BIOLOGICALLY-BASED FUNCTIONAL MECHANISMS OF MOTOR SKILL ACQUISITION

A Dissertation Presented

by

ASHVIN SHAH

Submitted to the Graduate School of the
University of Massachusetts Amherst in partial fulfillment
of the requirements for the degree of

DOCTOR OF PHILOSOPHY

September 2008

Neuroscience and Behavior Program

BIOLOGICALLY-BASED FUNCTIONAL MECHANISMS OF MOTOR SKILL ACQUISITION

A Dissertation Presented

by

ASHVIN SHAH

Approved as to style and content by:

____

Andrew G. Barto, Chair

____

Neil E. Berthier, Member

____

Andrew H. Fagg, Member

____

Richard E.A. van Emmerik, Member

____

Nancy G. Forger, Acting Director
Neuroscience and Behavior Program

For my parents.

ACKNOWLEDGMENTS

I thank my adviser, Andy Barto, and my unofficial co-adviser, Andy Fagg. Both helped shape my (often scattered) thoughts into coherent research directions and supplied a wealth of information for me to absorb (and sometimes forget). I also appreciate their patience and modesty. Once in a while, graduate students find that they lack important bits of knowledge. Of course, this rarely happened to me; on the rare occasion that it did, I was comfortable in asking questions. Such an environment is conducive to learning, and learning, besides being the focus of research in our lab, is the reason I came to graduate school.

I also thank my other two committee members, Neil Berthier and Richard van Emmerik, for reading through this thesis and for their insightful comments and discussions during my defense. The work presented in this thesis was made possible by grants from the National Institutes of Health (grant number NS 044393-01A1). I thank NIH and Jim Houk, Scott Grafton, and Peter Strick who, besides being part of the grant with Andy Barto, spent the time to read some very early descriptions of my ideas and provide extremely helpful comments.

Academic research environments greatly vary from place to place, and I am lucky to have worked where I did. From my undergrad days, I thank Fred Ellis, for introducing me to research in a physics lab (and the joys of liquid nitrogen), Rick Jensen, for introducing me to computational neuroscience, and Harry Sinnamon, for introducing me to experimental work on rats and reminding me that I am the primate and the rat is the rodent, a very important distinction to remember when dealing with rats. From my post-undergrad days, I thank Christiane Linster, for whom I attempted to build compartmental neuron models (unsuccessfully, due in part to typos in long lists of parameters), and Bruce Bean, for whom I attempted (unsuccessfully) to find a certain type of hard-to-find sodium channel. My experience in Bruce's lab also vastly increased my respect for research in experimental biology.

Thankfully, when I entered graduate school and a computer science lab, I never had to put on latex gloves again. I am also thankful for my peers. It is the rare computer science lab that welcomes one who thinks that java refers to coffee, latex refers to a rubber-like material, and Matlab scripts are a form of programming. Members of the Autonomous Learning Laboratory are fine researchers and finer people, and it's been a joy to work (and not work) with them. I am also happy to have discussed research (and non-research topics once in a great while) with members of the Laboratory of Perceptual Robotics, other members the Computer Science Department, members of the Neuroscience and Behavior Program, and fine friends from Western Massachusetts. I am sorry to omit individual names, but I thank you all for your companionship, past, present, and future.

ABSTRACT

BIOLOGICALLY-BASED FUNCTIONAL MECHANISMS OF MOTOR SKILL ACQUISITION

SEPTEMBER 2008

ASHVIN SHAH

B.A., WESLEYAN UNIVERSITY

Ph.D., UNIVERSITY OF MASSACHUSETTS AMHERST

Directed by: Professor Andrew G. Barto

The adage *practice makes perfect* makes for sound advice when learning a novel motor skill. Be it typing a new password or hitting a forehand in tennis, proficiency increases with experience. Behavioral changes associated with motor skill acquisition can be broken down into three broad categories: 1) movements are executed faster and become more coordinated, 2) they come to rely on sensory information gained while executing the task, rather than just sensory information used during initial stages of learning the task, and 3) they seem to be executed with less conscious thought and attention. In addition, neural activity changes: many imaging and neural recording studies suggest that with experience, control is transferred from cortical planning areas to the basal ganglia. The two areas are thought to employ different learning and control schemes. In general, planning can quickly take new information into account to make reasonable decisions, but its control mechanisms have large computational requirements. The basal ganglia use a simpler and less computationally expensive control scheme, but they require much experience before they can produce reasonable behavior.

In this thesis, I contribute to answering the question, "what goes on during practice?" More formally, I am interested in the mechanisms by which motor skills are acquired. I take a theoretical approach in that I hypothesize a multiple controller scheme, based on the learning and control mechanisms of cortical planning areas and the basal ganglia, and test it with simulations designed emulate generic motor skill tasks. Because skill proficiency increases with experience, I am particularly interested in the role of the experience-dependent mechanisms of the basal ganglia in motor skill

acquisition. Thus, learning mechanisms attributed to cortical areas are artificially restricted so that any change in model behavior is attributed to the learning mechanisms of the basal ganglia.

Model behaviors exhibit characteristics indicative of motor skills, supporting the plausibility of the multiple controller scheme as one used by our nervous system and suggesting that the learning mechanisms of the basal ganglia can contribute to developing most characteristics. In addition, I show how the strategies developed by the models are functionally advantageous, providing a reason why such a scheme may be used.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

The adage *practice makes perfect* makes for sound advice when learning a novel motor skill. The purpose of a motor skill (or any type of skill) is to accomplish some task in a proficient manner. For learning purposes, a task is often decomposed into a sequence of coarsely-defined subtasks. For example, in learning how to throw a baseball, it is helpful if the process is described as a sequence similar to the following: grasp ball, pull arm back, pivot body, accelerate elbow forward, allow wrist to extend back, snap wrist forward, release ball. "Step-by-step" instructions are used to describe many types of complex tasks. A description of a subtask is coarse in that it may specify the subtask goal on one spatial level (e.g., pull arm back), but it does not specify other variables needed to achieve the goal (e.g., the ideal configuration of the arm and upper body, the stiffness of the joints to accommodate the force of the ball, etc.). There is coarseness in the temporal domain as well: there is not a specification of subtasks to account for each moment in time. Because of the coarse description of subtasks, the learner must "fill in the blanks" by actually executing the movements required to accomplish the subtasks. "Practice" is the act of repeatedly solving the sequence of subtasks in order to attain proficiency in solving the overall task.

In this thesis, I contribute to answering the question "what goes on during practice?" More formally, I am interested in the computational mechanisms animals employ in acquiring motor skills. I approach this problem with the methods of computational, or theoretical, neuroscience in which I hypothesize computational mechanisms animals use in motor skill acquisition. I focus on *functional* aspects in that I attempt to elucidate what functional purposes they serve. The mechanisms are *biologically-based* in that they are based on mechanisms attributable to brain areas, as evidenced by literature in neuroscience encompassing a wide range of experimental techniques. In particular, because proficiency increases with practice, I focus on *experiential learning*, which occurs when interacting with the environment.

The rest of this Introduction is devoted to an overview of motor skills, what types of problems must be solved in their acquisition, and a brief discussion of theoretical neuroscience.

1.1 Motor skills

Many books on human motor control devote some discussion to what is meant by the term *motor skill* (cf. Kelso 1982, Chapter 2; Schmidt 1982, Chapter 3; Rosenbaum 1991, Chapter 3). Most researchers agree with Schmidt's broad definition that skills are movements that are learned and "dependent on practice and experience for their execution, as opposed to being genetically defined" (Schmidt 1982, pg. 20). Most researchers also agree that a skill refers to the ability to achieve some goal with proficiency, i.e., in a way that approaches optimality as defined by the task. The concept of proficiency is also used in Kelso (1982), which makes a distinction between *control* and *skill*. Control refers to the assignment of values to variables, e.g., the levels (values) at which to activate muscles (variables). A skill requires not merely that a goal be achieved, which can be accomplished in many ways, but that "the *optimal* value be assigned to the controller variables" (Kelso 1982, pg. 28). Inherent in the study of motor skills are the concepts of redundancy and optimal control.

Though helpful as general descriptions, the terms "learned" and "optimal" apply to many types of movements. Thus, the scope of behaviors that can be described as "motor skills" is broad to the point of being cumbersome (cf. Newell 1991). Rosenbaum et al. (2001) define a skill as "an ability that allows a goal to be achieved within some domain with increasing likelihood as a result of practice." "Optimal" in this case refers to reliability. Kelso (1982, Chapter 2), on the other hand, suggests that some of the most important characteristics of skills depend on the actual movements: they are accurate and precise in space and time (e.g., the tennis racquet head is positioned so that the ball hits its center at the right time), they are adaptable (a skilled tennis player can achieve space-time accuracy for a large variety of incoming ball trajectories), and the movements are consistent (the movements the player uses vary little given a particular incoming ball trajectory). Kelso notes that these characteristics are "far from inclusive" (pg. 23) and goes on to discuss how "optimal" can mean very different things depending on the task (e.g., a long-distance run versus a short sprint).

It is thus warranted to provide here a more constrained description of the types of movements on which this thesis focuses. Schmidt (1982, Chapter 3) discusses two broad dimensions under which movements can be classified: 1) *discrete-continuous*, and 2) *open-closed*. The first dimension deals with the structure of the movements. A discrete movement has a well-defined beginning and end, e.g., pressing a key on a keyboard. A continuous movement, e.g., walking or riding a bicycle, does not. Schmidt goes on to describe *serial movements* as "made up of a series of discrete [movements] strung together in time to make some 'whole'," and that "serial tasks can be thought of as a number of discrete [sub]tasks strung together..." (pg. 54). I focus on serial tasks in this thesis.

The open-closed dimension deals with the predictability of the environment during movement execution. Open skills are used in unpredictable environments and thus depend greatly on sensory feedback to guide movements, e.g., catching a ball on a windy day. Closed skills, on the other hand, are used in relatively predictable environments, e.g., typing on a keyboard. Sensory feedback is not needed to the same extent to execute the skill. I focus on tasks in predictable environments, and thus closed skills, in this thesis. (Also note that the terms "open" and "closed" as used by Schmidt are different than those as used in control theory, where "open loop" control refers to generating commands in the absence of feedback, while "closed loop" control refers to generating commands based on feedback.)

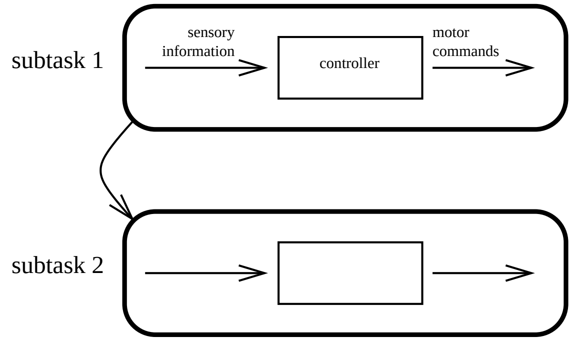

Figure 1.1. Schematic of two subtasks, where each subtask is accomplished by some controller generating motor commands based on sensory information.

1.2 Redundancy

As discussed in the beginning of this thesis, the specification of subtasks composing a motor skill is usually on a coarse level. It is very difficult, usually impossible, to offer a detailed description of all variables that must be controlled. Thus, the subtasks are often expressed in terms of goals that are clearly described and assumed to be solvable — at least to a reasonable extent — without further instruction. Also, the values of variables to be controlled may be different for different people. For example, a child, because he is small and not very strong, would hit a tennis ball so that a large proportion of its velocity is in the upward direction in order to clear the net. An experienced adult, on the other hand, would direct more energy towards the horizontal direction. A coarse description can be generalized to different people while a precise description is only useful if it is appropriate for the specific person.

The coarse specification results in *redundancy* in how each subtask is accomplished. Figure 1.1 is a schematic of a task decomposed into two subtasks. The process through which a subtask is accomplished can be divided into three parts: 1) motor commands, 2) controller, and 3) sensory information. A controller uses sensory information to determine what motor commands to generate. In a sense, there is redundancy in each of the three parts:

Motor command. There is redundancy in both the motor commands used to produce movement and the types of movements that accomplish a subtask. For example, many different patterns of muscle activity can produce the same torque on a joint, many different arm configurations can place the hand at a particular point in space, and the hand can take many different paths in moving from one spatial point to another.

Controller. Different types of control mechanisms can produce the same motor command. For example, one can swing a tennis racquet using sensory feedback mechanisms to track a ball. Alternatively, one can swing the tennis racquet in a very similar way through a purely feedforward control mechanism, without any reference to the actual position of the ball. (In the case of tennis, which requires open skills, such a control strategy is ill-advised. Nevertheless, it is available.)

Sensory information. Different types of sensory information can be used to guide movements. For example, when learning how to drive a manual transmission car, you might use the visual information provided by the tachometer to signal when to shift gears. However, after you gain some experience, you might use auditory information instead, allowing your visual resources to be directed towards the road.

Subtasks are usually devised and described so that, given our cognitive and planning abilities, we can accomplish them with little or no experience. However, the initial solution is rarely proficient enough to be considered a "skill." Redundancy in motor commands, controller type, and sensory information can be exploited to increase proficiency in the overall task. Exploitation of redundancy presents a problem the nervous system must solve. Resulting behavior defines *what* a motor skill is. The process by which redundancy is exploited is the process of motor skill acquisition. In this thesis, I view motor skill acquisition as a decision-making problem: *how* does our nervous system, with practice, settle upon particular choices in motor commands, controller, and sensory information, and *why* are such choices advantageous? To address these questions, I turn to the methods of theoretical neuroscience.

1.3 Theoretical neuroscience

Good experimental analayses discuss data within the context of some theoretical framework: Do the results support or contradict the theory? How do they modify the theory or provide further insights? Why did the results come out the way they did? Theoretical analyses attempt to formalize the theory, couched in the language of mathematics, and explore its consequences in detail (editorial, *Nature Neuroscience*, v. 12, pg. 1627, 2005). As the preface of the book *Theoretical Neuroscience* (Dayan and Abbott, 2001) explains, "[t]heoretcial analysis and computational modeling are important tools for characterizing what nervous systems do, determining how they function, and understanding why they operate in particular ways." The authors further suggest that the models, mathematical formulations of the systems to be analyzed and theories to be implemented, fall under three general types:

1. *descriptive*, in which data is described mathematically, addressing the *what* question described above,

2. *mechanistic*, in which the computational mechanisms thought to be used by the nervous system are implemented, addressing the *how*, and

3. *interpretive*, in which theoretical principles are used to address the *why*.

The types of models are not defined by the systems they study, but rather by the questions they ask. Models of each type can be used in examining behavior of ion channels, parts of neurons, whole neurons, small networks of neurons, systems of neurons, and so on all the way up to gross movements and decision making. Since I am interested in the how and why of motor skill acquisition, the work presented in this thesis uses a combination of mechanistic and interpretive modeling techniques. Since I focus on movements and how those movements are learned and executed, I model neural systems and how they control movement or make decisions.

One of the recognized founders of the field of computational neuroscience, David Marr, suggested that, when viewing the brain as a computational problem solver, the problem can be decomposed into three levels (Marr, 1982):

1. *computational problem*: what is being computed and why
2. *algorithmic problem*: how it is being computed
3. *implementation problem*: where in the brain it is being computed

(despite the similarity in language, the three levels are not meant to parallel the three types of models described in Dayan and Abbott 2001). Although the three levels can be viewed independently, they seldom are. In fact, in applying computational techniques to neuroscience, they must all be considered. At which level one "starts," though, reveals yet another dimension across which models vary. *Top-down* models focus on the functional mechanisms that explain behavior; the problems they address fall under the computational and algorithmic levels. *Bottom-up* models, on the other hand, focus on what types of behavior can be produced by the interaction of low level elements; the problems they address fall under the implementation and algorithmic levels.

Again, because I am interested in how and why motor skills are acquired, I use a top-down approach. I attempt to explain by what functional mechanisms motor skills can be acquired and why. The computational problem the nervous system solves is to find the motor commands, control method, and sensory information that produces the most proficient sequence of movements in solving a sequence of subtasks. I hypothesize how this problem is solved (the algorithmic problem). The hypothesis is based on known anatomical and physiological characteristics of the brain. Hence, I address the algorithmic problem by using the current best answers to the implementation problem.

1.4 Overview

The functional mechanisms I implement are biologically-based. Chapter 2 of this thesis provides a broad review of the neural mechanisms of motor control. The experimental work cited describes brain areas thought to be involved and what types of computations they can implement, providing us with a set of computational tools. In brief, cortical mechanisms can implement a general control method that is used to provide reasonable solutions to subtasks. Cortical mechanisms can also implement sophisticated planning techniques to provide even better solutions. In fact, many proposed solutions to the redundancy problem rely on such techniques, disregarding the contributions of other brain areas.

Because of the importance of practice in motor skill acquisition, brain areas that learn primarily through interaction with the environment — actually executing movements — can participate in solving the redundancy problem as well. Repetition is required, but it is provided by practice. There is evidence that as a motor skill is acquired, control is transferred from cortical planning areas to the basal ganglia (BG), which are thought to implement an experiential learning method. Chapter 2 describes the BG in detail, including what computations they are thought to employ, how they might participate in motor skill acquisition, and evidence that they are important for motor skill acquisition.

I then discuss how the functional mechanisms are used in acquiring motor skills. Briefly, early in learning, cortical planning mechanisms provide a reasonable solution to solving each subtask. As experience is gained through practice, control is transferred to the BG. One advantage of the theoretical approach is that certain assumptions or scenarios can be explicitly implemented. To show that the learning mechanisms of the BG can contribute in developing most behavioral characteristics indicative of motor skills, I artificially restrict the learning capabilities of cortical areas. Thus, any change in behavior is due to the learning and control mechanisms of the BG.

I further harness the flexibility and control afforded by the theoretical approach in the next three chapters, where I construct simulated environments, systems, and tasks that allow me to focus on the use of motor commands, controller type, and sensory information separately. Chapter 3 is dedicated to examining how redundancy in motor commands is exploited. This is done with the use of a simulated redundant "arm" that must hit a series of spatial goals (i.e., accomplish a sequence of subtasks). The arm is redundant in that it has more degrees of freedom to control than are necessary for the task: there are many ways by which the arm can hit each goal. Chapters 4 and 5 dispense with the arm as they do not focus on redundancy in motor commands. Rather, Chapter 4 uses a simple sequential decision task where decisions can be made by one of three different controllers that range from flexible yet computationally expensive to inflexible but computationally cheaper. The focus of Chapter 4 is on the circumstances under which each controller is advantageous. A similar task and model are used in Chapter 5, but different types of sensory information are available. Like the controllers of Chapter 4, the different types of sensory information have advantages and disadvantages, namely that some require more time or are costly, but deliver a precise estimate of the current situation. Others require less time or are less costly, but deliver a rather imprecise estimate.

Since most topics in the research of motor control began with an examination of pure behavior (as opposed to the biology that generates such behavior), behavioral characteristics are used to determine if and how redundancy is exploited. Exploitation of motor commands leads to behavior described as *coarticulation*. The use of the simplest possible controller where appropriate leads to behavior described as *automatization*. I use the term *sensory exploitation* to refer to the exploitation of redundancy in sensory information (but I am open to better terms).

The main contributions of this thesis are in the areas of neuroscience and psychology. First and foremost, this thesis describes a general framework by which motor skills are acquired. It also shows that, because of practice, the learning mechanisms of the BG (which require experience) can contribute to developing the behavioral characteristics of motor skills. In contrast, most accounts focus on how planning mechanisms attributable to cortical areas develop behavioral characteristics. I also discuss how the solutions devised by the BG are functionally advantageous.

A theoretical approach is not meant to merely validate a particular theory through simulation; it should focus on how behavior resulting from that theory differs from behavior due to other theories. In addition, it should suggest behavioral and/or neural predictions under certain experimental conditions. In each chapter, I discuss such predictions and why they might arise. A long-term goal of the work initiated in this thesis is to better understand how and why motor skills are acquired. A better understanding of the brain areas involved, and how damage to them affects behavior, can aid in the diagnosis and treatment of neural damage (Shadmehr and Krakauer, 2008). Finally, Chapter 6 provides a discussion of the general techniques used in each of the previous chapters and directions for extending those techniques.

CHAPTER 2

BACKGROUND

Considering that most, if not all, measurable behavior manifests itself in the form of movement, it is not surprising that the functional anatomy of motor control is complex. Current research continues to shape and modify our understanding of how movement is controlled and controversy persists at all levels of research. Voluntary movement is accomplished by the cooperation of three gross pathways: direct cortical control, cerebellar control, and basal ganglionic control. In this chapter I outline our current understanding of the functions of these brain areas as they relate to motor skill acquisition. While an exhaustive description of every aspect of motor control is beyond the scope of this thesis, a review of our current understanding is helpful in assessing the capabilities of biological systems.

2.1 Overview of the cortex and cerebellum

Cortex

Cortical control of behavior, i.e., what movements to execute and abstract decision-making in general, involves many areas. Research over the past decade or so has shown that primate behavior is similar to that predicted by Bayesian decision theory (Kording and Wolpert, 2006) and game theory (Glimcher, 2002), which involve incorporating statistics such as uncertainty and expected outcome. Many variables are represented in cortical areas (Yoshida and Ishii, 2006; Glimcher, 2002; Rangel et al., 2008), providing biological evidence that support their inclusion in decision-making. In general, cortical areas caudal of the central sulcus mediate the processing of sensory information and cortical areas rostral of the central sulcus mediate motor control (though this is not a hard segregation, e.g., Battaglia-Mayer et al. 2003).

As we move from the central sulcus towards the rostral end of the brain, cortical areas exhibit activity related to more abstract representations of behavior, such as a goal to be reached rather than muscle activity. This information can be used in planning, perhaps the single most human attribute. Planning, by definition, involves predicting future outcomes and making decisions based on those predictions. The area most associated with planning, the prefrontal cortex (PFC), lies at the most rostral part of the cortex. It is one of the phylogenetically youngest areas of the brain and plays a larger role in the control of behavior in primates than in other animals (which exhibit far less flexible behavior). Lesion and neuropsychiatric studies show that improper functioning of the PFC leads to inappropriate behavior and cognitive deficits (Goldman-Rakic and Selemon, 1997; Fuster, 1997). Areas of the PFC have extensive connections with higher order cortical areas involved with sensory, sensory association, and motor functions, making it able to represent both sensory signals and motor responses (Passingham et al., 2000; Barbas and Pandya, 1989) and affect behavior based on a task-relevant processed perception (Fuster, 2000; Duncan, 2001; Alexander et al., 1986). In addition, the PFC and the orbitofrontal cortex, often considered a part of the PFC, have connections with limbic regions (Rushworth et al., 2004; Tremblay and Schultz, 1999), the information of which can be used to evaluate behavior and signify the relevance of sensory information. To relate temporally separated information, the PFC demonstrates working memory capabilities illustrated by sustained neural activity (Goldman-Rakic, 1995).

The results of several studies support the role of the PFC in planning (for reviews, see Miller and Cohen 2001; Tanji and Hoshi 2008). For example, in a multi-step path planning task, the PFC represented short-term and long-term goals and actions (Saito et al., 2005; Muchiake et al., 2001), including sequential order (Shima et al., 2007). Tanaka et al. (2004) shows that the PFC represents expected future rewards and Hampton et al. (2006) uses brain imaging techniques to show that the PFC represents a model of the overall task rather than just immediate decisions.

Other cortical areas aid in making decisions and executing movements. As we move towards the central sulcus, cortical areas tend to represent activity more directly related to movement. The supplementary motor area (SMA) and preSMA are involved in the learning and execution of sequential movements (Tanji, 2001). Premotor areas (PMAs), including the ventral and dorsal premotor cortex (PMv and PMd, respectively), exhibit activity related to abstract representations of movement. Traditionally, the primary motor cortex (MI) has been implicated as the conduit through which the cortex controls movement. This is because MI neurons represent movement on a concrete level (such as muscle activity) as well as a more abstract level (such as direction of movement) (Kakei et al., 1999) and have direct connections with areas of brain stem and spinal cord. However, recent evidence complicates the matter. Other motor areas, such as PMv, PMd, and SMA, also have direct connections to brain stem and spinal cord (Dum and Strick, 2002), allowing for parallel control.

Each of the different cortical areas may participate in various aspects of motor learning such as motor skill acquisition and adaptation to novel environments (Sanes, 2003). For example, recent evidence shows that MI represent learned sequences of movements (Matsuzaka et al., 2007; Hatsopoulos et al., 2003). In short, cortical control of behavior and movement is a complex process, but the complexity allows cortical areas to process information and control movement in a variety of ways (Carson and Kelso, 2004).

Cerebellum

Cortical areas project to the cerebellum as well. Much of the input that the cerebellum receives is sensory-related, and the cerebellum projects mainly to motor areas of the cortex, brain stem, thalamus, and spinal cord. Signals are sent to the cerebellar cortex through two main pathways. In the first, nuclei in the spinal cord and brain stem send *mossy fibers* to the cerebellar cortex. These axons terminate on granule cells, which give rise to *parallel fibers*, which project parallel to the surface of the cerebellar cortex and make weak contacts on the dendrites of many Purkinje cells. This pathway is thought to carry sensory information from the peripheral nervous system and cortex. In the second pathway, more abstract sensory information, arising from the cortex, is sent to the inferior olivary nucleus (IO). The IO sends *climbing fibers* up to terminate on the soma of Purkinje cells. Unlike parallel fibers, a climbing fiber makes strong contacts on a small number of Purkinje cells. Purkinje cells send inhibitory projections to deep cerebellar nuclei, which send excitatory connections to the thalamus. The regular architecture and synaptic plasticity of the cerebellum led Marr (1969) and Albus (1971) to suggest that it can be trained to participate in controlling movement after learning to recognize patterns of input. Berthier et al. (1993) present a model which uses climbing fiber input to train the response of Purkinje cells to a particular pattern of parallel fiber activation. In addition, Kawato (1999) suggests that the cerebellum aids in motor control by creating an internal model to be used for predictions.

Lesion studies show that disruption of the cerebellum leads to awkward, uncoordinated movement, and the inability to adapt to dynamic environments. Seidler et al. (2002) show that the cerebellum may not be used in the learning of sequences in a serial reaction time (SRT) task, but may be used to aid in the execution of movements in the SRT and other tasks. Other recordings show that neural activity in the cerebellar cortex represents intended movement and error between actual movement and the goal (Kitazawa et al., 1998). These, and other aspects of its anatomy and physiology, lead some to suggest that the cerebellum plays a role in movement correction or supervised learning (Doya, 1999). However, the role of the cerebellum in motor control and learning continues to be a topic of current research and debate (for reviews, see Schweighofer et al. 2004, Garwicz 2002, Ito 2000, and Houk et al. 1996).

Functional role

Overall, the cortex and cerebellum participate in controlling and learning movement through complex distributed pathways and new research is always refining our view of these areas. Because I focus on the decision-making aspects of motor skill acquisition in this thesis, the functional mechanisms of cortical areas serve two main purposes: 1) to provide a representation of relevant sensory information from which to make decisions, and 2) to act as a general planner with which to make decisions. For similar reasons, the role of the cerebellum in my models is not to make decisions, but rather to ensure that decisions made by other areas (e.g., to hit a particular goal) are implemented. Thus, in the event that a movement does not accomplish the subtask for which it was selected, as may be the case in a stochastic environment or if a novel exploratory movement is chosen, the cerebellum aids in controlling movement to accomplish the subtask.

These simplifications are not meant to suggest that the cortex and cerebellum play minor roles in motor skill acquisition. On the contrary, decision-making in motor skill acquisition is manifested as movements, and many functions of cortical areas and the cerebellum serve to properly execute those movements. They play important roles and can account for many types of behaviors characteristic of motor skills. However, in this thesis, I claim that with practice the learning mechanisms of the basal ganglia can account for characteristics often attributed to cortical or cerebellar mechanisms. To show this, I artificially restrict the capabilities of the cortex and cerebellum to the general functions described in the previous paragraph in models presented in this thesis. The next section describes the architecture and physiology of the BG.

2.2 Anatomy and physiology of the basal ganglia

Pathways

The basal ganglia are a set of forebrain nuclei (see figure 2.1) that participate in motor control through multiple pathways and mechanisms (for reviews, see Bar-Gad et al. 2003, Mink 1996, Bolam et al. 2000, Graybiel and Kimura 1995, Graybiel 1998, and Graybiel 2005). The BG communicate with most areas of the cortex and motor areas of the thalamus and brain stem through cortico-ganglio-thalamic loops (Middleton and Strick, 2000; Hoover and Strick, 1993; Kelly and Strick, 2004), which I shall discuss in detail later. The striatum (consisting of the caudate, putamen, and nucleus accumbens) receives excitatory projections from cortex and thalamus and is considered to be the main input nucleus of the BG. Some corticostriatal projections are branches from corticothalamic and corticospinal projections (Zheng and Wilson, 2002), suggesting a mechanism by which the BG can receive an efference copy of motor commands generated by cortical areas. The striatum projects to other nuclei of the BG, including the substantia nigra pars compacta (SNpc), the external segment of the globus pallidus (GPe), and the internal segment of the globus pallidus (GPi). The GPi and another nucleus, the substantia nigra pars reticulata (SNpr), are typically grouped together because they share similar characteristics and compose the output nuclei of the BG. GPi/SNpr send inhibitory projections to areas of the thalamus and brain stem. Thus, the BG ultimately control movement through disinhibition of thalamic and brain stem areas.

The projection neurons of the striatum are called *medium spiny neurons* because of their size and the presence of spines on their dendrites. Medium spiny neurons are GABAergic (they communicate to other neurons with the chemical gamma-aminobutyric acid, GABA) — they send inhibitory projections to their targets. In addition to projection neurons, the striatum contains several types of inhibitory interneurons which may mediate lateral inhibition in the striatum. The most studied type are called tonically active interneurons (TANS) and, as the name suggests, exert a tonic inhibitory influence on spiny neurons.

The BG project to the thalamus and brain stem through three main pathways, described as follows ($\oplus$ means that the connection has an excitatory effect and $\ominus$ means that the connection has an inhibitory effect):

$\ominus$**Hyperdirect Pathway:** [Cortex $\rightarrow \oplus$ STN $\rightarrow \oplus$GPi/SNpr$\rightarrow \ominus$movement]

The cortex has direct excitatory connections to the subthalamic nucleus (STN)

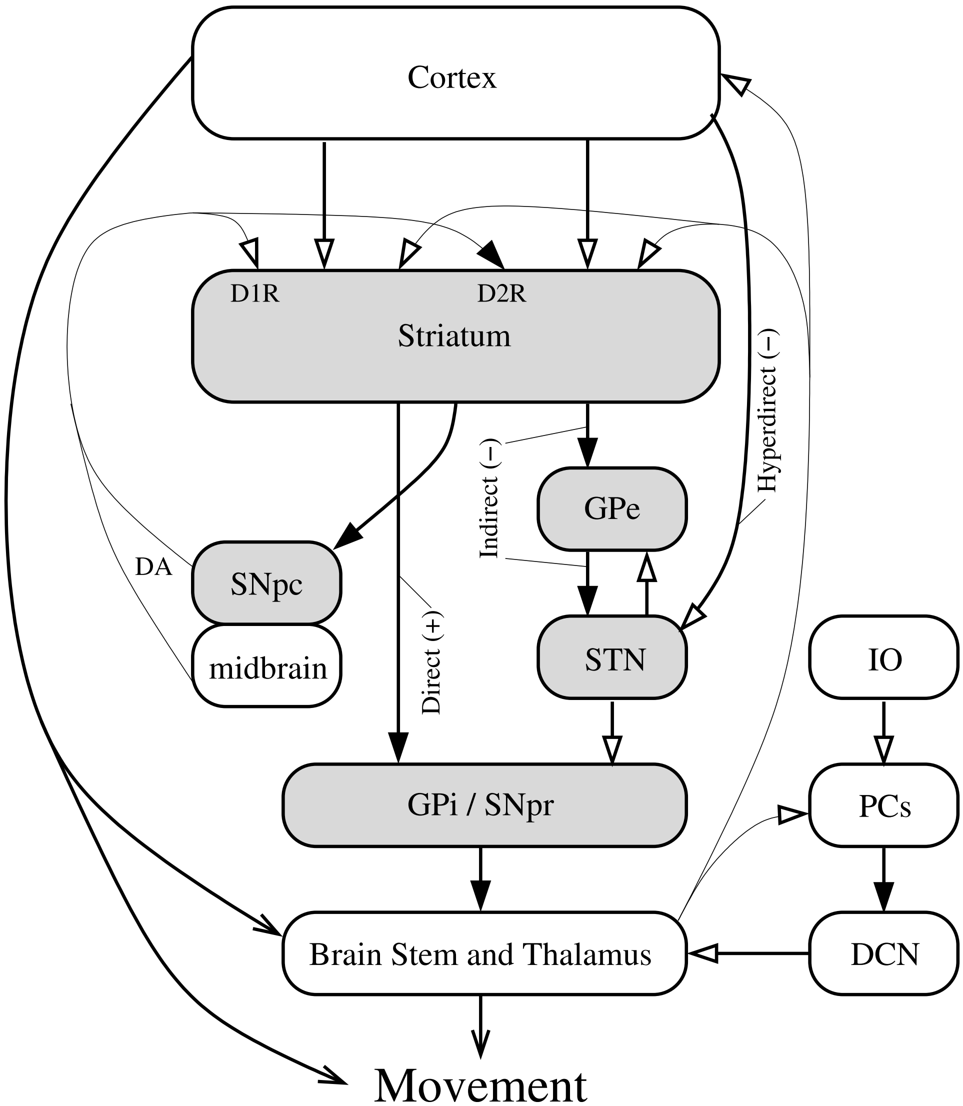

Figure 2.1. Gross representation of major pathways involved with movement. Basal ganglia nuclei are in shaded boxes. Excitatory connections are in unfilled closed arrows, inhibitory connections are in filled closed arrows, and mixed connections are in open arrows. Thin lines represent ascending connections for clarity. The "(+)" or "(-)" signs following the direct, indirect, and hyperdirect pathway labels indicated the individual pathway's affect on on movement. (+), disinhibits movement, (-), inhibits movement. SNpc, substantia nigra pars compacta; DA, dopamine, D1R, D1-type DA receptor; D2R, D2-type DA receptor; GPe, globus pallidus external segment; GPi, globus pallidus internal segment; SNpr, substantia nigra pars reticulata; STN, subthalamic nucleus; IO, inferior olivary nucleus; PCs, Purkinje cells of the cerebellum; DCN, deep cerebellar nuclei.

of the BG. The STN, in turn, has excitatory connections to the GPi/SNpr, which inhibits movement. Faster than direct or indirect.

$\oplus$**Direct Pathway** [Cortex $\rightarrow \oplus$ Striatum $\rightarrow \ominus$ GPi/SNpr$\rightarrow \ominus$movement]
Excites movement through disinhibition. The cortex excites the striatum, which inhibits the GPi/SNpr, which inhibits brainstem and thalamus circuits.

$\ominus$**Indirect Pathway** [Cortex $\rightarrow \oplus$ Striatum $\rightarrow \ominus$ GPe $\rightarrow \ominus$ STN $\rightarrow \oplus$ GPi/SNpr$\rightarrow$ $\ominus$movement]
Enhances inhibition of movement by disinhibiting the STN.

Movement is controlled by the BG through a balance between the three pathways.

In addition to the nuclei already mentioned, the BG include the substantia nigra pars campacta (SNpc), which consist of dopaminergic neurons and is adjacent to other groups of midbrain dopaminergic neurons. The striatum sends inhibitory projections to the SNpc, and the SNpc and midbrain dopaminergic neurons send projections that terminate on the striatum, including on corticostriatal synapses. Dopamine (DA) is part of the brain's reward processing system. However, DA also may have a more direct effect on BG activity. Striatal projection neurons contain either D1-like receptors (D1R, excitatory DA receptors) or D2-like receptors (D2R, inhibitory). D1R containing striatal neurons typically project through the direct pathway, while D2R containing striatal neurons typically project through the indirect pathway (Gerfen et al., 1990; Aubert et al., 2000). Thus, DA has a net effect of providing excitation to brain stem and thalamus. This effect partially explains the symptoms of the BG-associated movement disorders Parkinson's disease and Huntington's disease (Boraud et al., 2002; Obeso et al., 2002). Parkinson's disease is due to a degeneration of DA neurons in the SNpc. The decrease in DA to the striatum results in greater BG inhibition of the thalamus and brain stem and hence the hypokinetic symptoms of Parkinson's. Huntington's disease is accompanied by an overproduction of DA and hence a decrease of BG inhibition of the thalamus and brain stem, resulting in hyperkinetic symptoms.

The available research on the characteristics of the remaining nuclei of the BG is not as in depth as that of the striatum, and hence their descriptions are brief. The GPe sends inhibitory projections to non-TANS interneurons in the striatum, the STN, and GPi/SNpr, as well as to excitatory projection neurons of the STN. GPe neurons tend to exhibit high frequency activity interspersed with pauses in that activity. The STN neurons are excitatory and are tonically active, except during movement, when they fire short bursts. STN projections to GPi/SNpr have been characterized as diffuse (Parent and Hazrati, 1995; Gurney et al., 2001). The GPi/SNpr neurons are inhibitory and exhibit high frequency activity with no or few pauses.

In the following subsections, I describe in more detail characteristics of the striatum and pathways germane to motor skill acquisition.

Cortico-ganglio-thalamic loops

The pathways through the BG include cortex $\rightarrow$ BG $\rightarrow$ thalamus $\rightarrow$ cortex. Most parts of the cortex communicate with the BG through such cortico-ganglio-thalamic loops (Middleton and Strick, 2000; Hoover and Strick, 1993; Kelly and Strick, 2004; Alexander et al., 1986; Alexander and Crutcher, 1990), enabling the BG to affect movement based on the different types of information available through the highly processed representation of information in cortex. Many argue that these loops are segregated to a large degree — one part of the cortex typically does not communicate with another part of the cortex in a direct or nearly direct way (Takada et al., 1998; Tokuno et al., 1999; Parent and Hazrati, 1995; Hoover and Strick, 1993; Middleton and Strick, 2000). For instance, sensory areas of the cortex project mainly to the dorsolateral striatum, associative areas project to the central striatum (caudate), and limbic areas project to the ventromedial striatum (including the nucleus accumbens). Some experiments suggest that there is also little or no overlap between higher motor and executive motor information in the striatum (Takada et al., 2001).

The results of other studies suggest that there is some communication between the loops through overlapping corticostriatal projections. It has been shown that each striatal spiny neuron receives projections from thousands of cortical neurons (Bolam et al., 1993), and some striatal neurons respond to stimulation of multiple cortical areas (Nambu et al., 2002; Kimura et al., 1996; Yoshida et al., 1993). Parts of cortex that communicate with each other may provide overlapping inputs to the striatum. Graybiel showed that by stimulating an area of cortex, several non-contiguous patches in striatum were activated (Graybiel et al., 1994; Graybiel and Kimura, 1995). Flaherty and Graybiel (1991) show that there may be some overlap between MI and SI projections onto striatal neurons. In addition to convergence of cortical information to striatal spiny neurons, Ramanathan et al. (2002) shows that there is some convergence of MI and SI projections onto striatal interneurons, suggesting that intrinsic connections may also shape striatal spiny neuron activity.

There is also evidence for indirect communication between loops (Haber, 2003; McFarland and Haber, 2002). Connections from striatum to GPi/SNpr diverge to some degree; cortical areas receive information generated from other cortical areas in an indirect way. In addition, the dopaminergic pathways may provide a means by which different loops communicate with each other (Joel and Weiner, 1994, 2000; Haber et al., 2000).

The properties described in this section indicate that while the loops are segregated to a large degree, there is likely some communication between them. If we equate activity of a loop with a particular movement, then the segregation allows for two (or more) movements to be executed concurrently, while the communication allows for the movements to be modified to some degree to take into account preceding, concurrent, or subsequent movements.

Exploration

The inhibitory nature of the output nuclei of the BG allow them mediate movement exploration by selecting from and/or modifying motor commands as suggested by other brain areas. Many motor-related areas of the brain send excitatory projections to brain stem and thalamus circuits, leading to a convergence of movement-related excitation (Mink, 1996). The activation patterns may not be consonant —

different brain areas may elicit different motor commands. Without some form of inhibition, such a confluence would result in ataxic movements (Gurney et al., 2001; Mink, 1996). One role of the BG may be to provide the inhibition necessary to damp the existing motor signals so that only one set, or some combination of sets, of motor commands is used to generate movement. I refer to this as a form of *directed* exploration because movements are modified through the influence of other motor commands. In addition, through noise in neural activity (possibly mediated by the excitatory effects of DA), the BG may be able to mediate a form of *undirected* exploration — elicit novel motor commands based on those elicited by other motor areas of the brain, but modified based on noise rather than the influence of other motor commands. In both directed and undirected exploration, the BG is not faced with the task of generating motor commands; instead, they mediate the relatively easier problem of modifying existing motor commands.

There is evidence of weak proximal lateral inhibition in striatum spiny and interneurons (Bolam et al., 2000; Wilson and Oorschot, 2000). Lateral inhibition may enable activity selection mechanisms such as a "winner take all" system, in which the most active of neurons in a pool of active neurons eventually remains as the only active neuron. The weak nature of the lateral inhibition may allow for a softer competition, such as a softmax distribution, in which stronger activity patterns are selected with a higher probability than weaker activity patterns, or a "winner share all" mechanism, in which several patterns are allowed (Fukai and Tanaka, 1997). The proximal nature of the lateral inhibition prevents distant groups of spiny neurons from inhibiting each other, allowing for concurrent activation of multiple movements.

Dopamine and learning in the BG

Synaptic plasticity at the corticostriatal synapses can mediate evaluation of the exploratory processes described in the previous subsection and thus learning. Cortical and DA projections to the striatum form asymmetric synapses on the spines of spiny neurons (Schultz, 1998). Corticostriatal plasticity is dependent on DA (Centonze et al., 2001; Wickens et al., 2003) and occurs in the form of both long term potentiation (LTP) and long term depression (LTD), though the exact circumstances that elicit one or the other has not yet been determined. We can model synaptic plasticity on an abstract level as a three factored Hebbian form: activity of the striatal cell, activity of the cortical cell, and presence of DA. Some form of plasticity may also occur at the STN→pallidal synapses (Hanson and Jaeger, 2002), while the stability of synapses in the GPi may be dependent on DA (Ingham et al., 1997; Whone et al., 2003). However, these latter processes are not understood in great detail as of yet.

A widely accepted computational role of the DA signal is that of *reward prediction error* (Schultz 1998, but see also Niv et al. 2005), in which the activity of the DA neurons (in the SNpc or the ventral tegmental area, VTA) is approximated by the difference between the reward received and the reward expected. For example, if a reward occurred as expected, the DA neuron activity would be baseline. If the reward was higher than expected, it would be above baseline, and if it was lower than expected, it would be below baseline. Plasticity, and hence learning, only occurs if the reward is not as expected.

Reward-related learning may also mediate other forms of learning. Tonically active interneurons (TANS) in the striatum change their activity according to reward-predicting cues. In an operant conditioning task, when a rat is presented with a cue that predicts a reward, some TANS decrease in firing (Aosaki et al., 1994b,a). This removes the tonic inhibition to the striatal neurons, allowing them to disinhibit motor commands. The tonic firing of TANS can also be easily modified by extrastriatal sources (Aosaki et al., 1994b).

The parallel control of motor output by the direct and indirect pathways may also allow for different types of learning to occur. Striatal neurons in the indirect pathway express D2Rs and MAPK (Gerfen et al., 2002), which is implicated in synaptic plasticity (Thomas and Huganir, 2004; Sharma and Carew, 2004). One possible consequence of this is that long term learning (e.g., skill formation) may be modulated through the indirect pathway, but short term control is modulated through the direct pathway (Gerfen, 2004).

Finally, dopamine neurons have been shown to signal not just reward or reward prediction error, but also *salient* sensory events, such as unexpected or highly intense sensations (Horvitz, 2000, 2002). Through DA-mediated learning, the BG may also play a role in detecting novel or previously unattended stimuli.

Influence of the thalamus

Although BG activity is influenced strongly by corticostriatal projections, thalamostriatal projections may also affect BG activity (Smith et al., 2004). The thalamus has been traditionally thought of as an active conduit through which the cortex communicates with lower brain areas. It gaits and shapes ascending sensory information to the cortex and mediates descending motor information from the cortex, BG, and cerebellum. The thalamus is composed of many nuclei, some of which project to distinct areas of cortex and mediate specific functions. The ventral motor areas of the thalamus are thought to convey motor information from the BG and cerebellum to cortical motor areas, while the ventral sensory areas convey sensory information. The exact functions of many thalamic nuclei are not well understood; the nuclei seem to project diffusely to cortex. For example, a set of nuclei called the intralaminar nuclei receive projections from subcortical areas and cerebellum and project to limbic areas of the cortex and the BG. Intrathalamic connections help shape and integrate activity in the thalamus as well.

