexual act), are considered together with feelings occurrent on the achievement or failure of important longer-term goals (compare winning an olympic gold medal to losing a loved one). Such (poorly understood) phenomena are then linked with 'flows' of cathectic energy, and by extension, libido, without an increase in clarity.

The next section compares libido theory with the computational libidinal economy outlined in this document.

9.5.2 A comparison of CUE and libido

The properties of libido in Freudian metapsychology can be summarised as follows:

1. Dynamic: libido is generally a cause of change, bestowing and withdrawing its attachments (cathexes) to and from various mental objects according to the pleasure principle. It is 'hydraulic' in the sense that it flows between mental objects, an 'energy' in the sense that it is the 'fuel' that causes mental events to occur (such as directing attention and 'interest'), and 'striving' in the sense that it is the active representative of instincts (particularly sexual), seeking to achieve their conditions of satisfaction.

2. Quantitative: it flows and becomes attached to objects in definite quantities.

3. Attentional: libido is correlated with the direction of attention, for example an object with high libidinal energy is likely to gain conscious attention (unless repressed).

4. Motivational: libido is the psychic representative of (strictly speaking, sexual) organic drives or instincts, and represents their motivational 'push' in mental life. Attention and motivation are implicitly linked in Freudian metapsychology.

5. Non-intentional: libido is to be contrasted with objects that represent things. It does not refer.

6. Unclear relations with affect: the relations between libido theory and 'feelings' or simple affects, such as cognitive pleasure and unpleasure, are unclear.

7. Basic: or primitive, in the sense that the sources of libidinal energy are the instincts, which are deemed to be the innate representatives of our evolutionary heritage.

8. Adaptive: libido flows to and from mental objects according to the pleasure principle, that is libidinal energy transfers to those objects (and their associations) that are linked with positive outcomes, which are conditions that satisfy instincts.

9. Storable: libido is cathected to objects, where it can attached, stored or connected. It is distributed over objects in the Id.

The properties of the conative universal equivalent that circulates within a libidinal economy can be summarised as follows.

1. Passive: in contrast to libido, CUE is passive and operated upon. The main operator on CUE is exchange. CUE is not hydraulic, 'energising' or striving: the dynamism of the system is within motivational substates, not within the CUE that is exchanged. However, CUE does 'flow' within the system, via local exchanges. Also, CUE is computational or information-theoretic, in the sense that it is a control signal within an information processing architecture. This avoids the vitalistic connotations of Freud's metaphor of 'energy'. However, Freud did occasionally use an economic metaphor.

2. Quantitative: CUE is quantitative; it is exchanged and stored in definite quantities.

3. Ability to buy processing power: the possession of CUE by a substate is a dispositional ability to buy processing power. This includes the ability to construct motivators for attentive processing; hence, CUE is involved in the allocation of attention.

4. Motivational: the sources of CUE are untaught reinforcement mechanisms, which are examples of *a priori*, evolutionary sources of motivation.

5. Non-intentional: CUE does not refer, but is relational, specifying a partial ordering (in terms of possession of CUE) on system substates. CUE is internal economy alone.

6. Clear relations with simple affect: CUE has well-specified relations to simple affect, to be precise, cognitive valency, which is achievement pleasure or failure unpleasure. The quantitative intensity of valency, and its qualitative differentiation into pleasure or unpleasure is equated with the self-monitoring of credit assignment.

7. Basic: the sources of CUE are untaught reinforcers, which are held to be genetically specified.

8. Adaptive: exchanges of CUE perform credit assignment. The CLE includes untaught reinforcers that reward motivational substates according to the utility of their consequences. This increases their ability to buy processing power and dispositionally determine the behaviour of the system. Over time the system adapts to environmental circumstances. A net gain in value is linked to the achievement of basic goals and can be monitored as positive valency. This process is an example of the pleasure principle, that is CUE is transferred to those 'objects' that satisfy instincts.

9. Storable: CUE is stored with substates, and is distributed over the substates of the libidinal economy.

9.5.3 Summary

In summary, the CFHN and specification of a computational libidinal economy share a number of important features with Freudian libido theory. However, there are important differences, especially properties 1, 3 and 6. Freud believed that 'psychic energy' was a ubiquitous causal factor in psychological processes, whereas the CUE is currently narrower in scope: it is restricted to processes of attachment characterised by valenced states, where valency is a technical concept and is to be distinguished from other forms of pleasure and unpleasure. The theory has been motivated by different concerns and arguments (particularly the emphasis on requirements and design). Also, the CLE is better specified than libido theory, to the extent that it will be possible to develop implementations to help clarify the requirements and specification. Some existing architectures already operate on principles that conform to the CLE, albeit in a simplified manner. This is why the specification outlined in this chapter is called a computational libidinal economy: the name is intended to reflect a convergence of ideas from computer science, AI, psychology and economics.

9.6 Conclusion

This chapter used the CFH to build a specification of a computational libidinal economy, a multi-agent system that adapts through reinforcement and generates motives that are candidates for motive management. The AFP theory has been extended to explain valenced perturbances, the relation between reinforcement learning and some emotional states, and forms of achievement pleasure and failure unpleasure. The extended theory is a 'valenced attention filter penetration' (VAFP) theory of emotion.

Chapter 10

A circulation of value analysis of attachment and loss

In this section the VAFP theory is applied to a concrete emotional phenomenon: grief at the loss of a loved one. This is not a comprehensive 'theory of grief' but an application of a 'broad but shallow' architecture to the understanding of a complex cognitive phenomenon. It will be seen that a complete architecture can account better for the diversity of internal and external behaviour that can occur during grief than an explanation focusing only on some emotional mechanism.

10.1 A first-person account of loss

The following quotations (a) to (p) are based on extracts from a first-hand account of grief, taken from the internet and edited to disguise its origin. After each item preliminary comments are made (in italics), relating it to the architecture. These comments may appear gratuitous to some, but the aim is not to belittle an intense personal experience but to provide general remarks about the psychological phenomena in question. Later the comments are expanded upon.

(a) When the person finally passes on, the hurt is like no other. ...Now the memories of the good times seem to be like a film playing in my head.

This is one of many examples of partial loss of control of thought processes in emotional states: thoughts, motives and memories relating to the object of the *emotion intrude and may even dominate thought processes, making it hard to attend to other matters, even those judged to be urgent and important.*

(b) XXX called us the day after his friend died suddenly. He was broke, alone, grieving and gravely ill, staying at the hospice where he'd spent the last month by his friend's bedside. I flew to YYY and brought him home, where he spent the next ten weeks in hospital. He was *so* ill that it was hard even for me – a medically sophisticated person – to comprehend and accept.

New facts may not easily fit into one's belief system. This is linked to the pain they cause: not physical pain but mental pain.

(c) I started grieving the day I received the phone call, continued grieving the day I met him in the hospice in YYY, and all the time I sat by his bedside back in ZZZ. It tore me apart to see what this disease had done to him...

Grief can be both extended and highly disruptive. Notice that 'tore me apart' does not describe physical or physiological processes, but in a highly metaphorical way describes mental processes. Unpacking that sort of metaphor is part of the job of a deep theory of emotions.

(d) At least, that's all I wished for [his good health] silently over and over again as I sat next to him in the hospital.

Powerful new motivators sometimes cause futile behaviour.

(e) When he died, I was initially in shock.

Shock can be physical or mental. It is difficult to tell which was referred to here but it could easily have been both. One of the things to be explained is how mental events, for example learning about the death of a loved one, can produce profound physical effects.

(f) And then there are details like writing obits, funeral arrangements, meeting his family, etc., which keep you busy. And then the grief resumes.

Grief involves dispositions that may be temporarily ineffective because urgent and important tasks manage to hold attention. But during that time the disposition remains and when there is a chance it regains control. From this point of view it is misleading to say 'the grief resumes'. It was there all along, though its manifestations were temporarily absent.

(g) It has been two-phased for me. This summer I kept confronting the *unreality* of it all. XXX, dead? I was wracked with insomnia. Every night as I'd try to go to bed, indeed, during much of the day too, I would replay the details of the previous three months over and over and over again.

(h) Each recollection was full of almost indescribable pain, but it was also fresh with his presence, something I cherished and which I'd missed daily for almost four years before. I wish I'd had some 'good times' to replay too, but they were from an earlier time, …and they were crowded out by the intensity of the recent months.

This helps to bring out some of the complexity of human emotional states: mixtures of different states are not uncommon. Several different dispositions can be 'fighting' for control and the balance may shift from time to time. Not long ago, in a BBC radio interview, the captain of a women's yacht in a round the world race described her emotional state on arriving at the final port. It was an enormously complex mixture of: pride in achievement, sadness that it was all over, joy at the prospect of seeing loved ones again, delight at the prospect of eating (rations had been exhausted two days earlier), regret that they had not won the race, happy memories of teamwork and obstacles overcome, sadness that the team would now be parting, and so on.

(i) I managed to keep up a bit of a social life during this summer.

In a state of grief, interactions that would normally be taken for granted and enjoyed may be difficult.

(j) My overall mood was somber, but I was at least willing to be distracted by friends…

Sometimes when self control of thoughts is hard, external help is effective. A mood is a type of global state, which need not have a semantic focus, unlike emotions. Moods also need to be explained by a theory of the architecture of a mind.

(k) I became more and more of a hermit. Things lost their savour for me, and I withdrew.

Some of the high level control states, which form the personality, can be profoundly changed by grief. This may affect a wide range of preferences, choices, strategies, plans, and behaviour.

(l) Grief isn't something you can show for too long. People are uncomfortable with it. They will indulge you only for so long, so you just hold it inside, slog along and try to get on with life.

Besides the control problems which relate to tasks and goals that form part of normal life, the observation that grief has undesirable social consequences can generate a new second order control problem – namely, not allowing the grief to be shown or to intrude in social interactions. Many emotions generate second-order motivations relating to the control of those emotional states themselves.

(m) The holidays and the new year are a natural marker, and they've kept me busy and distracted me.

Another example of external help with the control problem.

(n) I try to reintegrate myself with my social circle, and start seeing friends again more regularly. But I'm worried; I don't want to let go of the grief. Sometimes I think it's all I have left of XXX.

Motivation in relation to the control of the emotional state may be very mixed. The griever who manages to control the grief and get on with life may also suffer feelings of guilt or regret because another motivator involving one's duty to the deceased seems to be violated, or because of the feeling that overcoming the grief would itself be a sort of loss of contact with the deceased.

(o) I don't want to enter his bedroom one day with indifference, and wonder how we might use the space. I don't want to stop crying every day when reminded of another piece of our time together. Well, I *do*, eventually, and I know I will, but I am not comfortable with this yet.

Another example of mixed and conflicting second order motivation relating to the state of grief.

(p) This was a gruelling, soul-grinding and exhausting year. It has been the worst year of my life.

Emotional states normally involve evaluations. Powerful emotions often arise out of 'intense' evaluations. The emotional states themselves can be evaluated and contribute to the judgement that something bad, or something good, is happening.

Those who have grieved or had close contact with a griever may empathise with these extracts. Others will also understand, for these are pervasive phenomena and play a major role in many works of art. The architectural theories presented and developed in this thesis can provide a provisional first draft explanation of certain characteristic cognitive features of mourning in terms of causal relations between underlying information processing mechanisms.

The following are among the surface phenomena of grieving, illustrated by the above extracts:

1. The continual and repeated interruption of attention by memories or thoughts relating to the friend's illness and death; that is, loss of 'normal' control of thought processes. See [a, d, g, h].

2. The difficulty of accepting the fact of the friend's illness and death. See [b, e, g].

3. The disruptive effect on normal, day-to-day functioning. See [c, e, g, k, p].

4. Periods of relative normality when grief is 'backgrounded', sometimes because external factors help one regain 'normal' control. See [f, i, j, m].

5. Attempts to 'fight' the grief. [g: 'try to go to bed', l: 'hold it inside, slog along and try to get on with life'].

6. Second order motivators, some of them involving evaluation of the grieving state as good or bad, including, in some cases, wishing the grief to continue. See [n, o].

7. The subjective 'pain' experienced by the mourner. There may be mental pain as well as bodily disturbances. See [a, h, p].

8. Crying. See [o].

These are not all the possible symptoms of long-term grief and neither are they unique to grief: excited anticipation of a long awaited event could also disrupt normal day-to-day functioning. Anger can also be 'backgrounded' when other demands control one's attention. Guilt feelings, or an undesirable infatuation may also be fought against.

Many theories of emotion concentrate either on the neural substrate, external behaviour or externally observable changes (facial expression, posture, muscular tension, sweating, etc.) In the VAFP theory these phenomena are deliberately ignored and regarded as only marginally relevant, since, in principle, they could occur without these other accompaniments, for instance, in beings from another planet whose mental functioning and social life were much like ours despite considerable bodily differences (Sloman, 1992).

10.2 Attachment structure

A raised surface can leave an impression on human skin; similarly, interaction with another person will leave an 'impression' on mentality. Before the advent of information processing architectures this metaphor could not be unpacked.

Bowlby's theory of attachment (Bowlby, 1991a; Bowlby, 1991b; Bowlby, 1979; Bowlby, 1988) attempts to explain how affectional bonds are created and the effects that occur when such bonds are broken. Although criticised in recent times (particularly the emphasis on maternal deprivation in childhood to explain subsequent problems in adulthood; see chapter three in (Smith & Cowie, 1991)) attachment theory is still used to account for both childhood and adult mourning in clinical psychology. Bowlby was highly sympathetic to a control system approach to psychology, although he did not attempt to design and build them (Bowlby, 1991a). The VAFP theory can extend his work and show how a distributed multi-component *structure of attachment* to an individual may develop within an agent architecture and influence subsequent processing. Being deeply entrenched at many levels within the control hierarchy it manifests itself in multifarious ways when its object dies.

The perceptual system and belief systems of an agent will include information about other agents, including information about how to recognize them and what behaviour to expect from them in various situations. This may also include evaluations such as 'X is a good person' or 'X is dependable'. Interaction with X will lead to creation of motive generactivators expressing motivational attitudes towards X. Over time, enduring control states pertaining to X will be generated that interact with higher level attitudes and personality traits.

For example, various preferential mechanisms may be set up ('prefer to be in the company of X'), which could function as motive comparators; or unfocused and abstract wishes ('wish X is always happy and well'); also desires ('desire to holiday with X sometime soon' or 'desire to spend more time with X'), hopes ('hope X likes me', 'hope X enjoys my company', 'hope that X will remain close') and aims ('maintain friendship with X', 'avoid arguments with X'). High level preferences may generate lower level motive generators: for example, preferring to be in the company of X could generate the aim to maintain the friendship of X. In other words, a diverse collection of control states with complex interrelations and dispositional powers will be created alongside factual information collected through interactions with X.

If interaction with another person satisfies the preconditions of various untaught and learnt reinforcers the CLE will assign credit to the responsible motivational substates. This is a positive feedback loop: existing substates determine behaviour that is rewarding, leading to an increase in their CUE, which, in turn, allows rewarded substates to buy more processing resources and have greater causal powers to dispositionally determine attention, and hence behaviour. The process will include the construction of libidinal generactivators that are specific to the particular person concerned. The generactivators will accumulate CUE if they produce motives that lead to successful outcomes. As substates gain in value they gain in the ability to commandeer libidinal resources, including forming links with producer substates, or 'employing' substates directly. A new 'branch of production' will appear, concerned with generating motives pertaining to the person, ensuring the attachment process continues, and various threat and opportunity detectors to attachment goals. In consequence, the attachment process will include moments of (libidinal) goal achievement or failure linked with the monitoring of credit assignment; that is, they will be characterised by valenced states. Of course, there will be other types of emotional state apart from valenced states: for example, there may be moments of disappointment, where an expected reward did not occur. The subject, therefore, over a certain period of time, enjoys rewarding interaction with another person who becomes an object of affection. The attachment structure, including the libidinal substates, mediates the mature relationship.

The evaluations, generactivators and motivators will be positive towards some individuals, negative or neutral towards others, and with varying degrees of strength. Depending on the particular combinations of evaluations and other attitudes towards an individual, the death of that individual may cause grief or some other kind of emotional state, or no emotional state. A prerequisite for grief is strong positive evaluation, though that is not sufficient, for the death of a person whom one admires or respects greatly need not cause grief. Something more is required, namely the 'instinctual, attachment bond' created by the other person satisfying the conditions of libidinal reinforcers, and the sort of entanglement of personalities commonly labelled as 'love'.