Thalamic activity represents processed information as well as direct information. The centre médian (CM) and parafascicular nucleus (Pf) intralaminar nuclei of the thalamus convey behaviorally significant sensory information to striatum, including unexpected sensory stimuli (Matsumoto et al., 2001). Neural response was found to be reward-independent yet necessary for the responses of TANS to rewarded stimuli in the striatum. Matsumoto et al. (2001) suggest that the CM-Pf complex aid in activating learned responses of striatal neurons; it is degenerated in Parkinson's and

Huntington's disease patients (Henderson et al., 2000; Smith et al., 2004), further supporting its importance in proper BG functioning.

Bistability of striatal neurons

Some striatal spiny neurons exhibit sustained activity (Hikosaka et al., 1989) and most exhibit bistable behavior (Wilson and Groves, 1981; Nicola et al., 2000; Wilson, 2008). In a "down" state, in which the resting potential is hyperpolarized, a spiny neuron will not be excited by weak inputs. In an "up" state, in which the resting potential is more depolarized, weak inputs may excite it. While controversy persists over whether bistability is a characteristic inherent to spiny neurons or the result of cortical inputs (Wilson, 2008; Kasanetz et al., 2006), the observed bistability allows for the influence of weak inputs to be diminished or amplified depending on what state the spiny neuron is in. The bistable properties may be modulated by DA (according to a modeling study by Gruber et al. 2003) or extrinsic connections (Wickens and Wilson, 1998).

Functional role

The anatomical and physiological characteristics of the BG enable them to facilitate several types of functional roles (Graybiel, 2005; Gurney et al., 2004), including pattern recognition (Shadmehr and Wise, 2005; Houk and Wise, 1995; Graybiel, 1998), dimensionality reduction (Bar-Gad et al., 2003; Joel et al., 2002), preparation for movement (Hikosaka et al., 2000), focused selection of motor activity (Hikosaka et al., 2000), and efficient mediation between competing decisions (Bogacz and Gurney, 2007).

Perhaps the most widely accepted role of the BG involves reward-related learning due to the rich DA projections to the striatum and DA-dependent plasticity at corticostriatal synapses. Cortical areas provide for a representation of relevant sensory information from which to select a movement and, through the planning processes described in the earlier section, *Cortex*, motor commands to accomplish a particular subtask. Through exploration and reward-mediated learning, the BG can learn to execute those movements and possibly find better ones.

The DA signal combined with DA-dependent plasticity may allow the BG to learn in ways similar to the algorithms of Reinforcement Learning (RL, Sutton and Barto 1998), a computational formulation of learning from the consequences of decisions executed, often referred to as *actions* in the RL literature. In essence, if an action is followed by a favorable outcome (e.g., a reward greater than the expected reward) the tendency to select that action is increased (cf. Thorndike 1911; in the language of psychology, that action is *reinforced*). Expected reward of actions may be represented in the activity of striatal neurons (Samejima et al., 2005).

One attractive feature of RL is that, unlike planning, a model of the environment is not necessarily required. Thus, learning is done by interacting with the environment — executing an action and observing the resulting change in environment. Houk et al. (1995) and Barto (1995) present models of how the BG might implement RL, and Doya (2007) reviews further connections between RL and behavior. I discuss RL in more detail later in this chapter. In addition, thalamostriatal projections may provide the BG with a representation of sensory information that is less processed and presumably occurs earlier in time than that of cortical areas. If such projections are weak, the bistable properties of striatal neurons would enable those projections to elicit movements selectively. Thus, a more efficient form of movement execution is possible with the machinery of the BG. In the next section, I review evidence that the BG and DA are important in motor skill acquisition.

2.3 BG in Motor Skill Acquisition

The previous section described aspects of anatomy and physiology that enable the BG to perform functions useful in motor skill acquisition. In this section, I describe experimental evidence that the BG does, in fact, play a role in motor skill acquisition. Descriptions are grouped by methodology.

Lesion and neural recording

An unlearned form of motor skill, termed a "syntatic chain" (Berridge et al., 1987) is expressed as a set sequence of grooming actions in rodents. Once the first part is initiated, the rest of the actions can be predicted with 85% accuracy. Lesion studies suggest that the striatum recruits and coordinates circuits involved with syntatic chains (Berridge and Fentress, 1987; Berridge, 1989a; Berridge and Whishaw, 1992; Cromwell and Berridge, 1996). Recording studies show that neurons in the striatum respond differently to grooming actions in isolation versus the same actions in the context of a syntatic chain (Aldridge et al., 1993; Aldridge and Berridge, 1998). Thus, the BG may be involved with this unlearned form of sequenced behavior.

The role of the BG in acquiring motor skills can be studied by training an animal to perform novel tasks. Lesion studies show that the striatum is important in learning complex visual stimulus response tasks (Reading et al., 1991), but not necessarily in simple unlearned movements (Aldridge et al., 1997). Neural recording studies also show that the BG are involved with learned behavior. When trained to perform movements at the onset of a cue, striatal and pallidal neurons have been shown to be selective for the cue signal in the context of the task, but not out of context (Gdowski et al., 2001; Kimura, 1986, 1990; Schultz and Romo, 1992; Romo et al., 1992; Gardiner and Kitai, 1992) or even if presented redundantly within context (Kermadi et al., 1993; Kemardi and Joseph, 1995). Similarly, striatal and pallidal neural activity responds to a movement during one task differently than the same movement during another task or outside of any task (Schultz and Romo, 1992; Romo et al., 1992; Gardiner and Kitai, 1992; Brotchie et al., 1991; Kimura et al., 1992). The context-specific behavior of BG neurons indicate that they participate in motor control not by merely controlling a specific movement, but by controlling that movement as part of a motor skill.

BG activity evolves as a task is learned as well. In learning a multi-step task, the proportion of striatal neurons in rat that display task-related activity increases as the rat learns the task (Jog et al., 1999). Some neurons became responsive to cue aspects of the task, while others became responsive to movement aspects of the task.

Imaging

Human imaging studies show that the BG play a role in the learning and execution of motor skills (Grafton et al., 1992; Hazeltine et al., 1997; Jenkins et al., 1994; Jueptner et al., 1997; Toni et al., 1998). Boecker et al. (1998) use positron emission tomography (PET) to show that regional cerebral blood flow (rCBF) in the anterior globus pallidus in humans increases as task complexity (e.g., number of movements to be made) increases. Grafton et al. (1995) had human subjects perform an SRT task under two conditions: as a single task or in conjunction with another, "distractor," task. Using PET imaging, they showed that different cortical areas were activated during the different conditions. One distinction was that the SMA was involved with the dual task condition and the PFC and PMAs were involved with the single task condition. In both conditions, the BG were activated, indicating that they play a role in learning and execution in SRT tasks. The cortical results suggest that when one can devote full attention to the task, as with the single task condition, planning areas are involved. Under the dual task condition, more automatic systems are used to perform the task.

Puttemans et al. (2005) used functional magnetic resonance imaging (fMRI) to show that the anterior cerebellum and putamen were the only brain structures they measured which increased in metabolic activity during the entire progression of a human learning a bimanual coordination task. Doyon and Benali (2005) review imaging evidence suggesting that, when a skill is well learned, the representation of the sequence is transferred from cortical areas of the brain to the corticostriatal synapses. Rauch et al. (1998) measured a decrease in metabolic activity in the thalamus during early learning stages of a sequential task. This may represent an increase in neural activity in the thalamus due to a decrease in GPi/SNpr inhibitory projections to the thalamus.

Disorders of the BG

Studying humans with disorders of the BG also helps us understand the BG's role in motor skill acquisition and performance. Parkinson's disease (PD) and Huntington's disease (HD) are the most common type of disorders studied. While the exact mechanisms by which PD and HD affect movement is not fully understood, they both involve degeneration of parts of the BG (as described earlier) and thus impairments in moving.

In a sequential button pushing task, normal subjects were able to execute a repeated set sequence faster than random ones (even if they were unaware of them), but patients with PD and HD did not perform any better on the repeated sequence (Jackson et al., 1995; Knopman and Nissen, 1991). PD patients performed worse than controls on other types of movement sequence tasks, including performing worse as the movement sequence grew longer and more complicated (Agostino et al., 1992) and in speaking tasks (Volkmann et al., 1992). PD inhibited performance on well-known tasks as well. In performing a sequence of two simple movements, PD patients performed each movement longer in the sequence than separately (Benecke et al., 1987) and also had problems performing two simple movement simultaneously (Benecke et al., 1986). Benecke and colleagues suggested that PD interfered with the ability to switch from one motor program to another efficiently and to perform two motor programs concurrently. Tyrone et al. (1999) studied the behavior of sign language users with PD versus normal sign language users in a finger spelling task. They found that the movements used by PD patients were less smooth and coordinated. Tyrone et al. (1999) hypothesized that PD patients adopted the strategy of reducing the motor demand by executing each movement separately, without taking into account the overall task.

BG in exploitation of sensory redundancy

The strong link between the BG and movement convinces most that, if the BG does participate in motor skill acquisition, it aids in exploiting redundancy in motor commands. The transfer of control as suggested by the imaging studies above suggest that they may play a role in redundancy in control as well. I discuss further evidence supporting this role in Chapters 3 and 4. However, the role of the BG in the exploitation of sensory redundancy is not as obvious. In this section I review evidence suggesting that the BG play a role in using sensory information in movement tasks. Since the behavioral experiments that elucidate this exploitation are complex, such evidence is in the form of brain imaging results with human subjects and performance of normal human subjects compared with those who suffer from diseases of the basal ganglia.

Debaere et al. (2003) had subjects make periodic hand movements with both hands, with a 90° phase difference between the hands. This phase shift is typically difficult to learn and is halfway between easily-learned phases shifts of 0° and 180°. In addition, the subjects made the movements with either normal visual feedback or augmented visual feedback, in the form of Lissajous figures, which plot the displacement of one hand versus another. fMRI shows that with the augmented visual feedback, the rCBF in the BG is less than that without the augmented visual feedback. In a similar task, Verschueren et al. (1997) shows that normal subjects perform better than PD patients without the augmented visual feedback, but that PD patients perform similar to normals with the augmented visual feedback. In addition, without augmented visual feedback, poor performance by PD patients differed from that of normal subjects. PD patients would revert to the more intuitive phase difference of 0°, in which the hands are synchronized, while normals would tend toward a phase difference of 180°, in which the hands are in antiphase. In general, normals would not revert to previously learned phase differences as much as PD patients would. These results demonstrate that the BG are important in learning novel complex movements and establishing a stable motor skill which relies on internal cues, which must be learned to some degree (as opposed to sensory information provided by the task). However, with appropriate external feedback, the role of the BG is not as prevalent.

Two other studies support the notion that the BG are involved with learning to initiate or control motor skills with internal cues. Taniwaki et al. (2003) used fMRI to show that, in a sequential finger movement task with human subjects, the cortico-ganglio-thalamic loop was used in self-paced, but not externally paced, movements. In a "connect the dots" type of task, subjects connected a sequence of squares on a screen with a stylus (with no immediate feedback such as an ink trail). After learning the task, the subjects were asked to execute the same movements but without the visual feedback of the squares on the screen. Normal subjects performed better than PD patients, showing that PD patients were not able to rely on internal cues like normal subjects were (Martin et al., 1994).

The BG may also be responsible for mediating the control of movement by subliminal external sensory cues. Aron et al. (2003) developed a task in which a cue on a computer screen indicated in which direction a human subject should point. Before the trigger cue, a subliminal cue (presented for 32 msec) was presented. If the subliminal cue was compatible (the same as the trigger cue), and the interstimulus interval (ISI) was short, the cue resulted in decreased reaction time after the trigger, but if the ISI was longer, the compatible subliminal cue resulted in a longer reaction time. The opposite was the case if the subliminal cue is incompatible. These results support the theory discussed in Mink (1996), which stated that the BG aids in inhibiting motor programs. If the human subject subliminally inhibited the motor response indicated by the cue during the long ISI trials, then reaction time when presented with that cue as a trigger would be longer. Likewise, if the competing motor program was inhibited, as is the case when the subliminal cue was incompatible, then the reaction time would be shorter. fMRI scans during this task show that the caudate and thalamus may mediate this effect. Also, the behavior of HD patients deviated from that of normal subjects.

Dopamine in motor skill acquisition

Neurochemical analyses show that DA is important for the learning and execution of motor skills. 6-hydroxydopamine (6OHDA), which destroys nigrostriatal projections, in the striatum disrupts the ability to complete a learned motor skill in the rat (Sabol et al., 1985). DA antagonists or 6OHDA in the striatum also decrease the percentage of completed syntatic chain grooming behavior in the rat (Berridge and Fentress, 1987; Berridge, 1989a,b). Targeted mutation studies show that D1R activation aids in completion of grooming behavior, while D2R activation disrupts grooming behavior (Bolivar et al., 1996; Cromwell et al., 1998; Berridge and Aldridge, 2000a,b). Cocaine and amphetamine (DA agonists), injected intraperitoneally (Canales and Graybiel, 2000) or directly into the striatum (Dickson et al., 1994), induced stereotypy, in which movements or sequences of movements are repeated without any external prompting. From the studies described in this paragraph, we can postulate that DA, though D1R's, is necessary for the maintenance and completion of motor skills. D2R's, on the other hand, may prevent the maintenance of motor skills (important to suppress inappropriate movements).

A study by Matsumoto et al. (1999) investigated the role of DA in the development and execution of learned motor skills. The authors infused 1-methyl-4-phenyl-1,2,3,6-tetrahydropyridine (MPTP) into the monkey striatum (unilaterally), to destroy the nigrostriatal DA projections to that area either before or after the monkey learned a sequential three button pushing task. After training, normal monkeys were able to quickly push the three buttons; monkeys treated with MPTP prior to learning exhibited slower movements than normals. Monkeys treated with MPTP after learning also exhibited slower movements than normals, but the affect was less than that of monkeys treated with MPTP prior to learning. In another permutation of the task, when reward was given after the second button push (after the monkey was trained with reward after the third button push), monkeys treated with MPTP quickly learned to not press the third button. Normal monkeys kept pressing the third button for a number of trials even though it wasn't necessary, showing that DA in the monkey striatum aided in encoding the sequence of movements necessary to push the three buttons as one motor skill.

Neurochemical studies also show that an optimal level of DA in striatum is important for the selection or exploration of appropriate behaviors (Graybiel and Rauch, 2000). DA antagonists disrupted the rats' ability to explore — try out different motor skills without being cued to do so — if the current one was not accomplishing the task (Cools, 1980). DA antagonists also resulted in inappropriate behavior selection in the rat (Pellis et al., 1993).

Summary

In this section, I reviewed experimental evidence using a range of techniques: neural recordings in animals, chemical and physical lesion in animals, neurochemical manipulations in animals, and brain imaging studies in humans with and without disorders of the BG. All studies coupled techniques with behavioral tasks. While the studies are not conclusive, partly due to the necessary lack of precision of some of the techniques, their conglomerate results strongly support the notion that the BG and DA play an intimate role in motor skill acquisition.

2.4 Functional Mechanisms

In the previous sections, I outlined anatomical and physiological properties which provide the cortex, BG, and related motor areas with tools that aid in the acquisition of motor skills. I have also described experimental studies showing that the BG and DA are critical in the learning and execution of motor skills. In this section, I discuss, on a conceptual level, how these functional mechanisms are used in this thesis. What immediately follows is a description of a generic motor skill acquisition task; the description also illustrates the general level of abstraction I use in this thesis. Following that, I discuss how the brain areas described earlier can be used to accomplish the task.

Generic task

As discussed in the introduction, the type of motor skills I investigate in this thesis involve solving a task that can decomposed into a sequence of discrete subtasks. Thus, the learner, or *agent*, must solve a *sequential decision problem* in that it must make a sequence of decisions to accomplish a task. What decision the agent makes depends on the environmental situation, or *state*, it is in. When it executes a decision, it may be transported to another state. Because decisions affect state, decisions are often referred to as *actions*. When the agent selects an action, it receives an immediate numerical *reward* that may be dependent on the action selected and state it winds up in. A reward can be considered analogous to the amount of effort and/or time required to make a movement. The typical task is for the agent to move to a goal state while maximizing reward.

Consider, as a simple example, a grid of squares (like a checkerboard). This grid comprises the *state space*, the space of all states (S). s corresponds to one of the squares and from each state, it must choose an action, a, from a set of available actions, A, that move it to another state. For example, A can consist of four actions: each moves the agent one square in one of the four cardinal directions (north, east, south, and west). (In some tasks, the set of available actions depends on state, i.e., some actions are available in only certain states.) Every time the agent selects an action, it receives a reward of -1; when it happens upon the square in the top right corner (the goal state), it also receives a reward of $+100$. Since it's goal is to maximize reward, the agent must learn to move towards the top right corner in as few steps as possible.

In this example, since there is only one goal, state is simply the position, or square, the agent is in. However, in other tasks, there may be more than one possible goal. In order to select the best action, the agent must have some representation of both position and goal; thus, both may be used to define state. As described under the functional roles of the cortex and cerebellum (page 10), one role of cortical mechanisms is to provide a representation of state, including position and goal from the checkerboard example. In addition, as described under functional roles of the basal ganglia (page 17), the thalamus may also provide a less rich representation, such as simply position in the checkerboard example. This distinction is examined in Chapter 4.

Planning and error correction

As described in the earlier section, *Overview of cortex and cerebellum*, cortical planning mechanisms can select actions through a planning process. In order to plan well, an accurate representation of the entire state space (e.g., the entire checkerboard), goal (e.g., which goal must be reached to accomplish a task, if there is more than one possible goal), and characteristics of the goal (e.g., at which specific square on the checkerboard the goal is located). Planning mechanisms can then search through possible trajectories of states so as to select the best action from the current state to achieve a particular goal. This type of planning is described in greater detail in Chapter 4 and is used in both Chapters 4 and 5.

Error correction, on the other hand, involves selecting actions to transition to the intended state. For example, if the agent intended to move to a particular square, but for some reason ended up in a different square, an error corrector can calculate the appropriate action to take to transition to the intended square. An error corrector serves much the same function as a planner in the checkerboard example. However, in environments where position is continuous (rather than a discrete square), an error corrector can be used to ensure the agent reaches the intended position. Again, a model of the environment is required (so as to be able to predict the effects of actions) as is an explicit representation of the intended state. An error corrector is described in greater detail and used in Chapter 3.

Although there is a distinction between planning and error correction, the two mechanisms serve the same general purpose in this thesis: given sufficient computational and representational resources, they can generate a reasonable solution to a particular subtask or task. Thus, they are useful for selecting actions (or making more complicated movements) during early exposure to a task.

Reinforcement Learning

As described under functional roles of the basal ganglia (page 17), the BG are thought to learn how valuable each action is in each state through reward-related learning mechanisms similar to those used in Reinforcement Learning (RL, Sutton and Barto 1998). An action is selected by comparing the relative values of each action from the current state. To behave "greedily," the agent selects the action corresponding to the highest value. To learn the values of other actions, the agent selects another action once in a while. The latter process is termed exploration and is essential for finding the best action for a task.

In terms of the generic task described above, the value of an action from a particular state is the sum of expected future rewards received when taking that action from that state and selecting the best possible actions — based on its current value estimates — after that. In the checkerboard example, if the current position of the agent was one square south of the goal, then the value of taking action north from that square would be $-1 + 100 = 99$. Similarly, if the agent was two squares south of the goal, the value of action north is 98.

The value of each action is learned through experience — trying out different actions, visiting different states, and observing the consequences. If the consequences are greater than expected (the current value), the value is increased; if less, the value is decreased. Thus, the values represent the predicted consequences. In Chapter 3, the consequence of an action is calculated by summing the rewards for each action taken for a task. If $Q(s, a)$ represents the value of action a taken from state s, then $Q(s, a)$ is updated by $\sum r_i$, where r_i is the reward received at step i. This form of update is often referred to as *Monte Carlo* update. In Chapters 4 and 5, the consequence of an action is calculated by the immediate reward received and value of the next action taken: $Q(s, a)$ is updated by $r + Q(s', a')$, where r is the immediate reward received and $Q(s', a')$ is the value of the next action taken from the next state visited. This general form of update, in which values are updated by other values, is referred to as *bootstrapping*. Sutton and Barto (1998) describe other types of algorithms for learning values. I describe the algorithms I use in more detail in their respective chapters.

Unlike the planning and error correction mechanisms described in the previous section, actions are selected through the relatively cheap (computationally) process of comparing the values of actions. In addition, since there is no planning or calculation of best action based on the environment or position of the goal, a model of the environment is not needed. Thus, the computational and representational resources required are less than those for planning and error correction.

On the other hand, RL mechanisms are not as useful for early learning of a task. The values of actions are typically initialized to some equal number or random numbers; thus, during early exposure to a task, even greedy actions result in poor performance. Exploration and practice — solving the task many times — is required to properly learn the values. Because proficiency in motor skill execution requires practice, there is an opportunity for the learning and control mechanisms of the BG to participate motor skill acquisition.

Transfer of control

As discussed in the previous two sections, planning and error correction mechanisms are useful during early exposure to a task, but RL mechanisms have advantages after experience is gained. Experimental evidence cited in the earlier section, *BG in Motor Skill Acquisition*, show that as the animal repeatedly accomplishes the task, the role of the BG is more prominent. In addition, some imaging data suggest that the role of cortical planning areas decrease as the animal repeatedly accomplishes the task. Other studies show how important the BG is for the learning and execution of motor skills. These data combined suggest that control mechanisms associated with cortical planning areas dominate control early in learning a motor skill, but mechanisms associated with the BG dominate control later.

The learning mechanisms of the BG require experience; cortical planning mechanisms provide experience early in learning in the form of reasonable behavior. Thus, after the values of actions are learned to some degree, the BG can take over control in a more efficient manner than cortical planning mechanisms. In addition, as discussed under the section *Exploration* (page 14), the BG can try out different actions or movements and use reward-related learning to evaluate them. Such exploration can lead to behavior indicative of motor skills.

The transfer just described corresponds to the "three stages of skill learning" theory of Fitts and Posner (1967). The first, *cognitive*, stage occurs early in learning and corresponds to the subject merely trying to ascertain the goal of the task. In my formulation, this information is given to a large degree and cortical planning mechanisms, associated with cognitive aspects of behavior, control behavior. During the *associative* stage, performance (e.g., speed of movements) increases due to adjustments made to the initial solution, including finding better motor commands and sensory information. In addition, explicit representations of the task exert less influence, suggesting that the subject is less "aware" of the intricacies of the movements. Most of this thesis focuses on changes in behavior that could be described as part of the associative phase. Finally, the *autonomous* stage is the result of much practice and describes behavior executed with very little conscious control. The results presented in Chapter 4 of my thesis can be applied to this phase.

CHAPTER 3

COARTICULATION

3.1 Redundancy

In the context of motor control, most animals, including humans, work with a redundant system. There are many joint configurations that enable one's hand to grasp an object, there are many patterns of muscle activity that will produce the same joint movements, there are many ways to accomplish a task, and so forth. For most tasks, there are many more degrees of freedom (DOFs) to be controlled than are required. Our central nervous system must solve an ill-posed problem in that there is rarely a unique solution.

A single task

While the "degrees of freedom problem" (Bernstein, 1967) makes control difficult, it also affords us the opportunity to maximize secondary objectives when accomplishing a task. In Figure 3.1A, the task is to move from the rectangle labeled *Start* to the ellipse in one movement, represented by an arrow. The fastest movement is represented by the arrow labeled *Fastest*. However, what if there were noise in the system so that the actual outcome of the movement is drawn from a probability distribution centered around the expected outcome? In this case, the solution labeled *Most Conservative* might be selected as deviations from its intended outcome would more likely result in the task being accomplished than that of other solutions.

Maximization of secondary objectives may lead to a unique solution. Stereotypical patterns at all levels of control (i.e., neural activity, muscle activity, joint movements, etc.), both within and across subjects, leads many researchers to believe that secondary objectives are maximized when we accomplish a task. Examples are numerous (cf. Flash and Sejnowski 2001; Engelbrecht 2001) and include minimization of muscular effort (Fagg et al., 2002; Pedotti et al., 1978; Collins, 1995; Mussa-Ivaldi et al., 1988; Bizzi et al., 1991), minimization of derivatives of control and kinematic variables (Flash and Hogan, 1985; Uno et al., 1989), and minimization of movement variability (Bays and Wolpert, 2007; Todorov, 2002; Harris and Wolpert, 1998). By comparing the solution generated by a model that maximizes one or more objectives with that of human and animal behavior, we can ascertain how prominent the objectives are. Each study referenced had successes and limitations.

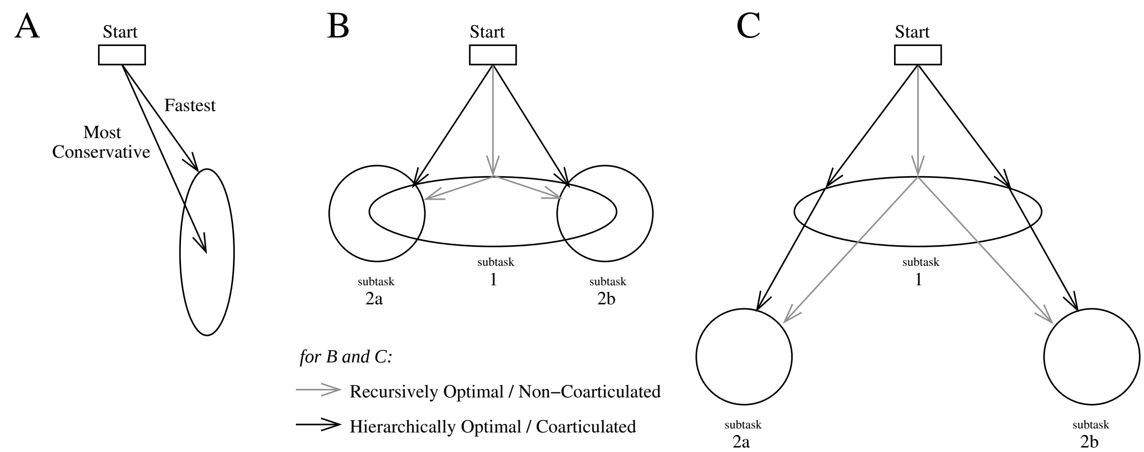

Figure 3.1. Schematic illustrating movement strategies in a two-dimensional space. Each ellipse and circle represents the set of possible movements that satisfy the primary objective of a subtask. Movements are represented by arrows.

A sequence of tasks

When there is an ordered sequence of tasks to be accomplished, additional objectives may be based on the entire sequence rather than just each individual task. (To minimize ambiguity, henceforth I will refer to each individual task as a *subtask* and the entire sequence as the *overall task*.) The influence of the overall task is seen by observing how each subtask is accomplished. Figures 3.1B and 3.1C each depict two sequences: 1) move from the *Start* rectangle to the ellipse labeled *subtask 1*, and from there to the circle labeled *subtask 2a*, and 2) from *Start* to *subtask 1* to *subtask 2b*. In Figure 3.1B, the solution set that achieves subtask 1 intersects with that for subtasks 2a and 2b. For simplicity, we will only consider minimizing path length in Figure 3.1. The grey arrows depict solutions that are *recursively optimal* (Dietterich, 2000) in that only each subtask, not the overall task, is considered. The black arrows depict *hierarchically optimal* solutions in that the overall task is considered.

Two differences between the recursively optimal and hierarchically optimal solutions are readily apparent: 1) in the recursively optimal solution, the solution to subtask 1 is the same for both overall tasks, whereas they're different in the hierarchically optimal solution; and 2) with the hierarchically optimal solution, the solution to subtask 1 is actually suboptimal when taking only subtask 1 into account. However, the sacrifice made in accomplishing subtask 1 pays off when accomplishing the second subtask. When humans and animals accomplish a sequence of subtasks, skilled behavior exhibits characteristics of hierarchical optimality, suggesting that evaluative feedback from the overall task is used in selecting movements. In the motor control literature, such behavior is described as *coarticulation*; below I review some behavioral examples.

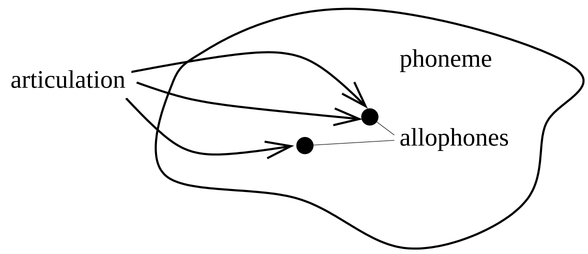

Figure 3.2. Schematic illustrating sound as two-dimensional points or sets of points. Allophones correspond to specific sounds and are shown as black circles. A phonemes corresponds to a set of sounds that serve some linguistic function and is shown as the amorphous shape. Articulations, which produce sounds, are shown as arrows.

3.2 Behavioral Examples of Coarticulation

The term *coarticulation* derives from studies of human speech production. Briefly, an *allophone* is a particular speech sound, a *phoneme* is a set of allophones that serve the same linguistic function (e.g., all allophones that signify an "n"), and *articulation* is the act of producing an allophone. The same phoneme, and in some cases the same allophone, can be produced through many different types of movements associated with speech organs (such as the tongue, lips, throat, and lungs); the organs producing the sound are called *articulators*. Figure 3.2 illustrates these concepts in a manner analogous to Figure 3.1.

The redundancy in phonemes and articulation leads to characteristics similar to those seen in hierarchical optimization: the particular allophone of a phoneme and how it is articulated depends on *context* — the preceding and subsequent phonemes to be articulated (Kent and Minifie, 1977; Abbs et al., 1984). For example, the "k" phoneme in "keep" and "cool" are different allophones, selected because of the subsequent phonemes of "ee" and "oo," respectively. In addition, because the "k" phoneme is articulated mostly by the throat, and the "ee" and "oo" phonemes are articulated mostly by shaping the lips and mouth, we activate both articulators at the same time for the different phonemes — coarticulation. Strictly speaking, coarticulation refers to the scenario seen in Figure 3.1B, where two tasks can be accomplished at once, and is a special case of phonetic influence, where phonemes and articulators are modified by context, including the scenarios in Figures 3.1B and 3.1C. Because of the similarities between the control of speech and the control of other sequential motor tasks, researchers have adopted the term coarticulation to describe both the modification of movement based on context (Figure 3.1C) and the special case of accomplishing two tasks with one movement (Figure 3.1B).

Coarticulation is seen at the levels of joint configuration and hand trajectory when tracking a known trajectory of targets in three-dimensional space (Breteler et al., 2003). Similar to the examples in Figure 3.1, the first target was the same but subsequent targets differed. Coarticulation was more pronounced with three targets than with two, suggesting that behavior shows more coarticulation effects when the task demands are greater. Similarly, as the number of DOFs with which to work increases (allowing for greater flexibility and presenting a more demanding control problem), coarticulation may be more pronounced. Jerde et al. (2003) examined the finger and hand movements of sign language users as they performed finger spelling tasks. Sign language exploits the many DOFs of the hand to create clearly distinguished hand postures. They found that, like movements used in speech production, how one finger-spells a letter depends greatly on context (the preceding and subsequent hand postures). Some types of behavior described as coarticulation may also be described as *prospective coding* or *anticipatory activity* in that the way a movement is executed may indicate what subsequent movements may be. For example, the way an object is grasped may indicate how one plans to use it (Johnson and Grafton, 2003; Cohen and Rosenbaum, 2004).

Coarticulation is also seen in how discrete effectors, such as fingers, are recruited. When a pianist plays an eight-note ascending scale, he plays the first three notes with his thumb, index, and middle fingers (in that order), and then crosses his thumb underneath the palm to play the fourth note with the thumb, using the index, middle, ring, and pinky fingers to play the remaining four notes. If the sequence was only four notes long, he likely would play the fourth note with his ring finger. Engel et al. (1997) describe in detail how a piano player plays a fixed sequence of keys differently depending on context; similar effects are seen in how violinists (Baader et al., 2005) and typists (Soechting and Flanders, 1992) recruit their fingers.

In the previous examples, coarticulation referred to how one accomplishes a subtask depending on context, analogous to Figure 3.1C. In some cases, two or more subtasks can be accomplished concurrently, as depicted in figure 3.1B. When learning to reach for an object with the intention of grasping it, a subject learns to open his hand while transporting it to the object. This act is referred to as *preshaping* or *prehension* (Hoff and Arbib, 1993; Jeannerod, 1981) and can be generalized to describe the behavior of some form of movement $i + 1$ occurring concurrently with movement i. A similar behavior is seen in bimanual coordination. Wiesendanger and Serrien (2001) review their studies in which a subject must open, and hold open, a drawer with one hand while reaching into it to manipulate a small object with the other hand. The subject learns to transport the object-manipulating hand along with the drawer-opening hand such that both hands reach the drawer at almost the same time. In these examples, the weak coupling of the degrees of freedom used in the two movements allow for them to be executed simultaneously.

3.3 Search Strategies

What strategies does the brain use to search for movements from the possibly infinite set of movements than can accomplish a task or sequence of subtasks? Below I discuss three theoretical studies that specifically investigate coarticulation.

Rosenbaum and colleagues (Rosenbaum et al., 1993, 1995, 1999) developed a series of models in which solutions are created from a linear combination of stored solutions, weighted by their error terms (based on a weighted sum of accuracy and specified secondary objectives). I refer to this set of models as the *Rosenbaum model*.

For a sequence of movements, Rosenbaum et al. (1995) suggest that one plans the entire sequence of movements explicitly before movement onset. In reaching for two sequential goals, their model finds the least cost configuration that reaches the second goal from the starting configuration, and then finds the configuration for the first goal that minimizes the cost it takes to move from the staring configuration and then to the second goal. The first goal is subordinate to the second goal. Rosenbaum et al. (1999) modified the model by creating new configurations based on a noisy version of a stored one deemed best for the task. Crucial to their model is the ability to store many candidate configurations, evaluate them off-line based on some error term, explicitly represent past and future configurations, and linearly combine them based on their errors for each goal. Although an overall task evaluation is not specified, such an evaluation could be easily integrated into their model.

Jordan and Rumelhart (Jordan, 1992; Jordan and Rumelhart, 1992; Jordan, 1990, 1988) used supervised learning neural network models to study the effects of secondary objectives on movement selection when hitting a sequence of goals. I refer to this set of models as the *Jordan model*. The error term includes a smoothness objective (Jordan, 1988), defined as the Euclidean distance between joint configuration at time t with the configuration at $t+1$. In addition, the model allows some configuration variables to be "flexible" depending on the current goal. Some variables were allowed to take on any value, be within a certain range of values, or be below or above a certain value for a particular subtask. The supervised learning algorithm exploits the added redundancy to specify the value of the variable so that it is positioned to best accomplish the task for which it is used. Inclusion of the smoothness objective and allowing the values of some variables to be flexible resulted in behavior described as coarticulation.

Guenther (1995) developed a model of speech production in which phonemes were represented as target regions in articulation space (analogous to joint configuration space). I refer to this model as the *Guenther model*. The transition from one phoneme to the next takes place along the shortest path. Because phonemes are sets of sounds, if the next phoneme does not require any change in a subset of articulatory variables (e.g., lip formation), that subset will not be changed. This leads to behavior described as coarticulation in the strictest sense. Another effect of using the shortest path is that the articulation of the current phoneme depends on the articulation of the preceding phoneme, an influence sometimes referred to as *carry-over coarticulation*. *Anticipatory coarticulation*, in which subsequent phonemes influence the current one, is produced by a planning process in which articulation of the current phoneme is restricted to coincide with the range of variables used in subsequent phonemes. Anticipatory coarticulation is produced by a different process than carry-over or strict coarticulation.

In all three models, movements are modified through the direct influence of surrounding movements — they are explicitly "blended" together, an understandable strategy considering the behavioral characteristics of coarticulation. I use the term *directed search* to refer to searching for better movements using knowledge of other movements and subtasks. However, the neural mechanisms that allow for blending have not been discussed in the models mentioned above. Do we explicitly blend movements together, or can a more general form of search contribute to the observed behavior?

A more general form of search and evaluation may have advantages over blending; blending assumes that the most similar movements result in the best solutions. Such a strategy is understandable when speed is important. However, Jerde et al. (2003) bring up an interesting point in their analysis of finger spelling: not all motor tasks have the same goals. In the case of sign language, speed is not the only, or even the primary, objective. The letters and concepts indicated by the hand and finger configurations must be *distinguishable*. Thus, if similar hand and finger configurations indicate two different letters, signing those letters in similar ways to enable a smooth transition is not desirable. Rather, one would want to augment the difference between the two letters and thus choose dissimilar hand and finger postures. These effects were seen in Jerde et al. (2003).

Action Modification

Because the neural mechanisms of blending are not understood, and because a more general search may have advantages, I suggest that the behavioral characteristics of coarticulation can arise from *undirected search*, in which search is not confined to directions dictated by an explicit representation of other movements or subtasks. In the rest of this chapter I use the term *action* to refer to *movement* to keep terminology within this thesis harmonious.

To implement undirected search, actions can be modified by varying the current best action in any direction (as opposed to a subset of directions, as would be the case in directed search) and, if the result is better, setting that as the best action. I refer to this process as *Action Modification*. Also, I define here a reward function as a function over all possible actions such that its value is the reward for executing that action. An example of a simple reward function is illustrated in Figure 3.3A, where each point represents a particular action, such as specification of an arm configuration, and height corresponds to reward. Action Modification may lead to an optimal solution if the reward function is of a fairly simple form, such as the convex shape illustrated in Figure 3.3A. The search for the best action is analogous to "climbing the hill" to reach the peak of the reward function. Figure 3.3B is a bird's eye view of a part of the hill and illustrates the concept of Action Modification. Thick arrows indicate modifications that result in better actions, while thin arrows indicate those that do not. Knowledge of the shape of the reward function would enable a much more efficient search in that all modifications could be directed toward the peak, as would be used in directed search. However, in many cases, such knowledge is not available. The search strategies discussed in the previous section substitute another function (e.g., the inverse of an error function based on some objective) for the true reward function. While directed search climbs the substitute function, it may not climb the true reward function. Because undirected search is not confined to search in directions dictated by a substitute function, and actions are evaluated based on reward, it may be able to find the peak of the true reward function. In addition, because motor skill acquisition requires repeatedly accomplishing a task (i.e., practicing), the opportunity

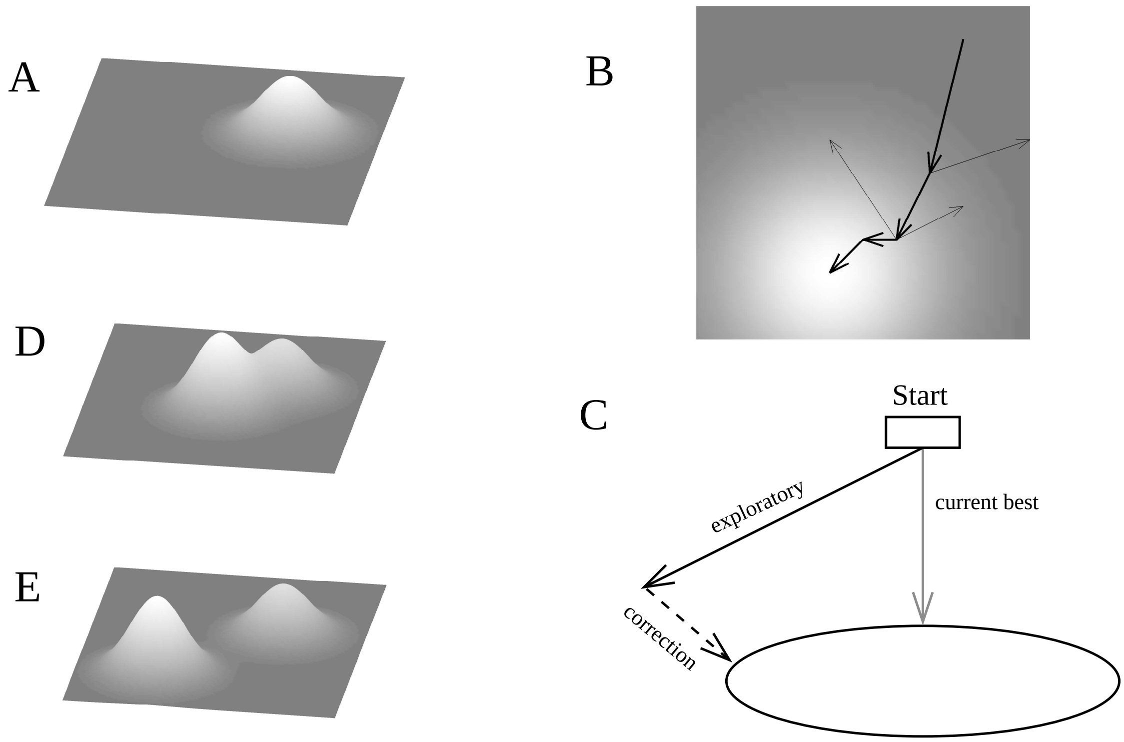

Figure 3.3. A,B,D, and E: examples reward functions. Each point represents an action, such as specification of arm configuration, and the higher the function / lighter the color, the greater the reward received for taking that action. B: Schematic of Action Modification. Thick arrows represent modifications (change from the action represented by the base of the arrow to the action represented by the tip) that result in greater rewards, thin arrows represents modifications that do not. C: follows same conventions as those in Figure 3.1.

exists for undirected search to participate in finding better actions. A model described in Rosenstein and Barto (2001) and Rosenstein (2003) uses a search process similar to that described in this section to show that unexpected solutions can be found by undirected search.

Ensuring subtask accomplishment

When searching for actions that best accomplish the overall task, we do not have to consider actions that do not accomplish the required subtasks. Some non-biological models of motor control, computer science, and robotics explicitly restrict search to solutions that do not interfere with the primary objective of the task. Examples include restricting exploration or modification to directions that bring the system closer to achieving the primary objective (Perkins and Barto, 2001; Perkins, 2002; Torres and Zipser, 2004), maximizing secondary objectives only if they don't interfere with the primary objective (Coelho and Grupen, 1997; Rohanimanesh et al., 2004), and explicitly projecting the actions for secondary objectives into the null space of the primary objective (Platt et al., 2002). (The term *null space* has a strict mathematical definition, but for our purposes, we can think of a null space of a goal as the space of all movements that accomplish that goal.) The exploration theory of Mink (1996) can be interpreted as the BG exploring only within the space of motor commands that achieve the goal. However, Mink does not explicitly suggest this and there is no non-speculative reason to think this restriction exists. While restrictive exploration can be useful, non-restrictive exploration cannot be ruled out. With the aid of the cortical and cerebellar functions (described on page 10), a mechanism does exist to ensure that the goal of the subtask is accomplished even if the exploration is not explicitly restricted to the null space.

In the context of Action Modification, the cortex and cerebellum can act as an error corrector, an abstract representation of which finds the shortest path from the current point to the goal point. Consider Figure 3.3C: the grey arrow represents the current best action, and the black arrow represents an exploratory action (such as those illustrated in Figure 3.3B). Since the exploratory action did not accomplish the goal, an error corrector is recruited to find a path toward the goal, illustrated by the dashed arrow. A set of models by Fagg and colleagues (Fagg et al., 1997a,b, 1998), described in Chapter 6, uses such an error correction mechanism. The combination of undirected search and error correction, in effect, searches the null space of a task.

Action Selection

If the reward function is complicated, action modification alone may not be sufficient to find the highest reward. Such is the case when the reward function is "bumpy," i.e., there are many points at which the second derivative is zero. Figure 3.3D illustrates an example of such a case. If variability was high enough, Action Modification alone may be enough to discover the highest peak in the reward function. The scenario in Figure 3.3E, though, presents an even more complicated reward function such that the variability in Action Modification would have to be very high to discover the optimal solution. Such a high variability may result in many poor actions.

A complicated reward function is readily apparent when multiple discrete effectors can accomplish a task, e.g., when the task calls for the selection of one of several fingers or one of two hands. The use of each effector can be modified to make it better, but it is unlikely that modification of one effector would smoothly lead to the use of another. Rather than rely solely on the modification of one action, several actions can be kept track of, e.g., the use of each effector. Each action undergoes its own modification and the learning agent selects from them. Thus, search occurs on more than one level. It is likely that for some tasks, one effector may have two radically different locally optimal solutions so that a multi-level action search is necessary to find the globally optimal solution. For the purpose of this chapter, though, I define Action Selection as the selection of one effector from a set.

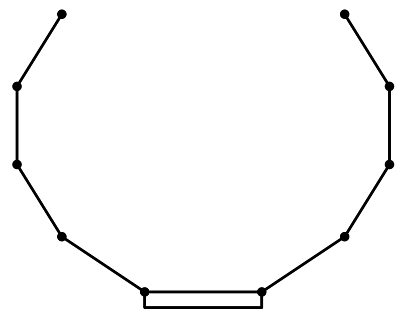

Figure 3.4. Schematic of “robot” used in simulations. Robot is planar and has a total of 10 DOFs. The base is mobile and can move vertically and horizontally. Each arm has four rotational joints, and no joint limits are imposed. Each arm link is one unit in length, and the base is a rectangular box which is one unit wide and 0.2 units high. This design is inspired by designs used in Jordan (1992); Jordan and Rumelhart (1992); Jordan (1990, 1988).

3.4 Hypotheses

In light of preceding discussion, I present two hypotheses

1. Undirected search, evaluated by overall task performance, can account for learned behavior described as coarticulation. Undirected search contrasts with previous theories, in which actions are modified through the influence of other actions or a planning process (directed search). In undirected search, the influence of other subtasks is felt by using only overall task performance as evaluation, a form of hierarchical optimization.

2. For some tasks, a multi-level exploration strategy, Action Modification and Action Selection, finds better solutions than either alone.

I investigate these hypotheses by implementing Action Modification and Action Selection with a simulated “robot.” The model is described next.

3.5 Model

I investigate how behavior described as coarticulation can occur with a simulated redundant system: a planar kinematic “robot” with two 4 DOF arms attached to a 2 DOF base (Figure 3.4). The arm DOFs are rotational joints and the base DOFs are orthogonal translational joints. The overall task for the robot is to hit a series of spatial goals (referred to as subtasks) with one of the two end-effectors (henceforth referred to as “hands” for brevity). Completion of the overall task constitutes a trial, and the locations and order of the goals to be hit are known. Each goal is a circle of fixed radius (0.1) centered on its defined location, which is represented as an (x, y) pair of coordinates. The goal is referred to as g and its location is $\mathbf{x}_g$. The robot design was chosen for the following reasons:

Redundancy: The redundancy allows it to display characteristics of coarticulation, both in how an arm is used and which arm is used to hit a goal.

Simplicity: The simplicity of the robot design allows us to avoid distractions which may accompany a more complicated system.

Similarity to animals: While the robot design is simple, we can draw analogies between its design and behavior to that of animals.

Similarity to other models: This design is based on that of the Jordan model.

Movement

The 10 DOFs of the robot are represented by its joint configuration, $\mathbf{q}$, a 10-element vector of which each element specifies the value of the corresponding joint. The robot's starting joint configuration is $\mathbf{q}_0$; it must choose a new joint configuration, $\mathbf{q}_g$, to which to move to hit goal g. Thus, for a sequence of three goals, the robot must take three actions to move from $\mathbf{q}_0 \rightarrow \mathbf{q}_1 \rightarrow \mathbf{q}_2 \rightarrow \mathbf{q}_3$. The robot moves from one configuration to the next in a step-wise manner:

$$\mathbf{q} \leftarrow \mathbf{q} + m\frac{\mathbf{q}_g - \mathbf{q}}{||\mathbf{q}_g - \mathbf{q}||},$$

where $||\cdot||$ refers to the Euclidean norm and m is a scalar, set to 0.01. The robot moves in the direction of $(\mathbf{q}_g - \mathbf{q})$ with a magnitude of m at each step of movement. For the sake of convenience, the number of steps the robot takes to make a movement is analogous to the amount of time the robot takes to make that movement, and the robot moves with a "constant velocity." For each movement step, a reward of -1 is incurred.

The learning agent chooses a hand by selecting an action, a, and target joint configuration ($\mathbf{q}_g$) for goal g. Movement begins and continues until one of two conditions are met:

1. the extrinsic position of the chosen hand ($\mathbf{x}^a$), calculated at each step via forward kinematics of $\mathbf{q}$, reaches its expected extrinsic position ($E[\mathbf{x}^a]$), calculated via forward kinematics of $\mathbf{q}_g$:

   $$||\mathbf{x}^a - E[\mathbf{x}^a]|| \leq \theta^a,$$

   where θ^a is the level of accuracy it must achieve and is set to 0.1 in the following simulations.

For a chosen hand (a),	
$\mathbf{x} \leftarrow F(\mathbf{q})$	calculate $\mathbf{x}$
while $\|\|\mathbf{x} - \mathbf{x}_g\|\| > \theta^g$	compare $\mathbf{x}$ with goal position
$\quad \mathbf{q} \leftarrow \mathbf{q} + \alpha \mathbf{J}^T(\mathbf{x}_g - \mathbf{x})$	modify $\mathbf{q}$ to decrease $\|\|\mathbf{x} - \mathbf{x}_g\|\|$
$\quad \mathbf{x} \leftarrow F(\mathbf{q})$	calculate $\mathbf{x}$ again, with new $\mathbf{q}$

Table 3.1. $\mathbf{A}(\mathbf{q}, a, \mathbf{x}_g)$: Iterative process that finds $\mathbf{q}$ such that the extrinsic position of the chosen hand ($\mathbf{x}$) is at the goal position ($\mathbf{x}_g$). This process affects the joint variables of the base and chosen arm; those of the other arm are not changed. $F(\mathbf{q})$ returns $\mathbf{x}$ via forward kinematics, $\mathbf{J}$ refers to the Jacobian matrix, the superscript T refers to the transpose, and α is a small positive number (set to 0.05 in these simulations).

2. the extrinsic position of the chosen hand reaches the current goal:

$$||\mathbf{x}^a - \mathbf{x}_g|| \leq \theta^g,$$

   where θ^g is the level of accuracy the agent must achieve, i.e., the radius of the goal, and is also set to 0.1.

The movement process described in this section is referred to as $\mathbf{Move}(\mathbf{q}, \mathbf{q}_g, a, \mathbf{x}_g)$. The sum of rewards received when moving from $\mathbf{q}$ to $\mathbf{q}_g$ is denoted r_{move}.

Planner

The *Planner*, $\mathbf{A}$, provides a mechanism by which a subtask can be accomplished from any joint configuration, but it does not take into account the overall task. As implemented in these simulations, $\mathbf{A}$ calculates a joint configuration based on the positional error between the chosen hand and the current goal. The error in extrinsic space is converted to a target joint configuration via an iterative process using a linear approximation (the Jacobian matrix, cf. Craig 2004), summarized in Table 3.1. The solution found by $\mathbf{A}$ is akin to taking the shortest route to a goal in extrinsic space, a form of recursive optimality. Although the transformation is non-linear, using a linear approximation in small increments allows us to find a target joint configuration such that $||\mathbf{x}^a - \mathbf{x}_g|| \leq \theta^g$. This iterative process is denoted as $\mathbf{A}(\mathbf{q}, a, \mathbf{x}_g)$ and is used to find an initial set of joint configurations and to make corrections if necessary.

Value-based controller

The *Value-based controller*, $\mathbf{B}$, executes three processes: *Action Selection*, *Action Modification*, and *Updates* the actions accordingly. These processes are described below and are summarized in Table 3.2.

Action Selection

Action Selection is analogous to discrete decision-making used in many Reinforcement Learning tasks (e.g., page 24). A look-up table is kept which specifies how valuable each action is in each state. The table is referred to as the Q-table and is $\|S\| \times \|A\|$, where S is the set of all states and A is the set of all actions. In these simulations, there are only two discrete actions: use the left hand or use the right hand. State can be as simple as just the current goal, but for tasks which use both hands, I use $s = (g, a_{g-1})$, where g is the goal to be hit and a_{g-1} is the action (hand) used for the previous goal. This representation is useful as the configuration of the robot when it uses its left hand to hit a goal will be very different than its configuration when it uses its right hand. Including the hand used to hit the previous goal captures much of the distinction without having to represent the actual configuration. Also, it only increases the state space from $\|G\|$ to $\|A\|\|G\| - (\|A\| - 1)$ (the state for the first goal does not include a_{g-1}). For the experiments I run, no further detail in state representation is required.

Each element in the Q-table, $Q(s, a)$, is the highest reward associated with selecting action a from state s. I adopt the typical ϵ-greedy exploration, in which the selected action is the $\text{argmax}_a Q(s, a)$ for $(1-\epsilon)$ proportion of the time, and a random action ϵ proportion of the time. ($0 \leq \epsilon \leq 1$ and is small. I use $\epsilon = 0.2$ here.)

Action Modification

Along with the Q-table is an $\|S\| \times \|A\|$ *configuration* table that stores $\mathbf{q}^*(s, a)$, the current best configuration for each state and action. When action a is chosen from state s, the robot uses a modified form of $\mathbf{q}^*(s, a)$ to attempt to hit the goal. Actions are modified according to the general scheme illustrated in Figure 3.3C — noise is added to the variables of the chosen configuration: $\tilde{\mathbf{q}} = \mathbf{q}^*(s, a) + \eta_\sigma$, where η_σ is a vector where each element is randomly chosen from a zero-mean Gaussian distribution with standard deviation of $\sigma = 0.05$ in these simulations. The robot moves from $\mathbf{q}$ toward $\tilde{\mathbf{q}}$ via $\mathbf{Move}(\mathbf{q}, \tilde{\mathbf{q}}, a, \mathbf{x}_g)$. If, upon completion of movement, $\mathbf{q}$ does not result in the selected hand hitting the goal, a corrective movement is made via $\mathbf{A}(\mathbf{q}, a, \mathbf{x}_g)$. The final configuration is denoted $\mathbf{q}'(s, a)$.

Update

The sum of the rewards incurred for making all movements necessary to accomplish the entire task, r, is recorded. For each state-action pair, (s, a), visited, if $r > Q(s, a)$, then $Q(s, a) \leftarrow r$ and $\mathbf{q}^*(s, a) \leftarrow \mathbf{q}'(s, a)$. Thus, actions are modified in a way best for the overall task, and the inclusion of $\mathbf{A}$ constrains search to movements that accomplish the subtasks.

3.6 Action Modification

The simulations in this section test the first hypothesis, which focuses on Action Modification alone. The robot must hit a sequence of three goals with just its right

$\mathbf{q} \leftarrow \mathbf{q}_0$	set the initial values of $\mathbf{q}, r$, and a
$r = 0$	
$a \leftarrow \emptyset$	
For $g = 1, ..., \|\|G\|\|$	
$a_{g-1} \leftarrow a$	
$s \leftarrow (g, a_{g-1})$	determine the state
$a \leftarrow \text{argmax}_a Q(s,a)$ ϵ-greedily	Action Selection
$\widetilde{\mathbf{q}} \leftarrow \mathbf{q}^*(s,a) + \eta_\sigma$	Action Modification
$\mathbf{q} \leftarrow \mathbf{Move}(\mathbf{q}, \widetilde{\mathbf{q}}, a, \mathbf{x}_g)$	Move
$r \leftarrow r + r_{move}$	record reward of movement
if $\|\|\mathbf{x}_a - \mathbf{x}_g\|\| > \theta^g$	Make correction if necessary
$\mathbf{q}^c \leftarrow \mathbf{A}(\mathbf{q}, a, \mathbf{x}_g)$	
$\mathbf{q} \leftarrow \mathbf{Move}(\mathbf{q}, \mathbf{q}^c, a, \mathbf{x}_g)$	
$r \leftarrow r + r_{move}$	
$\mathbf{q}'(s,a) \leftarrow \mathbf{q}$	record final configuration
For each (s,a) visited	Update
if $r > Q(s,a)$	
$Q(s,a) \leftarrow r$	
$\mathbf{q}^*(s,a) \leftarrow \mathbf{q}'(s,a)$	

Table 3.2. Summary of $\mathbf{B}$, the *Value-based* controller. Symbols used are defined in the text. In addition, $\mathbf{q}^c$ denotes a joint configuration found by $\mathbf{A}(\mathbf{q}, a, \mathbf{x}_g)$.

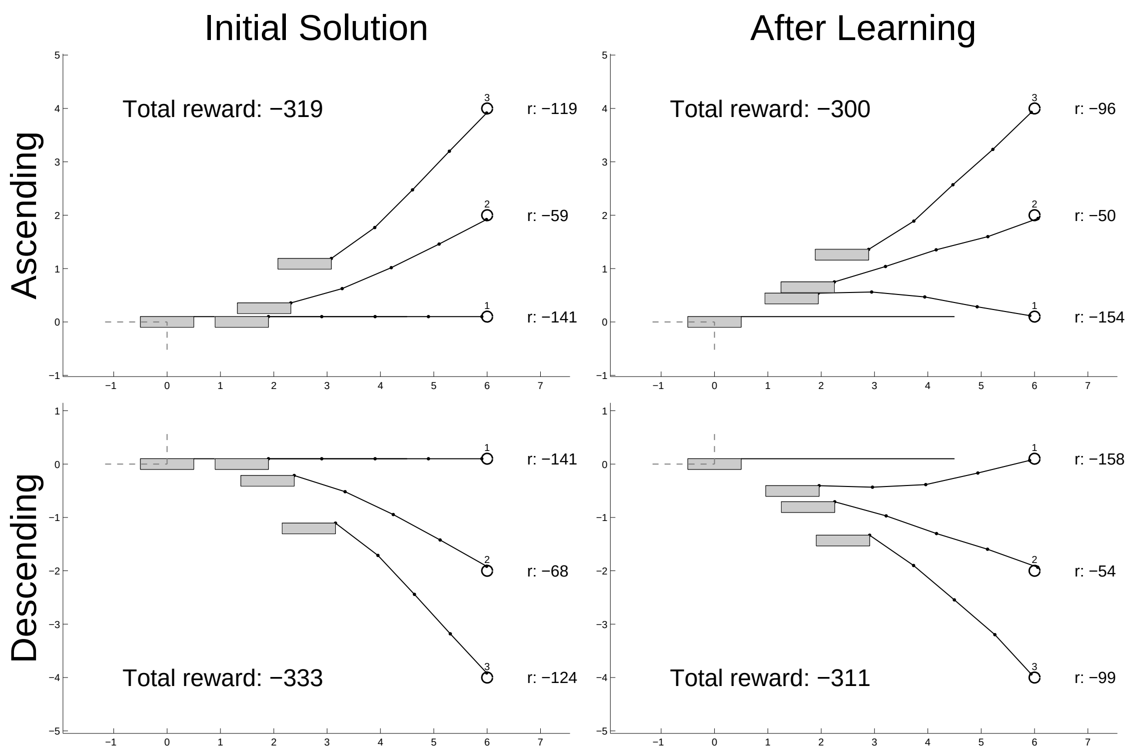

Figure 3.5. Illustration of joint configurations used to hit each goal. The order in which each goal is to be hit is indicated by a number above the goal. The reward for each movement is indicated to the right of each goal. The dashed lines meet at the point $(0, 0)$.

hand (there is no Action Selection). The starting configuration ($\mathbf{q}_0$) of the robot has its base centered at $(0, 0)$ and its right arm extended toward the right. The sequence of three goals are aligned vertically. In one task, the goals are *ascending*: $(6, 0.1)$, $(6, 2)$, and $(6, 4)$, in that order. In another task, the goals are *descending*: $(6, 0.1)$, $(6, -2)$, and $(6, -4)$. The position of the first goal, $(6, 0.1)$, is the same for both tasks.

For both sequences of goals (ascending and descending), the *Planner*, $\mathbf{A}$, was used to find the initial set of joint configurations to satisfy the overall task. The configurations are plotted on the left graphs of Figure 3.5, where the top graphs plot the solutions to the ascending task and the bottom graphs plot the solutions to the descending task. For each goal, g, the reward incurred for moving from $\mathbf{q}_{g-1}$ to $\mathbf{q}_g$ is noted; the sum of these is the total reward. The right graphs of Figure 3.5 show the set of joint configurations for the two tasks after Action Modification (for 5000 trials). The results shown are taken from one sample run; all other runs exhibited very similar results (not shown).

The rewards for each movement and for the overall task are indicated in Figure 3.5. For both tasks, Action Modification yields a better set of joint configurations than $\mathbf{A}$ alone did. The learned strategy used a $\mathbf{q}_1$ that was suboptimal in isolation but was better for the overall task. In addition, although the first goal was the same for both tasks, the joint configuration the robot used to hit the first goal differed between tasks — how the subtask was accomplished depended on context. Thus, the robot's behavior displayed characteristics of coarticulation, supporting hypothesis 1.

Comparison with other strategies

In this section, $\mathbf{q}_g^0$ denotes the configuration used to hit goal g as specified by the initial solution, and $\mathbf{q}_g$ denotes that as specified by the learned solution. The learned $\mathbf{q}_1$ could be interpreted, on a qualitative level, as a "blending" of the initial solutions: $\mathbf{q}_1^0$, $\mathbf{q}_2^0$, and $\mathbf{q}_3^0$. Such behavior results from the specification of the reward signal to be the negative of the number of times steps each movement took.

In the Rosenbaum, Jordan, and Guenther models, configurations used for one goal were modified based on representations of configurations used for other goals – blending was explicit. In contrast, my model does not explicitly impose such guidance. Rather, similar effects are the result of hierarchical optimization — using an evaluative signal based only on the overall task. In addition, the only constraint my model uses to modify movements is that each subtask is accomplished; the other three models limit search to directions toward configurations used for other subtasks. The lack of limitations used in my model may allow it to develop solutions that cannot be specified by explicit blending. In the next few subsections, I compare strategies developed by my model with strategies as suggested by the Rosenbaum, Jordan, and Guenther models.

Solutions as non-negative linear combinations of past solutions

When given a set of initial solutions for each goal, $\mathbf{q}_g^0$, blending could be interpreted as modifying each $\mathbf{q}_g$ toward some additive combination of every $\mathbf{q}_g^0$. In other words, blending suggests that the learned solution for goal g, $\mathbf{q}_g$, is a non-negative linear combination of all initial solutions:

$$\mathbf{q}_g = \sum_{g \in G} c_g \mathbf{q}_g^0,$$

such that each $c_g \geq 0$. To determine if my model follows such a strategy, $\sum_{g \in G} c_g \mathbf{q}_g^0$ was fit to each $\mathbf{q}_g$ for the ascending task (Figure 3.5, top right) as learned by my model. Coefficients were found by minimizing the following error function under the constraint that each coefficient is ≥ 0:

$$\left\| \mathbf{q}_g - \sum_{g \in G} c_g \mathbf{q}_g^0 \right\|^2 .$$

The fitting procedure ran for 300,000 iterations, or until the coefficients ceased to change at all, with a step-size of 0.001. Best fit results are displayed in Figure 3.6. For clarity, results are separated by goal. $\mathbf{q}_g^0$ for each goal is displayed in each graph in light grey; the best fit $\mathbf{q}_g$ is displayed in black with markers at each arm joint; the

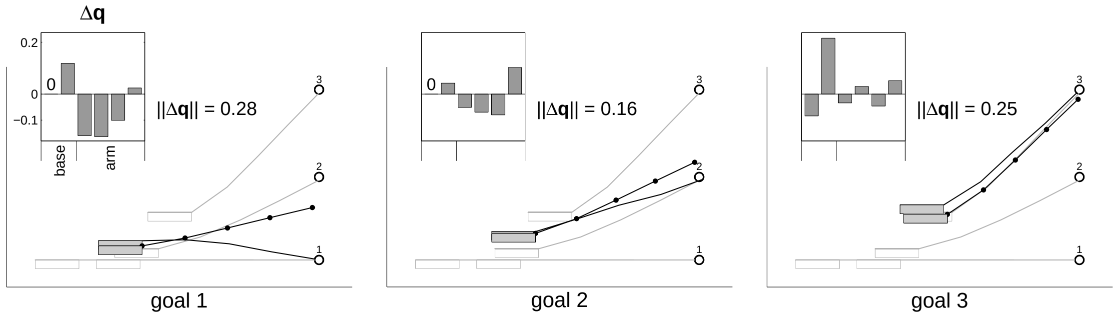

Figure 3.6. Illustration of the joint configurations used to accomplish the Ascending task. For each graph, the initial solution is drawn in light grey, the learned solution (Figure 3.5, top right) is drawn in black without markers at the joints, and the best fit solution (see text) is drawn in black with markers at the joints. The bar charts show the difference ($\Delta\mathbf{q}$) between the learned solution and the best fit configuration by joint variable. "0" indicates no difference. The four arm joint variables start proximally (shoulder) and end distally.

$\mathbf{q}_g$ as learned by my model is displayed without markers. Also plotted as bar graphs for each goal is $\Delta\mathbf{q}$, the element-by-element difference between the learned solution and the fitted solution; $\|\Delta\mathbf{q}\|$ is indicated as well.

Immediately apparent is the large difference between the learned solution for goal 1, $\mathbf{q}_1$, and the best fit configuration. This is because, with the initial solutions, the arm only curled upwards; however, in $\mathbf{q}_1$, the arm curled downward. The constraint that the coefficients be non-negative cannot capture the downward curl (the elements of $\mathbf{q}_1$ corresponding to the arm did not lie in the range of elements as specified by the initial solutions).

The Rosenbaum model modified earlier configurations based on later ones; such a modification would result in little or no change in $\mathbf{q}_3$. However, $\|\Delta\mathbf{q}\|$ for the first and third goals was almost double that for the second goal, indicating that the strategy developed by my model does not coincide with that of the Rosenbaum model.

The coefficients, c_g, were constrained to be non-negative so as to preserve the strategy of modifying one configuration toward another. Rosenbaum et al. (1999) used such a strategy in obstacle avoidance, in which movement was toward $(1-\kappa)\mathbf{q}_g+\kappa\mathbf{q}'$, where $\mathbf{q}'$ was a configuration that avoided an obstacle. κ varied over the course of the movement, depending on when the obstacle was expected to be encountered. However, it is conceivable that configurations may be modified toward an unrestricted linear combination of the initial solutions. Such a case yields best-fit configurations closely matching learned configurations: $\|\Delta\mathbf{q}\| = 0.067$, 0.127, and 0.110 for the three goals, respectively (best-fit configurations were visually very similar to learned configurations and thus were not plotted). Interestingly, $\|\Delta\mathbf{q}\|$ for the first goal was the lowest, further disagreeing with the strategy employed by Rosenbaum.

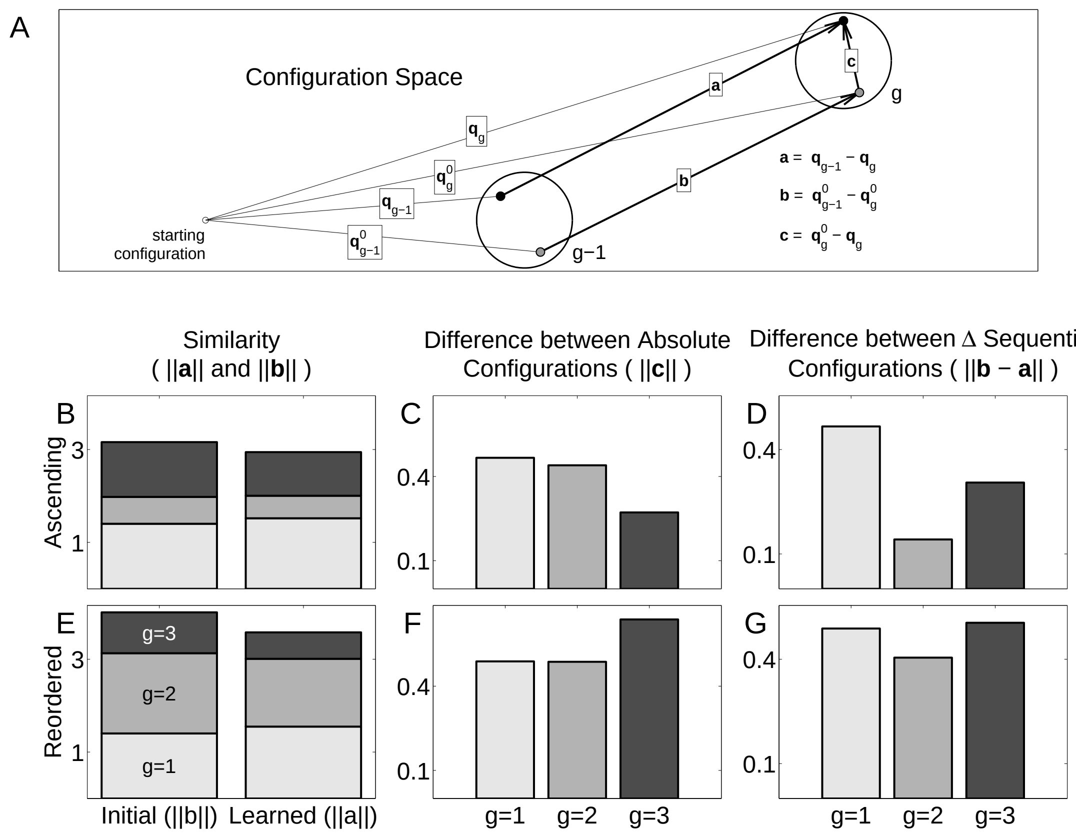

Figure 3.7. A: Representation of vectors in configuration space. For clarity, the vectors $\mathbf{q}_{g-1}^0$, $\mathbf{q}_g^0$, $\mathbf{q}_{g-1}$, and $\mathbf{q}_g$ do not have arrows. They start at the point labeled *starting configuration*. The circles represent subsets of joint configuration in which the end-effector hits the indicated goal. B through G: bar graphs indicating the Euclidean distance between vectors (see titles of graphs and text) for configurations corresponding to each goal. $g = 1$, light grey, $g = 2$, medium grey, $g = 3$, dark grey. A and E: stacked bar graphs.

Strategies represented by changes in joint configurations

Choosing configurations from the non-negative linear combination of initial solutions may impose overly stringent restrictions (though allowing negative coefficients expands the search space greatly). However, the general strategies suggested by the Rosenbaum, Jordan, and Guenther models can be employed without such restrictions. Rosenbaum modified earlier configurations according to later configurations, suggesting that learned solutions for goals early in the sequence would differ more from initial solutions than those for goals later in the sequence. In other words,

$$\left\|\mathbf{q}_{g-1}^0 - \mathbf{q}_{g-1}\right\| > \left\|\mathbf{q}_g^0 - \mathbf{q}_g\right\|$$

for all goals. Jordan imposed a secondary objective maximizing similarity between consecutive configurations. In other words, the objective was to minimize

$$\|\mathbf{q}_{g-1} - \mathbf{q}_g\|$$

for all goals. The Guenther model, through a planning process, followed a similar strategy. It is helpful to assign names to these difference vectors: for goal g,

$$\begin{aligned}\mathbf{a}_g = &\ \mathbf{q}_{g-1} - \mathbf{q}_g \\ \mathbf{b}_g = &\ \mathbf{q}^0_{g-1} - \mathbf{q}^0_g \\ \mathbf{c}_g = &\ \mathbf{q}^0_g - \mathbf{q}_g.\end{aligned}$$

These vectors are schematized in the Figure 3.7A. The large circles indicate the subsets of configuration space that accomplish goals $g-1$ (bottom center) and g (upper right).

The Jordan strategy is expressed as: $\sum_{g \in G} \|\mathbf{b}_g\| > \sum_{g \in G} \|\mathbf{a}_g\|$. The Rosenbaum strategy is expressed as: $\|\mathbf{c}_{g-1}\| > \|\mathbf{c}_g\|$. These quantities are plotted as bar graphs (Figures 3.7B and C) for each goal for the ascending task. The learned configurations are more similar, overall, than the configurations of the initial solution (Figure 3.7B), supporting Jordan's strategy. This is not surprising as movement in my model is a step-by-step transition from $\mathbf{q}_{g-1}$ to $\mathbf{q}_g$: the more similar $\mathbf{q}_{g-1}$ and $\mathbf{q}_g$ are, the less time the movement takes. Rosenbaum's strategy is also supported: $\|\mathbf{c}_g\|$ decreases from $g = 1$ to 2 to 3 (Figure 3.7C). The learned configuration for the last goal is more similar to the initial solution than that for the earlier goals.

"Movement direction" can be interpreted as the change in joint variables from one configuration to the next, represented by vectors $\mathbf{a}_g$ (for the learned solution) and $\mathbf{b}_g$ (for the initial solution). If movement direction for later goals changes less than movement direction for earlier goals, $\|\mathbf{b}_{g-1} - \mathbf{a}_{g-1}\| > \|\mathbf{b}_g - \mathbf{a}_g\|$. Figure 3.7D plots this quantity for each goal for the ascending task; it is lowest for goal 2, not goal 3.

The Jordan and Rosenbaum strategies are supported by my model for the ascending task. However, the three goals, which proceed from bottom to middle to top, possess a spatiotemporal pattern that may lend itself to such strategies. Such might not be the case if the temporal sequence of the goals was reordered to be bottom, top, middle. Figure 3.8 displays the initial (left) and learned (right) solutions for the *reordered* task, and the bottom row of bar graphs in Figure 3.7 refers to the reordered task. Again, and not surprisingly, the learned solution has more similarity than the initial solution (Figure 3.7E). However, the learned configuration for the last goal deviates from the initial solution more than that for the other two goals (Figure 3.7F), contradicting Rosenbaum's strategy. Also, the movement direction for the last goal is of greater magnitude than that of the other two goals (Figure 3.7G).

The analyses presented in this section shows that the undirected search used in my model generates configurations that follow a strategy different than that of the Rosenbaum model. On the other hand, the learned configurations from my model and the Jordan model follow similar strategies. Such a similarity may result from the similarity between the reward function I use and the the error function used in the Jordan model. Finally, the configurations found by my model cannot be generated by an additive blending of initial solutions.

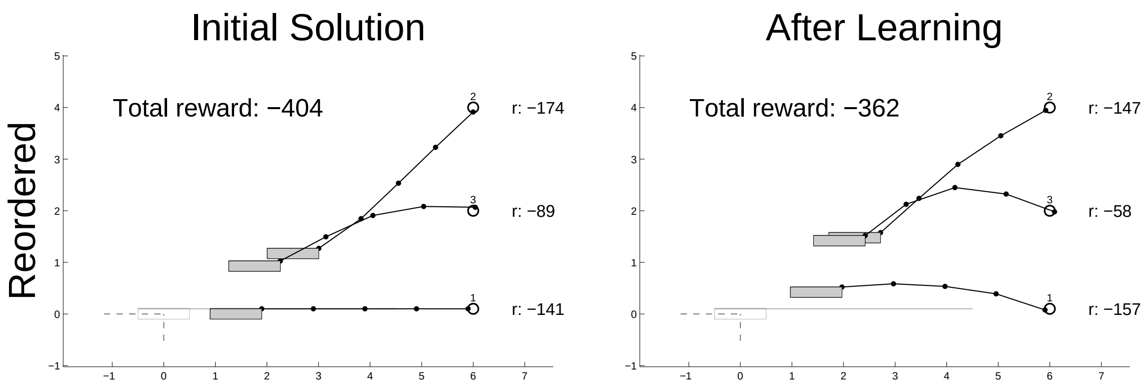

Figure 3.8. Follows the same conventions as in Figure 3.5.

3.7 Action Selection Experiments

In these experiments, the robot must hit a sequence of four goals using either hand for each goal. Because both hands are available, I refer to these tasks as *Biarticulate Tasks*. The availability of a second hand allows us to investigate three types of exploration, described briefly here and in detail in the following subsections.

Action Modification alone The values of all ten DOFs of the robot (including the base and both arms) are modified with zero-mean Gaussian noise. The hand closest to the next goal is chosen ϵ proportion of time. Action $\text{argmax}_a Q(s,a)$ is chosen the other $(1-\epsilon)$ proportion.

Action Modification and Action Selection This follows the scheme outlined in the section describing the *Value-based* controller, except that the arm not chosen for a goal undergoes no modification.

Leverage Redundant DOFs This does allow the arm not chosen for a goal to undergo modification.

Comparison of the three conditions illustrates the utility of the different types of exploration. Graphs displaying the robot's configurations in this section use the following convention: the right arm is drawn in black, the left arm is drawn in grey, and the arm of the hand chosen to hit a goal is indicated with markers on its four joints. Also, the starting configuration is all grey with no markers and the base is not colored in.

Action Modification alone

The starting configuration of the robot has its base centered at $(0,0)$ and both arms extended upward, tilted slightly medial so that the end-effectors occupy the same extrinsic location (to form a steeple-like pose, $\bigwedge$). The following sequence of four goals must be hit: $(1,3.7)$, $(5,3.7)$, $(6,2)$, and $(7,3.7)$ (*Biarticulate Task 1*). Either hand can be used for each goal. In addition, the base is not allowed to move vertically. The *Planner* was used to hit the series of goals with *just* its right hand and then with *just* its left hand. The right hand resulted in a more rewarding sequence of movements; thus, the initial solution used those configurations (top left of Figure 3.9).

Action Modification changes the joint configuration the robot uses to hit a goal. Thus, the values of all ten joints are changed; such modification results in a change in location for both hands. To allow each hand to be chosen with a non-zero probability, the hand *closest* to the next goal is chosen ϵ-proportion of the time (this specification was used in the Jordan model). (This does not necessarily guarantee that each hand will be used at some point during the simulations; rather, it guarantees that each hand will be used if the simulations ran forever.) The other $(1 - \epsilon)$ proportion of time, action $\text{argmax}_a Q(s, a)$ was chosen. Because the robot's hands occupy the same location at the starting configuration, each hand is chosen with equal probability ϵ-proportion of the time. The bottom four graphs of Figure 3.9 display selected learned solutions (from 20 different runs; in each run, the simulation ran for 10,000 trials). Because Action Modification changed the location of both hands, the robot did try different hands for each target once in a while. For some runs, the learned solution included using the left hand to hit one of the goals. Not surprisingly, in all cases Action Modification alone resulted in better solutions than the initial solution.

Although the left hand was chosen to hit each goal at different points during each run, the agent used just the right hand 9 runs out of 20 (lower right of Figure 3.9). On other runs, the agent did use the left hand for one of the goals. Starting from the most rewarding strategy to the least rewarding, the hand recruitment strategies are to use the left hand for $g = 3$ (middle left), $g = 1$ (middle right), and $g = 2$ (bottom left). In no runs did the agent use the left hand for the last goal.

The strategy of using just the right hand is worse than the other three strategies. However, Action Modification alone does not produce enough exploration to frequently place the left hand closer to the next goal. Because the Gaussian noise has no limits, the robot in my model can eventually stumble upon the best recruitment of hands. However, variability in real movements is not unlimited.

Is there a better hand recruitment strategy than the best found by Action Modification alone? Simply using the *Planner* to try out all possible sequences of hand recruitment reveals that using hands left, right, left, right (LRLR, plotted in Figure 3.9, top right) for the four goals results in a much higher reward than the strategy of just using the right hand. This suggests that explicitly trying out different hands for the goals — Action Selection — may find a better solution.

Action Modification and Action Selection

In this section, Action Modification alters just the joint variables of the base and the chosen arm; the joint variables of the other arm, referred to as *excess DOFs*, are not altered at all. I discuss the effects of leveraging the excess DOFs with Action Selection in the next section.

With Action Selection, in which a random action (hand) is chosen ϵ-proportion of the time, a better solution is found than with just Action Modification alone. Starting

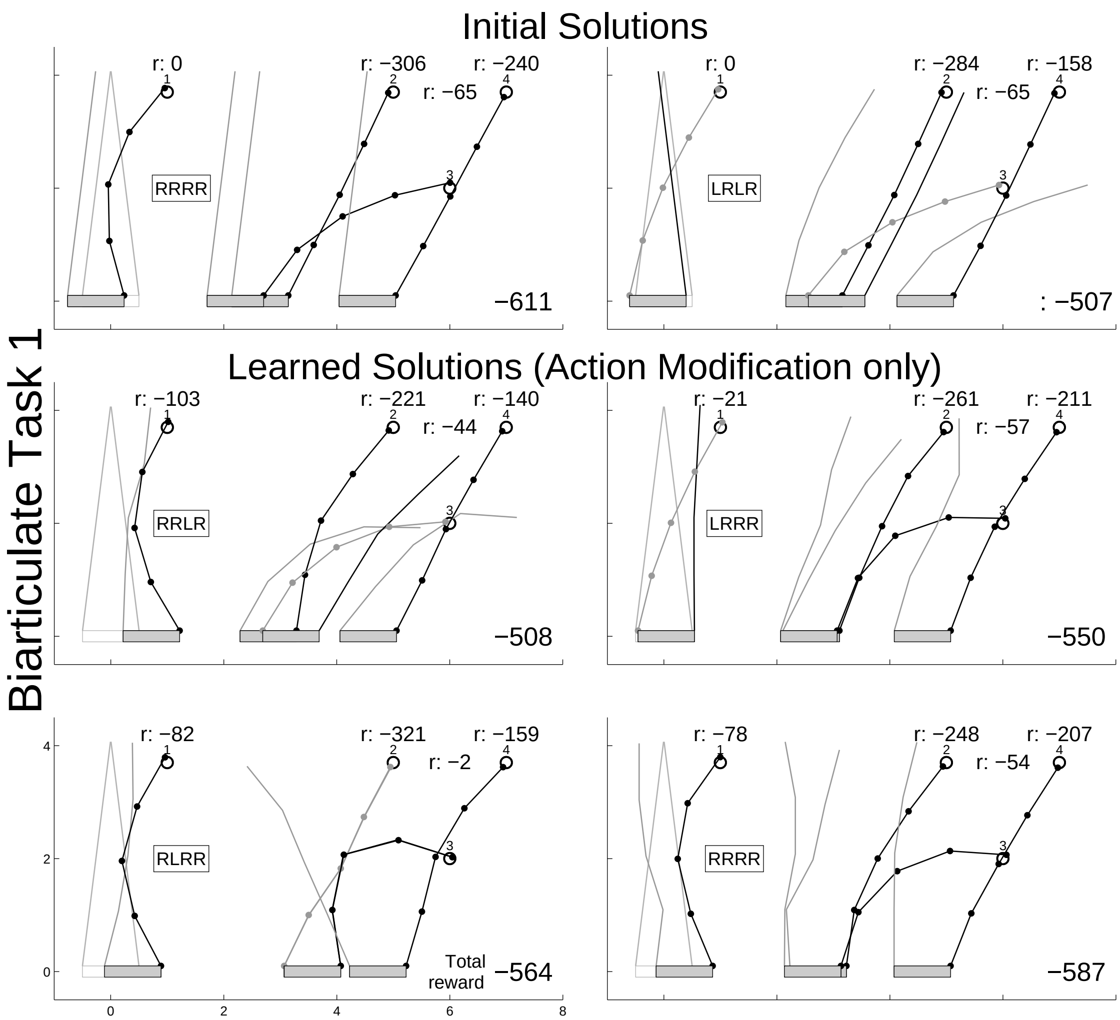

Figure 3.9. Follows similar conventions as in Figure 3.5. The right arm is drawn in black; the left arm is drawn in grey. The arm used to hit a goal is drawn with markers at the joints. Rewards for hitting each goal are indicated above the goal. The hand recruitment strategy (i.e., which hand was used to hit which target) is indicated by the box centered near point $(1, 2)$. The top two graphs indicate initial solutions using two different sequences of hand recruitment. The bottom four graphs indicated learned solutions using Action Modification alone (see text). Note that in some graphs (e.g., bottom right), because the robot did move very much from one configuration to the next, the arms for one of the configurations are hard to see.

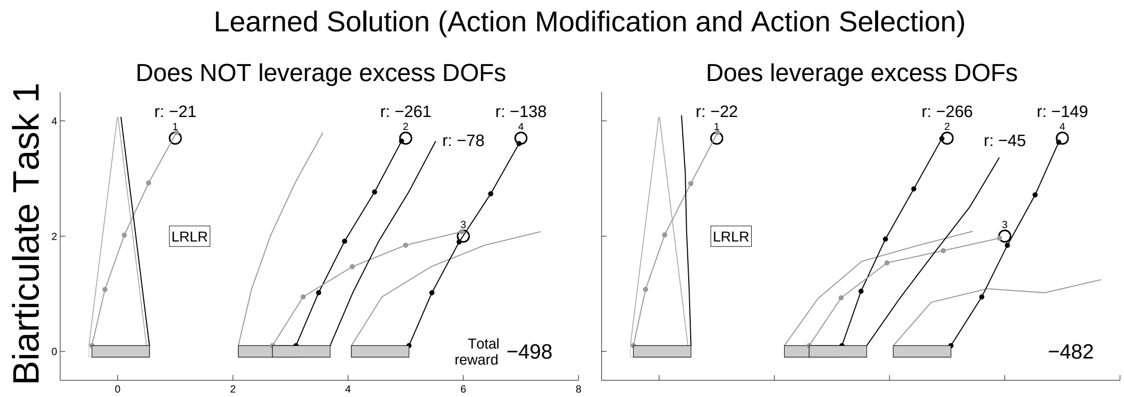

Figure 3.10. Follows the same conventions as in Figure 3.9.

with the same initial solution as that in the previous section (use just the right hand for all four goals), the learned solution adopts the strategy of alternating hands, LRLR, in all 20 runs (Figure 3.10, left). The configurations of the learned solution are visually very similar to those found by using just the *Planner* with the same hand recruitment pattern (Figure 3.9, top right). Action Modification, though, alters the configurations enough to find a better set of configurations. Thus, hypothesis 2 is supported: Action Selection and Action Modification finds better solutions than Action Modification alone.

The same strategy of hand recruitment can be found by simply trying out all possible sequences of hand recruitment with the *Planner*, an easy proposition with a small number of subtasks and actions. In a different type of task, Action Modification can alter the strategy of hand recruitment. In *Biarticulate Task 2*, the starting configuration of the robot and goals are similar to that for the Ascending task (Figure 3.5, top two graphs): the base is centered at $(0, 0)$ and its right arm is extended toward the right. However, the *left arm* is also extended toward the right, and the task is to hit the following sequence of four goals: $(6, 0.1)$, $(5.375, 1.6)$, $(6, 2)$, and $(6, 4)$, using either hand for each goal. Unlike Biarticulate Task 1, there are no restrictions on the joint variables. The *Planner* was used to hit the series of goals for every possible sequence of hand recruitment; the best initial solution used the right hand for each of the four goals (Figure 3.11, top left), while the second best strategy was to use RLRR ((Figure 3.11, top right).

After 20 runs of 10,000 trials each, two strategies were found: 1) use the right hand for all four goals (Figure 3.11, bottom left), which was the same as the strategy of the best initial solution, and 2) use the left hand for the second goal (Figure 3.11, middle left), which was the same as the strategy of the second best initial solution. However, all 20 runs only used the best initial solution (RRRR) as a starting point. Strategy 1 occurred 11 times, with a mean reward ($\pm$ standard deviation) of $-352.7(\pm 2.1)$. Strategy 2 occurred 9 times, with a mean reward of $-350(\pm 4.2)$. Even though the difference in reward is small, it is significant (two-tailed unpaired bootstrap test, $p < 0.05$, Cohen 1995). Thus, the combination of Action Selection

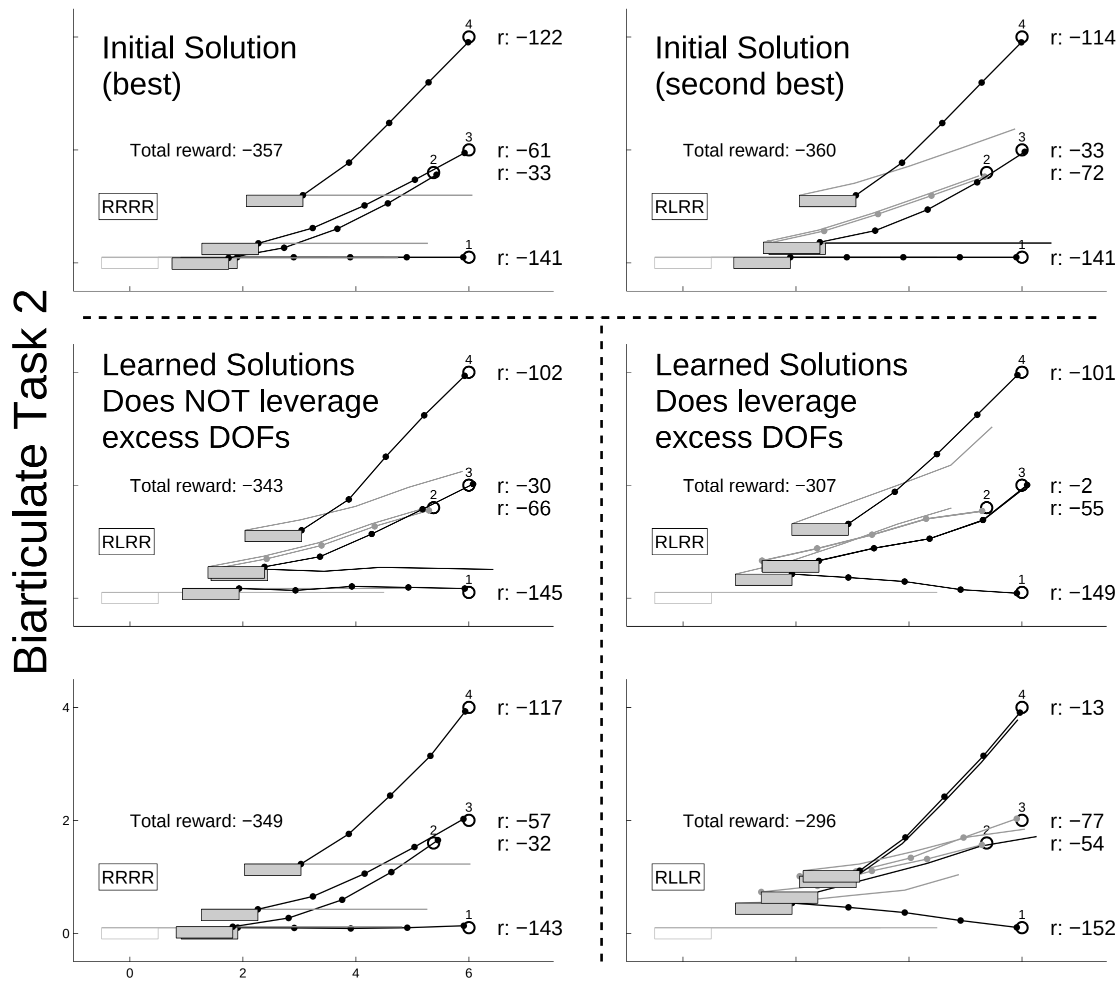

Figure 3.11. Follows the same conventions as in Figure 3.9. However, the reward for each movement is indicated to the right of the goals, as in Figure 3.5. Dashed lines separate three sets of solutions. The top set indicates initial solutions for two different sequences of hand recruitment. The bottom left set indicates learned solutions in which excess DOFs are not leveraged (see text). The bottom right set indicates learned solutions in which excess DOFs are leveraged.

and Action Modification resulted in a novel strategy of hand recruitment in just less than half the runs in this task; such a strategy was better than that of the best initial solution.