The kind of loving that potentially leads to grief, which is being called 'attachment', is a very complex mixture of states that develop over time through mutual interaction. It will involve many dispositions, including shorter-term dispositions that produce pleasurable feelings in the company of the person, displeasure when the person is absent or harmed, and so on. Besides feelings, attachment structures can generate new motivators relating to the person, for example when information is received about that person's needs, successes, failures, suffering, and so forth. All these new (dispositional) control states generated by the process of attachment will, over time, integrate into the existing control network.

These control states involve many dispositions, including potential influences on both preattentive and attentive processes. The former include (a) a tendency for new motives to be generated pertaining to X; (b) assigning relatively high insistence values to motives concerned with X, particularly if there is a serious problem involving X that needs urgent attention; (c) allowing the filtering mechanism to give preferential surfacing conditions to X-related motives, as all things relating to person X are deemed important (compare a mother and her baby); and (d) new links between phenomena involving X and the forms of pain and pleasure concerned with preserving or terminating current activities (e.g., pleasure while holding the loved one).

Effects on attentive, management processes include: (e) new dedicated decision procedures with regard to X (e.g., skewed importance, urgency and cost-benefit computations that raise the priority of X-related motives); (f) creation of unusually detailed (possibly unrealistic) predictive models about X's behaviour and preferences; (g) clusters of management procedures that manage X-related motives by combining model-based information, current and new goals to form new intentions; (g) a relatively high proportion of motives concerned with long, medium and short-term intentions relating to X, in various states, such as conditionally suspended, postponed, ongoing etc.; (h) an unusually high proportion of intentions that are long term mutual or joint plans predicated on the co-operation and continued proximity of X; and possibly (i) new motive conflicts pertaining to X, for example the combination of preferring to be in the company of X, wishing that X is happy and believing that X wishes to be alone, or loves another. Great novels and real human tragedies often depend on such conflicts and the processes they generate.

Metamanagement procedures may be generated or altered during the growth of attachment. For example, management tasks of the form 'decide whether to adopt motive M' may come to be handled as soon as possible if M pertains to X. Relatively more computational resources may come to be allotted to any decision procedure concerned with X. A host of plan libraries expressing the utility of certain actions for achieving goals with regard to X will be formed, which facilitate planning relating to X; and 'chunks' of actions that appear to be efficacious when dealing with X may be abnormally strongly reinforced and lead to stereotypical and positively valenced patterns of interaction.

Summary: an 'attachment structure' relating to an individual is a highly distributed collection of information stores and active components embedded in different parts of the architecture and linked to many other potential control states. When an attachment structure concerning individual X exists in an agent, almost any information about X is likely to trigger some internal reaction. In particular, information about good things or bad things happening to X may trigger reactions whose strength and pervasiveness depends on how good or bad they are. Death is a particularly bad event.

In this chapter the process of *detachment* (in which the attachment structure is gradually dismembered and possibly replaced by a new complex set of beliefs and motives relating to X, consistent with X no longer being alive) is not described. This drawn-out process is part of a self-control strategy for overcoming perturbance, albeit a long-term strategy that attempts a design change to achieve its ends. This process can be analysed into many sub-problems — for example, the structure of attachment would need to be inspected for the sources of perturbance, blame assigned, a modification of the structure selected, and repair work effected followed by some kind of verification process to check whether the modification had resulted in an improvement.

These are all extremely sketchy ideas that need to be developed in the light of a more detailed specification of the architecture and its 'learning' capabilities.

10.3 The information processing underlying loss

An architecturally grounded interpretation of the surface phenomena of grief is now given in terms of an attachment structure in the griever towards the deceased.

1. The continual and repeated interruption of attention by memories or thoughts relating to the friend's illness and death.

Perturbances have already been described: the VAFP theory permits perturbant states when heuristic mechanisms designed to prevent disturbance of resource limited processes continually 'let through' motivators and thoughts that divert attention from highly valued activities. Following bereavement, cyclic processes could occur, involving, among other things: motives relating to the dead person, generated by long term attachment structures, including desires for the person to be alive, or present, or unharmed; wishing one had done things that might have prevented the death; recalling that the person is dead; rejection of the motives as therefore inappropriate or futile; evaluating such rejection as undesirable; reminders of relevant information concerning the person, such as might be important if the rejected goals were being acted on. These and other interactions might all reverberate throughout the system because of the deeply entrenched information structures and the powerful triggering effect of news that the worst possible harm has already happened to the person.

Some of these events may set off a stream of deliberative thought (or meta-management processes) attempting to re-orientate extant desires, intentions and plans, to cope with the changed circumstances. This process could also trigger the recall of associated memories in the form of different sensory modalities (images, smells, sounds etc.), as well as triggering a host of embedded threat and opportunity detectors waiting in the wings.

The structure of attachment explains why motives relating to X are likely to disrupt attention. (a) X-related motives will be given high insistence values because the relationship with X is strongly positively rewarded, and therefore important, and X has suffered great harm. (b) Meta-management control processes ensure that motives and thoughts pertaining to X are always decided as soon as possible, so that such motives tend to grab attentive resources immediately. (c) Dedicated evaluation procedures rate X-related motives preferentially, assigning skewed importance, urgency and cost-benefit measures. (d) Predictive models, triggered by X-related motives, will consume computational resources by attempting to reason about X's needs and possible reactions to things. (e) In a resource-limited system, the proliferation of motives pertaining to X may 'crowd out' other motive generators.

Besides internal processes that spontaneously occur following the news of X's death, external reminders may trigger additional X-related processes: for example, driving past X's favourite restaurant or accidentally finding an old photograph, or hearing X mentioned in a conversation. In some environments such reminders will be frequent. Perceptual schemata looking out for the 'lost' individual may misidentify strangers as X. The association of places, objects and events with memories of the deceased may be powerful triggers for perturbant episodes, sustaining the period of mourning and making recovery difficult without a change of location.

To summarise: *If a structure of attachment to X exists then motives and thoughts pertaining to X will surface and successfully compete for attentive computational resources; news relating to X's death will therefore have a strong tendency to generate perturbant states. The agent's thought processes will be partly out of control.*

2. The difficulty of accepting the fact of the friend's illness and death.

Updating many entries in a large database of information can take time, including the time for restructuring and propagation (c.f. belief maintenance systems). This is one notion of the 'difficulty' of accepting new information.

Another factor is resistance to change. The agent has 'affective' grounds for wanting to believe that information about X's death is false. This could include long term high commitment intentions pertaining to X, involving mutual plans that have had resources expended on them, which the agent does not wish to regard as wasted. Besides uncomfortable evaluations, complete assimilation of the new information may require extensive resource-consuming cognitive reorganisation because the attachment structure is distributed and interwoven with other control states. Humans often seem to reject information, however reliable, if it requires extensive reorganisation of control states and value systems. This may be part of a good engineering design for intelligent agents in a mostly stable world.

Finally, the agent may know from past experience that the acceptance of such beliefs entails a long process of suffering and pain. Holding out hope that the information may turn out to be false is a management goal to delay the onset of this process.

3. The disruptive effect on normal, day-to-day functioning.

Perturbance involves disruption of the processes of motive management, and day-to-day goal processing may be adversely affected by management overload. It is difficult to plan a shopping trip or attend to what others are saying when distracted by futile regrets or painful thoughts and memories.

Besides cognitive disturbance, bereavement can cause physiological changes in the mourner, such as weight loss and excessive tiredness, which will contribute to a lack of efficacy; however, this is not addressed by the agent architecture.

4. Periods of relative normality when grief is 'backgrounded', sometimes because external factors help one regain 'normal' control.

When the management system is involved in new important and urgent tasks it sets the interrupt filter threshold so high that the conditions discussed above no longer hold, like the soldier or football player who is injured and yet feels no pain. When the external demands are removed, the threshold drops, and processes relating to the bereavement regain control (see section 5.5.3.2, which describes how MINDER1 supports occurrent and dispositional perturbances).

Another factor may be a general mood of depression that 'colours' motive processing during grief. A depressed mood is a global control state that 'scales-down' interaction with the environment. (There is no space for a full discussion of moods.)

5. Attempts to 'fight' the grief.

How can the mourner 'fight' the grief and 'try' to get on with life? 'Fighting' here refers to a kind of mental striving or conscious self-control, which is not always easy. It requires some way of suppressing perturbant states, which is often much harder than control of emotional expression (external symptoms).

When the bereaved is attempting to work yet thoughts are continually drawn to the recent death, a self-control mechanism (briefly sketched in sections 3.2.2 and 4.3.3, and a major function of metamanagement) may detect the perturbance and attempt to negate its disruptive effects. Until detachment has been achieved, such self-control is partial and transient: the mourning returns when fragments of the attachment structure are next triggered, by external or internal (possibly subconscious) processes.

One form of self-control uses the artificial 'deadening' of cognitive activity by alcohol or anti-depressants. Chemicals can alter the functioning of abstract machines through their effect on the neuro-physiological substrate they are implemented upon. Although understandable, this strategy could exacerbate the problem (drink can make people maudlin) or possibly slow down the process of detachment.

People are sometimes exhorted to try intentional suppression of perturbing thoughts, using internal imperatives: 'don't think about that', 'ignore that', 'put these thoughts out of mind' and so forth. This could also flow from a metamanagement process, spawned by a self-control mechanism, which rejects new motives pertaining to the deceased. Some people may have learnt how to raise their attention filter threshold deliberately. In practice instructions to oneself often fail.

Even if temporarily successful this may lead to a build up of motives waiting to surface, since the cause of perturbance remains. Sudden surfacing of these suppressed motives could produce breakdown of control if there is a drop in the filter threshold due to low management load. Thus a person experiencing grief might function normally at work during the day (when high management load sets a high filter threshold) only to break down at home later (when both load and threshold drop).

Even at home one can try to absorb oneself in attention grabbing, computationally expensive tasks. 'And then there are details like writing obits, funeral arrangements, meeting his family, etc., which keep you busy. And then the grief resumes.' Arranging the funeral *has* to be done (and therefore motives pertaining to it will be highly important) and can divert attention for a while, whereas attempting to read an interesting book fails. However, as the intensity of grief lessens through gradual dismantling of the attachment structure, thoughts and motives relating to X have lower insistence, allowing 'normal' tasks with low importance and urgency to hold attention and prevent perturbance, and permitting enjoyment of activities such as listening to music, playing games, reading books, or conversing with others.

Another coping strategy is the formation of a new affectional bond to replace the old one ('on the rebound') — an option that is not always available, especially to an older person who has lost a spouse. The formation of such a bond might be a way to avoid the lengthy and painful process of detaching the structure of attachment by finding a new use for it. This involves replacing the original referent and possibly other things, and will not be achieved easily because of the distributed mechanisms and links that constitute attachment. This might be connected with the phenomenon of 'projection', where the grieving person views the new person in terms of the old.

Some of these strategies may be ineffective, or may have undesirable side effects, including hindering the long term process of detachment. The pressure to find quick fixes may come from the culture, for example through the necessity to keep one's job. This could cause harm if the only satisfactory strategy for achieving internal reorganisation following death of a loved one is through the natural process of grieving.

Examples of strategies that might aid this process are acceptance (as opposed to suppression), via a meta-level goal to interpret the experience of grief positively, by understanding that grief is necessary and worthwhile. A supportive social circle may be required for this to work. The self-control of emotional expression ('hold it inside, slog along and try to get on with life') becomes necessary when friends or work colleagues are less prepared to make allowances. It is difficult to control facial expression and general demeanour. People can usually see through the attempt. Limitations on our ability to dissimulate may arise from requirements for successful social co-operation ((Sloman, 1992) and section 8.4.7.1).

> 6. Second order motivators, some of them involving evaluation of the grieving state as good or bad, including, in some cases, wishing the grief to continue. See [o, q].

Disruptive and painful processes can trigger a second order motive to end the state. But the mourner quoted wishes to preserve his grief: 'I don't want to let go of the grief. Sometimes I think it's all I have left of XXX.' And 'I don't want to stop crying every day as I'm reminded of another piece of our time together. Well, I *do*, eventually, and I know I will, but I am not comfortable with this yet.' Why should the mourner — paradoxically — wish the grief to continue? There are a number of elements at work here: (a) the knowledge that the period of intense grieving may be coming to an end, (b) the association between grieving and the recollection of memories of the deceased, (c) a conflict of meta-level motivation between wanting to stop grieving and not wanting to.

The architecture described can support such conflicting processes. For example, detachment may be occurring concurrently with a metamanagement process that ensures that the deceased is not forgotten, but remembered with the appropriate sadness. The meta-management process may have been constructed by a collection of high level control states constituting a self-image; for example, the mourner may view himself as somebody who loved the deceased very much and consequently *should* experience the appropriate amount of grief and heartache. (Cultural norms will affect this.)

Second order effects are to be expected within the framework of the architecture. The metamanagement system includes self-monitoring processes that allow high level motive generators to be triggered by the detection of internal states that require some change in management strategy. Some of these simply redirect attention and cause sensible evaluating, reasoning, deciding, and planning to occur. Others generate some new motives that are, for one reason or another, hard to achieve, and some that are rejected yet go on being reactivated and interrupting processing. Exactly how all this develops will vary from individual to individual and within an individual from one situation to another. Second order processes may be strongly influenced by a culture.

> 7. The subjective 'pain' experienced by the mourner. There may be mental pain as well as bodily disturbances.

Why is grief a subjectively painful 'soul-grinding' experience? The answer is that grief not only involves a structure of attachment perturbing attentive processing, but also involves valenced states and valenced perturbances. The VAFP theory can explain the nature of such painful and insistent thoughts.

The attachment structure is no longer adapted to its environment, and is no longer useful for mediating behaviour. A whole set of concerns is violated; for example, libidinal substates that detect threats to attachment plans will be activated, generating motives for attentive processing. The construction of a motive (semantic signalling) also involves the exchange of an amount of CUE (control signalling). If a motive pertaining to the dead person is generated by libidinal substates it becomes a candidate for surfacing. If it surfaces, management processes will decide that it is unsatisfiable, and reject it. Libidinal generactivators have persistent causal powers to interrupt attention due to previous reinforcement. It is only through generactivators constructing motives, and hence expending some of their accumulated CUE, that they lose their causal powers. Hence, as stated in section 9.4.4, libidinal generactivators need to construct candidates for attention in order to stop constructing candidates for attention (and eventually be deselected). Value must be expended in order to be lost. This is a 'crisis of overproduction', in which libidinal generactivators construct unsatisfiable motives that do not satisfy conditions of satisfaction. In addition, the death of the loved one may satisfy conditions of satisfaction of negative reinforcers, involving credit assignment processes that reduce the amount of CUE held by substates. In summary, libidinal substates will gradually lose their accumulated CUE.

The monitoring of credit assignment will detect net losses of CUE, generating mental states of negative valency, which are more or less intense, non-intentional states of cognitive 'unpleasure' or 'pain'. Therefore, the VAFP theory can ground the folk psychology intuition that some emotions involve a 'release': the accumulated value of libidinal substates is gradually expended in a hopeless attempt to satisfy libidinal reinforcers. This 'flow of value' continues until the CUE has been 'released' and the substates lose their causal powers. Negative valency ('painful thoughts'), therefore, may be a necessary consequence of adaptive change: the libidinal part of the structure of attachment, no longer useful for motive generation, loses its ability to buy processing power, grab attentive resources and dispositionally determine behaviour. This process is self-monitored as unpleasurable. (This is also one reason why grief differs from disappointment. The former case involves a loss of CUE that has actually been gained in the past, whereas the latter case involves a failure to gain CUE predicted to occur in the future.)

10.4 Crying as the plan of last resort

> 8. Crying. Can we provide a design-based account of the onset of crying? Why does wailing or howling occur in times of extreme emotional distress?

Infants cry when their desires are unsatisfied. This grabs the attention of an adult who will then normally attend to the baby's needs. For a baby in the helpless altricial stage crying is a *basic, genetically determined, plan*. It is the only way to satisfy its needs for food, warmth, milk and so forth. Later, crying is no longer necessary because there is a repertoire of plans and the agent can act independently.

Consider the following scenario. A loved one suddenly died a few weeks ago and you are in mourning. You are thinking about going shopping but your attention is disrupted by thoughts of the lost one. A perturbant state is manifest. Your thoughts are out of control and you cannot stop thinking about what you were going to do together, the things you miss sharing with them, that you are lonely and without a close friend, and so on. You stare at the wall and ruminate. In terms of the architecture the structure of attachment is generating motives that are surfacing through the attention filter to disrupt management processes. These motives, pertaining to the dead person, are unsatisfiable and therefore rejected. But their high insistence makes them surface again, foiling any plans for regaining control. You may feel mental pain as substates are negatively reinforced. In this situation, the basic plan of crying may be invoked because (a) no other plans from the repertoire exist for the surfacing motives, (b) the invocation conditions of the basic plan are situations where 'nothing can be done', situations identical to the altricial stage of development, (c) the basic plan has worked in the past to satisfy desires in such situations, and (d) it has general applicability, that is, was used to satisfy diverse and basic wants such as food, warmth, milk, and proximity to adults. *Crying is the plan of last resort,* and can be triggered by negatively valenced perturbant states.