Leverage excess DOFs

In the experiments from the previous section, the joint variables of the arm that wasn't chosen for a goal were not modified: $\mathbf{q}_g(o) = \mathbf{q}_{g-1}(o)$, where o is the set of indices of joint variables corresponding to the arm not chosen for goal g. However, as evidenced by behavioral studies of tasks that use multiple discrete end-effectors (Engel et al., 1997; Baader et al., 2005; Soechting and Flanders, 1992; Hoff and Arbib, 1993; Jeannerod, 1981; Wiesendanger and Serrien, 2001), the effector(s) not used for the current subtask can be positioned to better accomplish future subtasks. As with theories on single-effector coarticulation (e.g., the Rosenbaum, Jordan, and Guenther models), it is reasonable to suggest that some form of future movements influence how the DOFs of the other effectors are used.

In this section, Action Modification is used to modify all joint variables, including those of the arm not used to hit the current goal. Resulting strategies are illustrated in Figures 3.10, right (Biarticulate Task 1), and 3.11, bottom right and middle right (Biarticulate Task 2). In Biarticulate Task 1, the strategy of hand recruitment remained unchanged; in Biarticulate Task 2, allowing the other arm to move resulted in yet another strategy of hand recruitment: RLLR (Figure 3.11, lower right). In both cases, it appears as if the use of the other arm was influenced by its configuration in hitting future goals. Excess DOFs were recruited by a supervised learning or planning process in the Jordan and Guenther models, respectively. However, this strategy was not explicitly implemented in my model; rather, it was the result of undirected exploration and hierarchical optimization.

Summary of Action Selection strategies

Figure 3.12 charts the mean rewards of the three exploration conditions, hand recruitment strategies, and tasks discussed in this section. Dark bars indicate the mean reward for a particular exploration condition, while light bars show the different strategies of hand recruitment found. (For Biarticulate Task 1, the combination of Action Modification and Action Selection led to the same strategy of hand recruitment, LRLR, for all runs. Hence, the mean reward is represented by a dark bar.) The bar charts clearly show the effects of different exploration conditions: Action Modification alone does improve upon an initial solution, but it cannot be relied upon to find the best strategy of hand recruitment. Action Selection tries out different hands some proportion of the time, even if they're not considered a valuable choice. However, by choosing them, and with Action Modification, a better strategy of hand recruitment may be found. Finally, leveraging excess DOFs by positioning the arm not used for a goal into a configuration useful for subsequent goals results in even better solutions and possibly different strategies of hand recruitment.

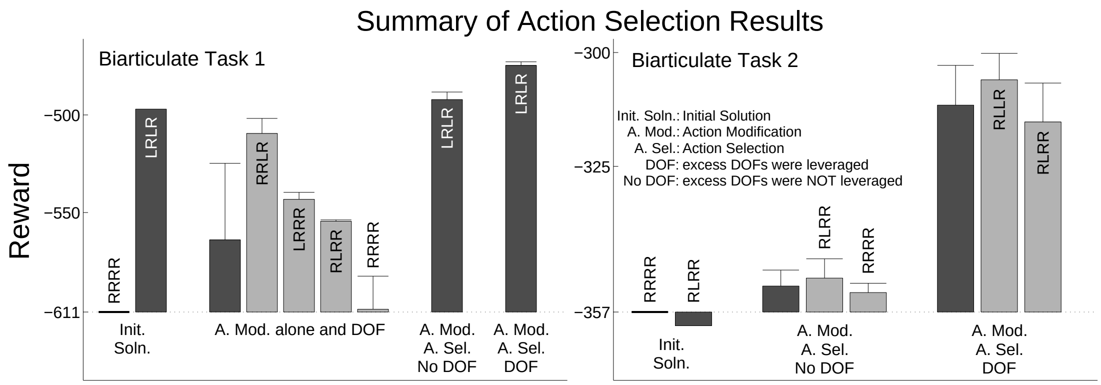

Figure 3.12. Mean rewards, including standard deviation, of all exploration conditions and tasks presented in this section. The particular strategy of hand recruitment is indicated on each bar. Unlabeled dark bars: mean of all rewards within an exploration condition, including all sets of hand recruitment patterns. If different patterns of hand recruitment were found within an exploration condition, the separate patterns are drawn with light bars. The higher the bar is, the more rewarding the strategy.

3.8 Discussion

The results presented in this chapter show that undirected exploration can produce behavior described as coarticulation. Crucial to this result is the use of hierarchical optimization, implemented here with a performance evaluation based only on the overall task. The undirected search used in my model is a more general way to search for solutions than (very reasonable) types of directed search, such as blending actions together, restricting search to directions optimizing specific secondary objectives, or planning. Two advantages of undirected search and hierarchical optimization are readily apparent.

Advantage 1) The best solutions might not lie in the direction dictated by directed search. The learned solutions of the Ascending task in the first set of experiments were not a non-negative linear combination of initial solutions. Planning requires an intimate knowledge of the system, environment, and task, and requires computational resources. In some cases, it might be better to simply try different variations out and observe the results. Also, while most instances of coarticulation can be described by directed search to some degree, there may be subtle differences. *Preshaping* (Hoff and Arbib, 1993; Jeannerod, 1981), mentioned at the beginning of this chapter, describes the act of opening one's hand while transporting it to an object to be grasped. How the hand opens during transport is different than how it opens if it was already at the object; coarticulation, in this case, it not simply initiating a subsequent action while executing the current one.

Advantage 2) Different tasks might have different objectives. Specification of secondary objectives may aid in search, but good learned performance requires that such specification is accurate. As discussed in Jerde et al. (2003), different secondary objectives seem to be optimized, depending on the task. A more general performance evaluation, such as the reward used in my model, can capture the different task demands.

A form of undirected search with hierarchical optimality was previously implemented on a three-link dynamic "weight lifting" robot arm (Rosenstein, 2003; Rosenstein and Barto, 2001). As with my model, initial solutions were generated and then allowed to vary depending on reward for the overall task. The robot found different types of learned solutions, and the learned solutions deviated greatly from initial solutions. In addition, solutions found for one weight were different than those found for another weight. Dynamics complicates the task significantly and makes planning or specification of secondary objectives a much harder problem.

There are, of course, some disadvantages. Although undirected search may be able to find solutions directed search cannot, it also looks for solutions in poor regions of action space. It is likely that some combination of directed search and undirected search is employed by our nervous system. The trade-off between directed search and undirected search is similar to the *exploitation-exploration* problem of Reinforcement Learning (Sutton and Barto, 1998; Barto and Dietterich, 2004). The values of actions must be estimated through experience — the agent must explore by trying out different actions in order to find the best one. However, such exploration will yield poor performance on some trials as the agent will inevitably try out poor actions. To avoid worse performance, the agent should exploit the knowledge it already has by selecting actions it estimates are the most valuable. Of course, then it cannot find potentially better actions.

An action selected from a particular state can be more rewarding than other actions through one of two ways: 1) in context of other actions selected in other states, as in biarticulate task 1 (Figures 3.9 and 3.10). The best strategy of hand recruitment as found in the learned solution (Figure 3.10) was the same as that found when the *Planner* was allowed to try out every possible strategy of hand recruitment (Figure 3.9, top right). 2) With the addition of Action Modification, where actions are modified and become more valuable, as in biarticulate task 2 (Figure 3.11). The best strategy of hand recruitment found in the learned solution was different than that found by the *Planner*. An action (hand) was modified to be more valuable than other actions, even within the same context.

I know of no other study in motor control which investigates the interplay between Action Modification and Action Selection. A similar interplay, though, is investigated in studies of hierarchical Reinforcement Learning, a field that studies the use of abstraction and hierarchy to better learn in large and complicated environments (cf., Barto and Mahadevan 2003). In this discussion, I essentially equate two types of hierarchy: 1) *options* (Precup et al., 1998; Sutton et al., 1999; Precup, 2000) and 2) *task decomposition* (Dietterich, 2000). While there are differences, such differences are beyond the scope of this discussion.

An option is a policy defined over a subset of the state space in which actions are selected to accomplish some subtask. For example, if the entire state space was all positions in a building which included a set of rooms, an option recruited from room 1 could be "move to the door." Rather than make a decision at each and every position encountered in the building, when the agent recruits an option, it makes one decision (e.g., move to the door) and executes that option's policy until the goal of the option has been achieved (e.g., the door is reached) before making another decision. Options are similar to actions. In task decomposition, a task is decomposed into a hierarchical set of subtasks. A decision to accomplish a particular subtask is similar to a decision to recruit a particular option.

An option or subtask is analogous to an action in my model, which is a specification of a joint configuration to which to move. Action Modification is analogous to modifying the policy used in the option or subtask. An action/option/subtask may not be considered valuable at first. However, modification may result in a better way to execute the action/option/subtask, resulting in an increase in its value. We see this effect in Figure 3.11, which shows how Action Modification leads to different recruitment of hands.

CHAPTER 4

AUTOMATIZATION

4.1 Automatic Behavior

We all have some concept of what "automatic movements" are. Anecdotal examples include typing a frequently-used password or even driving to work. The subjective feeling of executing a sequence of movements automatically is distinct from that of non-automatic movements, so much so that we are fairly certain that such a distinction exists in fact as well as in feeling. Automatic movements have been a subject of great study since the days when psychology used methods based more in philosophy than empirical science. According to most theories, the main characteristic of automatic movements is that they are executed involuntarily. They are elicited directly from sensation in a manner similar to the Cartesian reflex (*Treatise on Man*, Renè Descarte). In his seminal work, *The Principles of Psychology*, William James (James, 1890) described an automatic movement (or, a habit) as "mechanically, nothing but a reflex discharge" and suggested that "the most complex habits ... [are] nothing but concatenated discharges in the nerve-centres." The learned reflex description led to the theory of *stimulus-response* (SR) learning (Thorndike, 1911; Washburn, 1916; Watson, 1920), in which an action (response) is directly elicited by a stimulus (sensory cue).

The link between automatic movements and volition (or, lack thereof) is so strong that the existence of automatic movements was used in philosophical discussions of consciousness. For example, in his theory of dualism, Descartes (in *Meditations on First Philosophy*) suggests that the mind and body are distinct entities. The mind is a non-physical entity and responsible for volitional behavior, the body a physical entity and responsible for automatic behavior. In the first two chapters of *The Principles of Psychology*, where James discusses what psychology is and what the brain's functions are, James frequently discusses the relationship between automatic behavior and consciousness. Because volition and consciousness are such vague concepts, "an outside observer, unable to perceive the accompanying consciousness, might be wholly at a loss to discriminate between the automatic acts and those which volition escorted" (James, 1890). In short, automatic behavior is easy to perceive subjectively, so we assume it exists. Its existence has profound ramifications on our understanding of the mind. Unfortunately, its existence is hard to show and describe objectively. However, even if we constrain characterization to observable behavior, interesting traits associated with automatic movements can be gleaned.

Speed

Many of the behavioral studies use a type of task called the *serial reaction time* (SRT) task, used to assess capacity for learning sequences (Nissen and Bullemer, 1987; Keele et al., 2003; Matsuzaka et al., 2007). In a typical task, a subject must execute a specific action (e.g., press button 1) in response to a stimulus (e.g., the visual presentation of the numeral 1). The stimuli were presented randomly or in a set sequence. After training, the subject executed actions faster for the set sequence than for the random sequences, even if they were not aware of the set sequence. In some cases, reaction times were negative, i.e., the subject began the next action before he was cued to do so. Some of the increase in speed could be due to coarticulation effects (although Matsuzaka et al. 2007 reported no difference in movement kinematics) and the use of different sensory information (to be discussed in later chapters). However, the effects of automatization cannot be ruled out.

Interdependency

In automatic movements there is an interdependency between actions: observation of one action predicts another with great accuracy. Muchiake et al. (2001) trained monkeys to navigate a maze on a computer screen by using hand movements to move a cursor to a goal; shorter paths were rewarded more. Goal location varied between trials, but the starting point of the cursor was fixed. Some monkeys adopted a strategy of using "sub-goals:" if there were points common to the paths toward several goals, and alternate routes provided no advantage, the monkey would often follow points along the common path.

In a sequential button pushing task, Matsumoto et al. (1999) trained a monkey to execute a series of three button pushes in a set sequence on a 3x3 grid. After the task was well learned, the monkey was tested with "random trials," in which the third button in the sequence was located in one of three random locations. For several trials, the monkeys would continue to push the third button of the learned sequence even though another button was lit, and then push the lit button. This suggests that the original three-button sequence was executed as a integrated unit.

Berridge and colleagues investigated a type of grooming behavior in rodents, termed *syntactic chains* (SCs, Berridge et al. 1987), related to learned automatic movements. The SC consists of four phases of grooming: 1) the rat uses forelimb strokes to groom its vibrissae, 2) eyes, 3) ears, and 4) then licks its body. Within an SC, these actions occur in a set order, or sequence, with a probability $13,000$ times greater than chance. However, the same individual actions can occur outside an SC, and they appear in rodent behavior before development of the SCs, indicating that the SC is an integration of existing actions. Once the first part is initiated, the rest of the sequence can be predicted with 85% accuracy, suggesting that an SC is recruited as an integrated unit.

If the sequence of actions is learned as a unit, then the order of the actions is learned. However, if, instead, the transition probabilities of the elements are learned, then order is not important. For example, in both sequences, "ABAC" and "ACAB," the probabilities of B and C following A are 0.5 each. If, in an SRT task, a subject was trained on one sequence and tested on the other, there should be no difference in performance if only the transitions between adjacent elements is learned. Jackson et al. (1995) showed that the order, not the transition probabilities, is learned.

Goal independency

We associate the execution of a sequence of actions automatically as executing them without conscious thought. Another way to say this is that there is no decision-making process involved once the skill is initiated – the sequence of actions is executed without evaluating each action to decide how appropriate it is. In the previous section, I used the Matsumoto et al. (1999) study to support the claim that, when executed automatically, a sequence of actions is executed as a single unit. Part of the reasoning behind this is that the three button pushes were executed regardless of how useful they were — decision-making is eliminated from the process.

Dickinson (1985) explores this issue more directly with rats in an instrumental conditioning task. Dickinson trained rats to hit one of two levers in order to receive a reward, but he later devalued the stimuli used for the reward by, in a different context, decreasing the rats' motivation for them, pairing them with unpleasant stimuli, or other manipulations. After goal devaluation, the rats were placed in the original context and presented with the two levers. Rats that were trained for a long time on the original task continued to press the same lever, thus obtaining the (now devalued) "reward." After a few trials the rat learned to change its behavior. Rats that were trained for a shorter time on the original task immediately changed behavior. In the former case, the action was considered automatic (or, in Dickinson's terms, a *habit* or *response*). In the latter case, the action was elicited as a result of cognitive processes the explicitly paired the action with the reward. Yin and Knowlton (2006) review many manipulations and provide evidence that the basal ganglia are critical in the development of habits.

4.2 Theoretical account of automatization

Sequence learning

The second characteristic of automatic movements, the interdependency of actions, is so prominent that many theoretical accounts of automatic movements focus on the problem of *sequence learning*, in which one can predict an entire ordered sequence of elements given only the first few elements (cf. Dominey 2002). Most neural network or connectionist models of sequence learning use *recurrent connections*, in which some neurons' outputs also serve as their own inputs (directly or indirectly, cf. Doya 2002). They capture aspects of the history of the network, providing a form of context. Even before computational models of neural networks were developed, recurrent connections were thought to be responsible for automatic movements. James (1890) suggested that automatic movements are

> due to the presence [in the brain] of systems of reflex paths, so organized as to wake each other up successively - the impression produced by one muscular contraction serving as a stimulus to provoke the next, until a final impression inhibits the process and closes the chain.

While a simple stimulus-response chain (*SR chain*) may account for simple sequences of movements, many models exploit another property of some types of recurrent connections: *internally-generated dynamics*, in which the recurrent connections cause the activation levels of neurons of the network to change over time without any external influence (cf., Vogels et al. 2005; Guigon et al. 2002). For networks that use neurons whose activation levels decay without excitatory input, internally-generated dynamics are also responsible for *sustained activity*, in which a neuron's activation level reaches a stable non-resting value.

By using the context and internally-generated dynamics afforded by recurrent connections, neural networks are able to reproduce a variety of sequences. Most models represent sequences in one of two ways. First, the output of the neural network changes over time to represent the elements of the sequence in order (e.g., Dominey 1995; Berns and Sejnowski 1998). A network of this type essentially represents an SR-chain: the activation levels of the inputs (stimuli) change and the network is trained to produce the correct outputs (responses). Second, the temporal order of the sequence (or possible sequences) is represented as a spatial pattern of activation of the output units (e.g., Hopfield 1982; Beiser and Houk 1998). A network of this type relies on internally generated dynamics so that its activity evolves over time to settle on the correct representation. If trained properly, the network forms an *attractor state*, a pattern of activity that the network will evolve to even if faced with degraded initial inputs or externally applied perturbations.

Some neural network models are able to produce behavior more complicated than reproducing specific sequences. Botvinick and Plaut (Botvinick and Plaut, 2004, 2006, 2002) present a three-layered network model in which context is represented by internally-generated dynamics due to recurrent connections confined to the middle (or *hidden*) layer. The output layer represented actions, which are communicated to an environment, and the resulting change in environment is communicated to the input layer. The *Botvinick* model differs from the previously cited models in two ways: 1) the output layer only projects to the environment; no recurrent connections emanate from it, and 2) it is trained over several sequences, each one a solution to the same task. This flexibility is hard to capture with some architectures. The *Botvinick* model is able to learn that several sequences are equivalent; each sequence is used to accomplish the task at different trials. Most other models would treat each sequence as entirely separate entities.

Advantages of automatization

The models cited above show that simple computational mechanisms can learn sequences. However, their functional advantages are not immediately obvious and are not discussed in the presentation of the models (nor were they meant to be; the models focused on how sequence learning might be accomplished, not why such learning is advantageous). The implicit advantage is that simple computational mechanisms, rather than more complicated ones, are used to reproduce a learned sequence.

Below I discuss the advantages of the use of simple computational mechanisms. I also discuss how, and under what circumstances, simple computational mechanisms can be trained.

Historical perspective

The three observable characteristics of automatized movements are an increase in speed, interdependency of actions, and goal independency. While the functional advantage of speed is obvious, there are no functional advantages of the other two in of themselves (when considering execution of the sequence of actions in isolation of a greater context). Rather, these characteristics may emerge from the use of a mechanism to execute automatic movements that is simpler than that of non-automatic movements. It is the use of a simpler mechanism that has advantages. James (1890) suggests that a habit "diminishes fatigue" and "the conscious attention with which our acts are performed," and that "our lower centres know the order of these movements,... But our higher thought-centres know hardly anything about the matter." James further surmises that

> A strictly voluntary act has to be guided by idea, perception, and volition, throughout its whole course. In an habitual action, mere sensation is a sufficient guide, and the upper regions of brain and mind are set comparatively free.

In essence, "conscious thought" takes effort, and without automatic movements,

> we could not accomplish everyday tasks as each little act, like tying our shoes or dressing ourselves, would require so much conscious effort that we will be exhausted. ... The more of the details of our daily life we can hand over to the effortless custody of automatism, the more our higher powers of mind will be set free for their own proper work. There is no more miserable human being than one in whom nothing is habitual but indecision, and for whom the lighting of every cigar, the drinking of every cup, the time of rising and going to bed every day, and the beginning of every bit of work, are subjects of express volitional deliberation.

Again, James' arguments depend on some concept of volition and consciousness. Rather than try to define these concepts and that of "thoughtful effort," I focus instead on the computational resources a control mechanism requires to make a decision (e.g., to select an action).

Conceptual model

Consider the following task: a learning agent starts off in position $p_{t=0}$ and must reach a goal position, p_g. From each position, the agent can select one of several

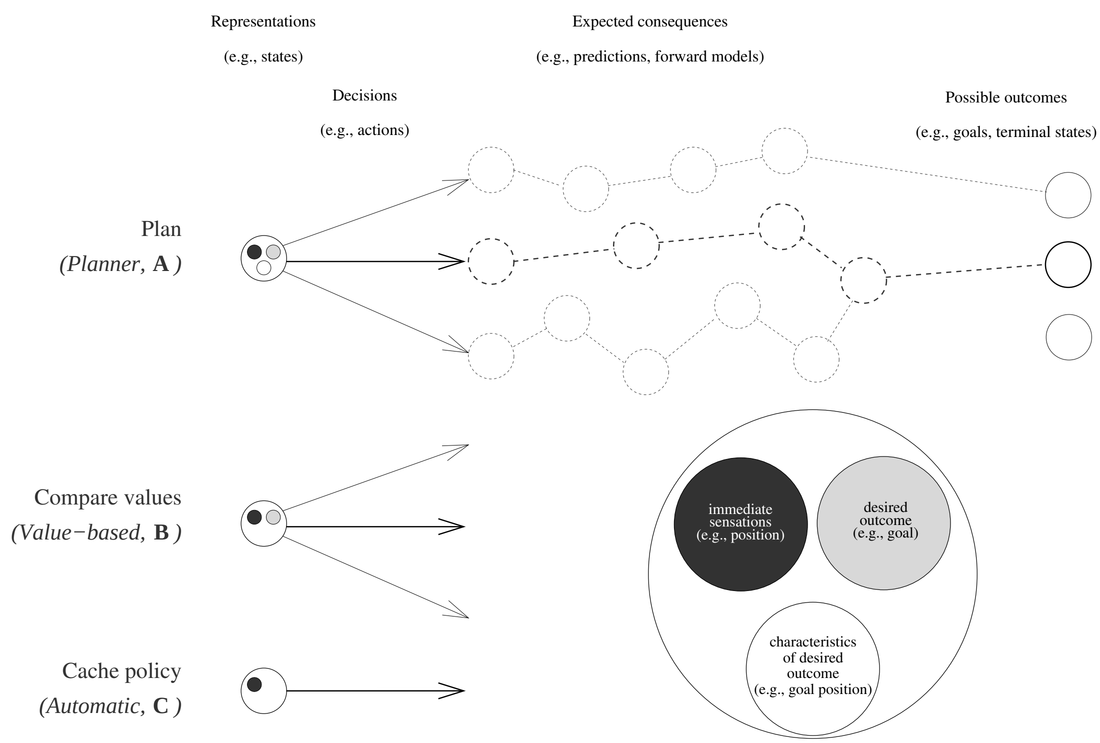

Figure 4.1. Illustration of three types of controllers (labels on left). Each controller uses some representation to select an action (arrows). The large circle (bottom right) indicates representation: the smaller circles within it label features of representation. Dashed circles and lines, used by the *Planner* (top), indicate expected representations and actions encountered. Solid circles used by the *Planner* (top right) indicate final possible consequences. The thick arrows (and thick dashed circles and lines) indicate best actions and consequences.

actions, a, which transports it to another position. Thus, the agent experiences the following chain of positions and actions:

$$p_{t=0} \rightarrow a_{t=0} \rightarrow p_{t+1} \rightarrow a_{t+1} \ldots \rightarrow p_{t+n} \rightarrow a_{t+n} \ldots \rightarrow p_g.$$

By what mechanism does the agent select an action? Figure 4.1 illustrates, on a conceptual level, three possible control mechanisms. The starting representation (corresponding to position $p_{t=0}$), referred to as a *state* in the figure, is depicted as a solid-lined circle on the left. The small circles within the larger circle represent immediate sensations (e.g., position, dark circle), a label for the desired outcome (e.g., goal, grey circle), and characteristics of the desired outcome (e.g., goal position, unfilled circle). From there, actions (arrows) can be selected.

A high level *planning* controller, depicted in the top part of Figure 4.1, explicitly considers the long-term consequences of selecting each possible action. The consequences include what the expected next position would be for each action, what action it would select from those positions, and so on until it reaches the goal. The consequences are depicted as dashed lines and circles. In order to plan, some representation of the current position, the goal, and the position of the goal must exist; hence, the state includes all three. The goal is the thick-lined circle at the right of the diagram, the best expected consequences are drawn with a thicker dashed line, and thus the best action is the thick arrow. Crucial to the planner's success is a model of the environment (i.e., all states, actions, transition probabilities, possible outcomes, etc). If the goal (and hence the task) changes frequently, the planner is very useful as it can adapt immediately. Each decision it makes is based on predicting its consequences; if those consequences change, so do the decisions. However, it also requires much computation, a rich representation, and an intimate knowledge of environment.

If, on the other hand, a particular goal is frequently encountered, use of the planner requires unnecessary computational and representational resources. Rather than predicting the consequences of each action, a simpler controller can keep track of how "valuable" each action is from each state. (As used in this description, "value" is intentionally vague so as to encompass any measure of what it means to achieve a goal optimally. As used in typical RL applications, value is the expected cumulative sum of rewards.) Value estimates are gained through experience; such experience can be generated by the planner, selecting actions via some reasonable initial policy, or even randomly selecting actions. Over time, the estimated value of each action from each state may be accurate enough to enable a *value-based* controller to select appropriate actions by comparing the estimated values of each action (middle part of Figure 4.1). The computational requirements are much less than those of the planner as the consequences of each action are not determined. In addition, the agent does not need a model of the environment; the value estimates are the only basis of comparison. Nor does the agent need a representation of the characteristics of the goal (e.g., goal position). However, it does require some minimal representation of goal, as the values of actions for one goal may be different than those for another. Thus, state includes current position and label for the goal. Finally, a representation of each action in each state is required as the agent compares the values of each action.

If the same action is selected in response to the same immediate sensation most of the time, regardless of goal, that policy (a mapping from sensation to action) can simply be cached, or "hard-coded" (bottom part of Figure 4.1). This controller requires the least computation as no alternative actions are even considered and uses a simpler state representation (position, but no representation of goal). The savings in representational and computational resources can be directed toward other tasks. A similar scheme has been suggested by Logan (1988) (see also Logan et al. 1999), who refers to complicated controllers as *algorithmic* and simpler controllers as *memory-based*.

The use of all three controllers, with the lower ones requiring more training, can enable an agent to accomplish a task immediately (assuming the planner has knowledge of the environment), but use simpler controllers as they are trained. The simplest controller produces behavior characteristic of automatized movements:

- If we assume that the fewer resources a controller needs to select an action, the less time it takes to execute that action, then simpler controllers select actions faster than more complicated ones.
- Because, by design, the goal is not represented with the simplest controller, when the same action is elicited in response to the same immediate sensations, trajectories common to multiple tasks will be cached.
- Also, again because goal is not represented, if the task changed, the mechanism in Figure 4.1C would select the same action even if it is no longer appropriate to do so.

I thus refer to the last controller as an *automatic controller*.

As opposed to most sequence learning accounts of automatization, which focus on how one type of controller can learn a given sequence, this model focuses on how different types of controllers are recruited to select the same actions. As a result, the model determines what actions can be controlled by the automatic controller — the sequences are *developed*, not given as training examples to be learned and then reproduced.

4.3 Hypotheses

I implement the conceptual model described in the previous section with a computational model, described in the next section. To keep the focus of the model on decision-making and to avoid complications that may arise with more realistic environments, the model is tested in a simple discrete-state discrete-action environment in which executing an action causes a transition from one state to another. Such environments can be represented in different ways. I use the "grid-world" representation common in the computational Reinforcement Learning literature (Sutton and Barto, 1998), shown in Figure 4.2. Although this representation suggests a maze to test navigational abilities, it is misleading to think of the grid-world in this way. It

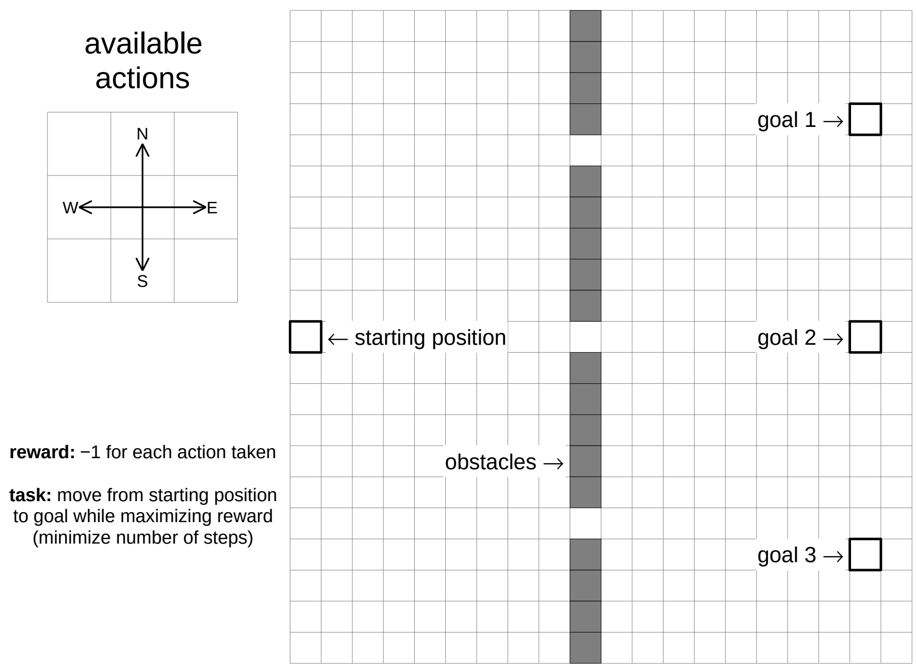

Figure 4.2. Representation of the "grid world" task used in the model. Each small square is a position; obstacles (solid grey squares), goals, and starting position are labeled. The effect of each action is illustrated in the top left mini-grid: N, north; S, south; E, east; W, west.

merely provides a visually-accessible representation of an abstract sequential decision task.

The underlying environment is a Markov decision process with states (p, g); the state space is factored into positions $p \in P$ (immediate sensations) and goals $g \in G$ (desired outcome). The characteristic of a goal is a particular position. The agent has available to it actions $a \in A$, which deterministically cause transitions from (p, g) to (p', g). No action causing a transition across the goal dimension exists.

At the start of each trial, the agent is in the starting position (labeled highlighted square on the left of the grid) and the goal for that trial is chosen randomly from a set of three (labeled highlighted squares on the right side). The four cardinal actions (north, south, east, and west) are available to the agent at each state (the effect of each action is shown in Figure 4.2). When the agent chooses action a, it incurs an immediate cost, represented as a negative numerical reward, $r = -1$. If the agent chooses an action that would cause a transition into an obstacle or off the grid, it receives the cost of the selected action and does not change positions. A trial ends when the agent reaches the position of the selected goal, at which point it receives reward $r = R_g$ (set to $+100$ in the first set of simulations, but I explore the effect of R_g on behavior later in the Results). The agent's objective is to maximize the cumulative reward over each trial by reaching the chosen goal with the smallest number of actions.

I hypothesize that the model will produce behaviors characteristic of automatic behavior. The relative "speed" of each controller is a design of the model in that it is assumed that simpler controllers select actions faster than more complicated controllers. In addition, if a contiguous sequence of actions as selected by the automatic controller is available, by design there will be an interdependency between them. Thus, I test the following two characteristics:

1. *Interdependency*: The automatic controller will select actions at positions at which the same actions are repeatedly selected by more complicated controllers, including where the same actions are selected for all three goals (as seen in Muchiake et al. 2001.

2. *Goal independency*: If a contiguous sequence of actions as selected by the automatic controller is available, it may still be recruited even if it results in a suboptimal strategy. However, such an availability will aid in learning a task, demonstrating the usefulness of the motor skill.

4.4 Model

As discussed in Chapter 2 of this thesis, the *Planner*, **A**, represents cortical planning mechanisms; the *Value-based* controller, **B**, represents the reward-mediated learning functions of the basal ganglia, and the *Automatic* controller, **C**, may be implemented by the thalamostriatal pathway. With experience, control is transferred from **A** to **B** to, if appropriate, **C**. Below I describe in detail the multiple controller model implementing these concepts.

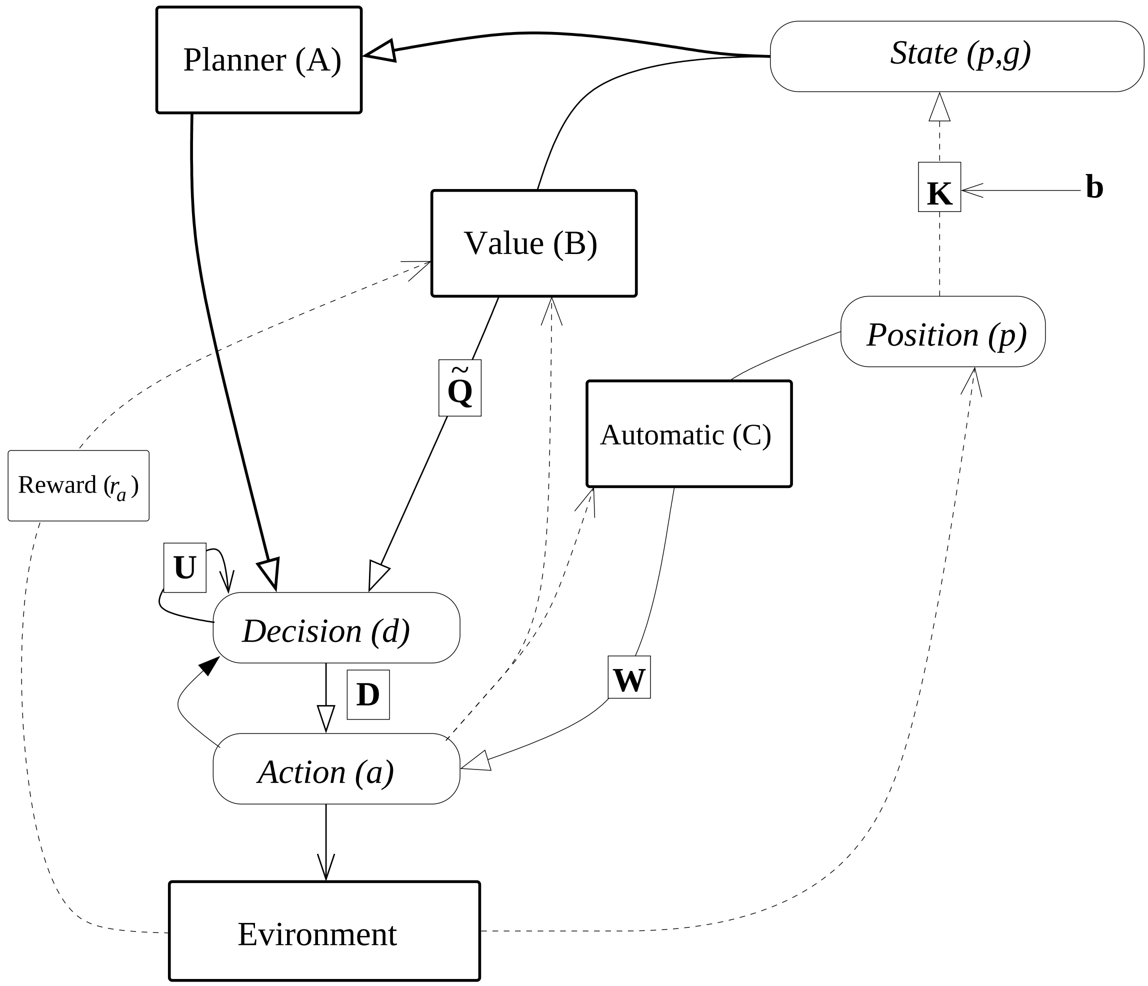

Figure 4.3. Architecture of the connectionist-style model. Unfilled closed arrows indicate excitatory connections, filled closed arrows indicate inhibitory connections, and open arrows indicate unrestricted connections. For clarity, ascending projections are shown with dashed lines. Arrays of neurons are represented by boxes with rounded corners and labeled with italics.

The multiple controller model is implemented with a connectionist-style model (Figure 4.3). Current position is represented by a $||P||$-element array of *Position* neurons where element p_i, corresponding to the current position, is 1 and all other $p_{j\neq i}$ are zero. State is represented by a $||P|| \times ||G||$-element array of *State* neurons; each neuron is labeled by (p_i, g_j). The activation of the *State* neurons is determined by $\mathbf{K}(\mathbf{p}, \mathbf{b})$, where $\mathbf{p}$ is a vector of the excitation levels of the *Position* neurons, $\mathbf{b}$ is a $||G||$-element goal vector, where $b_i = 1$ if g_i is the goal and all $b_{j\neq i}$ are zero, and $\mathbf{K}$ returns the outer product of its arguments. Thus, the state neuron corresponding to the current position and goal is 1 while all others are zero.

Actions are represented by an $||A||$-element array of *Action* neurons. When *Action* neuron a_i is excited to or beyond a threshold, θ (set to 5), the action corresponding to a_i is executed. Each *Action* neuron has an activation function of

$$f_a(x) = \begin{cases} y & \text{if } x < \theta \\ \theta & \text{otherwise,} \end{cases}$$

where x is the input and y is the resting activation level of the *Action* neurons, based on the bistable properties of striatal neurons (as discussed in Chapter 2, pg. 17). The striatal neurons can be in either an *upstate* ($y = 2.5$ in my implementation) or a *downstate* ($y = 0$). I discuss the ramifications of this later in the model description.

How the *Action* neurons are excited depends on the controller. The *Automatic* controller excites them directly; the *Planner* and *Value*-based controller excite them by exciting an $||A||$-element array of *Decision* neurons. Excitation of *Decision* neuron d_i corresponds to a decision to take the action represented by a_i. Each *Decision* neuron has an activation function of:

$$f_d(x) = \begin{cases} 0 & \text{if } x < 0 \\ x & \text{otherwise.} \end{cases}$$

The *Decision* neuron array constitutes a winner-take-all (WTA) network with the connection matrix $\mathbf{U}$: for all $i \neq j$, $u_{ij} = -1/||A||$, while each $u_{ii} = 1$. The *Decision* neurons project to the *Action* neurons via connections $\mathbf{D}$, which is merely the identity matrix in this implementation. When an action is taken (i.e., some $a_i > \theta$), the activation levels of the *Decision* neurons are set to zero (via inhibition from the *Action* neurons). Below I describe the three controllers.

Planner (A)

The *Planner* uses the current position, chosen goal, and goal position to select an action via the well-known heuristic search algorithm A* (Hart et al., 1968). Briefly, A* searches through possible positions (p') reachable from the current position (p). For each p', the cost incurred traveling from p to p' and the heuristic function of p' are calculated. The heuristic function I use is the negative of the Euclidean distance between p' and goal position (hence, the characteristics of the goal — its position — is required for the *Planner*). By searching through the "best" positions first (where the "best" positions are the ones for which the sum of the cost incurred and cost estimated by the heuristic function is least), A* finds the optimal trajectory from one position to another (assuming one exists) without spending too much time searching through more costly trajectories. The action moving the agent from p to the best next position is selected; in the case of ties, an action is chosen randomly from the set of best actions. This is not meant to be a realistic representation of cortical planning mechanisms. However, it captures the functional properties I wish to implement in **A**: provided a model of the environment, explicit knowledge of goal position, and sufficient computational resources, it suggests a reasonable action without any prior experience. When **A** selects an action, it excites the corresponding *Decision* neuron d_i to an excitation of θ, resulting in an excitation of *Action* neuron a_i to θ. The action is executed.

***Value*-based Controller (B)**

The *Value*-based controller uses the current position and chosen goal to select an action (but the goal's position is not used). To do so, a Q-table, of dimensions $||P|| \times ||G|| \times ||A||$, is used in which element $Q(p, g, a)$ estimates how valuable action a is in state (p, g). In this case, *value* refers to the expected cumulative reward received by taking action a from position p in order to reach goal g. The values are learned through direct experience. $Q(p, g, a)$ is updated via the Sarsa algorithm of Reinforcement Learning (*s*tate-*a*ction-*r*eward-*s*tate-*a*ction, Rummery and Niranjan 1994; Sutton and Barto 1998): en route to goal g, when action a is taken from position p, and then action a' is taken from the next position p',

$$Q(p, g, a) \leftarrow Q(p, g, a) + \alpha \ \left(r + \gamma Q(p', g, a') - Q(p, g, a)\right), \tag{4.1}$$

where α is a learning rate (set to 0.01) and γ is a discount factor (set to 1). $\mathbf{Q}$ is initialized to $\mathbf{0}$ (bold capital letters indicated matrices or multi-dimensional tables).

The *Value*-based controller is implemented as an excitatory mapping, $\widetilde{\mathbf{Q}}$, from *State* neurons to *Decision* neurons. $\widetilde{\mathbf{Q}}$ is initialized to $\mathbf{0}$ and is trained to represent the information contained in the Q-table — the values of each action from each state. I discuss why the Q-table is not used directly at the end of the description of the model. First, the Q-values are transformed into positive numbers normalized across actions via a soft-max-like function: for state (p, g) and all actions,

$$\psi(p, g, a) = \frac{e^{Q(p,g,a)/\tau}}{\sum\limits_{a \in A} e^{Q(p,g,a)/\tau}} \tag{4.2}$$

$$\mathbf{\Psi}(p, g, \cdot) = f_{\Psi}\left(\psi(p, g, \cdot)\right) \tag{4.3}$$

where τ is the temperature (set to 5), f_{Ψ} is a vector-valued function that sets each element of its argument vector to be the maximum of that element and 0.034 and then normalizes the vector, and $\mathbf{\Psi}(p, g, \cdot)$ and $\psi(p, g, \cdot)$ are $||A||$-element vectors, the elements of which correspond to the actions for state (p, g). Thus, after the normalization step, no element of $\mathbf{\Psi}$ is less than 0.03; other than that constraint, $\mathbf{\Psi}(p, g, \cdot)$ behaves similar to a soft-max in that the higher $Q(p, g, a_i)$ is relative to $Q(p, g, a_{j \neq i})$, the higher $\Psi(p, g, a_i)$ is.

$t_U = 0$
Calculate each $\tilde{d}_i$ for $i = 1, ..., ||A||$
while all $a_i < \theta$ and $t_U < t_U^{max}$
    $t_U = t_U + 1$
    each $d_i(t_U) \leftarrow f_d\left(\tilde{d}_i + \sum_j^{||A||} U_{ij}\tilde{d}_j(t_U - 1)\right)$
    each $a_i = f_a(d_i)$.

Table 4.1. The winner-take-all (WTA) circuit that comprises the *Decision* neuron array. t_U is the time step within the WTA, t_U^{max} is the maximum number of steps, and all other symbols are defined in the text.

$\mathbf{\Psi}$ is used to update the values of $\widetilde{\mathbf{Q}}$: when the agent is in position p, for state (p, g) and all actions,

$$\widetilde{Q}(p, g, a) \leftarrow f_d\left(\widetilde{Q}(p, g, a) + \alpha_q\left(\Psi(p, g, a) - \widetilde{Q}(p, g, a)\right)\right), \tag{4.4}$$

where α_q is a learning rate (set to 0.005) and f_d is defined as before. Note that $\widetilde{\mathbf{Q}}$ is not explicitly constrained to be normalized across the actions. In fact, it is initialized to $\mathbf{0}$ and its elements increase at a slow rate; thus, it isn't normalized during early stages of learning.

Each *Decision* neuron is excited by the *State* neurons as follows:

$$\tilde{d} = f_d\left(\sum_{p \in P}\sum_{g \in G}[(p, g)]\ \widetilde{Q}(p, g, a) + \eta_\sigma\right), \tag{4.5}$$

where $[(p, g)]$ is the activation level of *State* neuron (p, g), a is the action to which the *Decision* neuron corresponds, and η_σ is random number from a zero-mean Gaussian distribution with standard deviation σ (set to 0.15). (Note that $\tilde{d}$ can also be written simple as $f_d(\widetilde{Q}(p, g, a) + \eta_\sigma)$ when the agent is in state (p, g).) The notation $\tilde{d}$, as opposed to d, explicitly denotes the inclusion of η_σ. The WTA comprising the *Decision* neuron array is outlined in Table 4.1; I discuss its ramifications later in the description of the model.

The WTA circuit runs until an *Action* neuron is activated (some $a \geq \theta$) or a step number limit (t_U^{max}, set to 60) is reached (note that t_U, the time step within the WTA, is distinct from t, which is the time step in a trial). The use of η_σ causes the excitation of the *Decision* neurons to behave similar to a soft-max function in which the probability that action a is selected increases as the value of a relative to the other actions increases.

Because the values of $\widetilde{\mathbf{Q}}$ are initialized to $\mathbf{0}$ and are increased slowly, no a_i is excited to θ via $\mathbf{B}$ during early trials. Only after the values of $\widetilde{\mathbf{Q}}$ corresponding to state (p, g) are high enough to excite some a_i to θ is that action selected via $\mathbf{B}$.

Automatic Controller (C)

C selects actions based only on the current position and does *not* incorporate any representation of the goal. **C** is trained from experience generated by actions selected via **B**. When the agent is in position p and action a is selected via **B**, the weight of the association between p and a, $W(p,a)$, is modified according to a Hebbian-style (Hebb, 1949) learning rule as follows:

$$W(p,a) \leftarrow W(p,a) + \begin{cases} \alpha_+ & \text{if } a \text{ is the action taken} \\ \alpha_- & \text{for all actions not taken} \end{cases},$$

where α_+ is a small positive number (0.005) and α_- is a small negative number (-0.003). The elements of $\mathbf{W}$ are floored at zero and have a maximum value of W^{max} (set to 2.5 here). Also, if p is the goal position for that trial, the value of $W(p,a)$ is decreased by α_-. **C** selects actions via a simple mapping: $\mathbf{a} = f_a(\mathbf{W}\mathbf{p}+y)$, where $\mathbf{p}$ and $\mathbf{a}$ are vectors representing the excitation levels of the *Position* and *Action* neurons, respectively.

The setting of W^{max} to 2.5 and the bistable properties of striatal neurons gives the model a mechanism for "turning off" **C**. If the striatal neurons are in the down state ($y = 0$), the elements of $\mathbf{W}$ are not high enough to select an action. (See discussion in Chapter 2, pg. 17).

Arbitration

I suggest that at states for which the agent has little experience, **B** and **C** are not trained enough to select actions; thus **A** is used at these states. One could implement such an arbitration scheme by keeping count of the number of times the agent has visited each state; once a threshold is reached, **B** is enabled at that state. Similarly, when a higher threshold is reached, **C** is enabled. However, doing so requires some higher level "decision-maker" to explicitly choose which controller to use at each state. (Such a decision-maker would likely not be considered a high level cognitive process, but rather a tool to serve as a place holder for some other arbitration method not explicitly modeled.)

Rather than a higher level decision-maker, the arbitration scheme emerges from network architecture, network dynamics, and the WTA of the *Decision* neuron array. Because the **C** bypasses the *Decision* neuron array, and excites *Action* neurons directly, it will select an action first if $W(p,a)$ is strong enough. If not, **B** excites the *Decision* neuron array through $\widetilde{\mathbf{Q}}$. If the *Decision* neurons are not excited enough so that some *Action* neuron is not excited beyond θ within the time limit, only then does **A** select an action. It is assumed that, to perform the necessary computations, **A** takes longer than the time limit of the WTA.

Thus, incorporated in the model design is the assumption that the simplest controller selects an action fastest, provided it is trained enough to do so. The arbitration scheme is summarized as follows:

1. **C** attempts to select an action.

2. If no *Action* neuron is excited enough to implement the action, **B** is used.

3. If no *Action* neuron is excited enough to implement the action, **A** is used.

Further Details

Initialization of the Q-table

Most of the Q-values are initialized to zero. Those corresponding to a goal (i.e., where p is the position of goal g) are given a value of positive value, representing a reward for reaching a goal. This reward is denoted R_g and is discussed in detail in the next section.. Because a trial is terminated when the goal is reached, these values do not change. The choice of R_g may result in a *pessimistic initialization* in that the initial Q-values are less than their accurate values. Thus, as **A** selects action a from position p in order to reach goal g, **B** will learn to place a higher value on $Q(p, g, a)$ than that of the other actions. When **B** is trained enough to select actions, it will be biased to choose actions selected by **A** when it is first engaged. In contrast, with *optimistic initialization*, where Q-values are initialized to be more than their likely accurate values, Q-values will only decrease with experience and **B** will be biased to choose actions *not* selected previously. If performance while learning is not a factor, and there is plenty of time to explore all possibilities, optimistic initialization has advantages as it encourages exploration early in learning. However, these qualifiers are seldom met. Thus, for the first set of results presented later, I use a pessimistic initialization for the first set of results I present in the next section.

Why the Q-table is not used directly

$\widetilde{\mathbf{Q}}$ is trained by Ψ, a soft-max-like function of the Q-values. The Q-values are not used directly because they can potentially vary across a large range, include both positive and negative numbers, and will change drastically depending on the task and size of the environment. The Q-values capture experience (especially with a pessimistic initialization), but the weights within the WTA network would have to be tuned carefully. Ψ transforms the Q-values into values between 0 and 1, but, by definition, they are normalized across the actions — experience is not represented. Thus, I use Ψ to train $\widetilde{\mathbf{Q}}$, which represents experience with values between 0 and 1.

4.5 Development

As a reminder, the agent's task is to move from the starting position to one of three goals (chosen randomly at each trial) while maximizing reward (see Figure 4.2). The model was used to accomplish the task for 6000 trials over 20 independent runs. A trial ends when the agent reaches the chosen goal or if the agent has taken 1680 ($||P|| \times ||A||$) steps without reaching the goal (because of **A**, this rarely happens). In the first set of results, the Q-values corresponding to goal positions are $R_g = 100$, resulting in pessimistic initialization. Periodically, three "test" trials — one for each goal — are performed sequentially. All exploration and learning parameters

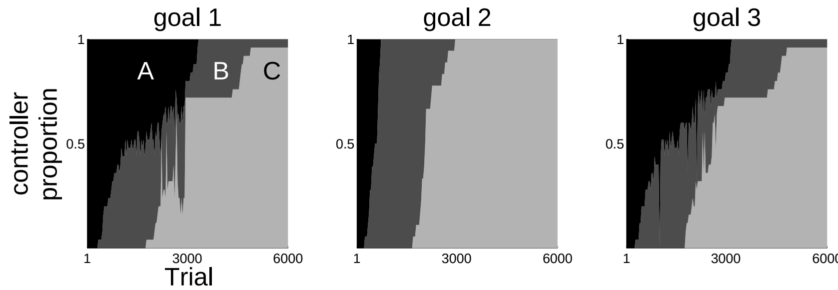

Figure 4.4. Proportion of actions chosen by each control for a given trial, presented in a manner similar to a stacked-bar graph (thus, each "bar" is of height 1). Black, **A**; dark grey, **B**; light grey, **C**. As training (trials) increases, control is shifted from **A** to **B** to, where appropriate, **C**.

are set to zero ("freezing" the system) and the agent's behavior is recorded. The graphs illustrating behavior are taken from these test trials. 14 out of the 20 runs displayed the behavior similar to the behavior to be described. Thus, presentation and discussion is restricted to just one run. Later, I discuss other types of behavior.

In the graphs to follow, selected actions are illustrated with an arrow pointing in the direction of movement. Actions specified by **A** are indicated by wedges (> for east), actions specified by **B** are indicated by a double arrow (⇒), and actions specified by **C** are indicated by a single arrow (→).

Progression of controller recruitment

Figure 4.4 illustrates, in a manner similar to stacked bar graphs, the proportion of actions selected by each controller the agent uses to reach each of the three goals as a function of trial for the sample run. In accordance with model design, **A** (black) dominates control during early trials, **B** (medium grey) selects a large portion of the actions during middle trials, and **C** (light grey) dominates control during later trials. The transfer of control from **A** to **B** to **C** occurs much quicker for goal 2; this result is unsurprising as there is only one optimal path from the starting position to goal 2, while there are many for goals 1 and 3. Therefore, **B** and **C** are trained quickly for goal 2.

Figure 4.5 illustrates model behavior at different trials for goal 3 (the bottom goal, only the bottom portion of the grid is displayed for visual brevity; behavior for the first goal follows a similar progression). During early trials (e.g., trial 10), because neither **B** nor **C** have been trained enough to select actions, **A** selects actions at all positions visited. **B** begins to select some actions at positions for which it has some experience (trials 633 and 1683, see highlighted regions). However, at positions for which it has little experience, **A** still selects actions (trial 993). As the agent gains more experience, **C** begins to select some actions at positions for which the same action is frequently selected by **B** (trial 1863, highlighted regions).

The effect of actions selected via **C** on other actions is profound. For example, at trial 2463, **A** (at the highlighted position) selects action S. The result is a few

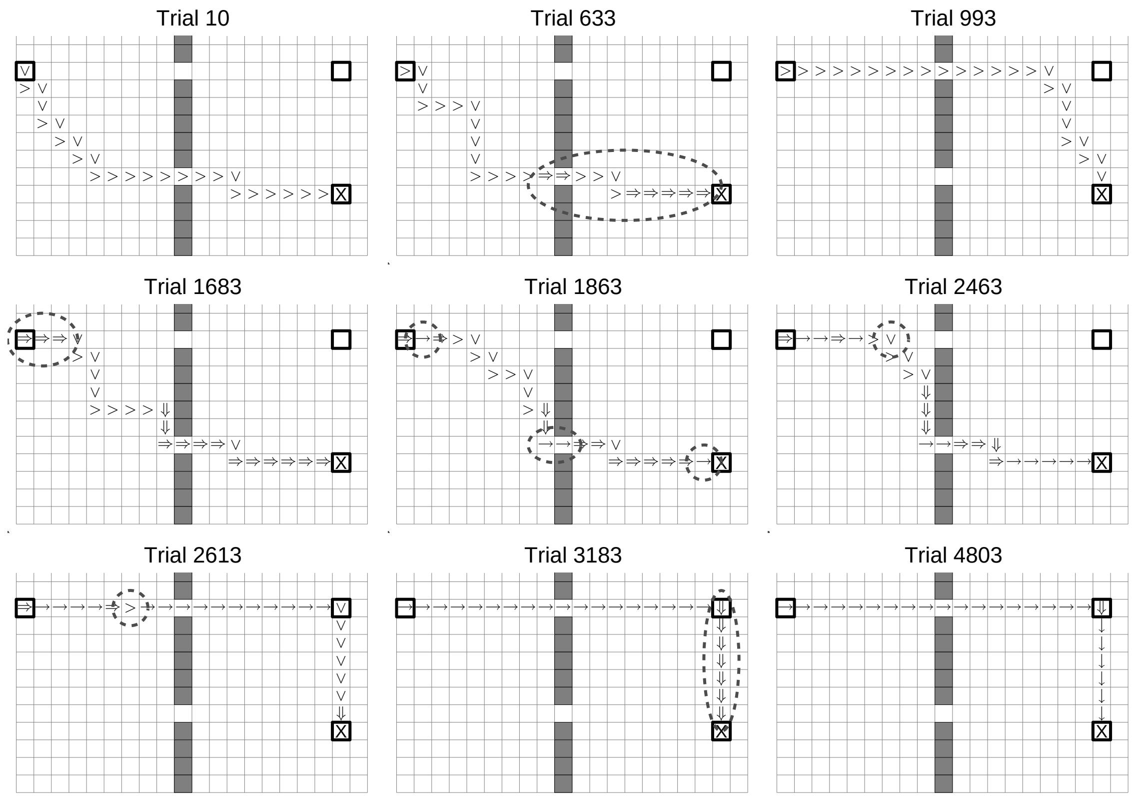

Figure 4.5. Examples, at different points in learning, of actions selected en route to goal 3. The controller that selected an action is indicated by the symbol used to represent that action: **A**, $>$ (east); **B**, $\Rightarrow$; **C** $\rightarrow$. Starting position and positions of goals 2 and 3 are highlighted by squares drawn with thick lines. Positions referred to in the text are highlighted with dark grey dashed ellipses. Only the lower portion of the grid is displayed for visual brevity.

“zig-zags” toward goal 3, using the lower doorway. However, at trial 2613, **A** selects E at the same position. Due to **C** selecting actions all the way to goal 2, the effect of E at the highlighted position is to move directly to goal 2. The single difference in decision (actions S and E) at same position results in radically different behavior towards goal 3.

By trial 3183, **C** is used to select action E at every position from the starting position to goal 2. This is because this path is along the optimal path to all three goals; hence, those positions are visited more frequently than most others and action E is chosen more frequently from those positions than other actions. At the goal 2 position, **B** is used to select either N (for goal 1) or S (for goal 3). At this point in learning, the agent has little experience with positions along the path from goal 2 to goal 3. At trial 2613, **A** is used to select actions; at trial 3183, **B** is used; finally, at trial 4803, **C** is used. The lack of experience the agent has at these positions is because it is very unlikely the agent will visit these positions by chance while maneuvering to goal 3. At almost every position in the environment, there are two equally optimal actions toward goal 3: E and S. For the agent to maneuver to goal 3 via goal 2, it would have to select action E at every position to goal 2. The recruitment of **C** to reach goal 2 forces the agent to use this path eventually.

Chunks

If **C** is trained enough to select actions at a contiguous sequence of positions (e.g., from the starting position to the second goal position), that sequence of actions is considered to be one “habit” or automatic sequence of actions. I refer to a contiguous sequence of actions chosen by **C** as a *chunk* (Graybiel 1998 used this term in a neuroscience paper on the basal ganglia, but it has a long history of use in the psychology literature). Figure 4.6 displays the chunks developed in this run. The central chunk, from the starting position to the position of the goal 2, is common along an optimal path to all three goals, supporting hypothesis 1. The other chunks were developed and used primarily in service to goals 1 or 3, but most could be used for all three goals if the agent happened to be in a position to use them (e.g., if the agent was in the lower half of the grid but trying to move toward goals 1 or 2). The only chunks that could not be used for all three goals are the two emanating from goal 2.

Figure 4.7 displays, by color, when **C** was recruited to select an action at each position (compare to Figure 4.6, positions at which **C** was not recruited are not marked). The lighter the color, the later the trial at which **C** was recruited. The earliest recorded trial at which **C** was recruited was at trial 1681 (labeled in Figure 4.7), at the center doorway position. The latest recorded trial was at trial 4891 (labeled), at the position just north of goal 2.

The general pattern of when **C** is recruited is similar to the pattern of when the relative Q-values for a particular state are accurate in RL. For a chunk leading to a goal, **C** is recruited at positions near the goal earlier than at positions further from the goal. **C** is also recruited earlier at doorway positions in the environment I use. Finally, the late learning of **C** at positions along the path from goal 2 to the other

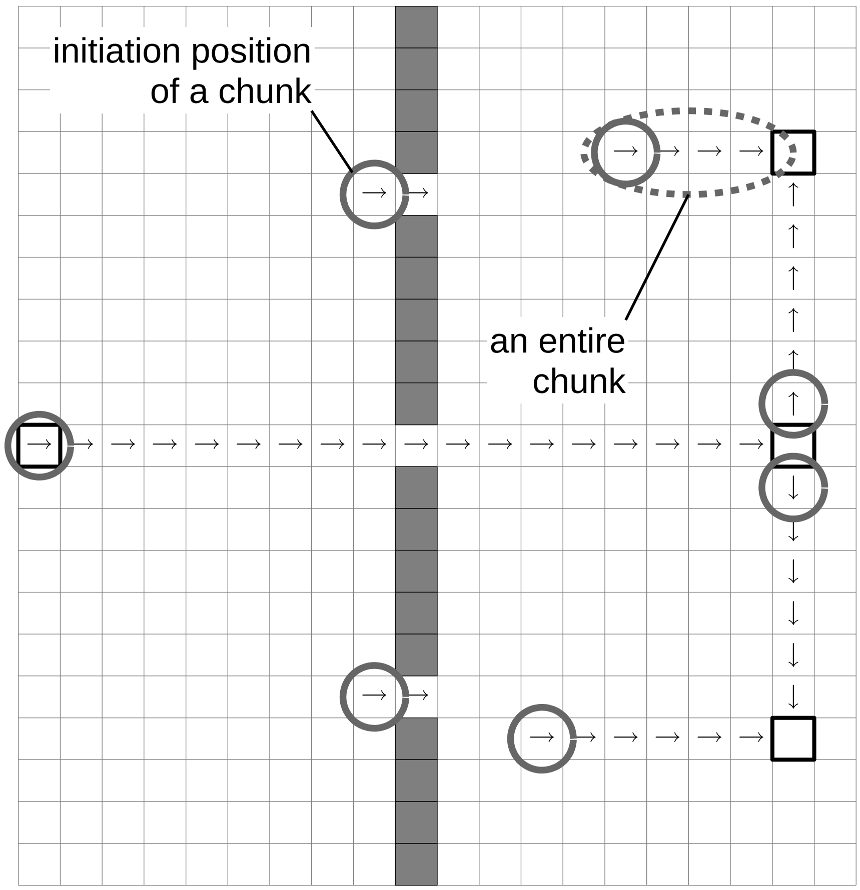

Figure 4.6. Chunks (contiguous sequence of actions selected by **C**) from a typical run. Initiation positions are highlighted by a dark grey open circle. An example of an entire chunk is indicated with a dashed ellipse (upper right). The overall pattern of chunks is an example of a type 1 chunk (see text, page 75).

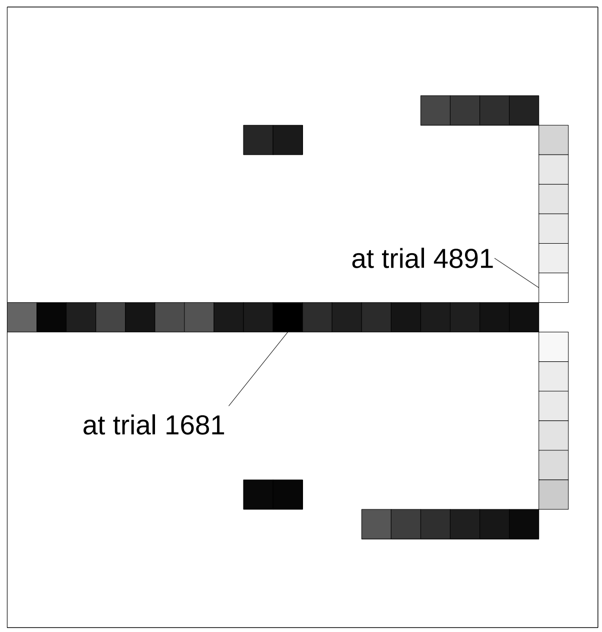

Figure 4.7. Points during learning that **C** was trained enough to select an action (compare to Figure 4.6) at different positions. Positions at which **C** was trained early are indicated by darker squares; those at which **C** was trained late are indicated by lighter squares. The earliest trial at which **C** was trained occurred at trial 1681; the latest at trial 4891 (labeled in the figure).

two goals is very clear in Figure 4.7. This model predicts that, early in learning, the agent will move towards each goal along routes that may be suboptimal for other goals (e.g., by taking action S from the starting positions). Only later in learning, as chunks are developed (influenced in part by moving to other goals), will the agent change strategy and use those chunks. Such changes in behavior may be drastic.

Effect of reward and exploration parameters on chunk development

$Q(p, g, a)$ is the expected sum of rewards received if the agent selects action a from position p en route to goal g. Because all Q-values are initialized to zero, the choice of R_g, the reward received when a goal is reached, affects how **B** explores different actions when it is trained enough to select actions (as discussed earlier, pg. 69). If $R_g = 0$, the accurate Q-value at every position are negative because each selected action is accompanied by a reward of $r = -1$. Thus, Q-values will only decrease with experience. **B** will be biased to choose actions *not* previously selected, a form of *optimistic initialization* because the Q-values are initialized to be greater than their accurate values. If R_g is much greater, the Q-values will be initialized *pessimistically* and thus **B** will be biased to follow select actions chosen by **A**. Because the exploration mechanism my model employs is similar to a soft-max selection, the more pessimistic the initial Q-values are, the less likely the agent will explore actions other than the ones with the highest Q-value.

Exploration may have an effect on chunk development. In my model, exploration is affected by three parameters:

1. R_g, as previously discussed
2. τ, the temperature of the soft-max (see equation 4.2)
3. σ, the width of the Gaussian noise applied to the *Decision* neurons (see equation 4.5)

To assess how chunk development is affected by exploration, the model was trained over the task for 6000 trials for 20 runs with each combination of the following parameter values: $R_g : 0, 25, 50, 100, 150$; $\tau : 1, 5, 10$; and $\sigma : 0.075, 0.15$. For each run, overall chunk development fell under one of three categories:

type 1 chunks: En route to a goal, **C** was used to select an action at every position except for the position of goal 2, at which point **B** was used for goals 1 and 3. **B** is never used for goal 2 as the first chunk leads directly to it. These are depicted in Figure 4.6.

type 2 chunks: En route to a goal, **C** was used to select an action at every position except for one, at which point **B** was used. For type 2 chunks, **B** was used even for goal 2. An example of type 2 chunks for $R_g = +100$ is illustrated in Figure 4.8, top left.

type 3 chunks: **B** was used to select an action at more than one position en route to a goal. An example of type 3 chunks for $R_g = +100$ is illustrated in Figure 4.8, top right.

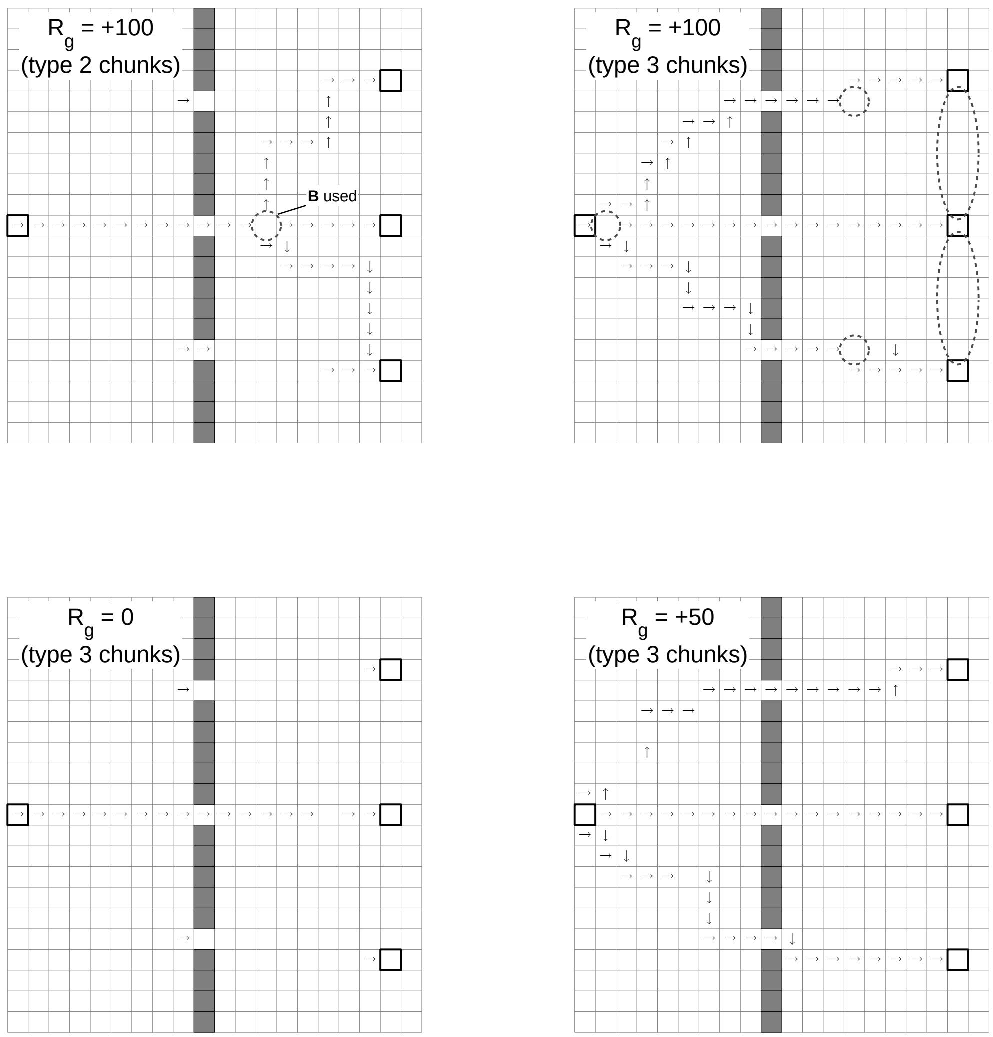

Figure 4.8. Examples of types 2 and 3 chunks. See text. Dashed circles and ellipses indicate positions at which **B** must be used, referred to in the text.

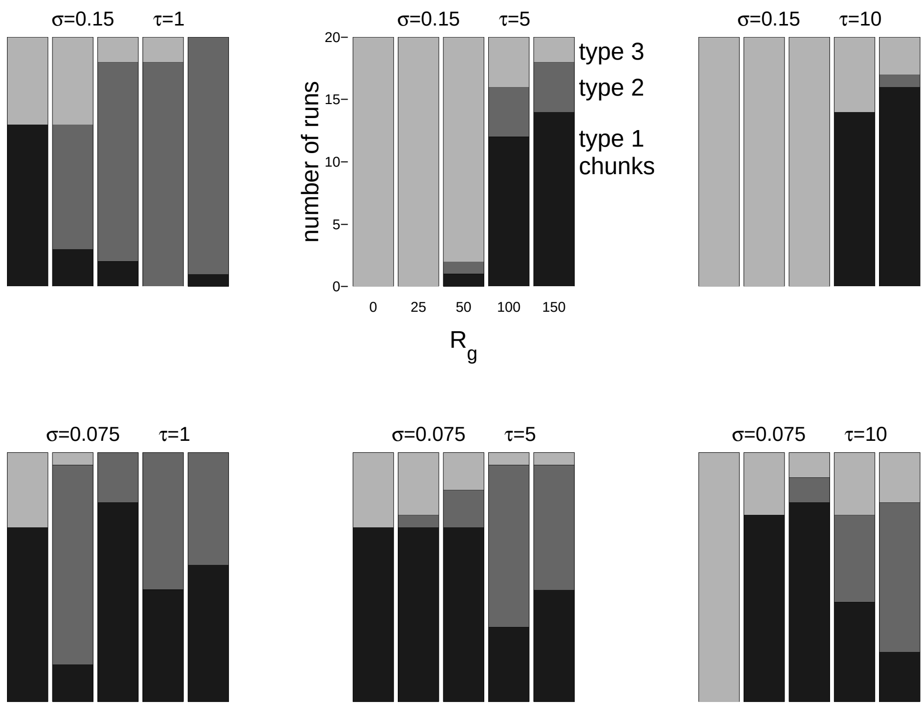

Figure 4.9. Stacked bar graphs showing the number of runs (out of 20) under each combination of R_g, τ, and σ (see text) that produced chunks of types 1 (black), 2 (dark grey) and 3 (light grey).

Figure 4.9 shows, as stacked bar graphs, the number of runs that used each type of chunk for each parameter combination after training for 6000 trials. Bars corresponding to a particular combination of τ and σ are grouped together (organized by R_g). Bars for type 1 runs are drawn in black; dark grey for type 2; light grey for type 3.

The top center group in Figure 4.9 corresponds to $\tau = 5$ and $\sigma = 0.15$, used for the previous results, and is annotated. For $R_g = 0, +25$, and $+50$, most of the chunks developed were of type 3. Examples of type 3 chunks for $R_g = 0$ and $+50$ are illustrated in Figure 4.8, bottom. For $R_g = +100$ and $+150$, most of the chunks developed were of type 1 and some were of types 2 and 3. As R_g increased, so did the likelihood that type 1 chunks will be developed within 6000 trials. If the model was allowed to run for longer trials, models trained with a low reward eventually develop type 1 chunks (not shown). A similar relationship to R_g is seen for $\tau = 10$ and $\sigma = 0.15$.

For lower levels of τ and σ, though, the relationship to R_g is different. In none of the other four combinations of τ and σ did an R_g of $+100$ or $+150$ result in more type 1 chunks than lower values of R_g. High values of R_g were accompanied with a large proportion of type 2 chunks; hence, they were still accompanied with a small proportion of type 1 chunks.

A general trend can be gleaned. When exploration early in learning is high (due to low values of R_g), type 1 chunks are more likely to be developed for low values of trial-independent exploration parameters (σ and τ) within 6000 trials. For high trial-independent parameters, type 1 chunks will eventually be developed (if learning progressed for more trials). When exploration early in learning is discouraged (due to high values of R_g), trial-independent exploration parameters must be high for type 1 chunks to be developed. For low parameter values, type 2 chunks are more likely to be developed. This suggests that **C** is trained too quickly for type 1 chunks to be developed.

Because **C** is used for the greatest proportion of action selection in type 1 chunks, I chose parameters that would produce type 1 chunks most of the time for the simulations presented previously. I chose $R_g = +100$ because I felt that **B** should be biased to select actions chosen by **A** when it is trained enough to do so. Higher values of R_g did not increase the proportion of type 1 chunks by much. I chose $\tau = 5$, as opposed to $\tau = 10$, for the same reason. In addition, very high values might overshadow the effects of other parameters and mechanisms. I chose $\sigma = 0.15$ because, simply, the lower value produced much fewer type 1 chunks.

Finally, although the biological equivalent of τ and σ are difficult to determine, the magnitude of the reward R_g for reaching a goal can be manipulated to some degree in experimental paradigms. This analysis shows that the value of R_g has a large effect on the types of chunks developed for all levels of τ and σ tested.

4.6 Chunk Use

Figure 4.6 shows the chunks developed after 6000 trials. Most theories of automatized movements suggest that they are recruited as a single integrated sequence of actions (Graybiel, 1998; Keele et al., 1995; Koch and Hoffman, 2000; Smith, 1999); the automatized movement can only be recruited at the beginning of the sequence. The bistable properties of striatal neurons provide a mechanism for this constraint (see Chapter 2, pg. 17). Effectiveness of **C**, as I have defined it, depends on *Action* neurons (representing striatal neurons) being in the upstate. If so, weak thalamostriatal projections, representing just position in my model, can elicit an action if the projection is strong enough. However, if the neurons are in a downstate, they are not strong enough and higher controllers, **B** or **A**, must select actions. Thus, **C** is effectively turned off when *Action* neurons are in a downstate. Recruitment of a chunk occurs when the stimulus for which the beginning of the chunk is trained is recognized, referred to as the *initiation position* (highlighted in Figure 4.6). If *Action* neurons are put in the upstate in the initiation position, the chunk is recruited. At the end of the chunk, the neurons transition to the downstate.

In the next set of experiments, I investigate how previously developed chunks are used. Figure 4.10 illustrates the new task, which deviates from the old one in the following ways:

- The starting position and position of goal 1 are slightly modified (indicated in Figure 4.10). This modification makes the upper doorway the best option for goal 1.

- The agent is pretrained with two chunks (indicated in Figure 4.10). It can only recruit the chunks at their initiation positions (grey circles). **C** is not trained any further.

- The resting activation level (y) of the *Action* neurons is 0. At every position, the agent can choose one of the four cardinal actions. In addition, at positions bordering the initiation positions of the chunks, the agent has another action: transition into the initiation position of the chunk and put the *Action* neurons in the upstate, thus recruiting that chunk. E.g., at the starting position, the agent has *five* actions available to it: N,S,E,W, and S^c, where S^c is an action to move south and put the *Action* neurons in the upstate. At the end of the chunk, the *Action* neurons are returned to the downstate.

- The WTA implementation of **B** is abandoned (described below).

- There is only one goal, but the agent does not know what its position is a priori. Because **A** requires goal position, it is disabled (described below).

The lack of **A** is not a realistic scenario, but it is conceivable that if the agent does not know where the goal is, or if there even is a goal, it would "wander around." Such behavior occurs without **A** and an untrained **B**. In addition, while planning mechanisms are well-developed in humans, they are much less so in other animals. Thus, controllers similar to **B** play a more dominant role in learning and behavior. Finally, most theoretical research on the learning mechanisms used by **B** (in Reinforcement Learning, Sutton and Barto 1998) do not include a planning controller such as **A**. The exclusion of **A** allows for connections to be made between my model and

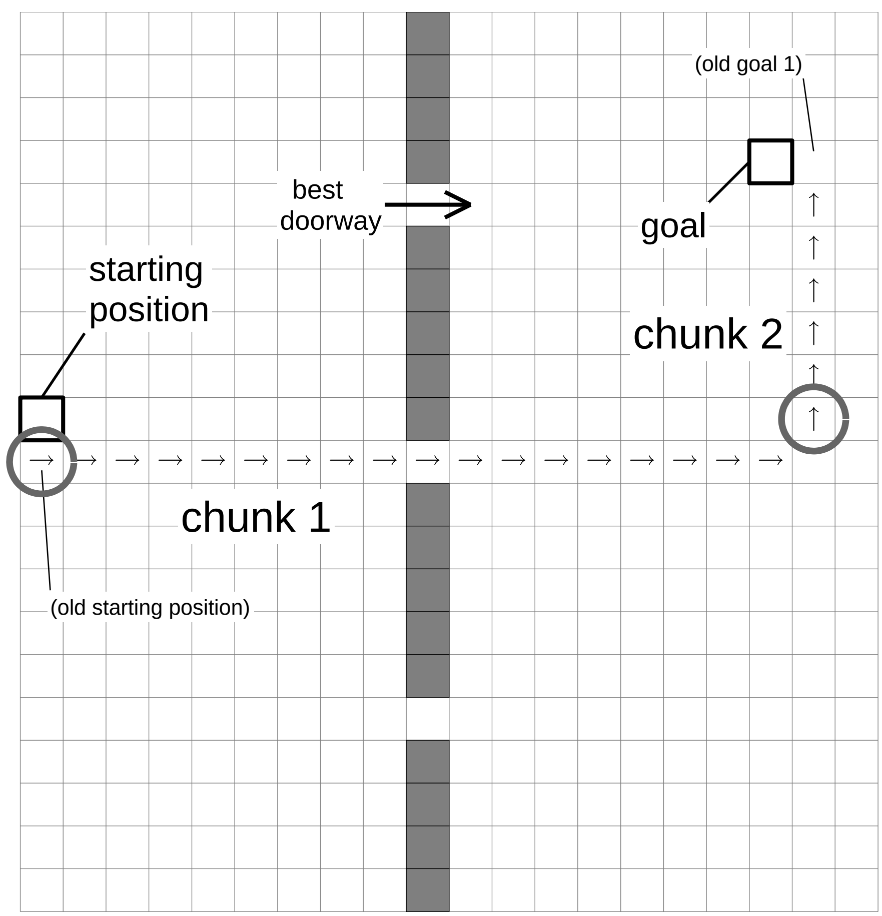

Figure 4.10. Illustration of task used to evaluate chunk use. Chunks 1 and 2 are labeled and can be recruited from their initiation positions (dark grey circles). Note that the starting position and position of the goal are slightly different than those indicated in Figure 4.2.

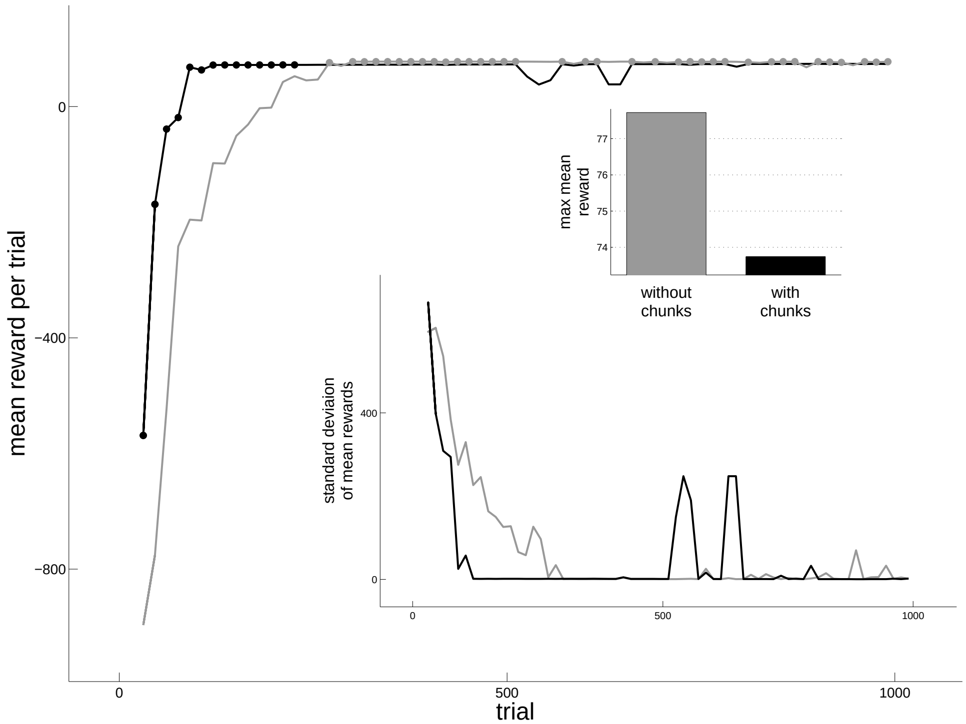

Figure 4.11. Mean rewards for conditions with chunks available (black) and without chunks (grey). Dots indicate points at which one was significantly greater than the other (see text). Insets: standard deviation, maximum mean reward.

some aspects of animal behavior and that of theoretical models; also, it allows us to focus on how chunks may aid in learning a task when no goal information is known beforehand.

To further focus the experiments on chunk use, the WTA implementation of the *Decision* neurons is abandoned. **B** selects actions ϵ-greedily: $(1-\epsilon)$ proportion of the time, the $\text{argmax}_a Q(p,a)$ is chosen (g is not included as there is only one goal), otherwise a random action is chosen ($0 \leq \epsilon \leq 1$ and is 0.1 in these experiments). **Q** is updated as follows:

$$Q(p,a) \leftarrow Q(p,a) + \alpha \ \left(R + \gamma Q(p',a') - Q(p,a)\right),$$

where p is the position at which **B** selects action a, p' is the position at which **B** was next used to select an action (a'), and R is the total cumulative reward received while moving from p to p'. In other words, only the values of actions selected via **B** are updated. The Q-values of the visited positions and actions while a chunk is executed are not updated.

The agent accomplished the new task for 1000 trials. 50 runs each of two conditions were performed: with and without chunks available. In every run in which chunks were available, the agent adopted the strategy of using both chunks en route to the goal, even though such a strategy is suboptimal (the center doorway was used). Every 15 trials, all exploration and learning parameters were set to zero and behavior for that trial was recorded.

Figure 4.11 plots the mean reward (and standard deviation and maximum mean reward) for the two conditions (black: with chunks; grey: without chunks). Points at which the difference between the mean rewards was significantly different (two-tailed unpaired bootstrap test, $p < 0.05$, Cohen 1995) are indicated with small circular markers, colored in with the color of the condition for which the mean reward was higher. During early trials, use of the chunks resulted in much better performance: the agent found a very good route to the goal. However, during later trials, the agent's dependence on the chunks prevented it from finding the better route: through the upper doorway. Thus, the use of chunks aided during early learning — solving the task — but led to suboptimal solutions; these results support hypothesis 2.

Effect of update rules on behavior

Types of update rules

In the previous simulation, Q-values were only updated if **B** was used to select an action. Such an assumption is reasonable; if the chunk is executed as a single contained unit, the actions within the chunk may not be evaluated. However, such an assumption is based on speculation rather than experimental data (of which there is none). To elucidate what, if any, effect on behavior the form of evaluating a chunk and actions chosen by **C** have, I have implemented alternative forms of updates in a modified task.

Figure 4.12 schematizes four update rules for the following generic sequence of states and actions:

- at state s_1, the agent, via **B**, chooses action a_1 and receives an immediate reward r_1. Action a_1 recruits a chunk by moving to state s_2 and placing the *Action* neurons in an upstate.

- The chunk is: $s_2 \rightarrow a_2 \rightarrow s_3 \rightarrow a_3 \rightarrow s_4 \rightarrow s_4$. Rewards of r_2, r_3, and r_4 are received while the chunk is executed.

- The chunk ends at state s_5, from which **B** is used to select action a_5 and the reward r_5 is received.

Thick open arrows represent actions selected via **B**, thin open arrows represent actions selected via **C**, and closed arrows indicate from which $Q(s', a')$ a $Q(s, a)$ is updated. Update rule 1 illustrates the rule that I used in the previous section for the case in which chunks are available; update rule 3 illustrates the rule for which chunks are unavailable (but the same sequence of states and actions are visited).

In update rule 2, $Q(s_1, a_1)$ is updated by $Q(s_5, a_5)$ and the sum of rewards received while transitioning from s_1 to s_5; this is the same as in update rule 1. However, the Q-values for the (s, a)'s visited while the chunk is executed are also updated, according to the next (s, a) visited. In other words, even though the value of the chunk is

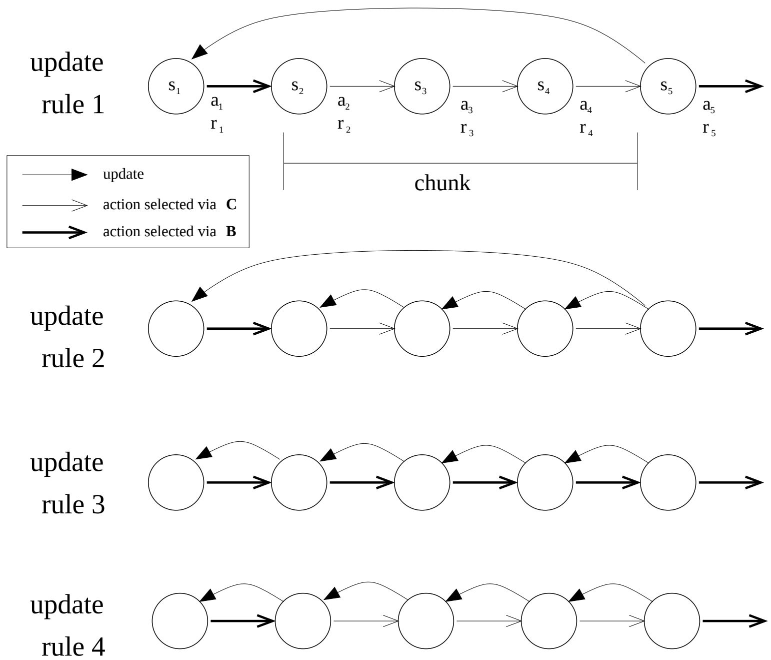

Figure 4.12. Schematics of the four update rules. Each circle represents a state. Thick open arrows: actions selected by **B**. Thin open arrows: actions selected by **C**. Closed arrows: updates.

updated according to the next action selected by **B**, the agent learns the values of the actions executed during the chunk.

In update rule 4, chunks are available, but each $Q(s, a)$ is updated according to the value of the next (s, a) visited, regardless of which controller was used to select the action. The only effect a chunk has is to change the exploratory behavior of the system — no exploratory actions are taken while the chunk is executed.

The value of the chunk (a_1) as updated according to rules 1 and 2 is updated more quickly than that as updated according to rule 4. This is because, under rule 4, $Q(s_1, a_1)$ is updated towards $r_1+Q(s_2, a_2)$, while under rules 1 and 2, $Q(s_1, a_1)$ is updated towards $\sum_{i=1}^{4} r_i+Q(s_5, a_5)$. With rule 4, the values of $Q(s_2, a_2), Q(s_3, a_3), Q(s_4, a_4)$, and $Q(s_5, a_5)$ must be accurate before $Q(s_1, a_1)$ can be accurate; under rules 1 and 2, only $Q(s_5, a_5)$ must be accurate. Hence, behavior under update rules 1 and 2 may include greater chunk usage than behavior under update rule 4.

Results

To assess behavior, the model accomplished the task under each update rule condition for 50 runs each, where each run consisted of 3000 trials. For trials 1 to 1000, chunks were available; for trials 1001 to 2000, chunks were unavailable; for trials 2001 to 3000, chunks were available again. Such an experimental paradigm helps, at least within the confines of this model, to assess the effect on behavior chunks and how they are updated have.