This is similar to Sartre's view of 'emotions' as an attempt to 'magically' transform the world (Sartre, 1934). From an infant's standpoint, crying achieves things by magic. As children grow their helplessness diminishes, along with the need to cry. (The picture becomes complicated later on, when the possibility of emotional deception is discovered — faking crying to manipulate others.) Also, the notion of 'basic plans' as control mechanisms of last resort bears important similarities to Kraemer's 'cascade hypothesis' (Kraemer, 1992), which states that control will 'cascade' down to 'genetically programmed neurobiological adaptive behaviours' if the organism is faced with 'disasters' (problematic situations) that its 'acquired behaviours' cannot deal with. However, Kraemer's approach is very different from ours, namely psychobiological as opposed to cognitive, and leads to seemingly different answers (biogenic amine system function as opposed to information processing). There is no space for a full comparison here, but a synthesis should be possible. For example, our notion of a basic plan makes no commitment to its implementation details. The integration of 'bottom-up' approaches with 'top-down' requirements-driven design work should yield fruitful, and not necessarily contradictory, results.

10.5 Discussion

Using the postulated architecture, a partial and provisional design-based interpretation of many aspects of human grief has been given. However, the theory is sketchy, incomplete and offered only as an initial step towards a comprehensive theory of emotions informed by the exploration of agent architectures in which control mechanisms are information based.

1 For each loved person, a structure of attachment develops, consisting of diverse distributed mechanisms and representations with varying powers of persistence and dispositional causal roles in determining behaviour. This is the affectional bond.

2 Removal of the referent renders the structure of attachment inappropriate for the control of behaviour.[1] Nevertheless, triggered by the news of the death and reverberating associations, the attachment structure generates thoughts and motives that surface and divert attentive resources, producing negatively valenced perturbant states.

3 Perturbant states disrupt normal functioning in resource-limited management processes.

4 The valenced component of perturbant states is the monitoring of a loss of credit. The mental pain is the result of attachment structure substates losing value and hence their causal powers to dispositionally determine the contents of attention and behaviour.

[1] Compare Oatley's treatment of grief: 'a whole repertoire of subplans and knowledge becomes useless' (Oatley, 1992).

5 The futility of other plans may trigger regression to a basic plan of crying.

6 Various self-control strategies may be instigated to overcome the perturbant states; however, the phenomenology of grief suggests that the causal powers of self-control mechanisms are limited.

7 Detachment takes time due to the deep and diffuse embedding of the attachment structure in the architecture. Extensive cognitive reorganisation and re-learning is required before the generation of perturbant states ceases, or drops to a manageable level. How long it takes will depend on details of the case. Some grief lasts as long as the griever.

Much work is still required, to deal with many difficulties and gaps in the design and the interpretation based on it. Problematic areas include mechanisms for: more sophisticated forms of learning, the variety of dispositional control states, and the processes which assemble and disassemble attachment structures,

Part IV

Emotional agents

Chapter 11

Conclusion

The work presented in this thesis constitutes a theory of valenced perturbant states, a theory specified in sufficient detail as to be implementable in a computer simulation. The main new theoretical results are the currency flow hypothesis, value as an ability to buy processing power, and valency as the self-monitoring of credit assignment. The VAFP theory builds upon previous work on the emotions, in particular the theoretical work of Sloman. It also places previous research, such as Oatley and Johnson-Laird's distinction between control and semantic signalling and Freud's concept of 'libidinal energy', in an illuminating new light. Section 1.2 lists in greater detail the main contributions of the work.

Some directions for future work are now discussed followed by concluding comments.

11.1 Limitations and future research directions

Sections 5.4.2 and 9.4.5 and chapter 10 discussed some limitations of the VAFP theory. In addition to the already mentioned limitations there a number of important research issues that need to be addressed by further design-based work.

- Designs need to be explored for adaptive autonomous agents that possess both reinforcement learning and motive management capabilities. The reinforcement learning component of such agents should progress from using relatively simple state-action rules to using sophisticated, minimally economic agents. Research into adaptive multi-agent systems may feed into the design of reinforcement learning algorithms and vice-versa. Such work will eventually decide the status of the currency flow hypothesis.

- A simple, prototype implementation of an agent exhibiting valenced perturbant states is required. This is well within the reach of current technology. Such an implementation would rightfully be called an emotional agent.

- A design-based theory of the self, self-reference, introspection, identity, and self-control is required. Such a theory would illuminate how an agent can have and can lose control over its thought processes. It would also help dispel confusions that arise from a reified conception of consciousness.

- The VAFP theory should be applied to an understanding of other emotional phenomena, such as excited anticipation, triumph, glee, disappointment, and so forth, and other related mental phenomena, such as obsession and depression. The theory, in the hands of experienced therapists, could perhaps help clinical practice, particularly if it grounds, clarifies or explains pre-existing intuitions. However, the deduction of robust counterfactual statements about human emotional problems is a long way off.

- The VAFP theory lends weight to those philosophical positions that support the possibility of a partial convergence between subjective phenomenology and causal relations between forms of representations in virtual machines. The theory that valency is the self-monitoring of credit assignment is perhaps the first example of a close *match* between phenomenology and information processing, although I am told that some theories of visual processing partially explain the phenomenology of colour perception. The structure and function of a scalar quantity representation of value explains both the structure and meaning of experienced valency. Valency is pleasurable or unpleasurable, more or less intense, and appraised as intrinsically 'good' or 'bad' because value refers to quantitative utility, and is gained or lost in definite amounts contingent on the achievement or failure of goals. A possibility for future work would be a detailed analysis of the assumptions behind and meaning of such a 'convergence' or 'match' and its implications for the philosophy of mind.

- A more comprehensive theory would consider evolutionary requirements and the reactive subsystems we share with other animals. This is required to explain the basis of other emotions, such as anger, fear and disgust. In addition, social requirements need to be considered to explain emotional expression and the function of emotions in human society (Aube & Senteni, 1996a; Aube & Senteni, 1996b).

- The VAFP theory (Sloman and Beaudoin's agent design plus valenced perturbances) is a candidate 'unified theory of cognition' (Newell, 1990) albeit in disguise. In terms of implementation and empirical validation it is not as developed as SOAR or ACT-R, but it has similar theoretical scope. However, the VAFP theory addresses different requirements to, say, SOAR, which is reflected in their very different architectures. SOAR does not manage multiple motives, and neither does it have different asynchronous levels of control. There is not the space for a full comparison here, but suffice to say that the VAFP theory introduces new requirements and new design possibilities to unified theories, in particular reinforcement learning, multiple motives and control of attention: VAFP is the heart to SOAR's head. Further work is required to develop the theory into a mature unified theory of cognition. That would require the collaboration of many people over many years.

11.2 The exploration of design-space

This thesis can be classified as belonging to the 'creative modelling' phase of science (Bhaskar, 1978), where competing models of generative mechanisms of surface phenomena are explored. Until we have a good 'design-space' of broad and increasingly deep agent architectures it will be difficult to move from hypothetical explanation to justified selection between hypotheses. The research community operating as parallel search in both empirical and theoretical directions will ultimately select between competing theories, using a variety of criteria including generality, simplicity, implementability, evolvability and applicability to other animals. But the theories need to be developed. The VAFP theory developed in this document is a first step, design-based explanation of a subset of emotional phenomena.

In order fully to understand the complexity of human behaviour new ontologies need to be proposed at the information processing level of description just as a designer specifies an architecture to meet some collection of behavioural requirements. By exploring families of related architectures and their strengths and weaknesses in relation to various niches, it may be possible to explain not only current human capabilities but also the evolution of human and non-human mental architectures in a variety of organisms. The work will also give new insights into what to expect if and when we design autonomous artificial agents, whose internal complexity will make detailed prediction of their behaviour very difficult.

That all this will be a long and difficult process, with continual revision of ideas, is not in question.

References

Agre, P. E. & Chapman, D. (1987). Pengi: an implementation of a theory of activity. In *Proceedings of the Sixth National Conference on Artificial Intelligence*, pages 268–272, Seattle. AAAI.

Aube, M. & Senteni, A. (1996a). Emotions as commitments operators: a foundation for control structure in multi-agent systems. In *Proceedings of the Seventh European Workshop on Modelling Autonomous Agents in a Multi-Agents World, MAAMAW '96, Lecture Notes in Artificial Intelligence*. Springer-Verlag.

Aube, M. & Senteni, A. (1996b). What are emotions for? commitments management and regulation within animals/animats encounters. In Maes, P., Mataric, M., Meyer, J.-A., Pollack, J., & Wilson, S. W. (Eds.), *From Animals to Animats IV, Proceedings of the Fourth International Conference on the Simulation of Adaptive Behavior*, pages 264–271. The MIT Press.

Axelrod, R. & Hamilton, W. D. (1981). The evolution of cooperation. *Science*, 211:1390–1396.

Baldwin, J. M. (1896). A new factor in evolution. *American Naturalist*, 30:441–451, 536–553. Reprinted in *Adaptive Individuals in Evolving Populations: Models and Algorithms*, edited by R. K. Belew and M. Mitchell (SFI Studies in the Sciences of Complexity, Proc. Vol. XXVI, Addison-Wesley, Reading, MA, 1996).

Balkenius, C. (1995). *Natural intelligence in artificial creatures.* Lund University Cognitive Studies 37.

Bates, J. (1994). The role of emotion in believable agents. *Communications of the ACM*, 37(7):122–125.

Bates, J., Loyall, A. B., & Reilly, W. S. (1991). Broad agents. In *Paper presented at AAAI spring symposium on integrated intelligent architectures.* (Available in SIGART BULLETIN, 2(4), Aug. 1991, pp. 38–40).

Baum, E. B. (1996). Toward a model of mind as a laissez-faire economy of idiots. In *Proceedings of the Thirteenth International Conference on Machine Learning.*

Beaudoin, L. P. (1994). *Goal processing in autonomous agents.* PhD thesis, School of Computer Science, The University of Birmingham.

Beaudoin, L. P. (1995). Personal communication.

Beaudoin, L. P. & Sloman, A. (1993). A study of motive processing and attention. In A.Sloman, D.Hogg, G.Humphreys, Partridge, D., & Ramsay, A. (Eds.), *Prospects for Artificial Intelligence*, pages 229–238. Amsterdam: IOS Press.

Beer, R. D., Chiel, H. J., & Sterling, L. S. (1990). A biological perspective on autonomous agent design. In Maes, P. (Ed.), *Designing autonomous agents: Theory and practice from biology to engineering and back*, pages 169–186. Amsterdam: Elsevier Science Publishers.

Benson, S. & Nilsson, N. J. (1995). Reacting, planning, and learning in an autonomous agent. In Furukawa, K., Michie, D., & Muggleton, S. (Eds.), *Machine Intelligence 14*. Oxford: The Clarendon Press.

Bhaskar, R. (1978). *A Realist Theory of Science*. Hassocks, Sussex: The Harvester Press Ltd.

Boardman, J., Griffin, J., & Murray, O. (1993). *The Oxford History of the Classical World*. Oxford: Oxford University Press.

Boddy, M. & Dean, T. (1989). Solving time-dependent planning problems. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence, vol. 2*.

Boden, M. A. (1972). *Purposive Explanation in Psychology*. Hassocks Sussex England: The Harvester Press.

Bond, A. H. (1990). A computational model for organization of cooperating intelligent agents. In *Proc. of the Conference on Office information Systems*, pages 21–30. Cambridge, MA.

Botelho, L. M. & Coelho, H. (1996). Emotion-based attention shift in autonomous agents. In Muller, J. P., Wooldridge, M. J., & Jennings, N. R. (Eds.), *Intelligent Agents III, Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages*, Heidelberg. Springer-Verlag.

Bowlby, J. (1979). *The Making and Breaking of Affectional Bonds*. Tavistock Publications.

Bowlby, J. (1988). *A Secure Base*. Routledge.

Bowlby, J. (1991a). *Attachment and Loss: Volume I*. Penguin Books. First published in 1969.

Bowlby, J. (1991b). *Attachment and Loss: Volume III.* Penguin Books. First published in 1980.

Braitenburg, V. (1984). *Vehicles, experiments in synthetic psychology.* MIT Press.

Bratman, M. E., Israel, D. J., & Pollack, M. E. (1988). Plans and resource-bounded practical reasoning. *Computational Intelligence*, 4:349–355.

Brooks, R. A. (1990). Elephants don't play chess. *Robotics and Autonomous Systems*, 6:3–15.

Brooks, R. A. (1991a). Integrated systems based on behaviours. *SIGART Bulletin*, 2(4):46–50.

Brooks, R. A. (1991b). Intelligence without representation. *Artificial Intelligence*, 47:139–159.

Calhoun, C. & Solomon, R. C. (1984). *What is an Emotion?* Oxford University Press.

Chalmers, D. (1996). *The Conscious Mind: In Search of a Fundamental Theory.* Oxford University Press.

Chapman, D. (1989). Penguins can make cake. *AI Magazine*, 10(4):45–50.

Chapman, D. (1990). Vision, instruction and action. Technical Report 1204, Massachusetts Institute of Technology, Artificial Intelligence Laboratory.

Charniak, E. & McDermott, D. (1985). *Introduction to Artificial Intelligence.* Addison-Wesley Publishing Company.

Cichosz, P. (1994). Reinforcement learning algorithms based on the methods of temporal differences. Master's thesis, Institute of Computer Science, Warsaw University of Technology.

Cohen, J. & Stewart, I. (1994). *The collapse of chaos, discovering simplicity in a complex world.* Viking.

Cziko, G. (1995). *Without Miracles, universal selection and the second Darwinian revolution.* Cambridge, Massachusetts: The MIT Press.

Davis, D. N. (1996). Reactive and motivational agents: towards a collective minder. In *Agent Theories, Architectures, and Languages, the Third International Workshop*, Budapest, Hungary. ECAI-96.

Davis, D. N., Sloman, A., & Poli, R. (1995). Simulating agents and their environments. *AISB Quarterly*.

Dawkins, R. (1986). *The blind watchmaker*. Harlow: Longman.

Dawkins, R. (1990). *The selfish gene*. New York: Oxford Unviversity Press.

Dawson, M. R. W. & Shamanski, K. S. (1994). Connectionism, confusion, and cognitive science. *The Journal of Intelligent Systems*, 4:215–262.

Dennett, D. C. (1978). *Brainstorms: Philosophical Essays on Mind and Psychology*. Cambridge, MA: MIT Press.

Dennett, D. C. (1991). *Consciousness Explained*. Allen Lane: Penguin Press.

Dennett, D. C. (1996). Do animals have beliefs? In Roitblat, H. L. & Meyer, J.-A. (Eds.), *Comparative approaches to cognitive science*, pages 111–118. Cambridge, Massachusetts: "A Bradford Book" The MIT Press.

d'Inverno, M. & Luck, M. (1996). Formalizing the contract net protocol as a goal-directed system. In de Velde, W. V. & Perram, J. W. (Eds.), *Agents Breaking Away, Proceedings of the 7th European Workshop on MAAMAW*, Lecture Notes on Artificial Intelligence, No. 1308, pages 72–85. Berlin: Springer.

Donnart, J. Y. & Meyer, J. A. (1994). A hierarchical classifier system implementing a motivationally autonomous animat. In *From Animals to Animats III, Proceedings of the Third International Conference on the Simulation of Adaptive Behavior*. MIT Press.

Dorigo, M. & Bersini, H. (1994). A comparison of q-learning and classifier systems. In *From Animals to Animats III: Proceedings of the Third Conference on Simulation of Adaptive Behaviour*. MIT Press.

Dorigo, M. & Colombetti, M. (1993). Robot shaping: developing situated agents through learning. Technical Report TR-92-040, International Computer Science Institute, Berkeley, CA. Revised version.

Dorigo, M., Maniezzo, V., & Colorni, A. (1996). The ant system: optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics*, 26(1):1–13.

Doyle, J. (1989). Reasoning, representation, and rational self-government. In Ras, Z. W. (Ed.), *Methodologies for intelligent systems*, pages 367–380. New York: Elsevier Science Publishing.

Doyle, J. (1994). A reasoning economy for planning and replanning. In *Technical papers of the ARPA Planning Initiative Workshop.*

Dyer, M. G. (1987). Emotions and their computations: Three computer models. *Cognition and Emotion*, 1(3):323–347.