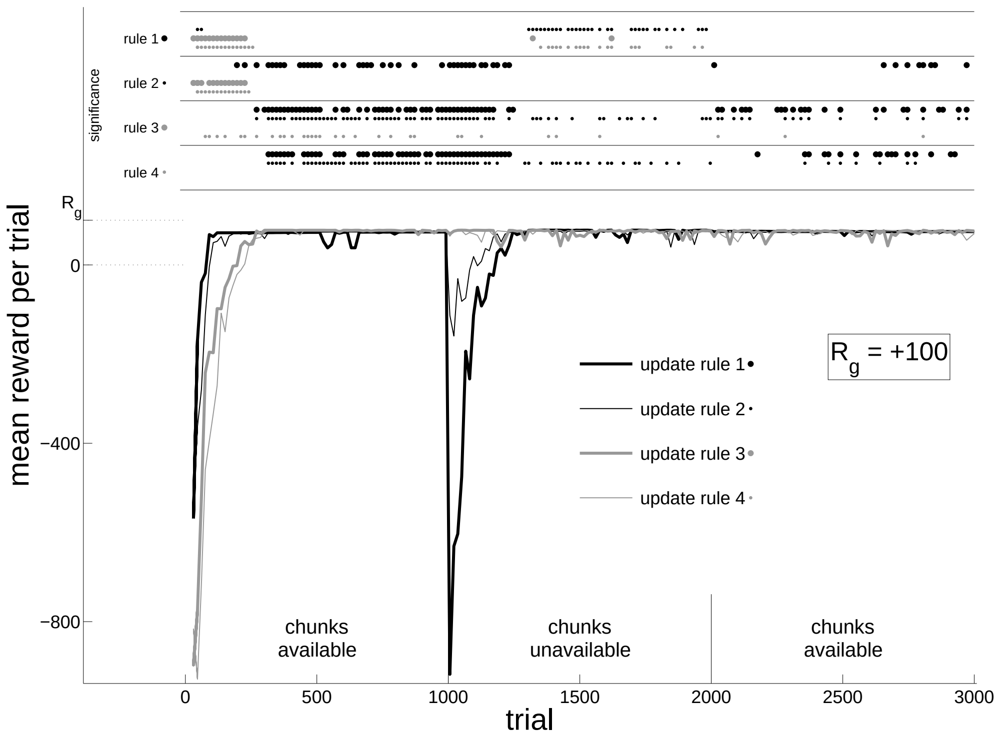

Figure 4.13. Mean rewards under each of the four update rule conditions (similar to Figure 4.11). The top rows of dots indicate at which trials the mean reward under each rule was significantly greater than that under the other rules. If such was the case, the dot corresponding to the other rule was plotted. Update rule 1, thick black line and large black dot; rule 2, thin black line and small black dot; rule 3, thick grey line and large grey dot; rule 4, thin grey line and small grey dot (indicated in legend in lower right). Chunks were unavailable during trials 1001 to 2000.

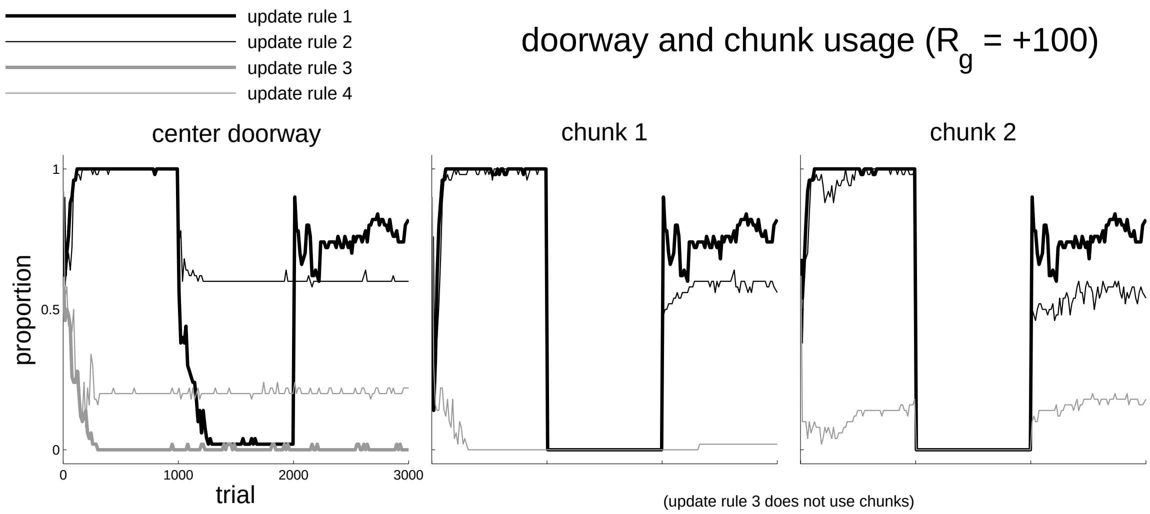

Figure 4.14. Proportion of runs, per trial, under each of the four update rules that used the center doorway (left), used chunk 1 (center), and used chunk 2 (right). Chunks were unavailable during trials 1001 to 2000. That of runs corresponding to a particular rule are labeled by line color and thickness (see legend in upper left; same convention as in Figure 4.13.

Figure 4.13 plots the mean reward for each condition. As in Figure 4.11, the mean reward under update rule 1 is drawn with a thick black line and that under update rule 3 is drawn with a thick grey line. In fact, the mean rewards are the same as that for Figure 4.11 for trials 1 to 1000 and update rules 1 and 3. In addition, the mean reward under update rule 2 is drawn with a thin black line and that under update rule 4 is drawn with a thin grey line. The top part of the figure indicates trials for which the mean reward for a condition was significantly different than that for another condition (two-tailed unpaired bootstrap test, Cohen 1995). Each condition is labeled by a closed circle (as indicated in the figure): large black circle for update rule 1; small black circle for rule 2; large grey circle for rule 3; and small grey circle for rule 4. For each rule, if its mean at a particular trial was greater than the mean for another rule, and the difference between the two was significant, the circle for the other rule is plotted. For example, the mean reward for rule 1 is significantly greater than the mean rewards for rules 3 and 4 during the first few hundred trials.

The mean rewards under update rules 1 and 2 were much greater than those for rules 3 and 4 during early trials. However, between trials ≈ 300 (where "$\approx$" indicates "approximately") and 1000, the mean rewards under rules 3 and 4 were greater. If we assume that behavior under rules 1 and 2 are more likely to recruit chunks, these results show that, although chunks helped in early performance, they led to a suboptimal strategy (as in Figure 4.11).

While further behavioral characteristics can be inferred from studying the mean rewards, it is easier to directly observe such behavior. Figure 4.14 plots, as a function of trial, the proportion of runs under each condition that used the center doorway (left), chunk 1 (middle) and chunk 2 (right). (Aside from possibly early trials, the

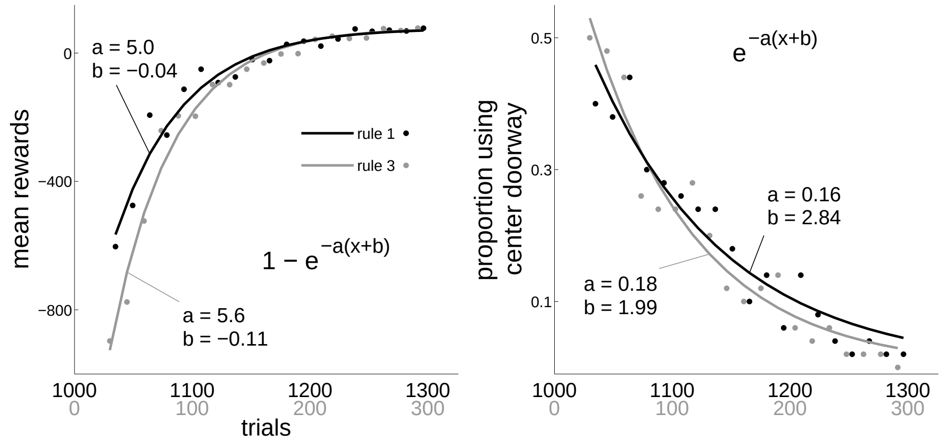

Figure 4.15. Comparison of mean rewards (left) and proportion of runs using the center doorway (right) under rule 1 during trials 1036 to 1306 (black) with that under rule 3 during trials 31 to 301 (grey, color of the trial number labels at bottom of graph correspond to the two rules). Mean rewards and behavior during the first 30 trials of each range were erratic and thus discarded. Dots, actual mean rewards and proportions; lines, best-fit curves of equations $y = 1 - e^{-a(x-t)}$ (mean rewards) and $y = e^{a(x+b)}$ (proportion) where x is trial. Because the fitting process was done to compare the shapes of the curves (quantified by the parameters a and b) under the two rules, mean trial numbers were transformed to be between 0 and 1; mean rewards were also transformed to be between 0 and 1 (they were scaled by the same amount). The fitting process ran for $1,000,000$ iterations and the sum of errors for each curve was < 0.035. Before plotting, the best-fit curves for the mean rewards were transformed so as to be of the same scale as the actual mean rewards. These graphs show that proportion using center doorway and increase in reward for the two rules are similar.

proportion of runs that used the upper doorway was $1-$ the proportion that used the center doorway.) The inference that behavior under rules 1 and 2 used chunks is confirmed by Figure 4.14 as almost all of the runs under these conditions used the center doorway and both chunks by trial 1000. Less than 1/4 of the runs under rule 4 used the center doorway, none used chunk 1, and less than 1/4 used chunk 2. Interestingly, although the agent under rule 4 did not use chunk 1 for any of the runs by trial 1000, about 1/4 of the runs did for the first few dozen trials. This likely led to the agent using the center doorway for some runs.

When chunks were no longer available (after trial 1000), performance under rules 1 and 2 suffered, though to different degrees. Mean reward under rule 1 dipped to early learning levels; the increase in mean reward (and behavior) under rule 1 for trials ≈ 1000 to 1300 was similar to that of rule 3 for trials ≈ 0 to 300 (Figure 4.15). This suggests that, under rule 1, the experience the agent gained for 1000 trials using the chunks had almost no effect when chunks were unavailable. The agent had to start learning from scratch, which is not surprising as the values for the vast majority of the actions the agent took were not updated under rule 1.

Mean reward under rule 2 just after trial 1000, on the other hand, did not dip as drastically as that of rule 1, likely because the Q-values of the actions taken while chunk 1 was executed were updated; the experience the agent gained while executing chunk 1 did allow it to find a path toward the goal fairly quickly after chunks were unavailable. However, that strategy is suboptimal. Where as behavior under update rule 1 eventually (by trial 1300) abandoned the center doorway, over half of the runs under update rule 2 continued to use the center doorway. The difference in strategy is reflected by the mean rewards: the mean reward for rule 1 was greater than that of rule 2 for trials $\approx$ 1300 to 2000, as were the mean rewards for rules 3 and 4. Where as the agent under rule 1 was able to find the better route (through the upper doorway), the agent under rule 2 was not for most runs.

Finally, the removal of chunks during trials 1001 to 2000 had an effect on behavior after trial 2000, when chunks were available again. The proportion of runs under rule 2 that used the center doorway remained the same (over half), but about 3/4 of the runs under rule 1 reverted to using the center doorway again, (using both chunks). The same cannot be said for behavior under rule 2, as the proportion using the center doorway, chunk 1, and chunk 2 did not match on a per trial basis as it did for rule 1. The decrease in proportion of runs using the chunks and center doorway for rule 1 is reflected in the mean rewards: compared to the latter 2/3 of the first 1000 trials, there are less trials during the latter 2/3 of the last 1000 trials for which the mean reward under rule 1 is significantly less than that of the other rules.

The results from this section show that not only does the availability of chunks have an effect on behavior, but behavior changes depending on how the experience gained while executing a chunk is used. In the next set of simulations, I show that the value of R_g also has an effect on behavior.

Effect of R_g on behavior

The simulations just described were run again with various values of R_g: 0, +25, +50, +150. Mean rewards and behavior for the four conditions are shown in Figures 4.16 and 4.17, which follow the same conventions as those in Figures 4.13 and 4.14, respectively. (For visual brevity, the y-axis for the graphs in Figure 4.16 was cutoff at -300 and annotation for both figures was decreased.) In general, as R_g increases, so does the effect of chunks on mean rewards and behavior.

Of particular note, when $R_g = 0$, the availability of chunks had very little effect (top left of Figures 4.16 and 4.17): chunks were not used. As explained in section 4.5 (page 75), as R_g decreases, exploration early in learning is increased. Thus, although the chunks do provide for a reasonable, easy to find, trajectory toward the goal, the values of the chunks are less than the values of actions not yet experienced early in learning. The agent is biased to explore other actions and thus finds the upper doorway.

For higher values of R_g, chunk availability does have an effect. The early increase in mean rewards due to chunks is greater as R_g increases. Figure 4.18 plots, as a function of R_g, the number of trials until mean reward is within 10 of the maximum mean reward for each of the four update rules for the first 1000 trials. The earlier this

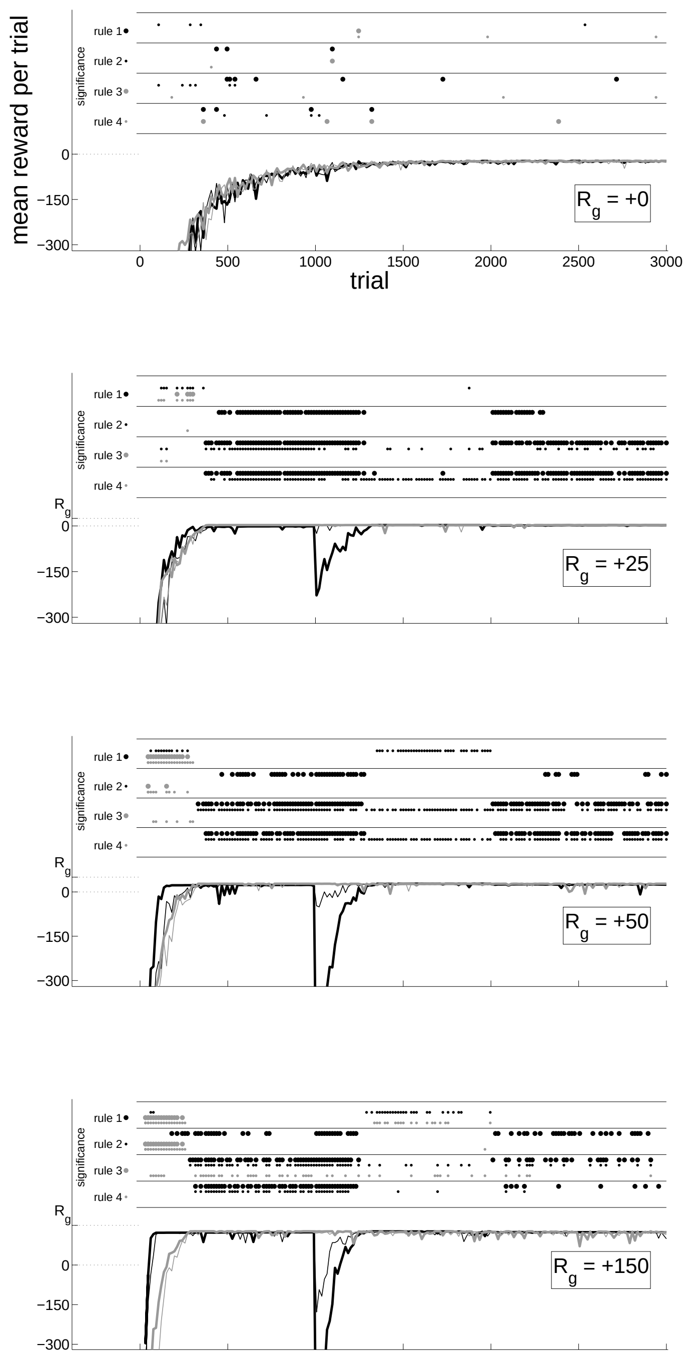

Figure 4.16. Mean rewards under each of the four rules for different values of R_g (indicated in each graph). Follows same conventions as in Figure 4.13. Graphs cutoff at mean reward = −300 for visual brevity.

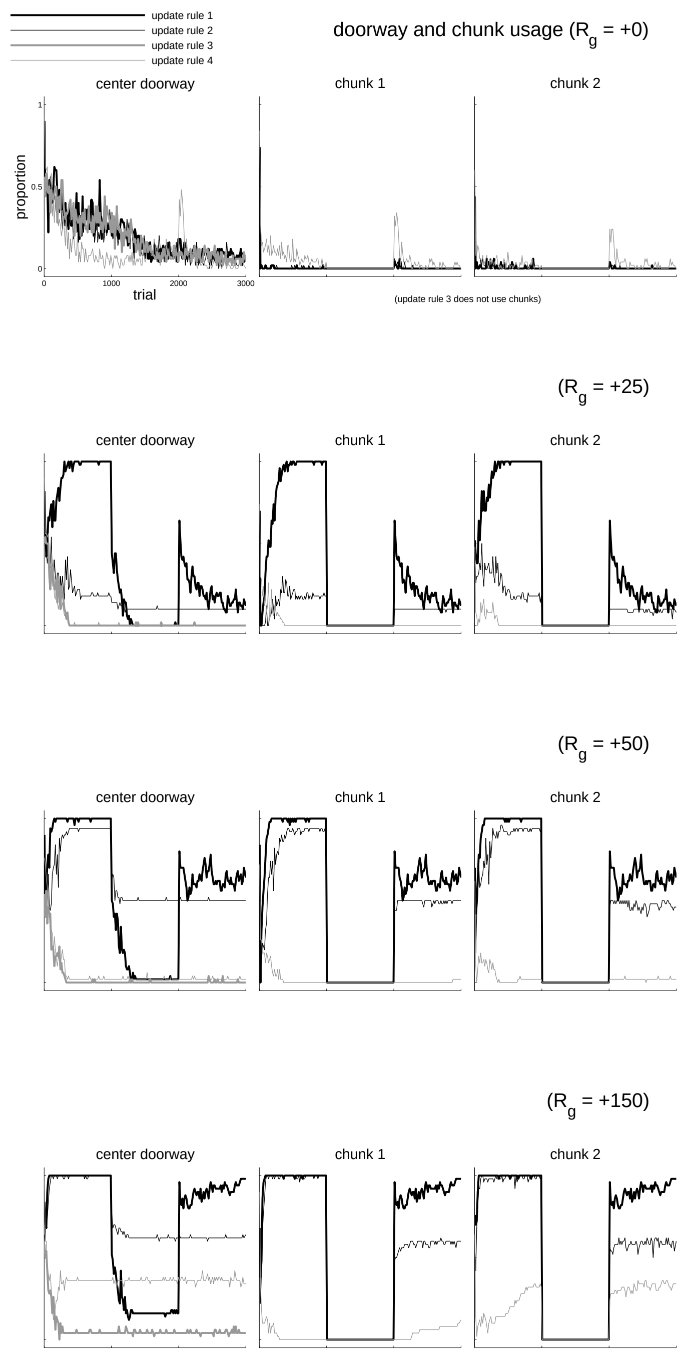

Figure 4.17. Behavior under each of the four rules for different values of R_g (indicated in each graph). Follows same conventions as in Figure 4.14.

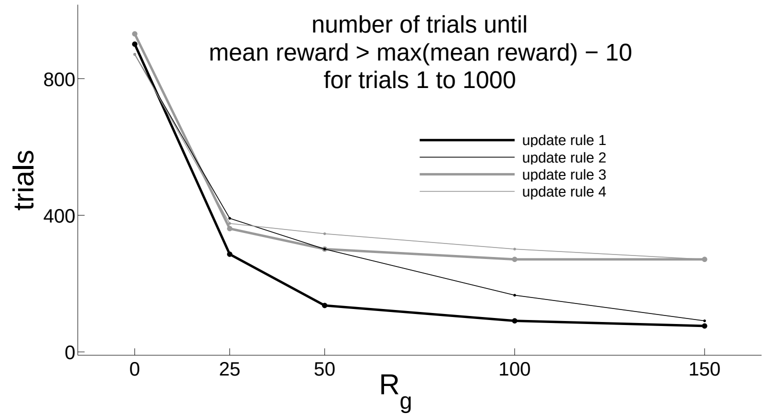

Figure 4.18. Trial at which mean reward is within 10 of the maximum mean reward for each of the four update rules for the first 1000 trials. The color and thickness of the lines correspond to the update rule, labeled and following the same convention as Figure 4.13.

occurs, the higher the rate of increase in mean reward. For update rules 1 and 2, as R_g increases, so does the rate of increase in mean rewards (compare to Figure 4.16). The effect was greater for rule 1 than for rule 2. For rules 3 and 4, the rate does not increase appreciably as R_g increases, suggesting that the value of R_g has little, if any, effect.

As with chunk development, gross behavioral changes occur when R_g is manipulated and chunks are available for recruitment. Such manipulation may be possible within experimental paradigms. In particular, one can infer if chunks are used and how experience gained while executing a chunk is used by observing the behavioral changes.

4.7 Discussion

Automatic movements are executed quickly and with little thought or attention. They display the following behavioral characteristics: an increase in speed (relative to non-automatic movements), an interdependency between actions, and an independency from goal. However, the underlying mechanisms of their development are difficult to ascertain. Are the changes in behavior due to subtle changes in movement parameters, as in coarticulation, or changes in the decision-making process — selecting actions? One advantage of computational models is that a specific theory can be implemented and resulting behavior can be analyzed and compared to experimental data.

In this chapter, I examined the decision-making aspect of automaticity and presented a multiple controller model that does account for some of the behavioral characteristics of automatic movements. The model implements the following scheme: early in learning a task, the *Planner*, **A**, which requires high computational and representational resources, selects actions based on planning. As the task is repeated, the *Value-based* controller, **B**, learns to place a high value on actions selected by **A**. **B** has lower computational and representational requirements as it selects actions by comparing the values of the available actions. Because of the lower requirements, **B** is assumed to select actions faster than **A** if it is trained enough to do so; thus, control is transferred. Finally, the *Automatic* controller, **C**, which has the least computational and representational requirements, caches the actions chosen frequently by **B** and selects actions faster than **B**. Actions chosen by **C** represent automatic behavior. I refer to a contiguous sequence of actions chosen by **C** as a chunk.

I examined two behavioral aspects of chunks: 1) the conditions under which they are developed. One approach I used that has not been studied in great detail in other models or experimental paradigms is the effect of training over more than one task in chunk development. Such training was crucial to types of chunks developed in the model. 2) The behavioral effect of the availability of chunks on solving a task. I also investigated different ways the experience gained while executing a chunk is used and the effects those had on behavior. One of the difficulties in studying automatic behavior is that different theories lead to very similar types of behavior. However, as evidenced by the results presented in this chapter, many of the experimental manipulations had a radical effect on behavior (e.g., the use of the doorways, rate of performance increase). Although the tasks and manipulations are unrealistic as presented in this chapter, they may be able to suggest experimental paradigms to further study the phenomenon of automaticity.

Relation to theoretical models

B uses learning mechanisms based on the algorithms of Reinforcement Learning (RL). Much of the RL literature includes theoretical analysis that may be applied to **B**. Also, the availability of chunks creates an additional hierarchical layer of actions; such hierarchy has also been studied in the RL literature in the form of *options* (Precup et al., 1998; Sutton et al., 1999; Precup, 2000), briefly discussed in the previous chapter. Rather than a "one-step" action, such as north or east, an option is a multi-step action designed to achieve a particular subgoal. For example, if the environment consisted of multiple rooms, an option might be to navigate toward a doorway.

The option is similar to a chunk in that, once learned, the multi-step behavior is recruited as a single entity and aids in learning and performance. The development of an option may also be similar to chunk development. Many chunks (Figure 4.6) were sequences of actions useful for all three goals. Such development is similar to the method of using *diverse density* to develop an option (McGovern and Barto, 2001; McGovern, 2002), in which the subgoal of an option was a state that different (diverse) trajectories visited frequently (dense).

Controllers **B** and **C** were trained in part according to actions selected via another controller, considered a form of *off-policy* learning (e.g., Sutton et al. 1998; Precup et al. 2000; Watkins 1989; Watkins and Dayan 1992). In most other examples of off-policy learning, actions taken while following one policy (e.g., en route to goal 1) are also evaluated in terms of another policy (e.g., en route to goal 2). For example, in *intra-option learning* (Sutton et al., 1998), the values for the option are updated for states and actions visited even if the option itself isn't recruited. Such a scheme can be helpful as any experience gained is used to improve different policies. However, there can be drawbacks under some circumstances. Some of the update rules I examined in section 4.6 (page 82) are similar to intra-option learning (though they're opposite: **B** updates it values according to actions selected by **C**). In those simulations, an on-policy only rule (type 1) was able to find a better strategy when chunks were removed, while an off-policy rule (type 2) was not.

Relation to similar multiple controller models

While some accounts automatic behavior focus on sequence learning (reviewed in section 4.2), other theories also suggest that different types of control mechanisms are used (cf., Logan 1988; Schall 2001). Below I review two models that share some similarities with the multiple controller model presented in this chapter.

Daw et al. (2005) present a computational model (henceforth referred to as the *Daw* model) in which a *Tree-search* controller, similar to **A**, and a *Cached-values* controller, similar to **B**, represent control mechanisms of the prefrontal cortex and striatum, respectively. Arbitration between the two controllers was based on the relative level of uncertainty of each controller — the controller with the least uncertainty selected actions. The uncertainty of their *Tree-search* controller decreased faster but had a higher lower limit than that of their *Cached-values* controller. Thus, similar to my model, the *Tree-search* controller dominated control early in learning, but the *Cached-values* controller dominated later. They showed that their model explained behavior seen in instrumental conditioning tasks with goal-devaluation.

The main difference between the *Daw* model and mine lies in the arbitration scheme. The scheme used in the *Daw* model makes functional sense as the controller with less uncertainty is the controller that better represents the environment and task. Uncertainty must be explicitly computed and a higher level decision-maker uses it to arbitrate between the two controllers. The scheme I use, on the other hand, arises naturally from the assumption that simpler controllers select actions faster than more complicated ones. **B** is able to select actions when it is able to excite *Decision* neurons enough for one of them to win a WTA. No higher level decision-maker is required. However, since only experience is required to train it enough to select actions, it is possible that **B** may make poor decisions when it begins to select actions. The use of **A**, the pessimistic initialization, the slow learning rate of **B**, and the relative simplicity of the task and environment I use (actions have deterministic consequences) prevents this from happening, but there is no inherent mechanism to prevent it under other circumstances. The scheme used in the *Daw* model does prevent their Cached-values controller from taking over inappropriately.

The experience-dependent and uncertainty-dependent arbitration schemes can be combined. According the Sarsa algorithm (equation 4.1, section 4.4), $Q(p, g, a)$ is updated to estimate $r + Q(p', g, a')$. The error between these two terms is referred to as the *temporal difference error*, δ (Sutton, 1988). If $Q(p, g, a)$ is not accurate, δ is high. In my model, the Q-values are used to train $\widetilde{\mathbf{Q}}$ (equation 4.4), with a constant learning rate of α_q. If the learning rate instead incorporated δ, such as $\alpha_q/f(\delta)$ (where

$f(\delta)$ returns the maximum of 1 or $|\delta|$), then Q-values would have to be accurate to some degree before the corresponding values of $\widetilde{\mathbf{Q}}$ could increase enough to allow **B** to select actions. In this chapter, I chose not to include δ as it made little difference in behavior (results not shown) for the task and environment I use. However, I revisit this issue in the next chapter.

Ashby et al. (2007) also attribute different processing capabilities to cortical areas and the basal ganglia, but in their model (referred to here as the *Ashby* model), a cortical pathway (functionally similar to **C**) learns to cache decisions made by the basal ganglia (BG, similar to **B**). (In addition, the *Ashby* model focuses on different types of behavior and tasks than the model I implemented.) In contrast, I attribute **C** to the thalamostriatal pathway (see Chapter 2). The two theories are not mutually exclusive as Hebbian-style learning is thought to mediate synaptic plasticity in many areas of the brain. Despite the differences in interpretation, the functional mechanisms of both the *Ashby* model and my model are similar. One controller learns the values of several actions via Reinforcement Learning and, through some competition mechanism, selects the action with the highest value more often than those of a lower value. The actions selected are cached by a simpler controller which directly selects the action without considering alternatives. In my model, I assume **B** takes longer to select an action than **C** because of the competition between actions. In the *Ashby* model, actions selected via the basal ganglia must traverse several synapses before that action is executed; the cortical pathway bypasses the BG, traverses fewer synapses, and thus executes the action faster.

Problems with simple controllers

Simple control mechanisms, such as that employed by **C** or recurrent networks, can account for many types of sequential behavior. However, many argue that they cannot account for more complicated types of behavior (Lashley, 1951; Balleine and Ostlund, 2007; Cooper and Shallice, 2006; Houghton and Hartley, 1995). (Some arguments use examples from reading, writing, and linguistics to illustrate objections, but the overall arguments often translate to other types of tasks.)

One of the main arguments against a simple control mechanism is that elements in a sequence often have associations with representational features other than immediate sensations. The phoneme used to pronounce a letter depends on the entire word it is in and even the language of origin of the word. The meaning of a word in a sentence often depends on the entire sentence, paragraph, or even higher hierarchical levels of context. The choice of actions for a given task may also depend on the greater context: late at night, one might choose to drive along a main road to go home, but during rush hour, one might choose back roads. These arguments do not argue against a simple control mechanism; rather, they argue against a simple representation. As discussed earlier (section 4.2), neural network architectures can provide for a much richer representation than is usually assumed, suggesting that simpler control mechanisms can generate more complicated behavior than is usually assumed.

Other aspects of some sequential behavior lead to the suggestion that selected actions are the result of a hierarchical control mechanism, in which a controller accomplishes a task by recruiting controllers that accomplish particular subtasks necessary to accomplish the overall task. Those subtasks, in turn, are accomplished by recruiting controllers that accomplish lower levels of subtasks, and so on (e.g., Sutton et al. 1999; Dietterich 2000; Cooper and Shallice 2006; see Botvinick 2008; Barto and Mahadevan 2003 for reviews). Such structure is attractive as it allows a complicated task to be decomposed into simpler parts and thus easier to understand and mimic. In some theories of hierarchical control, the high level controller, sometimes called a *schema* (Arbib, 2002; Cooper and Shallice, 2006), includes abstract features of the task to be executed. For example, the task of typing the word "look" may be implemented by a schema that includes some representation of a repeated letter. Such a schema accounts for the typo "lokk;" similarly, a schema with abstract features may account for the typo "wrapid writing." As further evidence for a hierarchical scheme, Lashley (1951) notes the ability of a bilingual speaker to directly translate a sentence from one language to another, observing the proper grammar and idiosyncrasies of each language. The thought communicated by the sentence, not the order of words, is the schema that dictates the structure of the sentence.

Hierarchical control is used in the multiple controller model presented in this chapter in the form of chunks. However, while we likely use hierarchical control in solving tasks, a single-level controller can account for some behavior normally attributed to hierarchical control. The *Botvinick* model (section 4.2, Botvinick and Plaut 2004, 2006) does not incorporate hierarchy, but the behavior it produces can be described in hierarchical terms. In addition, hierarchical representations may be difficult to map onto brain architecture (but see Botvinick 2008), while the link between neural network models and brain architecture is more apparent.

Finally, in the implementations presented in this chapter, the actions selected by **C** are "hard-coded." This is unrealistic as most examples of automatic movements can be modified or even abandoned: habits can be "broken" (with substantial effort in some cases). Indeed, when the goal is devalued in an instrumental learning task (Dickinson, 1985; Yin and Knowlton, 2006), the animal quickly learns to select another action. In one sense, such behavior can be explained by a controller similar to **B**, as **B** modifies Q-values while controlling behavior. However, in the case of goal devaluation, behavior changes very quickly. Such behavior may be better explained by recognizing that the action selected by **C** or **B** is no longer rewarding and "shifting control up" to **A**, which explicitly predicts the consequences of each action. Also, although I do not explore it in this thesis, the bistable properties of the *Action* neurons, which **C** excites directly, allows chunks to be modified. Control can be shifted up to **B** (by putting the *Action* neurons in the downstate) and, while **B** selects actions, **W** can be modified.

Summary

As evidenced by the discussions in this chapter, it is difficult to define what automatic behavior is and to prove (or disprove) its existence. Behavioral characteristics ascribed to automaticity can be generated by different control schemes (e.g., hierarchical versus non-hierarchical). Also, there is controversy over what behavior is considered automatic. Some suggest that behavior explained by controller like **B** is automatic (Daw et al., 2005; Yin and Knowlton, 2006), while others suggest that behavior explained by a controller like **C** is automatic (the model I present in this chapter, Ashby et al. 2007; Logan 1988).

Rather than base my theories purely on behavioral and physiological experiments, which at this point are inconclusive, I added another another dimension: functional advantage. In particular, I suggest that the simplest possible control scheme is used to select actions as it uses less computational and representational resources. In this chapter, I investigated under what circumstances simpler controllers are developed and recruited. I also provided examples of how behavior might change in measurable ways if the brain does use a multiple controller scheme similar to that presented in this chapter.

CHAPTER 5

SENSORY EXPLOITATION

5.1 Sensation, Perception, and State

According to the formalization of motor skills I used in this thesis, actions are selected based on state. In the previous two chapters state was delivered instantaneously with certainty (an agent knew exactly in which state it was at any time). However, state is a mental construct, a convenient interpretation of physical stimuli (e.g., electromagnetic waves or changes in air pressure) meant to represent aspects of the environment relevant to the current problem from which we can make decisions. Specialized cells (sensory receptors, e.g., rods and cones of the eye or cochlea of the ear) detect those physical stimuli, a process termed *sensation*. Those detections are communicated to other areas of the nervous system (e.g., visual or auditory cortex), which interpret them, and from there to even more areas (e.g., association cortices), which combine those interpretations. The separate process of interpreting sensations is called *perception*. Perception, along with memory of past perceptions, can lead to a representation of state.

The process by which state is estimated is (highly) schematized in Figure 5.1. Although a general review of sensation and perception is beyond the scope of this thesis, I highlight here some relevant considerations. First, a sensation can be of one of several modalities, the so-called "five" senses: vision, somatic, auditory, taste, and smell, though most researchers include balance as a sixth. Second, the type of physical stimuli available differentially affects the following characteristics of each modality:

- intensity: the magnitude of sensory receptor responses and corresponding perception,
- timing: how fast sensation and perception occurs (also affected by intensity),
- precision and accuracy: how well state can be estimated.

Because of these effects, different modalities (and their combinations) may be better suited for estimating state for different tasks and even different stages of learning a task.

In this chapter, I focus on the general characteristics of intensity, timing, and precision in state estimation. In particular, since sensation and perception take time to process, state estimation evolves over time from imprecise to precise, a process I term *sensory evolution*. Also, in many cases, the set of modalities used to estimate state changes over time; I refer to this process as *sensory transfer*. Below I describe behavioral evidence for both processes.

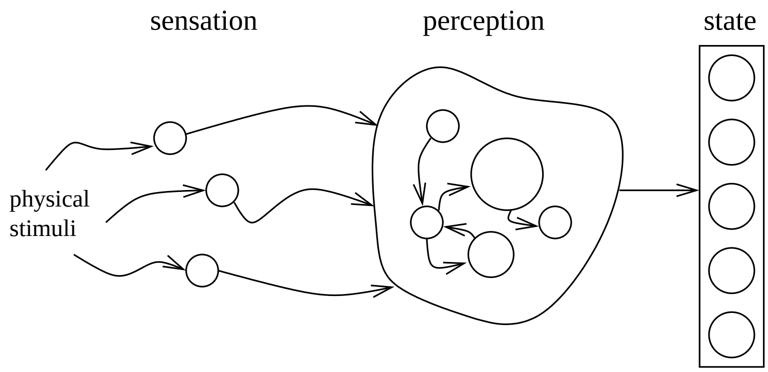

Figure 5.1. Schematic of how state is constructed from sensation and perception. The circles under "sensation" represent sensory receptors; the amorphous shape and circles under "perception" represent nervous system structures involved in perception; the circles under "state" represent possible states.

Sensory evolution

In most accounts of skill acquisition, it is assumed that the sensory information indicating the goal of a task is known a priori. However, it may take time to process sensory information to determine what the actual goal is with enough confidence to make a decision, particularly if the learner is trained over several goals. Sensory representation evolves over time. For tasks in which an animal (or human subject) has only one opportunity to achieve the goal, action is often delayed while the representation evolves to some degree of certainty. I refer to these types of tasks as *one-step tasks*. The most well-known example is the "moving dots task" of Britten et al. (1992) , where a subject (often a non-human primate) is presented with a number of moving dots on a computer screen. Some percentage of the dots move in the same direction (the *goal* direction; as the percentage increases, so does the *coherence* of the stimulus). The subject must look in the goal direction to receive a reward. As coherence decreased, goal direction was harder to ascertain; response time increased and accuracy decreased (a review of several studies involving this task is found in Opris and Bruce 2005). The implication of these results is that as the stimulus is harder perceive, the longer it takes to process it; the results of two other studies have similar implications: Archerfish presented with a low-contrast prey stimulus take longer to initiate movement than archerfish presented with a high-contrast stimulus (Schlegel and Schuster, 2008). Battaglia and Schrater (2007) conduct a goal-directed reaching experiment in which there is an explicit trade-off between perceptual certainty and movement accuracy. The actual goal location is not known, but rather must be estimated by the presence of dots drawn from a distribution centered on it. As time increases, more dots appear, and thus goal location can be estimated with greater certainty. However, movement must be made within a short time period – the cost of the decrease in perceptual uncertainty is a decrease in time allowed to make the movement, resulting in a less accurate one. Battaglia and Schrater (2007) show that the time point at which humans tend to initiate movement conforms with that predicted by statistical decision theory.

If, on the other hand, the task requires longer movements or more than one decision, there are opportunities to adjust the movements or make corrections after responding begins. In these *multi-step tasks*, it may make sense to act quickly even under significant uncertainty. For example, Ledoux (1998) discusses how, when we encounter a snake-like object (such as a stick) while on a walk, we may jump back immediately rather than wait to let our sensory processing better discriminate the object's identity. In the laboratory setting, Hudson et al. (2007) forced subjects to act under a fixed level of uncertainty in a goal-directed reaching task. Subjects began their reaches based on a given probability distribution over all possible goals; after one-third of the distance was traversed, the true goal was revealed. The initial direction of the subjects' movements was towards the mean of the probability distribution and then veered to the goal. This strategy shows that subjects make decisions that take the evolving sensory representation into account rather than waiting for it to resolve to an acceptable level of certainty.

Sensory transfer

The results of several studies show that using sensations of different modalities, and their combination, aids in motor control. Messier et al. (2003) suggest that, when performing movements, the central nervous system uses proprioceptive information to learn a forward model of the interaction forces generated when making multijoint movements. The model helps in producing appropriate muscle torques in anticipation of interaction forces to come. Deafferented patients avoid multijoint interaction torques at high speeds by freezing degrees of freedom (e.g., by locking the elbow joint) when possible. Tunik et al. (2003) suggest that vestibulospinal information aids in creating a model as well. In the absence of visual information, allowing the finger to touch the target increases accuracy of pointing to a target in a stable environment (Rao and Gordon, 2001). In a more complicated scenario (Lackner and DiZio, 2002, 1998), the environment was a rotating room, subjecting the participant to Coriolis forces, and the target was an LED under a clear table top. Thus, the subject, in absence of visual feedback, did not know if his reach was accurate even when he touched the table top. The Coriolis forces caused a deviation in the subjects reach, resulting in inaccurate movements. However, he was able to improve his accuracy substantially within a short number of trials. Improvement was greatly diminished if the subject was not allowed to touch the table top. The proprioceptive and tactile information gained when touching the table in the presence of the Coriolis forces may have enabled the subject to construct a model of the environment, and thus anticipate the error the forces caused. Sober and Sabes (2003) provide evidence that planning a movement trajectory involves visual information, while planing for the joint forces to implement that trajectory involves proprioceptive information. Ernst and Bulthoff (2004) review other studies that suggest that information from different modalities is combined in movement tasks.

Several studies, primarily from the lab of O. Hikosaka (Hikosaka et al., 1995), provide behavioral evidence that the brain learns a motor skill through two parallel control mechanisms - one in a spatial coordinate system (such as direction of movement) and one in a motor coordinate system (such as muscle activation). The former mechanism is general, robust, and can be used to control different effectors – either arm can be used to hit the sequence of buttons. The latter mechanism is specific to an effector – the learning cannot be transferred from one arm to the other. The task common to most of these studies is a sequential button pushing task, termed the "2x5 task," in which a monkey was presented with a 4x4 grid of LED buttons, two of which were lit. The monkey learned to push the buttons in the correct order, after which a second pair of buttons were presented, and so on until five pairs were presented. An ordered set of five of these two button sequences constitutes a "hyperset." As the monkeys practiced, performance, as measured by accuracy and speed, increased. The hyperset, although consisting of five pairs of two button sequences, was learned as one motor skill — when presented with the hyperset in reverse order, performance was similar to that of a novel hyperset (Rand et al., 1998). When tested with the opposite hand for a well trained hyperset (15 days of training), performance was similar to that of a novel hyperset (Rand et al., 1998), but when tested with the opposite hand for a moderately well trained hyperset (one day of training), performance was better than that of a novel hyperset (Rand et al., 2000). That learning was able to be transferred to the opposite hand during early learning, but not late learning, stages supports the hypothesis that early learning occurs in an abstract space, but with practice the skill is transferred to an intrinsic space (Hikosaka et al., 1999; Nakahara et al., 2001).

5.2 Using Sensory Information

As discussed in the previous section, uncertainty in state depends in part on the type of sensory modality used to estimate state. Learning under conditions of uncertainty is usually attributed to cortical planning systems. This is because such behavior is well-described by statistical decision-theoretic models that explicitly take uncertainty into account (e.g., Bayesian decision theory, Kording and Wolpert 2006; game theory, Glimcher 2002). For example, Wolpert (2007) reviews evidence that humans integrate information from different sensory modalities in a way similar to that suggested by the Kalman filter (Kalman, 1960), a control theory method for optimally (under certain conditions) combining information from different sources (e.g., sensory modalities) based on their relative precisions. Such integration leads to a more precise estimate. In addition, studies show that some a priori expectation of state (the *prior distribution*) is combined with immediate sensory information to influence state estimate (Tassinari et al., 2006; Kording and Wolpert, 2006). Besides behavioral evidence supporting the use of such models, many variables are represented in cortical areas (Yoshida and Ishii, 2006; Glimcher, 2002).

There are other ways the existence of multiple sensory modalities can be used to increase task performance in general. Consider the anecdotal example task of driving a manual transmission car (described in terms of the abstract discrete-state discrete-action tasks used in this thesis). Early in learning, the visual information of the tachometer is often used to estimate the revolutions per minute (RPM's) of the engine, the state from which an action (shift or don't shift) is selected. The visual modality is good for learning as it is intense (in that the visual reading is very clear and does not have to be learned), precise, accurate, and easily described by an instructor. However, it requires time in that the driver must divert his gaze from the road to the position of the display, and from there locate the needle and read the numbers. As experience with the task is gained, other modalities are used to estimate state: as RPM's increase, the whine of the engine increases in frequency (auditory modality), as do the vibrations of the car (somatic modality). These lead to a quicker state estimate in that the driver does not have to divert his gaze, but they must be learned: it is hard for an instructor to describe these perceptions, and they might be different for different cars. In addition, the precision in state estimate may be relatively low. However, in this task, a precise state estimate is not needed: shifting within a range of RPM's (e.g., 2000 to 3000) has very similar results, and the advantage of an earlier state estimate (to say nothing of not having to divert gaze) more than makes up for the lack of precision.

Lack of precision in task-irrelevant dimensions is seen in human behavior (Scholz and Schöner, 1999; Li et al., 1998; Todorov and Jordan, 2002). For example, Li et al. (1998) asked subjects to use their fingers to maintain a constant net force on a single pad. There was more variability in the force produced by each finger than there was in net force. Scholz and Schöner (1999) termed the dimensions in which variance was allowed to accumulate the *uncontrolled manifold*, and Todorov and Jordan (2002) use control theoretic methods to show how this strategy allows for more accuracy in dimensions relevant to the task.

The statistical decision and control theoretic methods described in the previous paragraphs can take all of these variables — intensity, timing, and precision — into account to plan for the optimal use of sensory information, and most researchers discuss behavior in terms of cortical planning areas. However, the efficacy with which they deal with uncertainty does not necessarily preclude the basal ganglia (BG) from contributing to behavior under such conditions. This is especially true during skill acquisition, where the task is accomplished repeatedly. Such repetition enables the experiential learning mechanisms of the BG to participate in learning. Theoretical research in Reinforcement Learning supports this assertion. Kaelbling et al. (1998) describe how uncertainty in state can be incorporated through the use of a *belief state*. A belief state is a probability distribution over all possible states, $\mathbf{b}$, such that $b(s)$ is the belief that state s is the actual state; Littman et al. (1995) review several learning algorithms based on $\mathbf{b}$ rather than s.

Though they may require cortical areas to calculate, Bayesian statistics can be incorporated into the estimation of the values of actions to influence exploration (Dearden et al., 1998) and into the estimate of the values of states (Mannor et al., 2004). Thus, learning mechanisms of the BG may be able to deal with uncertainty as well.

5.3 Hypotheses

Rather than speculate on how different sensory modalities estimate state, I focus on their effect on state, specifically how decisions are made under varying conditions of timing and precision of state. Intensity is considered in how it affects timing and precision. In the discrete-state discrete-action tasks I use in this thesis, state (p, g) is factored into two dimensions: position p and goal g; uncertainty and timing in each dimension is investigated separately.