Etzioni, O. (1993). Intelligence without robots (a reply to brooks). *Artificial Intelligence Magazine.*

Fehling, M. R., Altman, A. M., & Wilber, B. M. (1989). The heuristic control virtual machine: an implementation of the schemer computational model of reflective, real-time problem-solving. In *Blackboard Architectures and Applications.* Academic Press, Inc.

Ferguson, I. A. (1992). *TouringMachines: An architecture for dunamic, rational, mobile agents.* PhD thesis, University of Cambridge Computer Laboratory, University of Cambridge. Technical report No. 273.

Firby, R. J. (1987). An investigation into reactive planning in complex domains. In *Proceedings of the Sixth National Conference on Artificial Intelligence*, pages 202–206, Seattle. AAAI.

Firby, R. J. (1989). *Adaptive execution in complex dynamic worlds.* PhD thesis, Department of Computer Science, Yale University.

Franklin, S. & Graesser, A. (1996). Is it an agent, or just a program?: A taxonomy for autonomous agents. In *Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages.* Springer-Verlag.

Freud, S. (1987). *On sexuality: three essays on the theory of sexuality and other works*, volume VII of *The Pelican Freud Library.* Penguin Books.

Freud, S. (1991). *On metapychology: the theory of psychoanalysis*, volume III of *The Penguin Freud Library.* Penguin Books.

Frijda, N. H. (1986). *The Emotions.* Cambridge: Cambridge University Press.

Frijda, N. H. (1995). The irrationality of emotions. Paper read at the conference on Philosophy and Psychology, University of Oxford, 1995.

Frijda, N. H. & Swagerman, J. (1987). Can computers feel? theory and design of an emotional system. *Cognition and Emotion*, 1:235–257.

Gasser, L. (1991). Social conceptions of knowledge and action: Dai foundations and open systems semantics. *Artificial Intelligence*, 47:107–138.

Georgeff, M. P. & Ingrand, F. F. (1989). Decision-making in an embedded reasoning system. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, pages 972–978, Detroit, MI. IJCAI.

Georgeff, M. P. & Lansky, A. L. (1989). Reactive reasoning and planning. In *Proceedings of the Sixth National Conference on Artificial Intelligence*, pages 677–682, Seattle. AAAI.

Gerson, E. M. (1976). On 'quality of life'. *American Sociological Review*, 41:793–806.

Goldfarb, L. & Nigam, S. (1994). The unified learning paradigm: a foundation for ai. In Hanovar, V. & Uhr, L. (Eds.), *Artificial Intelligence and Neural Networks: Steps towards Principled Integration*. Boston, MA: Academic Press.

Gouldner, A. W. (1960). The norm of reciprocity: a preliminary statement. *American Sociological Review*, 25(2):161–178.

Green, O. H. (1992). *The Emotions, a philosophical theory*. Kluwer Academic Publishers.

Hanks, S. & Firby, R. J. (1990). Issues and architectures for planning and execution. In *Proceedings of a Workshop on Innovative Approaches to Planning, Scheduling and Control*, San Diego, CA. DARPA.

Hayes-Roth, B. (1990). Architectural foundations for real-time performance in intelligent agents. *Journal of Real-Time Systems*, 2:99–125.

Hayes-Roth, B. (1991a). Evaluation of integrated agent architectures. *SIGART Bulletin*, 2(4):82–84.

Hayes-Roth, B. (1991b). An integrated architecture for intelligent agents. *SIGART Bulletin*, 2(4):79–81.

Hayes-Roth, B. (1993a). An architecture for adaptive intelligent systems. Technical Report 93-19, Knowledge Systems Laboratory, Department of Computer Science, Stanford University.

Hayes-Roth, B. (1993b). Intelligent control. *Artificial Intelligence*, 59:213–220.

Heckhausen, H. & Kuhl, J. (1985). From wishes to action: the dead ends and short cuts on the long way to action. In Frese, M. & Sabini, J. (Eds.), *Goal Directed Behavior: The Concept of Action in Psychology*. Lawrence Erlbaum.

Hofstadter, D. & Dennett, D. C. (1981). *The Mind's I: Fantasies and Reflections on Self and Soul*. Brighton: The Harvester Press Ltd.

Hofstadter, D. R. (1979). *Godel, Escher, Bach: an Eternal Golden Braid*. Penguin Books.

Holland, J. H. (1975). *Adaption in natural and artificial systems*. The MIT Press.

Holland, J. H. (1986). Escaping brittleness: the possibilities of general-purpose learning algorithms applied to parallel rule-based systems. In Michalski, R. S., Carbonell, J. G., & Mitchell, T. M. (Eds.), *Machine learning, an artificial intelligence approach*. Los Altos, California: Morgan Kaufmann.

Holland, J. H. (1995). *Hidden Order, how adaptation builds complexity*. Helix Books.

Holland, J. H., Holyoak, K. J., Nisbett, R. E., & Thagard, P. R. (1986). *Induction: processes of inference, learning and discovery*. The MIT Press.

Horvitz, E. J., Gregory, F. C., & Heckerman, D. E. (1989). Reflection and action under scarce resources: theoretical principles and empirical study. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence, vol. 2*.

Humphreys, M. (1996). Action selection methods using reinforcement learning. In Maes, P., Mataric, M., Meyer, J.-A., Pollack, J., & Wilson, S. W. (Eds.), *From Animals to Animats IV, Proceedings of the Fourth International Conference on the Simulation of Adaptive Behavior*. The MIT Press.

Jennings, N. (1996). Coordination techniques for distributed artificial intelligence. In O'Hare, G. & Jennings, N. (Eds.), *Foundations of distributed artificial intelligence*. John Wiley & Sons.

Johnson-Laird, P. N. (1988). *The Computer and the Mind, an introduction to cognitive science*. Fontana Press.

Kaelbling, L. P., Littman, M. L., & Moore, A. W. (1995). Reinforcement learning: a survey. In *Practice and Future of Autonomous Agents, volume 1*.

Kagan, J. (1978). On emotion and its development: a working paper. In Lewis, M. & Rosenblum, L. A. (Eds.), *The Development of Affect*. New York and London: Plenum Press.

Kiss, G. (1992). Variable coupling of agents to their environment: combining situated and symbolic automata. In Werner, E. & Demazeau, Y. (Eds.), *Decentralized A.I.-3*. Elsevier Science Publishers.

Kittock, J. E. (1995). Emergent conventions and the structure of multiagent systems. In Nadel, L. & Stein, D. (Eds.), *1993 Lectures in Complex Systems: the proceedings of the 1993 Complex Systems Summer School*, Santa Fe Institute Studies in the Sciences of Complexity Lecture Volume VI. Santa Fe Institute, Addison-Wesley Publishing Co.

Kovacs, T. (1996). Evolving optimal populations with xcs classifier systems. Technical Report CSR-96-17, School of Computer Science, University of Birmingham.

Kraemer, G. W. (1992). A psychological theory of attachment. *Behavioral and Brain Sciences*, 15:493–541.

Kuhl, J. (1992). A theory of self-regulation: action versus state orientation, self-discrimination, and some applications. *Applied Psychology: An International Review*, 41(2):97–129.

Kuhl, J. & Kraska, K. (1989). Self-regulation and metamotivation. In Kanfer, R., Ackerman, P. L., & Cudek, R. (Eds.), *Abilities, motivation, and methodology: The Minnesota Symposium on Individual Differences*, pages 343–374. Hillsdale, NJ: Lawrence Erlbaum Associates Inc.

Laird, J. E., Newell, A., & Rosenbloom, P. S. (1987). Soar: An architecture for general intelligence. *Artificial Intelligence*, 33:1–64.

Lakatos, I. (1970). Falsification and the methodology of scientific research programmes. In Lakatos, I. & Musgrave, A. (Eds.), *Criticism and the Growth of Knowledge*, pages 91–196. Cambridge University Press.

LeDoux, J. E. (1994). Emotion, memory and the brain. *Scientific American*, pages 32–39.

Lesser, V. R., Pavlin, J., & Durfee, E. H. (1989). Approximate processing in real-time problem solving. In Clearwater, S. (Ed.), *Blackboard Architectures and*

Applications. Academic Press, Inc. Reprinted with permission from *AI Magazine*, vol. 9, no. 1, Spring 1988.

Lichtenstein, P. M. (1983). *An introduction to post-Keynesian and Marxian theories of value and price*. London and Basingstoke: Macmillan Press.

Liebniz, G. W. (1991). *Monadology*. Pittsburgh: 1991, University of Pittsburgh Press.

Loyall, A. B. & Bates, J. (1991). Hap – a reactive, adaptive architecture for agents. Technical Report CMU-CS-91-147, School of Computer Science, Carnegie Mellon University.

Lyons, D. M. & Hendriks, A. J. (1992). Reactive planning. In Shapiro, S. (Ed.), *Encyclopedia of Artificial Intelligence* (second ed.). John Wiley & Sons.

Mackintosh, N. J. (1983). *Conditioning and associative learning*. Oxford Psychology Series No. 3. Oxford: Oxford University Press.

MacLean, A., Young, R. M., Bellotti, V. M. E., & Morgan, T. P. (1991). Questions, options, and criteria: Elements of design space analysis. *Human-Computer Interaction*, 6:201–250.

Maes, P. (1989). The dynamics of action selection. In *Proceedings of the IJCAI-89 Conference*, Detroit.

Maes, P. (1990). Guest editorial: designing autonomous agents. In Maes, P. (Ed.), *Designing Autonomous Agents: theory and practice from biology to engineering and back*, pages 1–2. Amsterdam: Elsevier Science Publishers.

Maes, P. (1991). The agent network architecture. *SIGART Bulletin*, 2(4):115–120.

Marimon, R., McGrattan, E., & Sargent, T. J. (1990). Money as a medium of exchange in an economy with artificially intelligent agents. *Journal of Economic Dyanmics and Control*, 14:329–373.

Marjanovic, M., Scassellati, B., & Williamson, M. (1996). Self-taught visually-guided pointing for a humanoid robot. In Maes, P., Mataric, M., Meyer, J.-A., Pollack, J., & Wilson, S. W. (Eds.), *From Animals to Animats IV, Proceedings of the Fourth International Conference on the Simulation of Adaptive Behavior*, pages 35–44. The MIT Press.

Marx, K. (1887). *Capital, a critical analysis of capitalist production*, volume 1. Lawrence and Wishart. Lawrence and Wishart edition, 1970.

Marx, K. & Engels, F. (1968). *Karl Marx and Friedrich Engels selected works.* Lawrence and Wishart.

McCallum, A. R. (1996). Learning to use selective attention and short-term memory in sequential tasks. In Maes, P., Mataric, M., Meyer, J.-A., Pollack, J., & Wilson, S. W. (Eds.), *From Animals to Animats IV, Proceedings of the Fourth International Conference on the Simulation of Adaptive Behavior*. The MIT Press.

McCarthy, J. (1979). Ascribing mental qualities to machines. In Ringle, M. (Ed.), *Philosophical Perspectives in Artificial Intelligence*. Sussex: Harvester Press.

McCarthy, J. (1995). Making robots conscious of their mental states. In *AAAI Spring Symposium on Representing Mental States and Mechanisms.* Accessible via http://www-formal.stanford.edu/jmc/.

Meuller, R. E. (1990). *Daydreaming in humans and machines.* New Jersey: Ablex Publishing Corporation.

Miceli, M. & Castelfranchi, C. (1989). A cognitive approach to values. *Journal for the Theory of Social Behaviour*, 19:169–193.

Miller, G. A., Galanter, E., & Pribram, K. H. (1970). *Plans and the structure of behaviour.* Holt International Edition.

Minsky, M. L. (1987). *The Society of Mind.* London: William Heinemann Ltd.

Miranda, L. P. (1996). Deciding, planning and practical reasoning: elements towards a cognitive architecture. Technical Report CSRP-96-6, School of Computer Science and Cognitive Science Research Centre.

Moffat, D. & Frijda, N. H. (1995). Where there's a will there's an agent. In Wooldridge, M. & Jennings, N. (Eds.), *Intelligent Agents.* Berlin: Springer-Verlag.

Mook, D. G. (1987). *Motivation, the organization of action.* Edinburgh: Edinburgh University Press.

Morris, H. C. (1991). On the feasibility of computational artificial life – a reply to critics. In Meyer, J. A. & Wilson, S. W. (Eds.), *From Animals to Animats,* *Proceedings of the First International Conference on the Simulation of Adaptive Behavior*. MIT Press.

Muggleton, S. (1994). Logic and learning: Turing's legacy. In Furukawa, K., Michie, D., & Muggleton, S. (Eds.), *Machine Intelligence 13*. Oxford: Oxford University Press.

Nagel, E. & Newman, J. R. (1958). *Godel's proof*. London: Routledge and Kegan Paul.

Nagel, T. (1974). What is it like to be a bat? *The Philosophical Review*, LXXXIII(4):435–50.

Newell, A. (1990). *Unified Theories of Cognition*. Cambridge, MA: Harvard University Press.

Nietzsche, F. (1896). *Thus Spoke Zarathustra: a book for everyone and no one*. Harmondsworth: Penguin. Penguin Edition 1969. Translated with an introduction by R.J. Hollingdale.

Nilsson, N. J. (1984). Shakey the robot. Technical Report Technical note 323, SRI International, Menlo Park, California.

Nilsson, N. J. (1994). Teleo-reactive programs for agent control. *Journal of Artificial Intelligence Research*, 1:139–158.

Norman, T. J. (1994). Position paper: motivated goal and action selection. Presented at AISB workshop, Models or Behaviours, which way forward for robotics? Leeds, April 1994, and University College London research note RN/94/18.

Norman, T. J. (1996). *Motivation-based direction of planning attention in agents with goal-autonomy*. PhD thesis, Department of Computer Science, University College London.

Norman, T. J. & Long, D. P. (1995). Goal creation in motivated agents. In Wooldridge, M. J. & Jennings, N. R. (Eds.), *Intelligent Agents: Proceedings of the ECAI-94 Workshop on Agent Theories, Architectures, and Languages*, pages 277–290. Springer-Verlag. Volume 890 of Lecture Notes in Artificial Intelligence.

Oatley, K. (1992). *Best Laid Schemes*. Studies in Emotion and Social Interaction. Cambridge: Cambridge University Press.

Oatley, K. & Johnson-Laird, P. N. (1985). Sketch for a cognitive theory of emotions. Technical Report CSRP 045, School of Cognitive Science, University of Sussex.

Ortony, A., Clore, G. L., & Collins, A. (1988). *The Cognitive Structure of the Emotions*. New York: Cambridge University Press.

Palmer, S. E. & Kimchi, R. (1984). The information processing approach to cognition. In Knapp, T. J. & Robertson, L. C. (Eds.), *Approaches to Cognition: Contrasts and Controversies*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Panskepp, J. (1980). Les circuits des émotions. *Science et Vie, Horssérie*, 168:58–67. Translated by Michel Aubé.

Parsons, S. D. & Jennings, N. R. (1996). Negotiations through argumentation – a preliminary report. In *Proceedings of the Second International Conference on Multi-Agent Systems*.

Penrose, R. (1989). *The Emperor's New Mind: Concerning Computers, Minds and the Laws of Physics*. Oxford: Oxford University Press.

Pepper, S. C. (1958). *The Sources of Value*. University of California Press.

Pfeifer, R. (1994). The 'fungus eater approach' to emotion: a view from artificial intelligence. Cognitive Studies*: Bulletin of the Japanese Cognitive Science Society*, 1(2):42–57. Extended and revised version of an invited talk at *AISB-91*, Leeds, UK. Also available as a technical report from AI Lab, Institute for Informatics, University of Zurich-Irchel.

Popper, K. R. (1963). *Conjectures and Refutations: the growth of scientific knowledge*. London: Routledge & Kegan Paul.

Powers, W. T. (1988). *Living Control Systems, selected papers of William T. Powers*. Kentucky: The Control Systems Group.

Pryor, L. (1994). *Opportunities and planning in an unpredictable world*. PhD thesis, Northwestern University.

Pryor, L. & Collins, G. (1992). Reference features as guides to reasoning about opportunities. In *Proceedings of the Fourteenth Annual Conference of the Cognitive Science Society*. Lawrence Erlbaum Associates.

Pryor, L. & Collins, G. (1993). Cassandra: planning for contingencies. Technical Report 41, The Institute for Learning Sciences, Northwestern University.

Rao, A. S. & Georgeff, M. P. (1991a). An abstract architecture for rational agents. In *Proceedings of the Third International Conference on Knowledge Representation and Reasoning*, Boston.

Rao, A. S. & Georgeff, M. P. (1991b). Modeling rational agents within a BDI-architecture. In Allen, J., Fikes, R., & Sandewall, E. (Eds.), *Proceedings of the 2nd International Conference on Principles of Knowledge Representation and Reasoning*, pages 473–484, Cambridge, MA, USA. Morgan Kaufmann Publishers.

Read, T. & Sloman, A. (1993). The terminological pitfalls of studying emotion. In *Workshop on Architectures Underlying Motivation and Emotion WAUME93*, Birmingham, UK.

Read, T. R. (1995). *The use of systemic design to analyse Gray's theory of emotion.* PhD thesis, School of Computer Science, The University of Birmingham.

Reilly, W. S. (1993). Emotions as part of a broad agent architecture. In *Proceedings of the Workshop on Architectures Underlying Motivation and Emotion, WAUME93*, Birmingham, UK.

Rey, G. (1997). *Contemporary philosophy of mind: a contentiously classical approach.* Cambridge, Mass; Oxford: Blackwell.

Riolo, R. L. (1988). *A package of domain independent subroutines for implementing classifier systems in arbitrary, user-defined environments.* Logic of Computers Group, Division of Computer Science and Engineering, University of Michigan.

Rubin, I. I. (1988). *Essays on Marx's Theory of Value.* Montreal: Black Rose Books. Originally published 1928.

Ryle, G. (1949). *The Concept of Mind.* Hutchinson.

Sartre, J. P. (1934). *Esquisse d'une théorie phénomenlogique des émotions (The Emotions).* Philosophical Library. Paris: Hermann. Translated 1948.

Schaerf, A., Shoham, Y., & Tennenholtz, M. (1995). Adaptive load balancing: a study in multi-agent learning. *Journal of Artificial Intelligence Research*, 2:475–500.

Scherer, K. R. (1993). Studying the emotion-antecedent appraisal process: an expert system approach. *Cognition and Emotion*, 7:325–355.

Searle, J. R. (1980). Minds brains and programs. *The Behavioral and Brain Sciences*, 3(3).

Shing, E. (1996). *Computational mechanisms for constraining learning.* PhD thesis, School of Computer Science, The University of Birmingham.

Shoham, Y. & Tennenholtz, M. (1994). Co-learning and the evolution of social activity. Technical Report CS-TR-94-1511, Robotics Laboratory, Department of Computer Science, Stanford University.

Simon, H. A. (1967). Motivational and emotional controls of cognition. Reprinted in *Models of Thought,* Yale University Press, 29–38, 1979.

Simon, H. A. (1981a). Cognitive science: The newest science of the artificial. In Norman, D. A. (Ed.), *Perspectives on Cognitive Science*, pages 13–26. Hillsdale, New Jersey: Lawrence Erlbaum Associates.

Simon, H. A. (1981b). *The Sciences of the Artificial* (second ed.). The MIT Press.

Simon, H. A. (1982). Comments. In Clark, M. S. & Fiske, S. T. (Eds.), *Affect and cognition.* Hillsdale, New Jersey: Lawrence Erlbaum Associates.

Simon, H. A. (1990). A mechanism for social selection and successful altruism. *Science*, 250:1665–1668.

Simon, H. A. (1995). Artificial intelligence: an empirical science. *Artificial Intelligence*, 77:95–127.

Sloman, A. (1969). How to derive "better" from "is". *American Philosophical Quarterly*, 6(1):43–52.

Sloman, A. (1978). *The Computer Revolution in Philosophy: Philosophy, Science and Models of Mind.* Hassocks, Sussex: Harvester Press (and Humanities Press).

Sloman, A. (1985). Real time multiple-motive expert systems. In Merry, M. (Ed.), *Expert Systems 85*, pages 1–13. Cambridge: Cambridge University Press.

Sloman, A. (1987). Motives mechanisms and emotions. *Cognition and Emotion*, 1(3):217–234. Reprinted in M.A.Boden (ed), *The Philosophy of Artificial Intelligence*, OUP, 1990.

Sloman, A. (1990). Notes on consciousness. *AISB Quarterly*, (72):8–14. Also presented at Rockefeller foundation workshop on consciousness, Villa Serbelloni, Bellagio March 1990, organiser D.C.Dennett.

Sloman, A. (1992). Prolegomena to a theory of communication and affect. In Ortony, A., Slack, J., & Stock, O. (Eds.), *Communication from an Artificial Intelligence Perspective: Theoretical and Applied Issues*, pages 229–260. Heidelberg, Germany: Springer.

Sloman, A. (1993a). Definition of metamanagement. Internal Cognition and Affect project document.

Sloman, A. (1993b). The mind as a control system. In Hookway, C. & Peterson, D. (Eds.), *Philosophy and the Cognitive Sciences*, pages 69–110. Cambridge University Press.

Sloman, A. (1993c). Prospects for ai as the general science of intelligence. In A.Sloman, D.Hogg, G.Humphreys, Partridge, D., & Ramsay, A. (Eds.), *Prospects for Artificial Intelligence*, pages 1–10. Amsterdam: IOS Press.

Sloman, A. (1993d). Why we need to study architectures. In *Workshop on Architectures Underlying Motivation and Emotion WAUME93*, Birmingham UK.

Sloman, A. (1994a). Explorations in design space. In *Proceedings 11th European Conference on AI*, Amsterdam.

Sloman, A. (1994b). Semantics in an intelligent control system. *Philosophical Transactions of the Royal Society: Physical Sciences and Engineering*, 349(1689):43–58.

Sloman, A. (1995a). Exploring design space & niche space. In *Proc. 5th Scandinavian Conf. on AI, Trondheim*, Amsterdam. IOS Press.

Sloman, A. (1995b). Musings on the roles of logical and non-logical representations in intelligence. In Glasgow, J. & Hari Narayanan, C. (Eds.), *Diagrammatic Reasoning: Computational and Cognitive Perspectives*. AAAI Press.

Sloman, A. (1995c). Poprulebase help file. Available at URL ftp://ftp.cs.bham.ac.uk/pub/dist/poplog/prb/help/poprulebase.

Sloman, A. (1995d). Rulesystems help file. Available at URL ftp://ftp.cs.bham.ac.uk/pub/dist/poplog/prb/help/rulesystems.

Sloman, A. (1995e). Sim_agent help file. Available at URL ftp://ftp.cs.bham.ac.uk/pub/dist/poplog/sim/help/sim_agent.

Sloman, A. (1995f). Sim_agent web-page. Available at URL http://www.cs.bham.ac.uk/ axs/cog_affect/sim_agent.html.

Sloman, A. (1995g). What sort of control system is able to have a personality. Available at URL ftp://ftp.cs.bham.ac.uk/pub/groups/cog_affect/ Aaron.Sloman.vienna.ps.Z, (Presented at Workshop on Designing personalities for synthetic actors, Vienna, June 1995).

Sloman, A. (1996a). Personal communication.

Sloman, A. (1996b). Road crossing example. Personal communication.

Sloman, A. (1996c). A systems approach to consciousness. Lecture presented at the Royal Society of Arts on 26th Feb 1996. The slides used for the lecture available at
http://www.cs.bham.ac.uk/ axs/misc/consciousness.lecture.ps and a summary of the talk at
http://www.cs.bham.ac.uk/ axs/misc/consciousness.rsa.text.

Sloman, A. (1996d). Towards a general theory of representations. In Peterson, D. (Ed.), *Forms of Representation*. Intellect Books.

Sloman, A. (1996e). What kinds of machine can have emotions? Paper presented at the British Association Annual Festival. Summary of talk available at http://www.cs.bham.ac.uk/ axs/misc/baas.text.

Sloman, A. (1996f). What sort of architecture is required for a human-like agent? Technical Report CSRP-96-12, School of Computer Science and Cognitive Science Research Centre. Invited talk for Cognitive Modelling Workshop at AAAI 96.

Sloman, A., Beaudoin, L. P., & Wright, I. P. (1994). Computational modeling of motive-management processes. In Frijda, N. (Ed.), *Proceedings of the Conference of the International Society for Research in Emotions*, Cambridge. ISRE Publications.

Sloman, A. & Croucher, M. (1981). Why robots will have emotions. In *Proceedings of the Seventh International Joint Conference on Aritificial Intelligence*, Vancouver.

Sloman, A. & Poli, R. (1995). Sim_agent: a toolkit for exploring agent designs. In Wooldridge, M., Mueller, J., & Tambe, M. (Eds.), *Intelligent Agents Vol II*, pages 392–407. Springer-Verlag.

Sloman, A., Shing, E., Read, T., & Beaudoin, L. (1992). Six types of design decision. Notes from a *Cognition and Affect* project meeting. Department of Computer Science, University of Birmingham.

Smith, P. K. & Cowie, H. (1991). *Understanding children's development*. Oxford: Blackwell Publishers.

Smith, R. G. (1980). The contract net protocol: high-level communication and control in a distributed problem solver. *IEEE Transactions on Computers*, 29(12):1104–1113.

Smith, R. G. & Davis, R. (1981). Frameworks for cooperation in distributed problem solving. *IEEE Transactions on Systems, Man and Cybernetics*, 11(1):61–70.

Sonnemans, J. & Frijda, N. H. (1994). The structure of subjective emotional intensity. *Cognition and Emotion*, 8(4):329–350.

Steels, L. (1994). The artificial life roots of artificial intelligence. *Artificial Life Journal*, 1(1).

Strongman, K. T. (1987). *The Psychology of Emotion*. John Wiley and Sons Ltd.

Sutton, R. S. (1991). Dyna, an integrated architecure for learning, planning, and reacting. In *Working Notes of the 1991 AAAI Spring Symposium*, pages 151–155. Also in SIGART Bulletin, 2, pp. 160–163, 1991.

Toates, F. (1986). *Motivational systems*. Cambrdige University Press.

Watkins, C. & Dayan, P. (1992). Technical note: Q-learning. In *Machine Learning 8*, pages 279–292.

Webmaster, A. (1996). The act web. Home page for the ACT-R research group, Carnegie Mellon University. URL http://sands.psy.cmu.edu/ACT/act-home.html. ACT Webmaster at Webmaster@sands.psy.cmu.edu.

Weiss, G. (1995). Distributed reinforcement learning. *Robotics and Autonomous Systems*, 15:135–142.

Wellman, M. (1995). Market-oriented programming: some early lessons. In Clearwater, S. (Ed.), *Market-Based Control: A Paradigm for Distributed Resource Allocation*. World Scientific.

Wierzbicka, A. (1992). Defining emotion concepts. *Cognitive Science*, 16:539–581.

Wilson, S. W. (1994). Zcs: A zeroth level classifier system. *Evolutionary Computation*, 2(1):1–18.

Wilson, S. W. (1995). Classifier fitness based on accuracy. *Evolutionary Computation*, 3(2):149–185.

Wilson, S. W. & Goldberg, D. E. (1989). A critical review of classifier systems. In *Proceedings of the Third International Conference on Genetic Algorithms*, pages 244–255, Los Altos, California. Morgan Kaufmann.

Wooldridge, M. & Jennings, N. R. (1995). Agent theories, architectures, and languages. In Wooldridge, M. & Jennings, N. R. (Eds.), *Intelligent Agents*, pages 1–22. Berlin: Springer-Verlag.

Wright, I. P. (1994). An emotional agent: the detection and control of emergent states in autonomous resource-bounded agents. Technical Report RP-94-21, School of Computer Science and Cognitive Science Research Centre, University of Birmingham.

Wright, I. P. (1995). Cognition and currency flow. Unpublished research notes. Department of Computer Science, University of Birmingham.

Wright, I. P. (1996a). Design requirements for a computational libidinal economy. Technical Report CSRP-96-11, School of Computer Science and Cognitive Science Research Centre, University of Birmingham. Submitted to *Cognition and Emotion*.

Wright, I. P. (1996b). Reinforcement learning and animat emotions. In Maes, P., Mataric, M., Meyer, J.-A., Pollack, J., & Wilson, S. W. (Eds.), *From Animals to Animats IV, Proceedings of the Fourth International Conference on the Simulation of Adaptive Behavior*, pages 272–281. The MIT Press.

Wright, I. P. & Aube, M. (1997). The society of mind requires an economy of mind. Technical Report CSRP-97-6, School of Computer Science and Cognitive Science Research Centre, University of Birmingham. Submitted to *MAAMAW '97*.

Wright, I. P., Sloman, A., & Beaudoin, L. P. (1996). Towards a design based analysis of emotional episodes. *Philosophy Psychiatry and Psychology*, 3(2):101–137.

Part V

APPENDICES

Appendix A

Implementation details of Minder1

A.1 Motives

The general form of a motive is:

```
[MOTIVE motive      <motive descriptor>
        insistence  <insistence value {0..1}]>
        status      <status descriptor {sub,surfacing,suspended,active}>
        plan        <list of plan steps>
        trp         <list of current TR program actions>
        importance  <importance descriptor {normal,low}>
]
```

The seven types of motive in Minder1 are:

1. **[MOTIVE motive [recharge ?Obj] insistence ?insist status sub plan [] trp []]**: A motive to recharge the specified minibot. Generated if Minder1 believes that an object has low charge.

2. **[MOTIVE motive [enclose ?Obj] insistence ?insist status sub plan [] trp []]**: A motive to move the specified minibot to the northern area of the nursery behind the line of fences. Generated if the enclosure has been built and Minder1 believes a minibot is south of it. (Helps prevent minibots from falling into ditches.)

3. **[MOTIVE motive [dismiss ?Obj] insistence ?insist status sub plan [] trp []]**: A motive to move the specified minibot to the dismissal point. Generated

if Minder1 believes a minibot has fallen into a ditch.

4. **[MOTIVE motive [visit ?Obj] insistence ?insist status sub plan** [] **trp** []]: A motive to visit the specified ditch. Generated periodically. (Patrols the ditches to spot minibots that might fall into them.)

5. **[MOTIVE motive [no_maginot] insistence ?insist status sub plan** [] **trp** []]: A motive to build an enclosure. Generated if an enclosure has not been built.

6. **[MOTIVE motive [save ?Obj2 ?Obj1] insistence ?insist status sub plan** [] **trp** []]: A motive to move the specified minibot to a safe distance from the specified ditch. Generated if Minder1 believes a minibot is too close to a ditch.

7. **[MOTIVE motive [default] insistence ?insist status sub plan** [] **trp** []]: A motive to wander around the nursery. Always generated (a motive of last resort).

A.2 List of basic actions

Minder1 has a set of basic actions that are directly executable by effectors. However, they can fail; for example, Minder1 may attempt to move forward but cannot due to an obstacle. The basic actions are:

1. **move**: Move forward in current direction. Takes no arguments.

2. **rotate ?To**: Rotates to new direction as specified by argument.

3. **setspeed ?To**: Set new travelling speed as specified by argument.

4. **grab_object ?It**: Attempt to grab object identified by argument, for example "minibot2". (This action can fail if the object is not within reach.)

5. **drop ?It**: Drop object identified by argument.

6. **rotate_bar ?It ?To**: Rotate specified fence to specified direction. (Minder1 must be holding fence identified by first argument.)

7. **charge ?It ?With**: Charge specified object with specified object, for example "charge minibot2 gas1". (This action can fail if first object not held or second object not within reach.)

8. **dismiss ?It ?With**: Dismiss specified object at specified object, for example "dismiss minibot2 exit1".

A.3 List of prestored plans

Plans consist of teleo-reactive program (TRP) plan steps, and most plans have a single plan step. There is a unique plan for each motive, which is a major simplification. Listed below are the seven types of motive and their associated plans.

1. **motive** [**recharge ??Params**] **==>**
plan [[**charge_object ??Params 200**]]:

2. **motive** [**below_maginot ??Params**] **==>**
plan [[**take_object ??Params exit1**]]:

3. **motive** [**dismiss ??Params**] **==>**
plan [[**dismiss_object ??Params**]]:

4. **motive** [**visit ??Params**] **==>**
plan [[**drop**] [**goto_object ??Params**]]:

5. **motive** [**no_maginot ??Params**] **==>**
plan [[**make_wall 15 60 0 first**] [**make_wall 40 60 0 second**] [**make_wall 65 60 0 third**] [**make_wall 90 60 0 fourth**]]:

6. **motive** [**save ??Params**] **==>**
plan [[**put_safe ??Params**]]:

7. **motive** [**default ??Params**] **==>**
plan [[**drop**] [**search**]]:

Minder1 has two metaplans that specify internal management operations.

8. **DECIDE**: Determines the importance of a motive.