First, following the discussion under *sensory evolution*, an agent using a variant of the multiple controller model is presented with an evolving state representation in which precision across the goal dimension evolves over the course of a trial from imprecise to precise. There is a trade-off between time and precision in the goal dimension. Second, following the discussion under *sensory transfer*, an agent can choose to execute an action based on an imprecise state estimate or a precise one. However, precision comes with a cost, as the immediate reward incurred decreases with precision. There is a trade-off between reward and precision in the position dimension.

I describe the tasks and modifications of the multiple controller model of the previous chapter in more detail in subsequent sections. As described in the previous section, behavior under uncertainty is typically attributed to cortical planning areas. I hypothesize that the learning mechanisms of the BG can produce similar behavior. Specifically,

1. When presented with an evolving sensory representation, the agent will learn to move immediately in a direction appropriate for the belief state and task (an example such behavior is seen in Hudson et al. 2007).

2. When enabled to trade reward with precision at every decision, an agent will produce behavior similar to that seen in humans: use precise but costly state estimates in areas where precision is required, but imprecise and less costly state estimates elsewhere.

In addition to testing these hypotheses, I describe model behavior under different conditions of each task and discuss behavioral implications.

5.4 Sensory Evolution

Environment and Task

The task used is illustrated in Figure 5.2 and differs from those presented in the previous Chapter in that there are 9 actions from which to choose (the cardinal actions, diagonal actions, and a null action, which results in no movement), there are no obstacles, and the dimensions of the environment differ, including spatial positions of goals (of which there are five). Each trial begins with the agent in a fixed starting position (the same for every trial, indicated in Figure 5.2); the goal for that trial is chosen randomly from a fixed distribution (referred to as the *prior distribution*). I

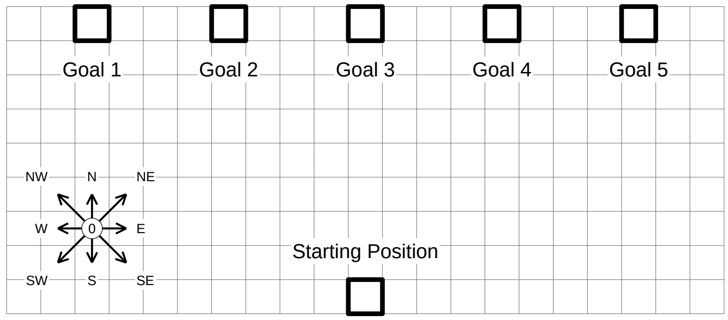

Figure 5.2. Representation of the "grid-world," a 21×9 grid of positions.

refer to the goal chosen for the trial as the *true* goal, g^*. When the agent chooses action a, it receives an immediate *action-dependent* cost ($r_a = -\sqrt{2}$ for the four diagonal actions and $= -1$ for all other actions, including the null action).

The agent's knowledge of goal is represented as a probability mass function over all possible goals, referred to as the *goal belief vector*, $\mathbf{b}$, with each component $b(g)$, specifying the agent's belief that goal g is the true goal and whose components sum to one. Over the course of a trial, $\mathbf{b}$ evolves such that $b(g^*)$ increases while all other $b(g)$ decrease; when $b(g^*) = 1$, goal belief evolution stops.

Importantly, goal belief evolution occurs independently of any action that the agent chooses and is assumed to occur through unmodeled sensory processing mechanisms. $\mathbf{b}$ is in the form of a hand-made distribution, in contrast to the agent creating $\mathbf{b}$ through some other method such as sampling. I do not attempt to investigate how sensory information is processed or evidence is accumulated. Rather, I present the agent with a simple form of an evolving goal belief and investigate how the agent makes decisions based on such a representation.

Types of Sensory Evolution

Learning agents accomplished the task under different prior distributions and types of goal belief evolution. I examine behavior under three types of prior distributions:

1. *flat*, where the probability of each of the five goals being the true goal is 0.2,
2. *biased*, where the probabilities of goals 1 through 3 being the true goal are each 0.1, that of goal 4 is 0.2, and that of goal 5 is 0.5,
3. and *two goal*, where the probabilities of goals 1 and 5 being the true goal are each 0.5 and those of the others are zero.

For each type of prior distribution, I examine six types of goal belief evolution: *slow*, *medium*, and *instant evolution* with *no delay* in evolution, and *slow*, *medium*, and *instant evolution* with a *delay* of four time steps before evolution begins. During the delay period, $\mathbf{b}$ was set as the prior distribution. Goal belief was always fully resolved

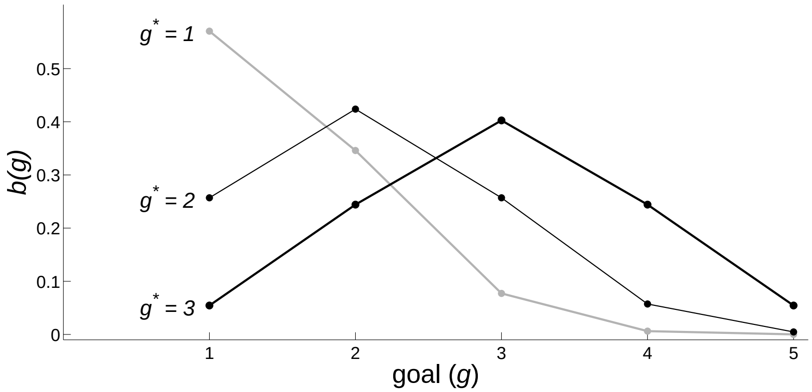

Figure 5.3. Goal belief ($\mathbf{b}$) for $\sigma = 1$ for $g^* = 1$ (grey line), 2 (think black) and 3 (thick black).

within the first 8 time steps of a trial (thus, as the agent approached the northern border of the environment, goal belief was fully resolved). Note that the *delay / instant evolution* type is used in Hudson et al. (2007). Also, for the conditions with no delay, the prior distribution is never explicitly represented in $\mathbf{b}$.

In tasks with the *flat* and *biased* prior distributions, for the purpose of calculating the goal belief vector, $\mathbf{b}$, goals 1 through 5 are assigned integer values (1 through 5, respectively). $b(g)$ is determined by a normal distribution centered on the value of true goal, g^*, with standard deviation σ. Since the integers are points, and the normal distribution is a continuous function, $\mathbf{b}$ is then normalized so its elements sum to 1. Figure 5.3 illustrates $\mathbf{b}$ with $\sigma = 1$ when $g^* = 1$ (grey line), $g^* = 2$ (thin black), and $g^* = 3$ (thick black). $\mathbf{b}$ for $g^* = 4$ and $g^* = 5$ are symmetrical with $\mathbf{b}$ for $g^* = 2$ and $g^* = 1$, respectively, and thus are not shown.

Sensory representation evolves by setting σ (by hand) to decrease over time to $\sigma = 0$, at which point I set $b(g^*) = 1$ and all other $b(g) = 0$. Figure 5.4, left, shows the decrease of σ for each type of sensory evolution. For the delayed cases, $\mathbf{b}$ is defined by the prior distribution for the first four time steps and then is determined by the normal distribution as described above. In tasks with the *two goal* prior distribution, I simply set the value of $b(g^*)$ (Figure 5.4, right), and the belief of the other goal is $1 - b(g^*)$.

Figure 5.5 illustrates the actual goal belief as it evolves for each of the six types of evolution under the *flat* prior distribution. In the figure, $\mathbf{b}$ at a time step is represented as five squares aligned horizontally (one for each goal); the squares are shaded in grey according to $b(g)$, where the darker the square, the closer $b(g)$ is to 1. These graphs, and graphs in the results section, are presented so that the slowest case of goal belief evolution is in the lower left corner and the fastest case is in the upper right corner. The bottom right of Figure 5.5 illustrates the *biased* prior distribution, and Figure 5.6 illustrates the six types of goal belief evolution for the *two goal* prior distribution.

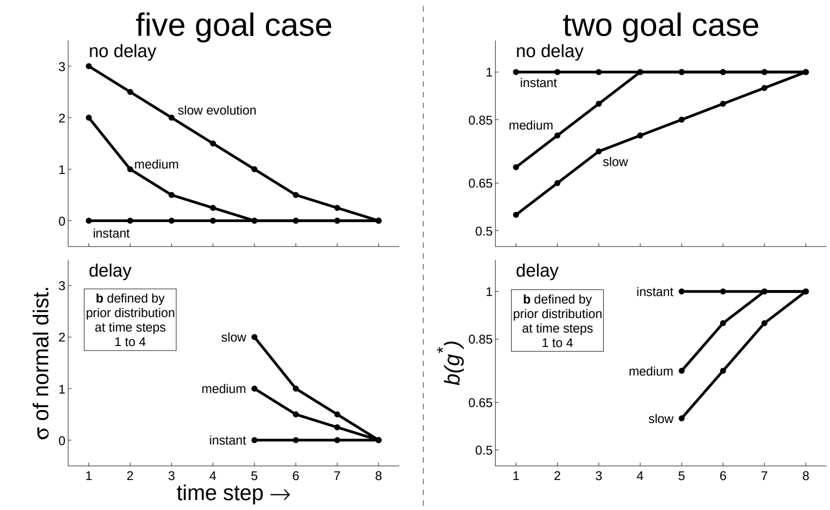

Figure 5.4. Left graphs: Goal belief evolution for the case when all five goals are selected with a non-zero probability (*flat* and *biased* priors). σ, which defines the width of $\mathbf{b}$, decreases to zero. Right graphs: Goal belief evolution for the *two goal* prior. Plotted is $b(g^*)$; $b(g)$ for the other goal is $1 - b(g^*)$. Top graphs: Evolution for the *no delay* conditions. Bottom graphs: Evolution for the *delay* conditions.

Multiple Controller Model

The multiple controller model used by the agents differs from that presented in the previous chapter. First, the *Automatic* controller is disabled, as I am interested in the strategies the learning component of the *Value*-based controller develops. Second, to incorporate $\mathbf{b}$ in the learning and determination of the values of each action, equations 4.1 and 4.4 from the previous chapter are replaced with, respectively,

$$Q(p,g,a) \leftarrow Q(p,g,a) + \alpha\, b(g) \left(r_a + \sum_{g' \in G} b(g') Q(p', g', a') - Q(p,g,a) \right) \quad (5.1)$$

$$\widetilde{Q}(p,g,a) \leftarrow f_d \left(\widetilde{Q}(p,g,a) + \alpha_q b(g) \left(\Psi(p,g,a) - \widetilde{Q}(p,g,a) \right) \right), \quad (5.2)$$

where g' is an index (not the next goal) and α_q is a learning rate (set to 0.001). In addition, to accommodate the different dimensions of the environment, the temperature in Equation 4.2 is set to 0.3. Referring to Figure 3 from the previous Chapter, $[(p_i, g_j)]$, the value of *State* neuron (p_i, g_j), is simply $b(g_j)$ when the agent is in the position represented by p_i. All neurons corresponding to other positions have zero value.

This representation is similar to that which is used in machine learning research in partially observable domains (Littman et al., 1995; Kaelbling et al., 1998). Such a formulation is neurally plausible: If we consider $\widetilde{Q}(p,g,a)$ to be the weight of the connection from *State* neuron (p,g) to the *Decision* neuron that implements action a, then incorporating $\mathbf{b}$ is analogous to activating *State* neuron (p,g) by $b(g)$ instead of 1.

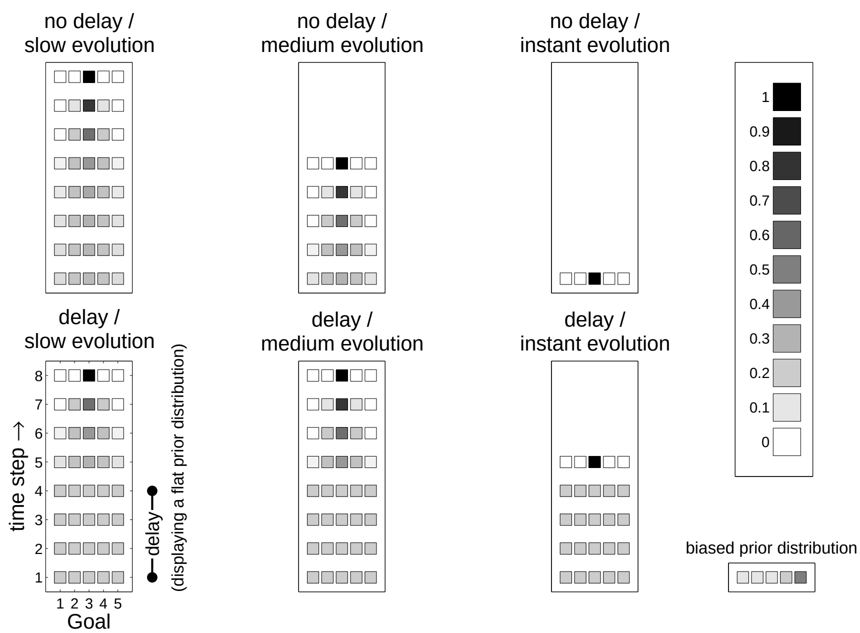

Figure 5.5. Illustration of each of the six types of goal belief evolution for the *flat* prior distribution. **b** is represented as five horizontally-aligned squares, shaded according to $b(g)$. Time progresses from bottom to top for each type. The top row illustrates the fully resolved **b**; **b** at later time steps is also fully resolved. Shown is the case for $g* = 3$. Under *delay* types of evolution, the first four time steps illustrate the *flat* prior distribution. The *biased* prior is illustrated in the bottom right.

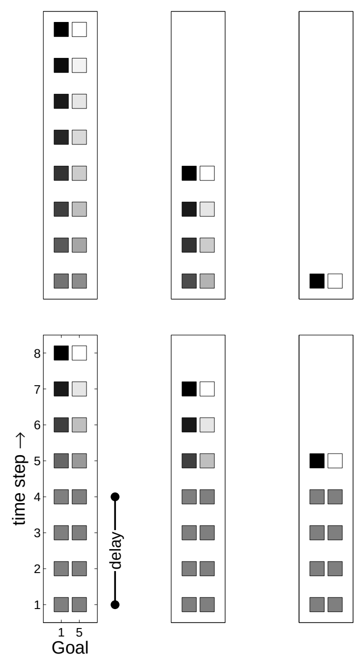

Figure 5.6. Goal belief evolution under the *two goal* prior distribution. Follows the same conventions as Figure 5.5, except goals 2,3,and 4 are not represented (as their belief is always zero). The arrangement of the types of evolution are the same as that in Figure 5.5, but are not labeled for brevity.

In total, I examine model behavior under 18 conditions (3 prior distributions and 6 types of goal belief evolution). 20 runs for each condition were performed, where a "run" consisted of having the agent accomplish the task for 30,000 trials. I examine three facets of behavior. First, I examine in detail the progression of behavior — how behavior changed with experience — under the *no delay / slow evolution / flat prior* condition. Second, I describe learned behavior under the different conditions and show that the model learned to select actions appropriate for the goal belief and prior distribution: under uncertain conditions, actions towards the mean of the prior distribution were taken. Third, I exposed agents trained under one condition to another type of condition; the conditions under which they were trained affected their strategies.

Results

Many of the graphs I present show model behavior for a particular condition, goal, and trial. Behaviors were taken from "test" trials (performed periodically for each goal during a run), during which all exploration and learning parameters were set to zero. Most graphs are a representation of the grid-world (see Figure 5.2). Unless otherwise noted, the grey-scale coloring of a position indicates the the proportion of the 20 runs for which that position was visited (greater proportions are darker, positions not visited are not plotted).

Progression of behavior

For the *no delay / slow evolution / flat prior* condition, behavior at four different points in learning for each of the five goals is shown in Figure 5.7. Early in learning (e.g., trial 1, bottom of Figure 5.7), by design, the *Value*-based controller, **B**, was not trained enough to select actions. Through the *Planner*, **A**, the agents waited until goal belief was fully resolved (by selecting the null action) and then took the optimal path towards each goal (for goals 2 and 4, there are several optimal paths).

As experience was gained, **B** selected a greater proportion of the actions. Figure 5.8 (top left) plots the proportion of actions selected by **B** as a function of trial for goals 1, 2, and 3 (that of goals 4 and 5 were very similar to that of goals 2 and 1, respectively, and thus are not shown). Note that for early trials (before 1200), **B** selected a greater proportion of actions for goals 1 and 3 than for goal 2. This is because, enroute to goal 2, several paths were traversed while behavior was controlled by **A**. Because the positions along those paths were visited less frequently than the positions along the paths for goals 1 and 3, **B** was trained at a slower rate for goal 2. **B** was also able to explore — select actions other than those considered most valuable — and thus began to select actions before goal belief was fully resolved. Under uncertainty in goal belief, the middle path (path of positions from the starting position to goal 3) was experienced and deemed valuable. Thus, **B** learned to immediately move north (towards the mean of the prior distribution) from the starting position for all trials, including those for which the true goal was 1 or 5.

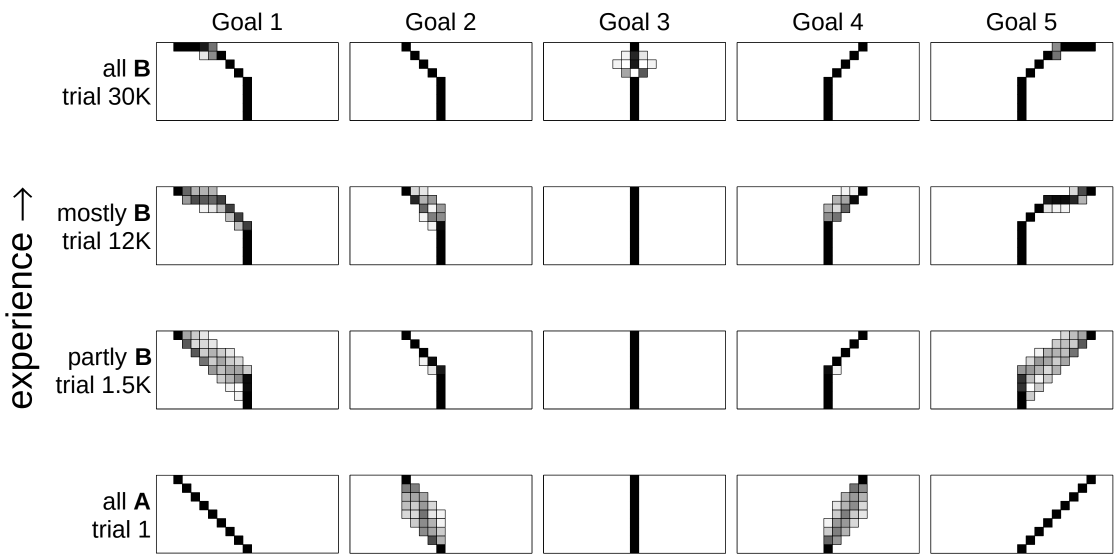

Figure 5.7. Illustration of behavior across all 20 runs for the *no delay / slow evolution / flat prior* condition at different points in learning (labeled on the left). Each rectangle is a representation of the grid-world (Figure 5.2. Shaded squares indicate positions visited; the darker the shading, the greater the proportion of the 20 runs visited that position. Positions not visited are not marked.

After trial 1200, for goal 1, the proportion of actions selected by **B** decreased and then rose again. This is because, early in learning, the agents had little experience with the middle path enroute to goal 1 (and positions between the middle path and goal 1), particularly when goal belief was resolved to some degree. Thus, **A** was used to select actions until **B** was trained. This behavior is more clearly seen in Figure 5.8 (bottom), which plots, for each position, how early in learning **B** was able to select an action for each goal (darker greys indicate earlier in learning).

Note that for goals 1 and 5, **B** was trained along the path directly from the starting position to the goal early in learning. This shows that **B** initially followed the behavior dictated by **A**. Figures 5.7 and 5.8 (bottom) show that, with experience, behavior for goals 1 and 5 deviated from moving straight to the goals to moving along the middle path; the difference between initial behavior and learned behavior increased with experience. This effect is seen more clearly in Figure 5.8 (top right), which plots, for goal 1, the average distance between the path taken to goal 1 and the path straight from the starting position to goal 1 (the *direct path*) as a function of trial number (see figure caption for more details). These results show that, early in learning, as **B** assumes control as some positions, it initially waits until goal belief is resolved to some degree and then follows the strategy prescribed by **A**. Later in learning, **B** learns to move earlier in time (i.e., when goal belief is less precise) towards the mean of the prior distribution.

The general progression of behavior described in this section is seen for all other conditions of goal belief evolution and prior distributions. Rather than plot behavior

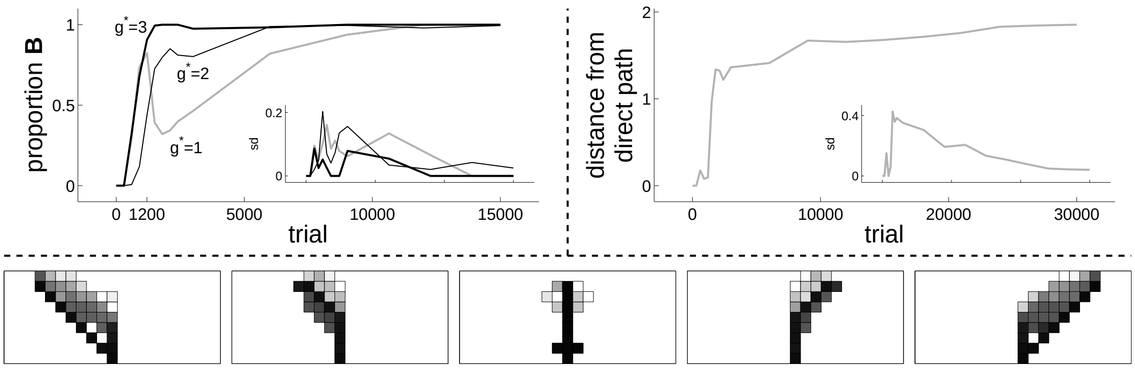

Figure 5.8. Top left: Mean (across the 20 runs) proportion of actions chosen by **B** as a function of trial for the *no delay / slow evolution / flat prior* condition. Shown are proportions enroute to goal 1 (grey line), goal 2 (thin black), and goal 3 (thick black). Inset indicates standard deviation (s.d.). Top right: Mean (across the 20 runs) distance between the chosen path and the direct path (the line from the starting position to goal 1) as a function of trial. Inset indicates s.d. For each run, distance was the mean distance between each position visited and the closest position along the direct path. Each position was only counted once (e.g., when **A** controlled behavior, the agent "visited" the starting position until goal belief was fully resolved; the starting position was only counted once). Bottom: Earliest recorded trial that **B** selected an action from each position. The darker the shading, the earlier the trial. Positions at which **B** never selected an action are not marked.

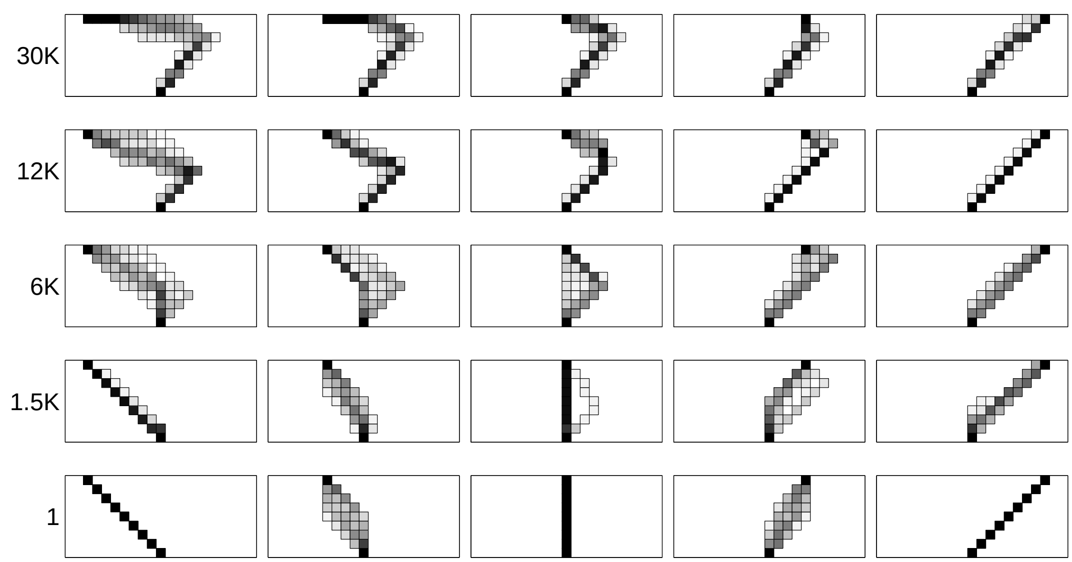

Figure 5.9. Follows same conventions as Figure 5.7, except this illustrates the *no delay / slow evolution / biased* prior condition.

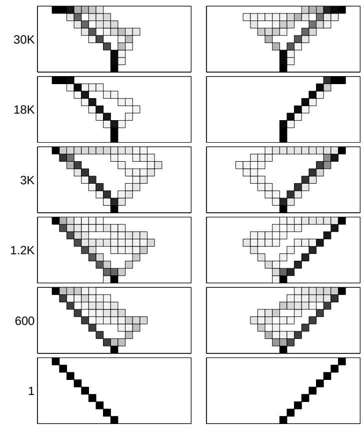

Figure 5.10. Follows same conventions as Figure 5.7, except this illustrates the *no delay / slow evolution / two goal* prior condition. Also, since there are only two goals, behavior for goal 1 is on the left and behavior for goal 5 is on the right.

for all other conditions, I include two more noteworthy illustrations: Figures 5.9 and 5.10 show, in a manner similar to Figure 5.7, behavior for the evolution type *no delay / slow evolution* with the *biased* and *two goal* prior distributions, respectively. For the *biased* prior, for which goal 5 was chosen as the true goal 50% of the time, the agents' behaviors gradually changed (with experience) from waiting until goal belief was resolved and then moving directly towards a specific goal from the starting position to moving immediately towards the mean of the prior distribution. For the *two goal* prior, though, a "Y" shape was seen in the distribution of positions visited for both goals as experience was gained. This is because goals 2, 3, and 4 were never selected as the true goals; hence, the agents required more experience to discover the strategy of moving along the middle path as goal belief was uncertain, and deviations from the middle path resulted in actions towards one of the two goals.

Thus, for all prior distributions, the agents gradually learned to move immediately towards the mean of the prior distribution rather than wait for goal belief to resolve. Another general trend is also seen: the behavioral effects of the evolving goal belief is greater for goals further away from the center of prior distribution. Thus, for brevity, presentation in the rest of this paper is restricted to behavior for goal 1.

Learned behavior

As the top parts of Figures 5.7, 5.9, and 5.10 show, the trained agent selected actions towards the mean of the prior distribution when goal belief was uncertain. As goal belief resolved, actions towards the true goal were taken. The type of goal belief evolution under which the agents were trained affected their behaviors. Figure 5.11 illustrates the learned behavior for all six types of goal belief evolution under the *flat* prior distribution for goal 1. In every case except *no delay / instant evolution* (top right), action north was selected for the first 3 or 4 time steps. For both the *no delay* and *delay* types of evolution, as the rate of evolution decreased, the longer the agent

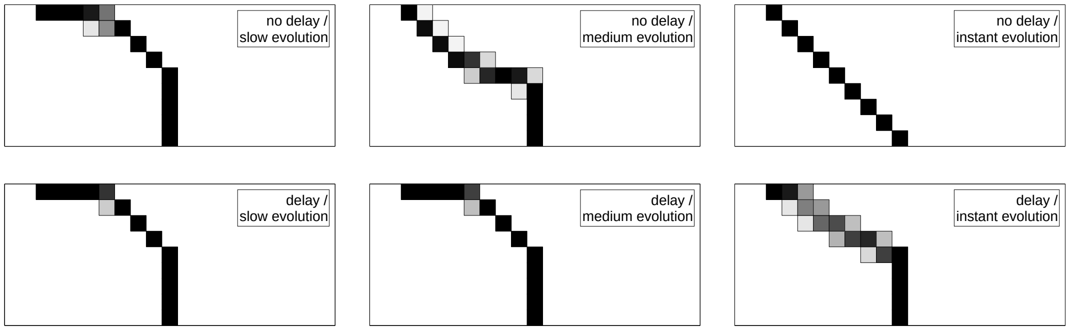

Figure 5.11. Learned behavior for each of the six goal belief conditions (labeled in each graph) for the *flat* prior distribution. Follows same shading conventions as Figure 5.7.

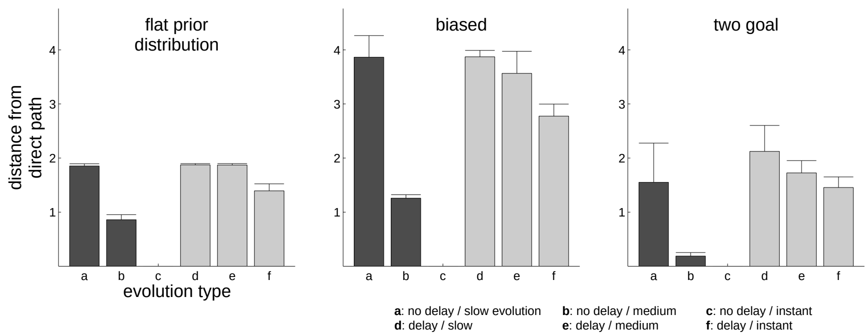

Figure 5.12. Mean (across the 20 runs) distance between learned path for goal 1 and the direct path for all 18 conditions, grouped by prior distribution (labeled on top of each graph). Labels for evolution type are indicated in the lower right. Evolution types with no delay are colored in dark grey; evolution types with delay are in light grey. Standard deviation (s.d.) is plotted as error bars; if s.d. was < 0.01, it was not plotted.

choose actions towards the mean of the prior distribution. This behavior is also seen in Figure 5.12 (left), which plots (as bar graphs) the average distance between the path taken to goal 1 (as controlled by a trained agent) and the direct path for each of the six types of goal evolution.

The type of prior distribution used also affected behavior. Figures 5.13 and 5.14 illustrate learned behavior for all six type of goal belief evolution under the *biased* and *two goal* prior distributions, respectively. The same trend is seen: the faster goal belief resolved, the less time the agents spent moving towards the mean of the distributions. Figure 5.12 (middle and right) also shows this trend. Note that actions towards the mean of the prior distribution were taken even when the prior distribution was not explicitly represented in the goal belief (observe behavior under the *no delay / slow* and *no delay / medium* conditions).

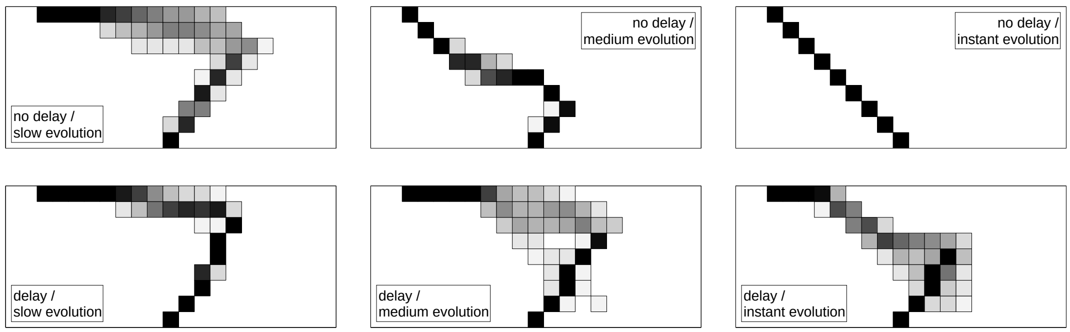

Figure 5.13. Learned behavior for the *biased* prior. Follows same conventions as 5.11.

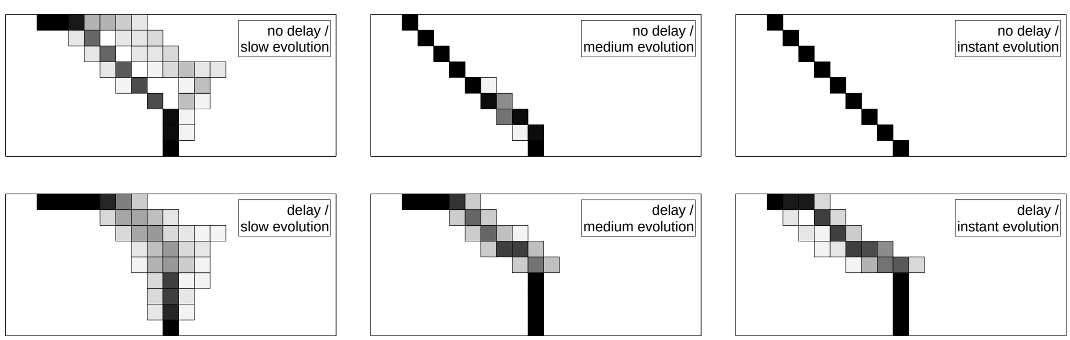

Figure 5.14. Learned behavior for the *two goal* prior. Follows same conventions as 5.11.

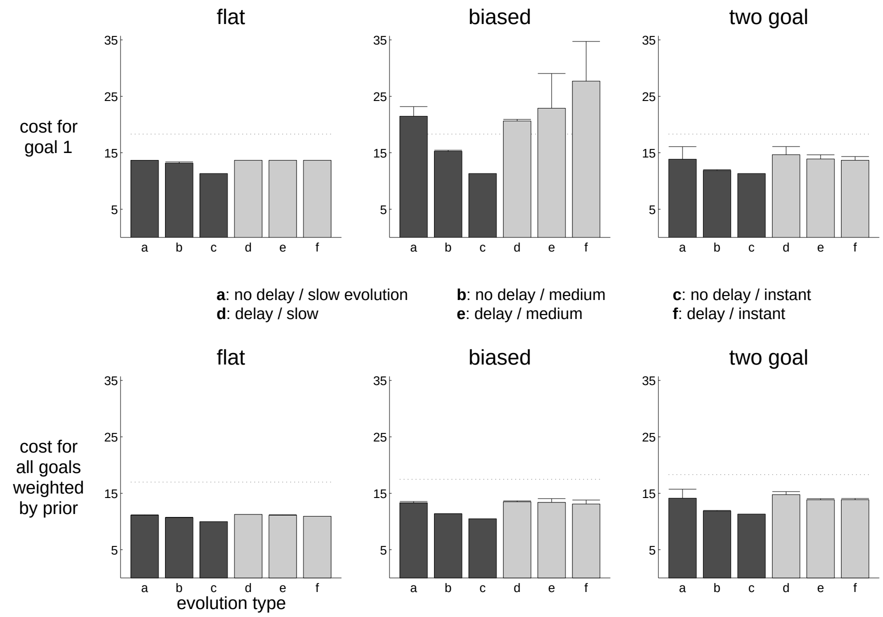

Figure 5.15. Mean (across the 20 runs) cost ($-\sum r_a$) for the learned behavior for all 18 conditions. Follows same conventions as Figure 5.12. Top: Cost for goal 1 only. Bottom: Mean cost over all goals, weighted by the prior for each goal.

In general, learned behavior as controlled by **B** incurred less cost ($-\sum r_a$) than behavior as controlled be **A** (Figure 5.15, top; cost under **A** is drawn as a dotted horizontal line). As seen in Figure 5.15 (top) under the *biased* prior, the strategy of moving towards the mean of the prior may be worse than simply using **A** when the goal is considered on its own. However, when taking into account the prior distribution (i.e., an average of the cost to all goals weighted by the prior distribution), behavior under **B** incurs less cost than behavior under **A** (Figure 5.15, bottom). Thus, behavior was more costly when considering goal 1 in isolation, but better on average considering the whole task.

The results of this section show that learned behavior for each of the three prior distributions follow the same general trends. Learned behavior, as developed and controlled by **B**, is to move immediately towards the mean of the prior distribution, even when the prior distribution is not explicitly represented in the goal belief. Movement deviates from the towards the mean to the true goal as goal belief resolves. The faster goal belief resolves, the faster movement towards the true goal occurs. Finally, while such behavior may result in a strategy that is worse than behavior as controlled by **A** when considering a goal in isolation, it is better when considering all goals.

Effect of training under one condition when presented with another

The condition under which an agent was trained affects its behavior even when exposed to a different condition. Figure 5.16 (top half) plots the behavior of agents trained under a *slow* evolution but tested with a fully resolved goal belief (i.e., of

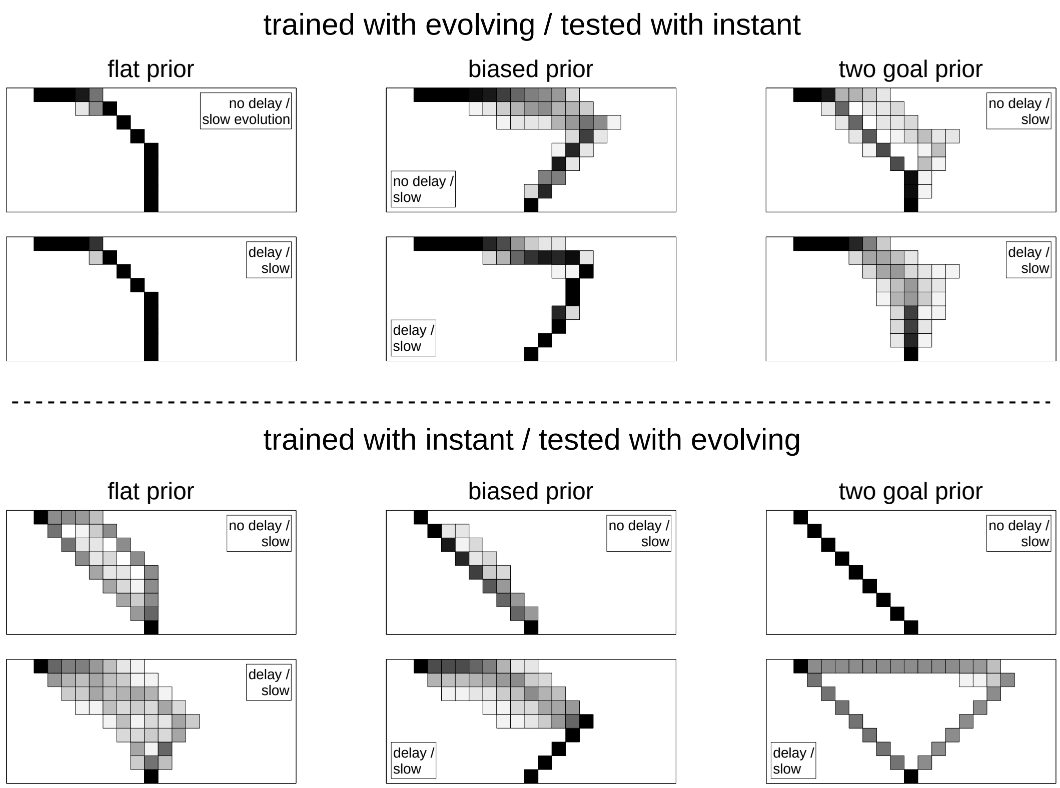

Figure 5.16. Agents trained under one condition were tested (for one trial with no learning or exploration) with another condition. Goal 1 is the true goal in cases. Shown is the proportion of runs that visited each position (Follows same conventions as Figure 5.11). Top half: labels indicate conditions under which the agents were trained, plotted is their behavior when given a fully resolve goal belief. Bottom half: All agents were trained under a fully resolved goal belief; labels indicate the conditions under which they were tested.

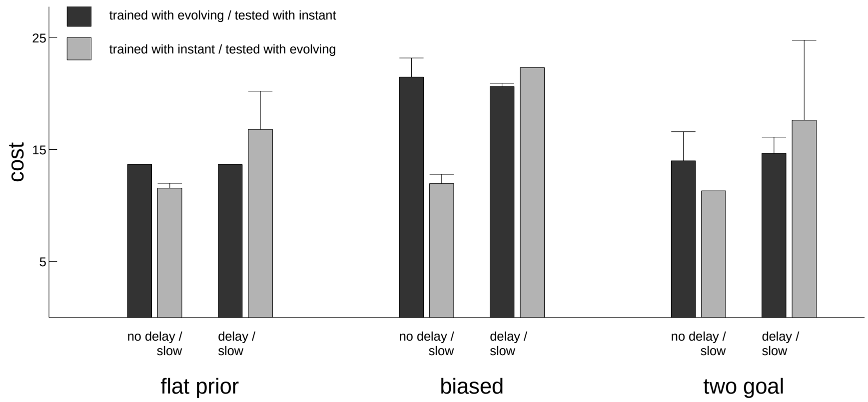

Figure 5.17. Mean cost of behaviors plotted in Figure 5.16. Dark grey bars correspond to the top half of Figure 5.16, light grey with the bottom half. Error bars show standard deviation.

type *no delay / instant evolution*), for both the *no delay* and *delay* cases and for all three prior distributions. For brevity, I do not present all possible combinations of training / testing conditions and only examine behavior when goal 1 is the true goal. Behavior was very similar (though not identical) to behavior when tested under the goal belief conditions for which they were trained (compare with the left two graphs of Figures 5.11, 5.13, and 5.14). Details on how the comparison was conducted are provided in the figure caption.

On the other hand, agents trained with a fully resolved goal belief exhibited very different behavior when tested with an evolving goal belief (Figure 5.16, bottom half). The most noteworthy trend is that the representation of the prior distribution in goal belief (which occurs under the *delay* condition) profoundly affects behavior. For the *flat* prior distribution, behavior during the delay period tended towards the middle path (due to the influence of goals 2, 3, and 4 on the weighted values of the actions), but with a high variance. For the *biased* prior, actions straight towards goal 5 (which had a prior belief of 0.5) were selected. For both distributions, when the delay period ended, actions towards goal 1 were selected. Under the *two goal* prior distribution, the agents moved straight towards goal 1 or goal 5 (and never moved along the middle path). In the *no delay* case for all prior distributions, behavior tended towards goal 1 from the starting position, displaying some variance due to the unresolved goal belief.

These results demonstrate the inflexibility of the *Value*-based controller. Behavior was learned from experience. When conditions (e.g., sensory information) changed, resulting behavior as controlled by **B** was not appropriate for the new conditions. The results also show that agents tested with goal belief that evolves faster than that for which they were trained retain much of their behavior. Agents tested with a goal belief that evolves slower, on the other hand, display a greater variance in behavior.

Figure 5.17 plots the mean cost of every condition illustrated in Figure 5.16. When there was a delay in goal evolution, agents trained with an evolving representation but tested with a fully resolved belief performed better than agents trained with a fully resolved belief but tested with an evolving representation. The opposite was true when there was no delay.

5.5 Sensory Transfer

Environment and Task

In this section, I investigate what I term *sensory transfer*, in which an agent learns to use sensory modalities with intensity, timing, and precision qualities different than those for which it was trained. To briefly describe the model, an agent can choose an action from a particular group of actions, analogous to making a movement based on sensory information from a particular modality. The actions within a group share the same trade-off between precision and reward: some groups are precise but costly while others are imprecise by less costly. For example, using the driving task described earlier, the driver can choose to estimate RPM's based on the tachometer, which takes time and attention but reveals an accurate and precise state estimate, or estimate RPM's based on auditory or somatic information, which takes less time but may be imprecise. Thus, a stream of sensations and perceptions, including those from different modalities, arrive over a short period of time. In this thesis, for the sake of simplicity, I assume that information that arrives earlier is less precise than information that arrives later.

The task used is presented in Figure 5.18. As in Chapter 4, the agent can move in four directions (termed *base actions* in this section). According to the transition dynamics of the previous chapter, execution of base action a results in a deterministic transition from position p to p'. In this section the agent can also specify a sensory modality, k, from which to estimate state. Rather than explicitly model the passage of time, as in the previous section, for ease of presentation reward is a surrogate for time. The choice of k affects the immediate reward received and the level of precision in state estimate. I implement four modalities, hereafter referred to simply as *Action Groups* to make clear that the putative *effects* of sensory modalities, rather than the modalities themselves, are modeled. Figure 5.19 illustrates the four *Action Groups*, and Figure 5.20 (left) illustrates the sixteen actions — four base actions and four *Action Groups*.