9. **GET_PLAN**: Retrieves the correct plan for a motive from the plan library.

A.4 List of TR programs

Each TRP consists of a set of production rules. Rule conditions match items of knowledge and rule actions can be calls to further TRPs, recursive calls to the same TRP, internal operations that schedule basic actions, or assertions of temporary beliefs to facilitate 'reasoning', for example reasoning about available fences when building an enclosure. Each TRP name and associated arguments is listed, followed by a short description of its function and the basic actions and other TRPs it may call.

1. **TRP_goto ?X ?Y**: Move to the specified coordinates. Uses [move, setspeed, rotate].

2. **TRP_amble ?X ?Y**: Move to the specified coordinates while avoiding ob-

stacles. Calls [1, 2].

3. **TRP_goto_object ?Obj**: Move to the specified object. Calls [2, 7].

4. **TRP_grab_object ?Obj**: Pick up the specified object. Uses [grab_object]; calls [3, 6].

5. **TRP_take_object ?Obj1 ?Obj2**: Take the specified object to the second specified object. Calls [3, 4].

6. **TRP_drop ?Obj**: Drop the specified object. Uses [drop].

7. **TRP_search**: Search the nursery. Calls [2].

8. **TRP_place_bar ?Obj ?X ?Y ?Heading**: Place specified fence at specified coordinates in specified direction. Uses [rotate_bar]; calls [6, 2].

9. **TRP_grab_wall**: Pick up a fence that can serve as a wall. Calls [6, 4, 7].

10. **TRP_make_wall ?X ?Y ?Heading ?Side**: Place a fence so to serve as a wall. Calls [8, 9].

11. **TRP_charge_object ?Obj ?Level**: Charge specified object to specified level. Uses [charge]; calls [6, 5].

12. **TRP_dismiss_object ?Obj**: Remove specified object from the nursery. Uses [dismiss]; calls [5].

13. **TRP_put_safe ?Obj1 ?Obj2**: Remove specified object to a safe distance from specified ditch. Calls [6, 2, 4].

A.5 Example trace

The following short trace shows the state transitions of perturbing motives involved in a cycle of 'rumination' (see section 4.3.2).

```
======================= end of cycle 641 ===================
 ** [[Surfacing --
                [MOTIVE motive
                        [dismiss minibot4]
                        insistence 0.15 status sub plan
                        [[decide] [get_plan]]
                        trp
                        [done decide]
                        importance low]]]
 ** [[Diving --
             [MOTIVE motive
                     [dismiss minibot9]
```

```
                    insistence 0 status suspended plan
                    [[decide] [get_plan]]
                    trp
                    [done decide]
                    importance low]]]
** [[Management
     rejects --
     [MOTIVE motive
             [dismiss minibot7]
             insistence 0.15 status active plan
             [[decide] [get_plan]]
             trp
             [done decide]
             importance low]]]

======================= end of cycle 642 ==================
** [[Surfacing --
               [MOTIVE motive
                       [dismiss minibot9]
                       insistence 0.15 status sub plan
...
...
```

A.6 Example code

Example code is included to show how to implement mechanisms using the SIM_AGENT syntax and production rule semantics. The first example shows some code that generates a motive if Minder1 has a belief that any minibot is too near a ditch. The second example shows how the teleoreactive program described in figure 5.3 was actually implemented. Comments are provided throughout.

An example generactivator:

```
;;; G_near_ditch----------------------------------------------------

;;; rules-----------------------------------------------------------

;;; This generactivator is composed of one ruleset that is itself
;;; composed of a number of rules. Each rule contains a condition
```

```
;;; part that can match with items in Minder1's database or memory
;;; store, and an action part that can place new items in the database.

define :ruleset G_near_ditch;

rule grule_near_ditch_remove_r1
;;; This rule removes a generated motive if its rationale no longer
;;; holds, that is the minibot has fallen into the ditch (it is too
;;; late to do anything about it!)

    [MOTIVE motive [save ?Obj2 ?Obj1] ==][ ->> Motive ]
        ;;; This line matches a declarative representation of a motive
        ;;; in Minder1's database.
    [belief == name ?Obj1 type minibot status dead ==]
        ;;; And a belief about a minibot of the same name as the
        ;;; matched motive.
==>
    [DEL ?Motive]
        ;;; If the conditions match items in the database then
        ;;; there is a motive to save a minibot that has already
        ;;; fallen into a ditch; therefore remove it.

rule grule_near_ditch_remove_r2
;;; This rule removes a generated motive if the insistence is
;;; computed to be zero, that is the minibot has moved out of
;;; the danger zone of its own accord.

    [MOTIVE motive [save ?Obj2 ?Obj1] insistence ?Insistence ==]
        [ ->> Motive ]
    [WHERE Insistence = 0 ]
==>
    [DEL ?Motive]

rule grule_near_ditch_add
;;; This rule generates a motive if there is a minibot too close
;;; to a ditch, and a motive to save it has not already been
;;; generated.

    [belief == name ?Obj1 type minibot status alive == x ?X y ?Y ==
        held false ==]
    [belief == name ?Obj2 type ditch == x ?X1 y ?Y1 == poly_space ?Polygon ==]
    [WHERE
        is_near_ditch(Polygon, X1, Y1, X, Y, near_ditch_proximity) ]
    [NOT MOTIVE motive [save ?Obj2 ?Obj1] ==]
==>
```

```
    [LVARS insist]
    [POP11 near_ditch_insistence(Polygon, X1, Y1, X, Y) -> insist; ]
        ;;; The insistence value is computed for this motive
        ;;; (near_ditch_insistence is a function defined elsewhere).
    [MOTIVE motive [save ?Obj2 ?Obj1] insistence ?insist status sub]
        ;;; Place the new motive in the database: it is of status sub
        ;;; and becomes a candidate for surfacing.

rule grule_near_ditch_change
;;; This rule recomputes the insistence values of previously generated
;;; motives (the reactivation part of generactivation).

    [MOTIVE motive [save ?Obj2 ?Obj1] insistence ?Insistence ==]
    [belief == name ?Obj1 type minibot status alive == x ?X y ?Y ==
        held false ==]
    [belief == name ?Obj2 type ditch == x ?X1 y ?Y1 == poly_space ?Polygon ==]
    [LVARS insist]
    [WHERE
        near_ditch_insistence(Polygon, X1, Y1, X, Y) -> insist;
        insist /= Insistence ]
==>
    [MODIFY 1 insistence ?insist]
        ;;; Modify the insistence value of the existing motive.

enddefine;
```

An example TR program:

```
;;; TRP_charge_object ----------------------------------------------

;;; rules -------------------------------------------------------------

;;; Note that these rules satisfy the regression condition; therefore,
;;; the order of the rules matter. For instance, the topmost rule
;;; checks that the action has been accomplished and simply notes this
;;; fact (the null action in figure~\ref{fig:trp}).

define :ruleset TRP_charge_object;

rule TR_charge_object_r1
    [MOTIVE motive == status active == trp [charge_object ?Obj ?Level] ==]
        ;;; There exists a motive to charge an object.
    [NOT held ?Obj]
        ;;; Minder1 is not holding the object.
    [new_sense_datum == name ?Obj == charge ?Level2 ==]
    [WHERE Level2 >= Level ]
        ;;; Minder1 can sense that the object has been recharged to
```

```
        ;;; the appropriate level.
     ==>
    [MODIFY 1 trp [done charge_object ?Obj ?Level]]
        ;;; Therefore, the task has been accomplished.
    [STOP]

rule TR_charge_object_r2
    [MOTIVE motive == status active == trp [charge_object ?Obj ?Level] ==]
        ;;; There exists a motive to charge an object.
    [held ?Obj]
        ;;; Minder1 is holding the object.
    [new_sense_datum == name ?Obj == charge ?Level2 ==]
    [WHERE Level2 >= Level ]
        ;;; Minder1 can sense that the object has been recharged to
        ;;; the appropriate level.
     ==>
    [MODIFY 1 trp [drop ?Obj]]
    [POP11
        prb_run(TRP_drop, sim_myself.sim_data, false);
    ]
        ;;; Therefore, drop the object because it is sufficiently
        ;;; charged. The call to prb_run is a recursive call to
        ;;; another TR program that drops the specified object.
    [STOP]

rule TR_charge_object_r3
    [MOTIVE motive == status active == trp [charge_object ?Obj ?Level] ==]
        ;;; There exists a motive to charge an object.
    [held ?Obj]
        ;;; Minder1 is holding the object.
    [new_sense_datum == name ?Obj == x ?X y ?Y ==]
    [new_sense_datum == name gas1 == x ?XX y ?YY ==]
    [WHERE close_enough(X, Y, XX, YY) ]
        ;;; Minder1 can sense that the object is near enough to the
        ;;; recharge point (gas1) for recharging.
     ==>
    [closure charge ?Obj gas1]
        ;;; Call a primitive action to charge the object at the
        ;;; recharge point. Note that this action will be repeated
        ;;; on subsequent cycles until rule r2 evaluates to true.
    [MODIFY 1 trp [stop]]
    [STOP]

rule TR_charge_object_r4
    [MOTIVE motive == status active == trp [charge_object ?Obj ?Level] ==]
```

```
        ;;; There exists a motive to charge an object.
     ==>
    [MODIFY 1 trp [take_object ?Obj gas1]]
    [POP11
        prb_run(TRP_take_object, sim_myself.sim_data, false);
    ]
        ;;; Recursive call to another TR program, take_object, that
        ;;; will itself call other TR programs that will attempt
        ;;; to locate the specified object, pick it up, and take it to
        ;;; the recharge point (or search for the recharge point it
        ;;; its location is unknown).
    [STOP]

enddefine;
```

Appendix B

The labour theory of value and the ability to buy processing power

The statement that value is an ability to buy processing power is based on Marx's labour theory of value (LTV), but is agnostic about its eventual status as a theory of economic value. A good, but technical, exposition of the LTV can be found in (Rubin, 1988).

B.1 The labour theory of value

The LTV holds that the exchange-value (or, simply, value) of a commodity represents the amount of social labour that is necessary to make it. A number of points need to be immediately made.

First, the value of a commodity is not its price. The price of a particular commodity at a particular moment in time can be determined by many factors, for instance local supply and demand. These factors cause prices to regularly deviate from values. In a simple commodity economy [1] it is the value of commodities that prices tend to gravitate towards. This is not to imply that the value of a commodity is static.

Supply and demand or costs of production alone cannot explain the prices of

[1] A simple commodity economy is a theoretical idealisation that allows economists to reason about value and price without needing to take into account such complications as price-fixing, cartels, monopolies, taxation, trade tariffs and so forth.

commodities: for example, the supply and demand of steering wheels and cars may be identical, but their prices will differ; explaining the difference of prices in terms of a difference in costs of production only postpones a consideration of what determines the price of the component commodities, including the cost of labour.

Second, value does not represent actual labour but abstract, homogenised, *social* labour. For example, a shoemaker might have an unproductive day due to interruptions and take twelve hours to produce an average pair of shoes. The actual labour expended in the production of the shoes is high, but the shoes will not sell at a correspondingly high value. That would be an absurdity, encouraging producers to idle. Rather, it is social labour that counts. The social labour necessary to make a commodity is the share of the total labour of society required to make the pair of shoes given current production techniques and the average degree of skill and intensity prevalent at the time. A commodity made with old, slow methods and lazy and idle workers will not have a high value.

Third, the LTV explains how the total labour of society is allocated and reallocated to different branches of production. A theory of prices can then be based on a theory of value, but not vice-versa.

Fourth, the LTV is an empirical, not normative, theory of value. Marx did not hold that the value of a commodity *should* reflect how much effort was expended on its production; rather, he held that the value of a commodity *does* reflect the amount of social labour expended on its production.

A number of obvious objections to the LTV are normally raised, notably the value of land and the value of unique, or rare commodities such as famous paintings. Land has an economic value despite not being made by people, which appears to contradict the LTV, and famous paintings have a value that appears to have nothing to do with labour-time but everything to do with desire or demand. Brief answers to these objections are as follows. Land is not a commodity because it is not continually produced by combining raw materials with labour, unlike shoes, clocks and food. Similarly, a famous painting, although originally made, is not a commodity because it is a unique object: economic agents do not compete to produce Mona Lisas. The mass produced printed copies of the Mona Lisa are commodities, but the real thing is not.

So, things can have prices without being commodities with values. The LTV does not attempt to explain the prices of all kinds of things, but the value of commodities. An explanation of the price of non-commodities, such as a famous paintings, would need to consider a market in things that serve as temporary stores of value, supported by a social group with a stable convention of viewing the non-commodities as things of worth; for example, the market in art or the market in antiques. The social conventions may embody rules that are specific to the particular objects; for example, if a painting is pronounced fake by a respected art historian it will lose its price.

B.2 Labour power and processing power

The ultimate fate of the LTV will be decided by economists. The currency flow hypothesis makes no commitment to theories of what determines the quantity of the value of commodities, but does view value as performing the function of an ability to buy labour or processing power.

For example, the owner of a famous painting may decide to sell in order to realise its value. The money received from the painting may now be put to use. Money can be exchanged for commodities. In capitalist economies, people functioning as workers are also commodities: they are bought in the labour market. Whether a thing is purchased or a person is purchased for a certain period of the day, an amount of labour power has been assigned to the purchaser. That the labour power has already been expended and is in the form of a commodity, or will be expended and is in the form of a commodity-maker, is a secondary matter. In both cases, processing resources have been bought.

Appendix C

Reinforcement learning and animat emotions

This paper appeared in 'From Animals to Animats IV, Proceedings of the Fourth International Conference on the Simulation of Adaptive Behavior', 1996, edited by Pattie Maes, Maja Mataric, Jean-Arcady Meyer, Jordan Pollack and Stewart W. Wilson, pages 272–281, and published by the MIT Press.

Abstract

Emotional states, such as happiness or sadness, pose particular problems for information processing theories of mind. Hedonic components of states, unlike cognitive components, lack representational content. Research within Artificial Life, in particular the investigation of adaptive agent architectures, provides insights into the dynamic relationship between motivation, the ability of control sub-states to gain access to limited processing resources, and prototype emotional states. Holland's learning classifier system provides a concrete example of this relationship, demonstrating simple 'emotion-like' states, much as a thermostat demonstrates simple 'belief-like' and 'desire-like' states.

This leads to the conclusion that valency, a particular form of pleasure or displeasure, is a self-monitored process of credit-assignment. The importance of the movement of a domain-independent representation of utility within adaptive architectures is stressed. Existing information processing theories of emotion can be enriched by a 'circulation of value' design hypothesis. Implications for the development of emotional animats are considered.

C.1 Introduction

Motivation and emotion are often conspicuously absent from information processing theories of mind. For example, Allen Newell's description of the SOAR architecture (Newell, 1990; Laird, Newell & Rosenbloom, 1987), the most advanced candidate for a unified theory of cognition, lists motivation and emotion as missing elements that need to be included in a more comprehensive theory.

Research into complete agent designs, such as 'animat' research within the field of Artificial Life (ALife), forces designers to attempt to integrate motivation, learning, sensing and acting within a single agent design to produce adaptive behaviour. If properly conceptualised, this work yields insights into the dynamic relationship between motivation, the ability of control sub-states to gain access to limited processing resources, and prototype emotional states. In particular, Holland's learning classifier system, a type of complex adaptive system often used in ALife research, provides a concrete, if simplified, example of this relationship.

A theoretical conclusion follows: the feeling component of some emotional states arises from the self-monitoring of a process of credit-assignment occurring within motivational subsystems. This conclusion enriches previous information processing theories of emotion and has implications for ALife research.

First, reasons are given why emotion may be thought 'difficult' for information processing theories.

C.2 Thinking refers but feelings just are

Cognitive representations can denote or refer to states-of-affairs that exist in an agent's domain. This is the physical symbol system hypothesis – the hypothesis that physical systems can implement symbols that contain information that denotes (Simon, 1995). For example, an animat within a simulated domain may possess information about other agents in the environment, including their type, location, or speed. Such information sub-states of the animat are causally linked to their referents: the representation of the speed of agent A will alter if it is perceived that agent A has altered its speed. This is a simple example: referential links can be very indirect in more complex information processing systems. The principle, however, is conceptually clear and forms a basis of information processing theories of mind: *thinking refers*.

The emotions, however, differ from 'cold' cognition: they can be 'hot', often involving feelings of pleasure or displeasure with associated intensities. Unlike 'straightforward' representational thinking, an emotional state has *both* a representational content, e.g., a state of happiness *about* passing one's exams, and a hedonic, or *valenced* content, e.g., the particular form of intense pleasure one is experiencing. The hedonic component does not represent a state-of-affairs: *feelings just 'are'*.

Many information processing theories of emotion tend to avoid an explanation of the hedonic components of emotional states by concentrating on the semantics of representational components (e.g., Dyer's BORIS system (Dyer, 1987), Frijda and Swagerman's ACRES system (Frijda & Swagerman, 1987), and Pfeifer's FEELER system, reviewed in (Pfeifer, 1994). Alternatively, feelings are brushed under the physiological carpet by assuming that all valenced states arise from perceptions of bodily states. For example, Herbert Simon in his seminal paper on motivation and emotion (Simon, 1967), outlines a view of 'feelings' that closely resembles William James' peripheric theory of the emotions: '... sudden intense stimuli often produce large effects on the autonomic nervous system, commonly of an "arousal" and "energy marshaling" nature. It is to these effects that the label "emotion" is generally attached'; and '... the feelings reported are produced, in turn, by internal stimuli resulting from the arousal of the autonomic system'. It is difficult to conceive how this view of valency could account for the mental pain associated with, for example, grief, which does not necessarily require bodily arousal or disturbance.

Therefore, there appear to be at least two reasons why explanations of hedonic states are generally absent from cognitive theories: first, valenced components appear not to conform to the representational model that supports cognition; and second, their possible functional role is unclear: what can these states possibly *do* if they do not represent? Why do such diverse and complex states such as happiness, sadness, glee, triumph, grief, despair, intense disappointment etc. have hedonic components?

C.2.1 A preliminary definition of valency

Such 'states' are phenomenologically highly variegated (compare your memories of being angry with being happy), with different causal antecedents (e.g., being slighted before your peers, or winning an Olympic gold medal) and different consequences (such as a desire for revenge or rest). The folk-psychological concepts that refer to these states play a communicative role between agents, yet ultimately derive from sophisticated internal self-monitoring mechanisms, a kind of 'internal perception' (Wright, Sloman & Beaudoin, to appear). This ability gives rise to the method of introspection or phenomenological analysis in psychology. Much theoretical work in the emotions draws on introspection, and this kind of knowledge places important constraints on possible theories. However, there is no reason to believe that the knowledge gained from our internal perception is any less fallible than that gained from 'external' perception. In the absence of good theories of the underlying mechanisms, care is required when employing phenomenological concepts.

A preliminary division can be made between *dispositional* and *occurrent* emotional states (Ryle, 1949; Green, 1992). A dispositional state is a latent state that may manifest in appropriate circumstances, such as the brittleness of a wine glass, whereas an occurrent state is a *running* state, such as the process of a wine glass breaking. For example, a man who has lost a parent may function normally at work (dispositional grief), only to break down in the evening (dispositional state manifests as an occurrent emotion).

There are two distinguishable components of an occurrent emotional state: *intentional* and *non-intentional*. The intentional component of an emotional state is what the state is about. A person is angry, disappointed, or ecstatic 'about' a perceived state of affairs[1]. This state of affairs may exist in the agent's environment, or entirely within cognition (as in the case of the mathematician irritated with himself for being unable to solve an equation). The intentional component has representational content.

The non-intentional component of an emotional state is often referred to as its 'hedonic tone', *feeling*, or *valency*. 'Feeling' is an ill-defined word, for it can cover such diverse sensations as one's cheeks burning with embarrassment, an itch on the left ear, or the mental happiness associated with triumph. 'Hedonic tone' is similarly semantically overloaded: it can be used to refer to the enjoyable sensation of a full stomach after a large and hearty meal. The word valency, if given a suitable definition, can avoid such confusion. Before giving such a definition we require some more distinctions.

A division can be made between *physiological* forms of pleasure and displeasure, and *cognitive* valency. For example, the (self-monitored) 'itchiness' on my left ear is a form of displeasure linked to information concerning bodily location. In contrast, the (self-monitored) mental pain of intense grief is a form of displeasure linked to information about a loved one's death. There can be no 'pain receptors' for this kind of displeasure, unlike the nerves that detect a pin pricking one's finger.

[1] Moods can be considered an exception.

To illustrate: an athlete may be experiencing the occurrent emotional state of triumph while standing on the winner's podium. The intentional component of her state includes thoughts pertaining to her achieved goals; the non-intentional component *includes* the feeling of increased heart-rate or arousal, the warm sun beating on her brow, *and* a valenced state of cognitive pleasure not located on or in the body.

'Happiness' and 'sadness' provide the clearest examples of valency. The discussion, therefore, will restrict itself to these emotional states. Particular examples of happiness are triumph, glee, joy, ecstasy, gladness, love; and of sadness, despair, disappointment, grief, and sorrow. In a similar way to object-oriented programming languages, a preliminary taxonomy can be constructed in which 'happiness' and 'sadness' define classes of emotions with *valence* and *intention* slots. Particular instances of these generic emotions have additional slots that define finer-grained attributes. In this way, a hierarchy of sub-types of the emotions 'happiness' and 'sadness' is formed (see (Ortony, Clore & Collins, 1988) for a similar treatment). Consequently, the terms 'happiness' and 'sadness', in this paper, refer to generic, abstract definitions. For the sake of brevity, it will be stated that a person is 'happy' when a desired goal is achieved, and 'sad' when failure occurs in achieving a desired goal. This is an oversimplification: concrete emotional states are rarely this straightforward or as simple.

Unlike the intentional component of an emotional state, valency does *not* represent a state of affairs. It can differ qualitatively in only a very restricted sense, i.e. it can be *either* pleasurable or displeasurable, and allow quantitative degrees of *intensity*: the valency can be very displeasurable or only mildly so. From a phenomenological perspective, valency is a 'brute fact'[2] of one's present state, unlike beliefs that can be true or false, or goals that have been achieved or not.

A preliminary definition of valency can now be provided: *Valency* is a form of pleasure or displeasure not located in the body, and is *a* non-intentional component of occurrent emotional states of happiness or sadness.

Our problem is to provide a theoretical account of valency.

[2]This term borrowed from (Chalmers, 1996).

C.3 A design-based answer: thermostats and classifiers

There are almost as many theories of emotion as emotion theorists. The field, as has often been remarked, is characterised by terminological confusion (Kagan, 1978; Read & Sloman, 1993) and riven by differing 'schools of thought' (Pfeifer, 1994).

Approaches to the study of emotions can be very broadly categorised as *semantics*-based, *phenomena*-based and *design*-based (Sloman, 1992). Semantics-based theories analyse the use of language to uncover implicit assumptions underlying emotion words (e.g., (Wierzbicka, 1992)). Phenomena-based theories assume that emotions are a well-specified category and attempt to correlate contemporaneous and measurable phenomena with the occurrence of an emotion, such as physiological changes (an early example is William James' theory – see (Calhoun & Solomon, 1984); for a comprehensive review of many phenomena-based theories, see (Strongman, 1987)).

The design-based approach, in contrast, takes the stance of an *engineer* attempting to build a system that exhibits the phenomena to be explained, and is a 'rational reconstruction' of the practice of Artificial Intelligence, considered as the general science of intelligent systems, both natural and artificial (Sloman, 1993c). The methodology involves exploring an abstract space of possible requirements for functioning agents (*niche-space*) and the space of possible designs for such agents (*design-space*) and the mappings between them (Sloman, 1995a). This is an iterative process and the research strategies can vary: they may be top-down, bottom-up or 'middle-out'. All are potentially useful.

Research within the field of Artificial Life is an example of the design-based approach, and is characterised by the investigation of complete agents that integrate many capabilities, such as sensing, action selection, acting and (particularly) adaptation to a simulated or real niche. A consequence of such a methodology is that the analytic isolation of emotions from other cognitive phenomena, often characteristic of semantic and phenomena-based approaches, can be directly avoided by the investigation of complete systems. This methodological course is likely to bear the most fruit and have the most relevance for unified theories of cognition.

The development of negative feedback control systems demonstrated that simple, materially embodied systems can have sub-states with different functional roles, in particular 'belief-like' and 'desire-like' control sub-states (McCarthy, 1979; Sloman, 1993b; Powers, 1988; Braitenburg, 1984). For example, the belief-like sub-state of a thermostat is the curvature of its bi-metallic strip, which alters in accordance with the ambient temperature of a room; the desire-like sub-state is the setting of the control knob. Negative feedback ensures that the temperature of the room stabilises around the control knob setting, i.e. the thermostat 'acts' in the world to achieve its 'desire'. Of course, a thermostat does not have sufficient architectural complexity required to support human beliefs and desires, but it is an illustrative 'limiting case' (Sloman, 1993b).

The simple thermostat implements a function that maps an input temperature to an output signal, which controls a heater. It does not learn. We can move through design-space adding architectural complexity to the negative feedback loop, such as varying the kinds of sub-states, the number and variety of sub-states, the functional differentiation of sub-states, and the kinds of causal influences on sub-states, such as whether the machine can change its own desire-like states, and so on (see Sloman, 93a for an extended discussion).

Holland's classifier system (Holland, 1995; Holland, 1975; Holland, Holyoak, Nisbett & Thagard, 1986; Riolo, 1988) is one of a class of relatively well-understood machine learning algorithms. Unlike the thermostat, it is sufficiently complex to exhibit prototypes of 'emotion-like' states, much as the thermostat exhibits prototypes of 'belief-like' and 'desire-like' sub-states. An analysis of its functioning reveals an important role for non-intentional representations. First, a brief description of a classifier system.

C.3.1 The learning classifier system

A classifier system consists of a *performance system*, and *credit-assignment* and *rule discovery* algorithms.[3] The performance system consists of a *classifier list* that consists of a set of condition-action rules called *classifiers*, a *message list* that holds current messages (as per the working memory of production systems), an *input interface* that provides the classifier system with information about its environment in the required form, and an *output interface* that translates action messages into world events. The *basic cycle* of the performance system matches messages in the message list (including sensory messages) with classifiers, which then post their actions 'back' to the message list. Many classifiers may become active and fire in parallel. Any current action messages are sent to the output interface. The performance system is a universal machine; that is, any computable function can be implemented as a collection of classifiers.

[3] This specification abridged from (Riolo, 1988). Readers unfamiliar with classifier systems should consult (Holland, 1995).

The performance system alone cannot learn. The credit-assignment algorithm in the classifier system is a *bucket-brigade* (*bb*) algorithm. This algorithm introduces competition between classifiers based on a quantitative 'strength'. Each classifier that has its condition activated by a message *bids* to post its action part to the message list. Only the highest bidders are allowed to post their actions. The bid of a classifier depends on its strength[4], which is a measure of the classifier's 'usefulness' to the system. The higher the strength of a classifier the more likely it will win the competitive bidding round and post a message. The behaviour of the classifier system, therefore, can be modified by changing the strengths associated with individual classifiers. If the strength of the classifiers that tend to lead to 'useful' behaviour can be increased, and the strength of the classifiers that tend to lead to 'useless' behaviour can be decreased, the system will learn to produce more useful behaviour. The *bb* is designed to bring about these types of changes in strength.

The basis for the *bb* is information from *reinforcement mechanisms* about whether the classifier system as a whole is behaving correctly. This is achieved via *rewards*, i.e. the system will receive positive reward when it behaves correctly and negative reward when it behaves incorrectly. More often than not neither positive or negative reward will be received. This is a type of *reinforcement learning* (for a survey, see (Kaelbling, Littman & Moore, 1995), which derives its name from behaviourist theories of animal learning (e.g., (Mackintosh, 1983)). The phrase 'credit assignment' will be preferred over 'reinforcement learning': the former emphasises internal mechanisms, whereas the latter is often associated with external operations.

When a reward is received the *bb* adds the reward value to the strength of all classifiers currently active, thereby changing the strength of classifiers *directly* associated in time with useful behaviour. Also, when a classifier is activated it 'pays' the amount it bid to the antecedent classifier that produced the message it matched. The strength of the active classifier is decreased by its bid amount. In this way, the *bb* acts to increase the strength of classifiers *indirectly* involved in the production of useful behaviour. This is a partial solution to the *temporal credit assignment problem*, which is the problem of determining which antecedent rules to strengthen given that there can be long delays between antecedent classifiers and the resultant

[4]This is a simplification for the sake of brevity. The bid of a classifier can depend on at least three factors: strength, the 'specificity' of the classifier, which is a measure of its relevance to a particular set of messages, and 'support', which allows internal messages to have differential importance.

rewarding act. The *bb* allows reward to 'circulate back' through the system (in a similar way to the 'back-propagation' of an error signal in artificial neural networks). Chains of high strength classifiers, performing useful computation, can emerge from such a scheme.

An alternative name for the quantitative measure associated with each individual classifier is *value*. This switch in terminology can be better understood by recalling that the *bb* was originally inspired by an economic metaphor (Holland, Holyoak, Nisbett & Thagard, 1986), in which classifiers are agents (consuming and producing messages) who possess a certain amount of money ('strength' or value) which they exchange for commodities at the market (the message list or blackboard). Much as money mirrors the flow of commodities in a simple commodity economy, *value mirrors the flow of messages* in the learning classifier system.

The *bb* can lead to improved system behaviour through the selection of some classifiers over others; however, it cannot create entirely new classifiers. A genetic algorithm (*ga*) implements rule discovery. Periodically, classifiers of high strength are selected as parents. The genetic operators of *crossover* and *mutation* are applied to classifiers considered as chromosome strings. The resultant offspring are placed in the classifier list. Normally, the *ga* is applied less frequently than the *bb* otherwise new classifiers will not have had sufficient time to be evaluated.

To summarise: within a classifier system *selection on value* occurs twice: in bidding rounds of classifiers competing for messages, and in rule discovery where high value classifiers generate offspring and low value classifiers are eventually removed from the system altogether. The combined effects of *circulation of value* via the *bb* and a double selection on value by competition and rule discovery allows the classifier system to reallocate classifier rules from unrewarding to rewarding processing. This ability makes it an adaptive system.

The classifier system, when embedded in a simulated or real environment, can construct classifiers that produce satisficing behaviour given the constraints and guidance of reinforcement mechanisms. It has been extensively used in the ALife community (Steels, 1994), for example in developing control programs for robots through supervised learning (Dorigo, Maniezzo & Colorni, 1996).

C.3.2 The intentional component of classifier states

The internal state of an implemented and *running* classifier system is continually changing. We will denote the classifier system's internal state at time step t as C_t, and define it as the joint operation of the basic cycle and *bb* considered as one indivisible moment. C_t has two distinguishable components: *intentional* and *non-intentional*.

The intentional component of C_t is the message list containing messages with representational content. The simplest example of the representational content of messages can be found in (Holland, 1995). We can imagine an artificial frog embedded in an environment of real or simulated flies. The control program for *simfrog* is a classifier system with an adapted set of classifiers. The *simfrog* has an eye sensor, which forms part of the classifier system's input interface. The eye can detect a number of attributes of any fly within range. Attributes could include whether the fly is moving, what colour it is, its size and proximity. If a fly is detected the eye sensor posts a message to the message list that encodes this information. This sensory message is the result of a *mapping* between a state of the environment and a sub-state of *simfrog*. It is in this sense that messages have representational content. Internal messages, less directly linked to sensing or acting, will have more complex representational roles within the system. The *semantics* of messages depends on the dynamic relationship between message and environment. For example, the sensory message may match a classifier that posts an action message that results in *simfrog* throwing its sticky tongue in the direction of the detected fly. The meaning of the message, therefore, would be an impoverished version of 'eat that fly!'.

C.3.3 The non-intentional component of classifier states

The non-intentional component of C_t is the circulation of value between antecedent classifiers, messages and matching classifiers. Value is exchanged for messages, i.e. matching classifiers pay the 'owners' of messages an amount of value, via the *bb*. Unlike messages, the value that circulates has no representational content.

The values associated with classifiers specify a probabilistic partial ordering on the classifier list. The ordering is partial because only some classifiers will bid for the same message. The ordering is probabilistic because the classifier selection mechanism is stochastic.

For example, both classifiers c_i and c_j match message m; c_i has value v_i, and c_j has value v_j, with $\mathrm{v}_i > \mathrm{v}_j$. In competitive bidding for message m, c_i will out bid c_j more often than not, i.e. there is a probabilistic total ordering on the set $A = \{c_i, c_j\}$. Other classifiers in the classifier list, such as c_k, *never* compete against any $c \in A$; consequently, no ordering holds between them[5]. Classifier values,

[5]In classifier systems with maximum size message lists the situation can become more complex,

therefore, specify a partial and probabilistic relation of utility over the classifier list.

Value is internal economy alone. It does not represent anything within or external to the classifier system; rather, it specifies a *relation* between classifiers. It is this property of value that helps to make the classifier system a *domain-independent* learning algorithm: the representation of utility does not alter from domain to domain.

The classifier system tightly integrates impoverished conceptions of cognition, conation and affect.

The cognitive engine is the performance system consisting of condition-action rules and a global blackboard (cf. the production system and working memory of Newell's SOAR architecture).

Conation, or 'motivational force', is also represented in the classifier system. The value of a classifier is its dispositional and relative ability to fire and post a message. Whether the implementation of the classifier system is truly parallel (with perhaps separate processors for each classifier), or only simulates parallelism, *the value of a classifier is an ability to buy processing power.* A high value classifier will be more likely to win bidding rounds, be processed, and post its action part. An internal sensory message, for example a message that *simfrog*'s energy is below a danger threshold, may match the first in a chain of high value classifiers that instigate a search for flies in the environment. The high value of such a processing chain will make it unlikely that other rules will out bid and switch processing to other ends. In this impoverished sense, value is motivational force.

Finally, the operation of the bucket-brigade, which alters the 'buying power' of sets of classifiers, involves losses or gains of value that are ultimately derived from reinforcement mechanisms, i.e. information regarding the 'goodness' or 'badness' of agent-environment situations. Positive and negative reinforcement are often linked to pleasure and pain.

C.3.4 Self-monitoring of credit-assignment

We now add a simple 'self-monitoring' mechanism to the classifier system. The mechanism is required to monitor the circulation of value and send its output to a suitable device. For current purposes self-monitoring need play no functional role within the classifier system, so the device can be a computer screen that displays results to an ALife engineer.

as classifiers compete for space in addition to competing for messages.

C_t involves a *net* exchange of value, denoted V_t, from matching classifiers to antecedent classifiers. The self-monitoring mechanism records each V_t over a specified time period, say $t = 1 \ldots n$, and displays the *change* in value, denoted δV_t, which is exchanged from one time step to the next, where $\delta V_t = V_{t+1} - V_t$. δV_t can be either:

- Positive, implying (a) a net increase in the utility of antecedent classifiers, and (b) currently active classifiers are likely to lead to positively rewarding consequences;
- Negative, implying (a) a net decrease in the utility of antecedent classifiers, and (b) currently active classifiers are likely to lead to negatively rewarding consequences; or
- Zero, implying no net change in the utility of antecedent classifiers.

Therefore, the self-monitoring of $C_1 \ldots C_n$ will display a rate of change of value with both sign and magnitude. We now connect the output of self-monitoring to *simfrog*'s skin, which can change colour. If δV_t is zero *simfrog* remains *green*, if δV_t is positive he displays *yellow* with an intensity $|\delta V_t|$, and if δV_t is negative he displays *blue* with intensity $|\delta V_t|$. When *simfrog* catches and eats a fly he will blush bright yellow as innate reinforcement mechanisms strongly positively reward antecedent classifiers. If *simfrog* possessed more sophisticated reflective capabilities he might wonder why he has beliefs that refer *and* an odd quantitative intensity that is either positive or negative but doesn't seem to be 'about' anything or serve any apparent purpose. Depending on philosophical prejudice, one might be tempted to say that *simfrog feels* happy, sad or indifferent depending on circumstance.

Some caveats are in order. A classifier system is limited in many ways. It does not have an explicit memory store. It tends to be an entirely reactive system with no representation of goals. It does not anticipate, or perform prior search within a world model before acting. In real-world applications it can be difficult for a classifier system to learn appropriate behaviours (Wilson & Goldberg, 1989). Also, the classifier system is not a fixed architecture but continues to evolve, e.g. the recent introduction of rule discovery based on the accuracy of classifier predictions (Wilson, 1995). However, by abstracting from implementation details we can examine certain *design principles* embodied in the classifier system. The following section examines the implications of such design principles for information processing theories of human emotions.

C.4 Circulation of value

There are two kinds of finite computational[6] resource in the classifier system: (a) the total *information* capacity of the performance system, which consists of the set of 'if-then' rules, a (usually fixed) number of classifiers that perform simple computations, and (b) *processing* limits, which is the amount of parallel computation allowed per time step; for example, a maximum of ten classifiers may be allowed to fire during each basic cycle. For current purposes, information limits will be placed to one side.

C.4.1 Processing limits: emotions as interrupts

Existing information processing theories of emotion agree on the importance of *processing* resource limits in accounting for emotional states. For example, Simon's 'interrupt theory' associates emotional states with the interruption of a resource-bound, high level, attentive system due to new, perhaps urgent motives: 'The theory explains how a basically serial information processor endowed with multiple needs behaves adaptively and survives in an environment that presents unpredictable threats and opportunities. The explanation is built on two central mechanisms: 1. A goal-terminating mechanism [goal executor] ... [and] 2. An interruption mechanism, that is, emotion allows the processor to respond to urgent needs in real time.' (Simon, 1967; Simon, 1982). An interrupting stimulus, such as the presence of a predator, disrupts ongoing goal processing in the serial processor and substitutes new goals to deal with the new situation producing, amongst other things, emotional behaviour, e.g. the flight-fight-fright response.

Aaron Sloman's attention filter penetration theory (Sloman & Croucher, 1981; Sloman, 1987; Sloman, 1992) extends the interrupt theory by introducing new, architectural detail implicit in Simon's paper. Two types of processing are distinguished: pre-attentive, highly parallel and automatic motive generation processes, and attentive, resource-bound 'motive management' processes that exhibit a limited degree of parallelism. The concept of *insistence* is introduced, which is the dispositional ability of a motivator to 'surface' through a variable threshold filter and disrupt attentive processing. The *intensity* of a motivator is its ability to 'keep hold' of attention once surfaced. A common characteristic of many emotional states is the phenomenon of *perturbance* (Beaudoin, 1994), which occurs when a motive has been postponed or rejected but nevertheless keeps resurfacing to disrupt ongoing,

[6] A classifier system animat embedded in an environment will also have 'physical' resource constraints, such as the number and kind of effectors and detectors.

motive processing. The concept of perturbance has been extensively used to provide an architectural account of grief or 'loss' (Wright, Sloman & Beaudoin, 1996). As in Simon's theory, 'emotional' interrupt mechanisms are needed to maintain reactivity to important events in a system with finite processing resources.

Both interrupt theories rely (implicitly in Simon's case) on a distinction between *control* and *semantic* signals in information processing architectures. Semantic signalling is the propagation of information that has representational content, whereas control signalling does not refer or have semantic content but performs a control function, such as changing the control flow of the system, or putting it into a distinct kind of processing state[7]. It is the non-representational nature of control signalling that is held to account for the non-intentional, 'feeling' component of emotional states. However, a theoretical explanation of valency is absent: reasons why some control signals are pleasurable or displeasurable, i.e. possess a qualitative dimension of valency that can be either positive or negative, and also vary in quantitative intensity, are not provided.

Keith Oatley and Philip Johnson-Laird's (Oatley, 1992; Oatley & Johnson-Laird, 1985; Johnson-Laird, 1988) communicative theory addresses this question by introducing basic, irreducible and phylogenetically older architectural control signals. 'Each goal and plan has a monitoring mechanism that evaluates events relevant to it. When a substantial change of probability occurs of achieving an important goal or subgoal, the monitoring mechanism broadcasts to the whole cognitive system a signal that can set it into readiness to respond to this change. Humans experience these signals and the states of readiness they induce as emotions' (Oatley, 1992). Control signals, of which 'happiness' and 'sadness' are examples, communicate significant junctures of plans to other cognitive subsystems. Emotional states, therefore, are viewed as a design solution to certain problems of the transition between plans in systems with multiple goals. The functional role of valenced signals is to enforce state transitions by interrupting a central processing system; for example, the 'sadness' control signal, broadcast when a major plan fails, causes a state transition to search for a new plan. On this view, control signals differ in valency because they differ in their functional roles.

[7]The following analogy may help capture the distinction. Imagine trains travelling on a complex network of tracks. Postal trains contain mail (semantic content) with destination addresses on the envelopes. These trains travel to the destinations and deposit the mail (the information). However, a different kind of train, a 'control signal' train, can travel through the network altering the points of the tracks. This has the effect of changing the topology of the network, i.e. trains will continue to deposit their mail but will use different routes.

All three theories lack a consideration of *adaptive* architectures, i.e. architectures able to modify themselves to improve their behaviour. Explicitly considering adaptation provides a new functional role for control signalling.

C.4.2 Adaptation: changes in the ability to buy processing power

The phrase *circulation of value* denotes a general design principle, and refers to any mechanism that (i) alters the dispositional ability of agents in a multi-agent system to gain access to limited processing resources, via (ii) exchanges of a quantitative, domain-independent representation of utility that mirrors the flow of agent products[8]. A particular example of circulation of value is the bucket-brigade algorithm, where an 'agent' is a single classifier, 'multi-agent system' is the set of competing and cooperating classifiers[9], and 'agent products' are messages.

The circulation of value is a pattern of flow of control signals. Such signals have no semantic content and propagate around the system altering control flow. Additionally, classifier bids attempt to 'grab' processing resources and may 'interrupt' current processing causing internal or external behaviour to take different routes. A quantitative representation of utility, therefore, need not contradict the interrupt function identified by Simon and Sloman, nor the control signal function identified by Oatley and Johnson-Laird. The local exchange of value between classifiers can generate the negative and positive control signals of the communicative theory: instead of two signals, we now have one. This is a more parsimonious state of affairs. But, additionally, there is a new, previously unidentified functional role for the control signal: the circulation of value implements a type of *adaptation*. This is not inductive learning of new hypotheses about a domain, but an ordering and reordering of the utility of control sub-states to dispositionally determine behaviour. In a

[8]This definition is very general; for example, it applies to currency flow in simple commodity economies (limited processing resources would correspond to the available labour-power of the social system). However, *circulation* of value does not usefully apply to tabular reinforcement learning algorithms, such as Q-learning, which use a quantitative representation of value without internal exchanges of value. An XCS classifier system (Wilson, 1995) that allows chains of classifiers to form will exhibit circulation of value; however, the distinction between strength and accuracy of pay-off prediction may afford more sophisticated self-monitoring; in particular, the difference between predicted and actual pay-off may help introduce *expectancies* into the discussion. Such an analysis is beyond the scope of this paper.

[9]To be precise, a set of classifiers is not a multi-agent system, as the definition of an agent normally includes a requirement for beliefs and desires, whereas a classifier is only a production rule.

classifier system, the circulation of value adaptively changes the ability of classifiers to buy processing power.

The design principle of circulation of value opens up the possibility of architectures that generate valenced states far removed from physiology. High level cognitive processes may be saturated with value, allowing the production of semantic messages coupled with losses or gains in quantitative value. For example, negatively valenced states, such as grief, may, amongst other things, involve the gradual loss of the accumulated value of a structure of attachment (Bowlby, 1979; Bowlby, 1988; Wright, Sloman & Beaudoin, 1996) towards a loved one. Negative valency may be a necessary consequence of adaptive change: the structure of attachment, no longer useful for motive generation, loses its ability to buy processing power, grab attentive resources and dispositionally determine behaviour. This process is self-monitored as displeasurable.

These considerations lead to a *design hypothesis* of a quantitative representation of utility that circulates within a subset of cognition. The next step is the construction of a circulation of value theory of emotions that builds on previous work. This will require another iteration of the design-based approach. The design-space of autonomous agent architectures that integrate circulation of value learning mechanisms with more complex forms of motive management needs to be explored. The capabilities, properties and explanatory power of such designs can then be examined. Further comparisons can then be made between designs and psychological phenomena. Preliminary efforts in this direction are reported in (Wright, 1995). In addition, it will be necessary to investigate the results and theories of neuropsychology in order to map the postulated mechanisms onto the neural substrate. If some form of circulation of value is found to occur in human brains this would add credence to such a theory. However, if the further exploration of adaptive architectures finds no use for circulation of value mechanisms then the hypothesis will need to be modified or replaced: ultimately, it is an empirical question.

However, even at this early stage it is possible to make some theoretical claims. Given the above considerations, and our preliminary definition of valency, we can state a corresponding architectural process that gives rise to it: *valency is a self-monitored process of credit-assignment*. The hedonic, or 'feeling' component of many emotional states arises from architectures that employ a circulation of value mechanism. (A type of) 'feeling' is (a type of) adaptation, and 'hot' cognition – forms of pleasure or displeasure involved in short term and long term control – need pose no insurmountable problems for information processing theories of emotion.

C.5 Towards emotional animats

The emotions encompass a broad range of human experience, whereas valency is a component of only some emotional states. To approach a real understanding of human emotions will require an investigation of more sophisticated architectures satisfying more complex requirements. This is long-term research. Yet it can be said with confidence that simple animats already exhibit simple 'emotion-like' states, as long as we take care over definitions and avoid hyperbole. 'Broad but shallow' agent designs can be illuminating. For example, (Balkenius, 1995) provides a clear discussion on relations between simple motivations and simple emotions based on experiments with animats. Also, the simulation of societies of competing and co-operating adaptive agents will impose new requirements on architectures and give rise to new kinds of internal state. (Aube & Senteni, 1996a) view (more complex) emotional states as 'commitment operators' that manage resources in multi-agent systems.

An important difference between such design-based (e.g., see (Beaudoin & Sloman, 1993; Beaudoin, 1994; Sloman, Beaudoin & Wright, 1994; Pfeifer, 1994)) approaches and previous computer simulations of emotions is that they move from requirements for *complete agents* to possible designs, and do not directly 'program in' correlates of emotions. In this way, design features are linked to niche features providing explanations of *why* emotional states are present in nature. A small example is provided in this paper: a requirement for adaptability entails, at some level, credit-assignment mechanisms that use a domain-independent representation of utility or value, a kind of internal 'common currency'. Such a representation does not refer but is relational, and can be gained or lost depending on whether actions are successful or unsuccessful in leading to rewarding consequences. These kinds of processes are self-monitored as non-intentional states, or 'feelings', which are quantitative in nature and either pleasurable or displeasurable.

Those philosophically inclined may doubt that animats constructed in this way 'really' experience their non-intentional states. Is it *anything to be like* a simulated animat? This kind of question is experimentally undecidable, and has no engineering consequences whatever. It is therefore scientifically uninteresting.

C.6 Conclusion

A subset of emotional states was examined and the concept of valency – the non-intentional component of occurrent emotional states of happiness or sadness – was introduced. Valency is a preliminary definition of a subset of the states colloquially referred to as 'feelings'. The internal state of a complex adaptive system, the learning classifier system, able to function as a complete agent within a niche, was examined. Much like the simple thermostat exhibits prototypes of 'belief-like' and 'desire-like' sub-states, the classifier system was found to exhibit simple examples of valenced states.

The design principle of circulation of value was abstracted from the classifier system and employed to overcome existing inadequacies in information processing theories of emotion. The circulation of value mechanism is more parsimonious than Oatley and Johnson-Laird's control signalling, and, more importantly, it introduces a basic form of adaptation. These new considerations build upon and do not contradict previous theories. A theoretical conclusion is that, to a first approximation, valency is a self-monitored process of credit-assignment. 'Feeling' is the self-monitoring of adaptation; that is, the non-intentional component of generic 'happiness' and 'sadness' states includes a movement of internal value, which functions to alter the dispositional ability of control sub-states to buy processing power and determine behaviour. Such a process is self-monitored as a 'brute' feeling because value does not refer, unlike beliefs that can be true or false, or goals that can be achieved or not.

To approach the complexity of concrete, human emotional states will require many more iterations of the design-based approach.

Acknowledgments

My thanks to Aaron Sloman, Chris Complin, Tim Kovacs, members of the Cognition and Affect and EEBIC (Evolutionary and Emergent Behaviour Intelligence and Computation) research groups at Birmingham, and the SAB96 referees for their helpful comments.

Appendix D

Abbreviations

AI Artificial Intelligence

AFP Sloman's Attention Filter Penentration theory of the emotions

AMAS Adaptive Multi-Agent System(s)

BDI Belief-Desire-Intention

C&AP Cognition and Affect Project

CFH Currency Flow Hypothesis

CFHN Currency Flow Hypothesis for Natural reinforcement learners

CLE Computational Libidinal Economy

COM Oatley and Johnson-Laird's COMmunicative theory of the emotions

CPU Central Processing Unit

CR Frijda's Concern Realisation theory of the emotions

CUE Conative Universal Equivalent

KA Knowledge Areas

MAS Multi-Agent System(s)

PRS Procedural Reasoning System

RAP Reactive Action Package(s)

RL machine Reinforcement Learning

TR Teleo-Reactive

TRP Teleo-Reactive Program

VAFP Valenced Attention Filter Penetration theory of the emotions