Following Figure 5.19, if an action from *Action Group* 4 is selected, the current position is estimated with exact precision and transition to p' is deterministic. The cost for this precision is an immediate reward of $r_k = -4$. The only difference between actions from *Action Group* 4 and actions from Chapter 4 is the immediate reward received. However, agents can also select actions that are less costly but less precise. In these cases, to implement imprecision, an action taken transports the agent one position in the intended direction from a position chosen randomly from a set centered around the current position. If the action is from *Action Group* 3, the set is all positions within one step of the current position and $r_k - 3$. Similarly, for actions from *Action Groups* 2 and 1, the set is all positions within two and three steps from the current position, respectively, and $r_k - 2$ and -1, respectively. In all cases, the position from which the agent is moved is chosen randomly from a uniform

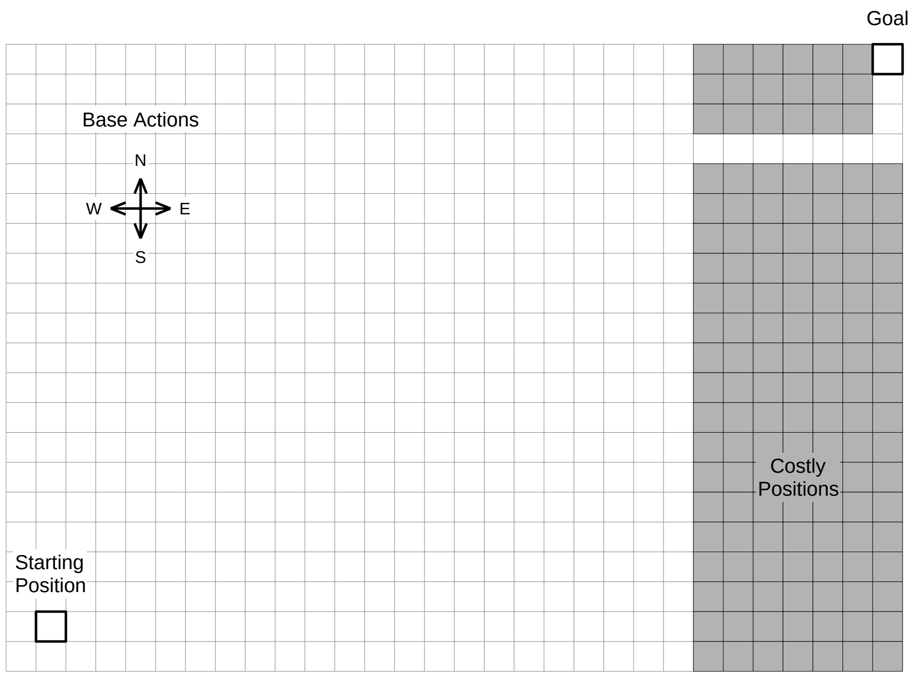

Figure 5.18. "Grid-World" for the *sensory transfer* experiments. There are 630 positions arranged in a 21×30 grid. A transition into a "costly position" (shaded grey) results in a reward of -50.

distribution over all possible positions (as defined by the *Action Group*). Thus, choice of *Action Group* represents a trade-off between reward and precision.

Unlike previous tasks, there is only one goal, and a subset of positions (shaded in grey in Figure 5.18) are "costly:" transitioning into them incurs an immediate reward of -50. Thus, immediate reward received includes r_p, which is 0 except for at the costly positions. Note that there is a narrow path through the costly positions to the goal. A trial begins with the agent in the starting position and ends when it has reached the goal position, valued at $+200$, or a time step limit of 200 has been reached.

Model

To focus on the relative advantages of different *Action Groups*, I depart from the multiple controller model entirely and use a pared-down version of the *Value*-based controller. Following typical simple Reinforcement Learning models, at each time step, the agent can choose from a set of available actions. The value of each available action is computed and the argmax_a is chosen $(1-\epsilon)$ proportion of the time, where $\epsilon = 0.1$ (this scheme is referred to as $\epsilon-$greedy, Sutton and Barto 1998). Otherwise, an action is chosen randomly from the set of available actions.

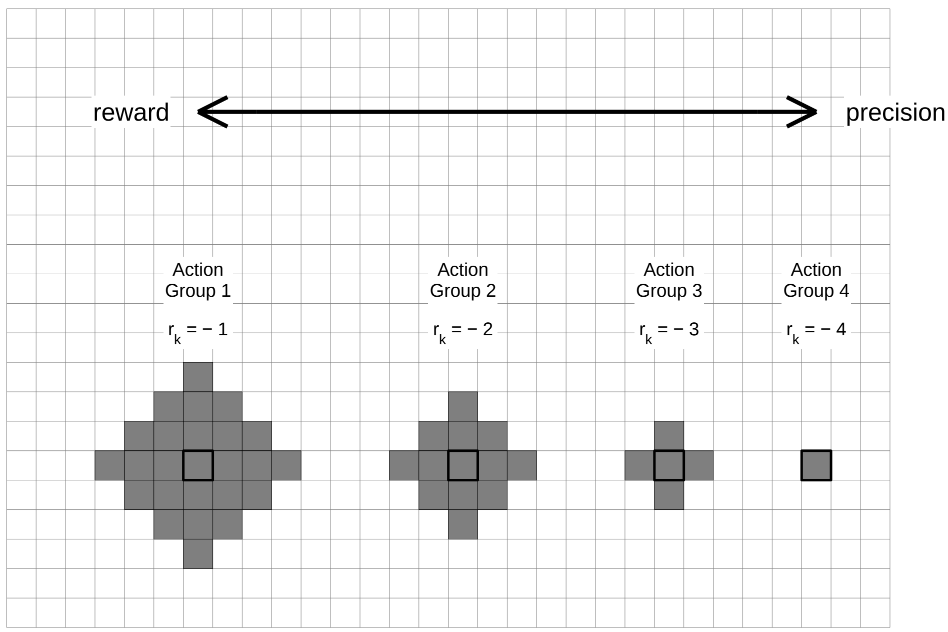

Figure 5.19. Illustration of the four *Action Groups*. When the agent is in position p, indicated by the thick-lined square in the center of each group, selection of an action transitions the agent from any shaded position within the group with an equal probability. If the agent is near a wall, the probability of positions that would be off the environment is distributed evenly to remaining positions.

Update in Q-values

Actions are labeled according to the base action to which they correspond and the *Action Group* to which they belong: a^k, where a is the base action (north, south, east, or west) and k is the *Action Group* (see Figures 5.18 and 5.20, left). The precision of action a^k is represented by a *position belief vector*, $\mathbf{b}$, similar to the *goal belief vector* from the previous section. $\mathbf{b}$ is determined by the actual current position, p, and the action group, k, of the selected action (Figure 5.19). As with the update rule for the previous section, the update of the Q-values take $\mathbf{b}$ into account. For all positions,

$$Q(p, a^k) \leftarrow Q(p, a^k) + \alpha\ b(p) \left(r_k + r_{p'_0} + \sum_{p' \in P} b(p') Q(p', a'^{k'}) - Q(p, a^k) \right), \qquad (5.3)$$

where $\alpha = 0.1$, p' and $a'^{k'}$ indicate the next position and action, respectively, r_k is the immediate reward for selecting an action from *Action Group* k, $r_{p'_0}$ is the immediate reward for transitioning into position p'_0 (where p'_0 is the *actual* position the agent transitioned into), and goal g is left out of the notation because there is only one goal.

Action saliency

Not all sixteen actions are immediately available, necessarily. In the driving example, the driver initially used visual information to estimate state, but he learned to use auditory and somatic information instead. Visual information was used because it is intense; the intensity makes it *salient* in that it is easily noticeable and is known to hold task-relevant information. The same cannot be said for auditory / somatic perceptions — their relevance to the task must be learned. However, once learned, they offer a better alternative to visual information. As discussed in the beginning of this chapter and in the Introduction, one part of motor skill acquisition is to learn what sensory information to use to best accomplish the task. The sensory information used to initially learn a task may not be the best sensory information to use to execute the movements after much practice. As discussed in Chapter 2 (page 16), dopamine may be able to signal the saliency of sensory information.

To formalize these concepts, I introduce a *saliency* measure for each position-action pair: $\nu(p, a^k)$ $(0 \leq \nu \leq 1)$. In order for an action to be available, its saliency must be above a threshold ($\theta_\nu = 0.8$ for the results presented here). To account for imprecision in position, the saliency of action a^k is computed as

$$\sum_{p \in P} b(p) \nu(p, a^k).$$

Initially, only the most precise actions are given a saliency of 1 (i.e., $\nu(p, a^k)$ for $k = 4$ and all p and a are initialized to 1) and all other actions are initialized to 0. Thus, the agent is restricted to use only the most precise actions. However, the saliencies of less precise actions can increase and thus become available. The next paragraphs describe this process.

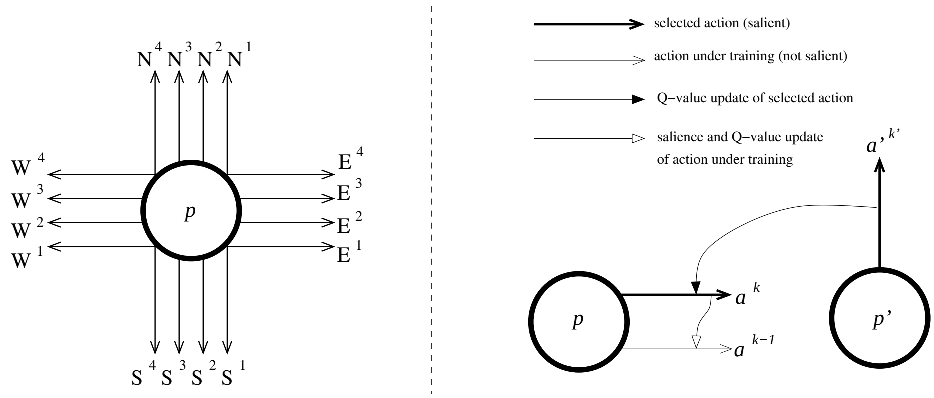

Figure 5.20. Left: schematic of the sixteen actions potentially available at each position. Right: Schematic of the updates of $Q(p, a^k)$, $Q(p, a^{k-1})$, and $\nu(p, a^{k-1})$.

When action a^k is chosen, the Q-values and saliencies associated with it train those of a^{k-1} (i.e., the base action taken from a *Action Group* k trains the base action for the less precise *Action Group*, $k-1$). For ease of explanation, consider the case where state estimate is exact: $b(p) = 1$ and all other $b(p') = 0$; thus $\mathbf{b}$ is left out of the next few terms. Figure 5.20 (right) illustrates the update of $Q(p, a^k)$, $Q(p, a^{k-1})$, and $\nu(p, a^{k-1})$. $Q(p, a^{k-1})$ is updated towards $Q(p, a^k)$. The accuracy of $Q(p, a^{k-1})$, δ_q, is simply the difference between the two. $\nu(p, a^{k-1})$ is updated towards some function of $\nu(p, a^k)/\delta_q$ with a learning rate also inversely proportional to δ_q. Thus, if $Q(p, a^{k-1})$ is not accurate, $\nu(p, a^{k-1})$ will not increase by much. While the accurate value of action a^{k-1} taken from position p is different than that of a^k, $Q(p, a^{k-1})$ is updated towards $Q(p, a^k)$. Thus, before action a^{k-1} becomes available from position p, $Q(p, a^{k-1})$ is near $Q(p, a^k)$.

The above equations are modified to take $\mathbf{b}$ into account as follows:

$$\begin{aligned}
\delta_q &\leftarrow \sum_{p \in P} b(p) Q(p, a^k) - \sum_{p \in P} b(p) Q(p, a^{k-1}) \\
Q(p, a^{k-1}) &\leftarrow Q(p, a^{k-1}) + \alpha \; b(p) \delta_q \text{ for all positions,}
\end{aligned}$$

which is similar to equation 5.3. δ_q, the accuracy of the estimated value of action a^{k-1}, is used to update the saliency of action a^{k-1} as follows:

$$\begin{aligned}
\delta_\nu &\leftarrow \frac{1}{\max(|\kappa \delta_q|, 1)} \sum_{p \in P} b(p) \nu(p, a^k) - \sum_{p \in P} b(p) \nu(p, a^{k-1}) \\
\nu(p, a^{k-1}) &\leftarrow \nu(p, a^{k-1}) + \frac{\alpha_\nu}{\max(|\kappa \delta_q|, 1)} b(p) \delta_\nu \text{ for all positions,}
\end{aligned}$$

where ν is bounded by 0 and 1, α_ν is a learning rate (set to 0.1 as well) and κ, $0 \leq \kappa \leq 1$, weighs the importance of accuracy in Q-values. In the experiments to be described, several values of κ are used.

Action selection

Finally, action selection takes both saliency and position belief into account. $(1-\epsilon)$ proportion of the time, the highest-valued action is chosen: for every base action a and *Action Group* k, the chosen action is

$$\text{argmax}_{a^k} \sum_{p \in P} b(p)\nu(p, a^k)Q(p, a^k).$$

The other ϵ proportion of time, an action is selected randomly from the set of actions such that the following quantity is $\geq \theta_\nu$:

$$\sum_{p \in P} b(p)\nu(p, a^k).$$

Experiments

I examined how learning agents accomplished the task with the learning and control mechanisms described above under five different conditions:

1. *Group 4 only*, in which only actions from *Action Group* 4 were allowed. α_ν was set to 0. Thus, precision was exact for every action chosen.

2. *Flat*, where ν for every p and a^k was initialized to 1. Thus, all sixteen actions were available from the beginning.

3. $\kappa = 0$, in which case accuracy was given a zero weight in updating ν for actions. Thus, ν was updated quickly based purely on experience.

4. $\kappa = 0.25$, in which case accuracy was given a moderate weight.

5. $\kappa = 0.75$, in which case accuracy was given a high weight.

20 runs of each condition were performed, where a run consisted of having the agent accomplish the task for 200,000 trials.

Results

As with the previous section, all results are taken from "test trials," where all learning, exploration, and stochasticity in the environment were set to zero.

Learning

Figure 5.21 plots the mean reward across the 20 runs for each condition for the first 30,000 trials. Note that because of the reward structure of the environment and task (r_k ranges from -1 to -4 and transition into a costly position incurs $r_p = -50$), mean rewards are largely negative during early trials. The reader's attention should be drawn to the shape of the reward curves, i.e., the number of trials it takes before mean reward increases substantially. Immediately evident is the advantage of restricting action selection to salient actions for early performance. Mean reward under the

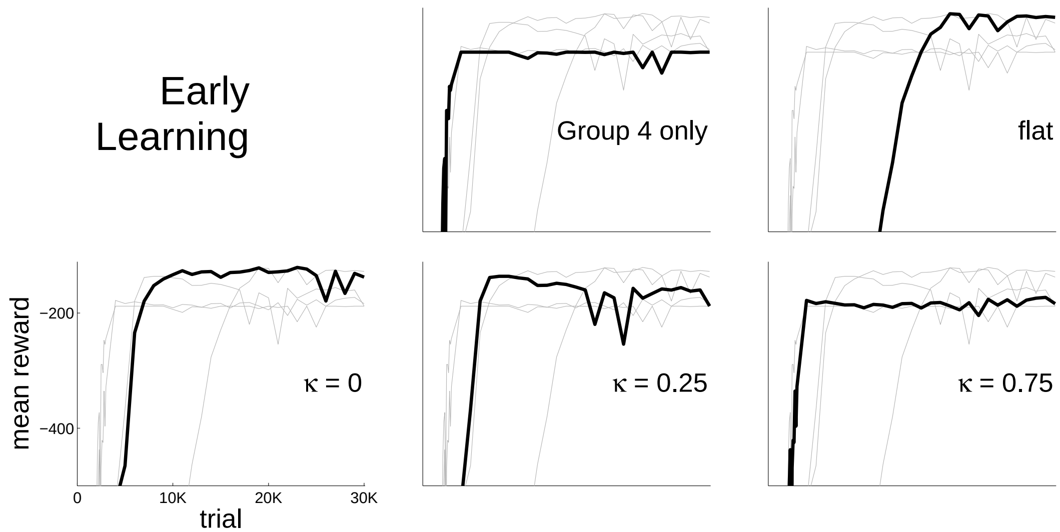

Figure 5.21. Mean reward (across the 20 runs) for each condition for the first 30,000 trials. Each plot draws the mean reward for the labeled condition in black and the mean rewards of the all other conditions in grey for comparison.

flat condition took roughly 15,000 trials to reach -200. Conditions where accuracy is moderately ($\kappa = 0.25$) or weakly weighted in training ν took roughly half that, and conditions where accuracy is highly weighted ($\kappa = 0.75$) or only *Action Group* 4 actions are allowed required even fewer trials. These results suggest that limiting exploration during early trials enabled the agents to accomplish the task more quickly.

The advantage in early performance may come at a cost. Figure 5.22 displays mean reward for all 200,000 trials. While the mean rewards of agents trained under conditions *Group 4 only* and $\kappa = 0.75$ increased the fastest during early trials, they also displayed the slowest increase in performance after the first 30,000 trials. The *flat* condition, on the other hand, increased the most, while conditions $\kappa = 0$ and $\kappa = 0.25$ had intermediate increases.

Strategy

Agents from all conditions learned to use the path through the costly positions and displayed a generally similar strategy of positions visited and use of *Action Groups* (save for the *Group 4 only* condition). Figure 5.23 displays learned behavior in a manner similar to Figure 5.11 — positions visited by each run are marked and shaded according to the proportion of runs that visited that position (darker is closer to 1). As a reminder, plotted are learned behaviors under an entirely greedy policy with no stochasticity in the environment — all learning, exploration, and stochasticity parameters have been set to zero.

The right four columns illustrate the proportion of runs that selected an action from each *Action Group* at each position. For all conditions (except *Group 4 only*),

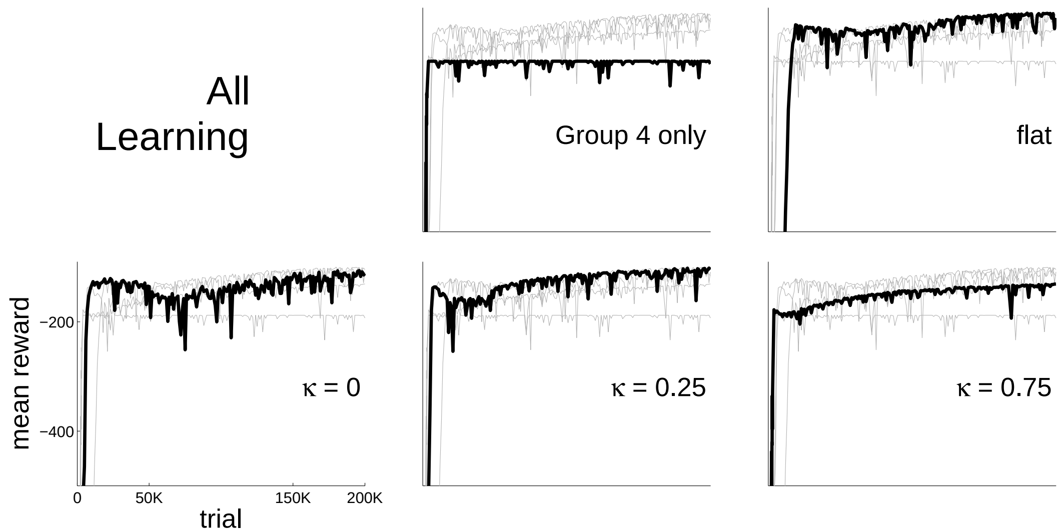

Figure 5.22. Follows same conventions as Figure 5.21, but plots mean reward for all 200,000 trials.

actions from *Action Group* $k = 1$ were used at many positions between the starting position and the entrance to the path (referred to as the *left portion* of the environment). At positions near the path entrance and in the path, actions from $k = 4$ were used. Agents trained under the *flat* condition use actions from progressively lower groups at positions farther from the path entrance. Agents trained under the other conditions seem to be developing that behavior. Such strategies make sense in that the most (immediately) rewarding but least precise *Action Groups* were used in areas of the environment where precision does not matter, while closer to the costly positions, more precise and costly actions were chosen.

To observe overall behavioral strategy, the left column of Figure 5.23 displays the same information but without separating *Action Groups*. As would be expected, there is variance in positions visited in the left portion of the environment as there are no diagonal base actions (there are many equally optimal paths to the goal). However, visual inspection indicates that agents that trained under conditions *flat*, $\kappa = 0$, and $\kappa = 0.25$ were more consistent in the choice of positions enroute to the goal. Figure 5.24 (top left) quantifies the consistency by plotting, as bar graphs, the mean density of positions for each condition. Mean density was calculated as follows: for each condition, the proportion of runs that visited each position was summed; that sum was divided by the number of positions visited. Positions not visited were not included. While difference in mean density was not great, that for conditions *flat* and $\kappa = 0.25$ were significantly greater than densities for conditions *Group 4 only* and $\kappa = 0.75$, and density for condition $kappa = 0$ was significantly greater than that for condition $\kappa = 0.75$ as well (two-tailed unpaired boostrap state, $p < 0.05$, Cohen 1995). Thus, agents trained under conditions for which less precise actions were easily

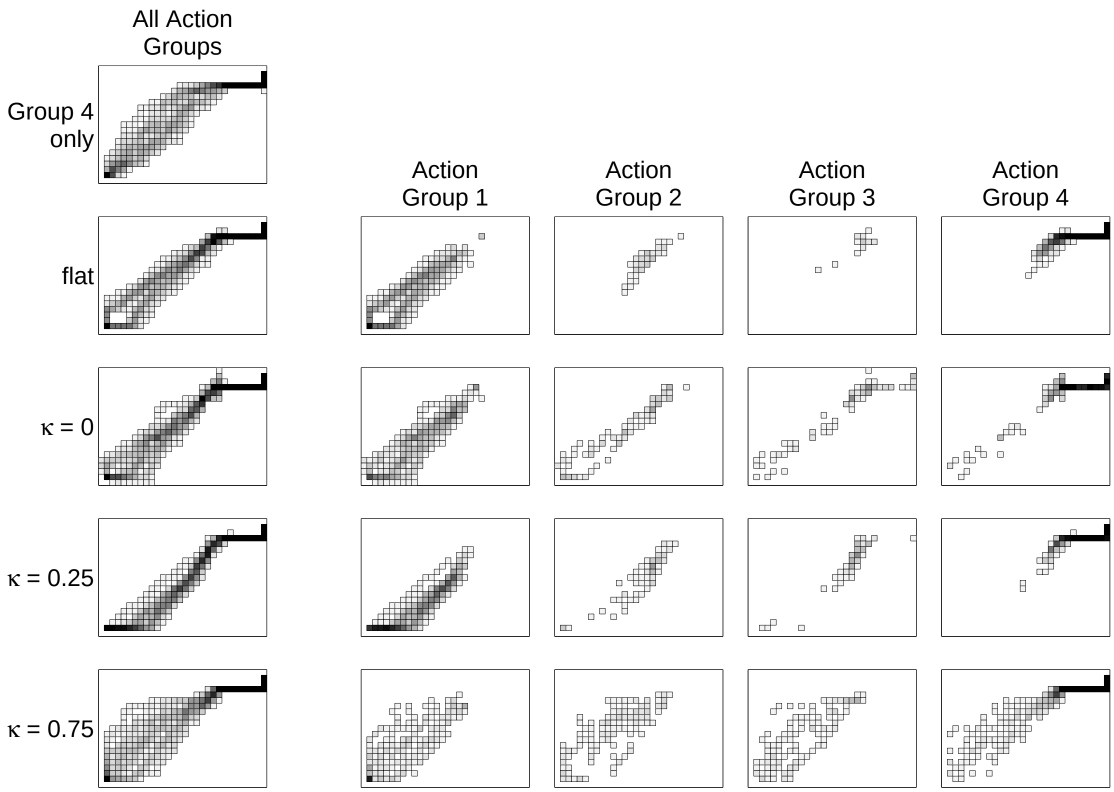

Figure 5.23. Each plot is a representation of the environment (Figure 5.18). Visited positions are drawn and shaded by the proportion of runs that visited that position (the darker the shading, the closer the proportion is to 1). Left column ("All Action Groups"): positions visited by all 20 runs for each of the five conditions. Remaining columns: proportion of positions from which an action from the *Action Group* labeled at top was selected.

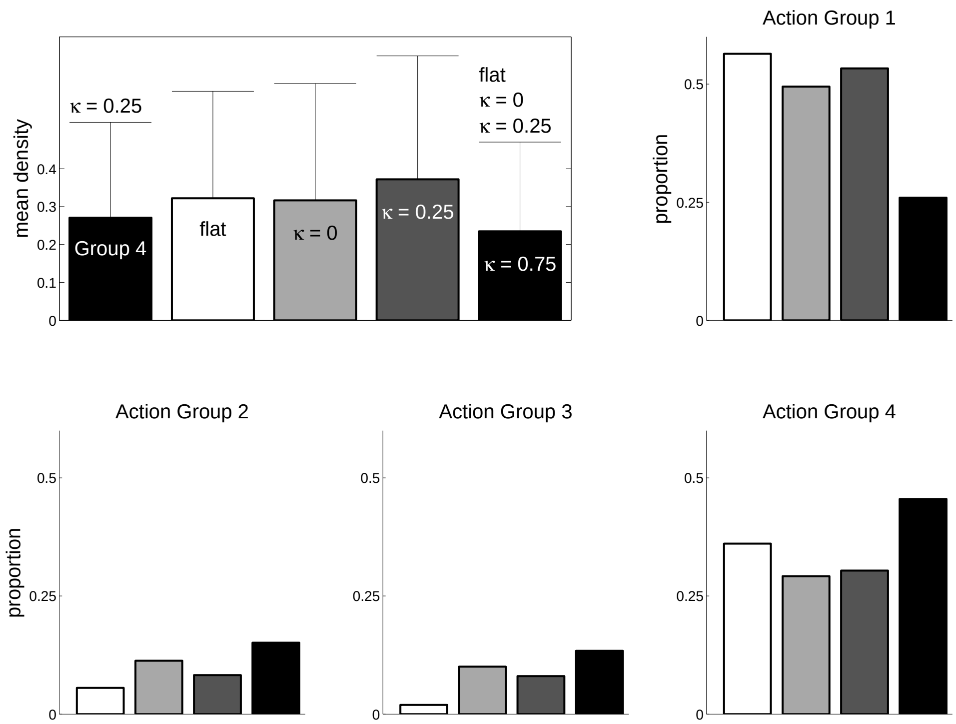

Figure 5.24. Top left: the mean (across the 20 runs) proportion (density) of positions visited under each condition (labeled in each bar). Error bars show standard deviation. The text above the bars for conditions *Group 4 only* and $\kappa = 0.75$ indicates which conditions had a significantly higher mean than those conditions. Remaining plots: proportion of actions selected that came from each *Action Group* (labeled in each plot). Bar colors are labeled in the mean density plot (top left). That for condition *Group 4 only* is not shown since 100% of the actions were from *Action Group* 4.

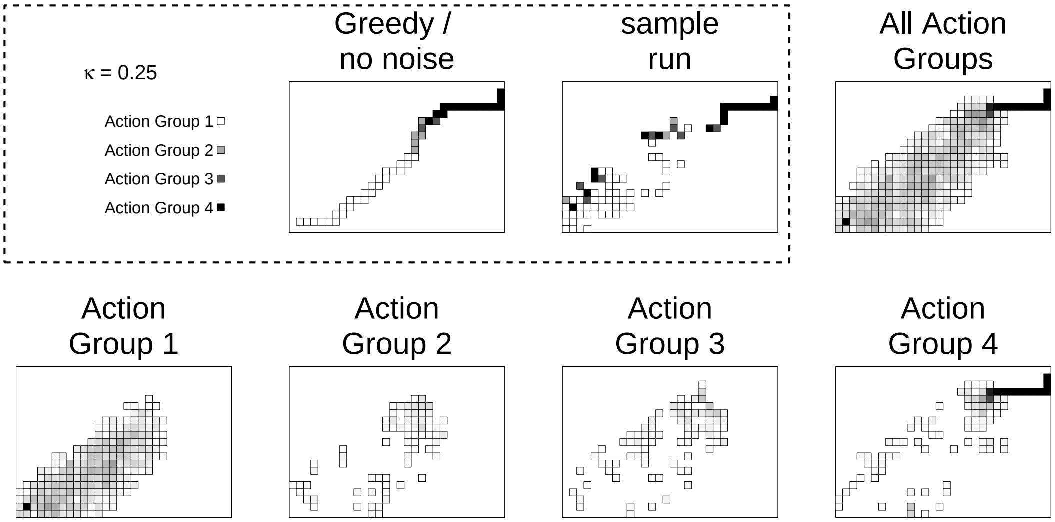

Figure 5.25. Behavior following the Q-values and ν-values from a specific run from condition $\kappa = 0.25$. The greedy policy is shown in the "Greedy / no noise" plot, while a single sample following the greedy policy but with stochasticity in the environment is shown under the plot labeled "sample run." For these two plots (which are within the dashed-line box), shading of the position indicates to which *Action Group* actions selected from each position belonged. The rest of the plots show behavior under the same conditions as the sample run, but for a conglomerate of 40 sample runs; they follow the same conventions as in Figure 5.23.

trained developed a more consistent strategy than agents trained under conditions for which less precise actions were difficult to train or unavailable.

To determine if, indeed, less precise actions were used if available, Figure 5.24 (top right and bottom row) plots the proportion of actions under each of the four *Action Groups* for the four unrestricted conditions (that for agents trained under condition *Group 4 only* are not included). Agents trained under condition $\kappa = 0.75$ used the least percentage of actions from *Group 1* and the most from *Group 4*. Thus, the use of actions from *Group 1*, the least costly but least precise actions, led to a greater consistency in paths taken to the goal.

Behavior

Although the use of imprecise actions led to a consistent *strategy* in the left portion of the environment, actual behavior when stochasticity of the environment is taken into account was highly variable. Figure 5.25 (top left, labeled "Greedy / no noise") plots the strategy, i.e., the greedy policy, of a particular run from an agent trained under the $\kappa = 0.25$ condition. Note that for the two plots in the dashed-line box, shading indicates to which *Action Group* an action chosen from each position belongs. Consistent with the previous section, the agent chose actions from *Action Group* 1 at positions in the left portion of the environment, chose actions from *Action Groups* 2

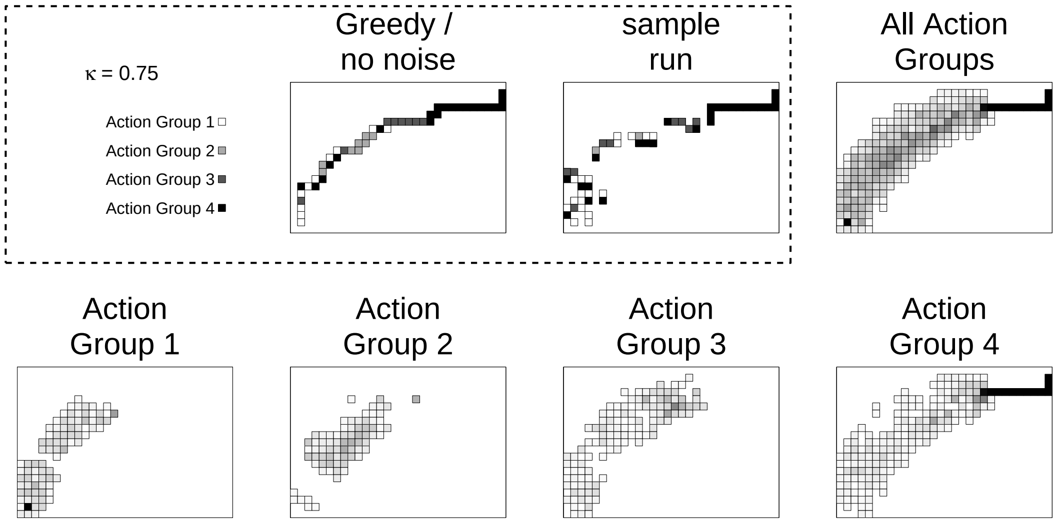

Figure 5.26. Follows same conventions as in Figure 5.25, but for condition $\kappa = 0.75$.

and 3 near the entrance to the path, and chose actions from *Action Group* 1 closer to the path entrance and along the path to the goal.

Figure 5.25 (top middle, labeled "sample run") illustrates a sample run for an agent following the greedy policy (no exploration or learning) but in a stochastic environment (and thus the effects of the imprecisions of the selected actions are revealed). Figure 5.25 (top right) shows resulting conglomerate behavior after 40 sample runs (Figure 5.25, bottom row, illustrates the same information separated by *Action Group*). Variance in behavior in the left portion of the environment was high, while behavior along the path towards the goal was very consistent. Thus, at areas of the environment where precision is not important, precision was sacrificed in favor of more rewarding actions. Such behavior is in agreement with the general motor control strategy (e.g., Todorov and Jordan 2002) of allowing variance to accumulate in task-irrelevant dimensions. However, in this case, such behavior is explained by choice of rewarding yet imprecise actions, analogous to using sensory modalities that estimate state imprecisely.

In contrast, a behavioral strategy that used a lower proportion of *Action Group* 1 actions (as is the case for a sample run from condition $\kappa = 0.75$, Figure 5.26) resulted in less variance. The mean ($\pm$ s.d.) density of positions visited (corresponding to the top right graph, labeled "All Action Groups," of Figures 5.25 and 5.26) is 0.18 ($\pm$0.2) for condition $\kappa = 0.25$, and 0.24 ($\pm$0.23) for condition $\kappa = 0.75$); the difference is significant (two-tailed unpaired boostrap test $p < 0.01$, Cohen 1995).

5.6 Discussion

As discussed at the beginning of this chapter, I model the effects of sensation and perception as leading to a redundancy in state estimate in which there is a trade-off between precision and reward or timing. Most accounts of motor control suggest that cortical planning mechanisms, as opposed to the simpler scheme used by the basal ganglia, are responsible for incorporating uncertainty into behavioral strategy. In this chapter I show that, when a task is repeatedly solved, it is possible for the learning and control mechanisms of the BG to produce behavior that takes uncertainty in state representation into account. In this chapter, state is factored into a position dimension and a goal dimension; I examine uncertainty in each dimension separately.

Sensory evolution

In the experiments dealing with *sensory evolution*, uncertainty in goal evolved over the first eight time steps of a trial from representing each possible goal with a non-zero probability to representing the true goal with certainty. An agent was presented with a variety of types of goal belief evolution and prior distributions (from which goals were chosen). Behaviors exhibited by trained models conformed with that exhibited by humans during a reaching task under an evolving goal representation (Hudson et al., 2007): while belief was uncertain, movement was towards the mean of the prior distribution. As goal belief resolved, movement veered towards the goal. This strategy held true under different prior distributions in both my model and human behavior; it also held true even when the prior distribution was not explicitly represented in the goal belief used by the model.

How behavior under such conditions is learned has not been described (to my knowledge) in the experimental literature. The simulation results have implications regarding the progression of behavior while learning a task and also how behavior differs when trained under different types of goal belief evolution. Briefly, behavior will progress gradually, over the course of learning, from waiting until goal belief is fully resolved and then moving straight towards the true goal to immediately moving towards the mean of the prior distribution (Figures 5.7, 5.9, and 5.10). Deviation from a direction towards the mean of the prior to a direction towards the true goal will occur earlier in a trial as goal belief resolves more quickly (Figures 5.11, 5.13, and 5.14). The different types of goal evolution used in this chapter serve as surrogates for different sets of goal stimuli, each with different perceptual qualities. The dependence of behavior on goal stimuli displayed in the model offers a way to indirectly assess the perceptual qualities of a stimulus.

Perhaps more interesting are the results summarized in Figure 5.16, which plots behavior as controlled by an agent trained under one type of sensory evolution but tested under a different type. This approximates the scenario of learning a task with goal stimuli of one type of perceptual quality, but then asked to accomplish the task with goal stimuli of a different type of perceptual quality. Agents trained under an evolving goal representation but tested with a fully resolved one did not change their behaviors (Figure 5.16, top half). Such a strategy is suboptimal considering the given representation. Also, it indicates that, when the goal stimulus is more easily resolved than the stimuli under which an agent was trained, the relative perceptual clarity had little effect on behavior. In the opposite case, in which agents were trained with a fully resolved goal representation but tested with an evolving one, behavior differed greatly (Figure 5.16, bottom half), indicating that the relative perceptual opaqueness had an effect on behavior.

While there was a fair amount of variance in behavior under some conditions, comparison of performance (Figure 5.17) suggests that, when there is no representation of a prior distribution (as when there is no delay in goal evolution), it is better to train with goal stimuli that are more easily perceived than the expected goal stimuli under testing conditions. On the other hand, when there is some representation of the prior distribution, it is better to train with less discernible goal stimuli, especially for cases when the prior distribution is similar to that of the *two goal* prior. This is because, under the *two goal* prior distribution, only goals 1 and 5 were represented and selected. Therefore, when trained under a fully resolved goal belief, the agents had little experience along the middle path. Consequently, during the delay period for an evolving goal belief, either action northeast or northwest was selected from the starting position. In the latter case, when the delay period ended, the agents were halfway to goal 5. Since, enroute to goal 1, the agents had little experience with positions along the path directly towards goal 5, actions towards goal 1 were not taken until goal belief was almost fully resolved (i.e., the agents had almost reached goal 5).

As behavior in the trained model is controlled by **B**, which uses Q-values to make decisions, the type of sensory representation under which an agent was trained also affects the Q-values it uses. Such a dependence is a design of the the learning mechanisms of **B** (Equation 5.1). As striatal neurons have been suggested to represent Q-values (Samejima et al., 2005), I would expect that striatal neuron activity would also exhibit such a dependence. In particular, when subjects are trained with an evolving goal representation and tested with a fully resolved goal representation, I would expect striatal neuron activity to be very similar (as would behavior, Figure 5.16, top half). Specifically, striatal neural activity would exhibit a high value for action north from the starting position even if the representation of goal 1 was easily perceived. In the opposite case (Figure 5.16, bottom half), though, I would expect striatal neuron activity to be very different. Specifically, striatal neural activity would exhibit action-values weighted by goal belief. An extreme example of this is inferred from the lower right graph of Figure 5.16, which suggests that actions northwest and northeast would be equal in value, while all other actions (including north) would be near zero in value.

Sensory transfer

In the experiments dealing with *sensory transfer*, different sensory modalities were not modeled explicitly; rather, their effects on state estimate were modeled. At each time step, the agent had a choice of executing an action based on imprecise estimates of position or based on precise ones; the higher the precision, the more costly the action. Agents developed a strategy of using imprecise actions in areas of the environment where precision was unimportant — there were no obstacles or costly positions. However, at areas of the environment near costly positions, precise actions were used.

The availability of imprecise actions led to behavior consistent with general motor control strategies. Agents that used a greater proportion of imprecise actions developed a strategy of following a desired path to the goal that was more consistent than that of agents that used a lower proportion of imprecise actions. Thus, *strategy* was more stereotyped when imprecise actions were used. Variance in behavior has been proposed as an objective to be minimized by control strategies in planning in stochastic environments (Harris and Wolpert, 1998). The use of imprecise actions led to an actual increase in variance in *behavior* in areas of state space where precision was not important. This strategy also conforms with a general motor control strategy, that of allowing variance in task irrelevant dimensions (Todorov and Jordan, 2002). However, while in this model such behavior is due to choice of *Action Group*, Todorov and Jordan (2002) show how such behavior minimizes variance in task-relevant dimensions.

I also showed that restricting the initial set of available actions to precise but costly ones, and allowing progressively imprecise but less costly actions to become available as their saliencies increased, improved early learning — the agents were able to learn the task more quickly than agents that had all actions available initially. This strategy is similar to that of *freezing degrees of freedom* (DOF's) (Bernstein, 1967), in which one limits the number of variables to be controlled to facilitate learning. As discussed in Chapter 3 of this thesis, excess DOF's presents our nervous system with an ill-posed problem in that there is no unique solution; freezing DOF's alleviates this problem. For example, human infants have been shown to "lock" their elbow joints in reaching, but, over the course of the first year of life and beyond, progressively allow their elbow joints and other DOF's to contribute to arm movements (Berthier and Keen, 2006; Berthier et al., 1999). Thus, arm movements early in life were not smooth, but were relatively easy to control, while arm movements later in life were much smoother. Berthier et al. (2005) show the utility of this strategy in a theoretical model of reaching.

While it was meant to explain the use of sensory information, the formulation of the *Action Groups* in this model can also be applied to the observation that variability in movements increases as the magnitude of control signals increase (Fitts, 1954; Engelbrecht et al., 2003; van Beers et al., 2004). In other words, larger and faster movements are less precise. If we equate reward with speed, then use of actions from *Action Group* 1 is analogous to moving faster.

Behavioral strategies as produced by the model were developed to take into account the effects of the use of different sensory modalities; such strategies coincide with behavior described to take into account motor variables. However, while I did not explicitly investigate it in this thesis, the use of different sensory modalities allows for additional types of motor behavior. For example, if actions from *Action Group* 1 were analogous to moving using visual information, then transfer of sensory control to actions from other *Action Groups* allows for visual information to be directed elsewhere. In the car driving example, the driver can shift gears while directing his gaze on the road.

Summary

Many theoretical accounts of motor behavior and decision-making assume a static representation of state — state is either precise or at a fixed-level of uncertainty in a sequential-decision task. In this chapter, the effects of sensation and perception on state estimate were modeled by introducing redundancy in state. Essentially, agents were able to trade precision for reward (either directly, as in *sensory transfer*, or by acting to move quickly, as in *sensory evolution*). A similar type of trade-off is the focus of some artificial intelligence models (Zilberstein and Russell, 1993; Zilberstein, 1994; Grass and Zilberstein, 1997; Hansen et al., 1996), which showed how to best trade quality of sensory information and abstraction with computation time in solving simulated tasks. Learning mechanisms attributable to the basal ganglia were able to take into account uncertainty in state representation to produce model behavior that follows similar strategies as human behavior, suggesting the the basal ganglia can contribute to such learning.

CHAPTER 6

DISCUSSION

In the previous three chapters, I presented a general account of motor skill acquisition and showed how the learning and control mechanisms of the basal ganglia (BG), realized with methods from Reinforcement Learning, can produce behavior indicative of motor skills. I focused on the BG because of the prominent role practice plays in motor skill acquisition — the learner must repeatedly interact with the environment in order to gain proficiency.

Chapter 3 showed how the behavioral characteristic of *coarticulation* was achieved through hierarchical optimization — using only total task performance, rather than considering performance during each subtask, as an evaluative measure. Also, the undirected exploration strategy used in my model produced behavior different than a directed exploration or planning strategy, used in most other accounts of coarticulation. The advantages of undirected search were discussed in the discussion section of Chapter 3.

Chapter 4 suggested that the notion of *automatic* could be explained by the richness of the state representation used to make a decision and the computational sophistication of the controller that makes it. Such a notion is similar to most accounts of *habits* (Daw et al., 2005; Yin and Knowlton, 2006; Dickinson, 1985; James, 1890). I also demonstrated the functional advantages of using an automatic controller to select a sequence of actions and how the use of such a controller, along with how experience gained while using the controller is used, leads to gross changes in behavior.

Chapter 5 described the utility of *sensory exploitation*, described differences in behavior due to differences in sensory information, and showed how the BG can learn and control such behavior. The use of the BG contrasts with most explanations of behavior under different sensory conditions, which suggest that cortical planning mechanisms are responsible for most behavioral change.

Many theories of motor skill acquisition focus on how planning mechanisms attributed to cortical areas contribute to developing behavior indicative of motor skills. The work presented in this thesis suggests an alternative method to such development: by using the experiential learning mechanisms of the BG. Thus, behavior indicative of motor skills is not due to just cortical planning mechanisms. In addition, there may be circumstances for which planning mechanisms cannot be used effectively, e.g., when attention must be devoted to other tasks or when an accurate model of the environment and task is difficult to construct. Because the learning mechanisms of the BG are less sophisticated than those of planning mechanisms, they may not require the same attentional and computational resources. Also, they do not require an accurate model of the environment. Even when cortical mechanisms cannot be used effectively, many aspects of motor skills can be acquired through the BG. In addition, I discussed the implications of the functional mechanisms used in each chapter in relation to theoretical research focusing on similar problems. Such a discussion aids in determining why motor skills are useful.

The purpose of this work was to present a basic theoretical framework through which motor skills can be acquired and to show how the learning and control mechanisms of the BG can participate in all aspects of acquisition. However, this work is not meant to be an exhaustive, detailed account of motor skill acquisition or motor control in general, and a few general restrictions were imposed in this study. First, as discussed in the Introduction, the term *motor skill* can be applied to a wide range of behaviors. I focused on serial discrete tasks in mostly closed environments. Techniques used in this thesis can be applied to other types of tasks, particularly if one views a periodic continuous task (e.g., walking) as simply repeating the same serial discrete task. Second, as discussed in Chapter 2, the learning and control mechanisms of areas other than the BG were artificially restricted to show that the mechanisms of the BG can participate in most aspects of motor skill acquisition. Third, the functional mechanisms presented in this thesis were segregated absolutely. However, it is likely that our nervous system employs a scheme that is closer to a continuum of control mechanisms. The segregation is useful in a theoretical model so that the contributions of each control scheme is readily apparent. I discuss how these and other restrictions can be lifted later in this chapter.

Nevertheless, although each behavioral characteristic of a motor skill was examined separately, a similar framework was used to gain proficiency in each task. In essence, planning mechanisms were used to provide a reasonable initial solution to each subtask, and the learning and control mechanisms of the BG were used to improve upon those decisions. Such a progression is supported by experimental research (discussed in Chapter 2) and is similar to the theory of skill acquisition suggested by Fitts and Posner (1967). The use of multiple controllers in solving tasks is not a new idea, but there is some debate as to how to best model the different controllers, how different brain areas participate in behavioral control, and how to best arbitrate between the control signals in general. In the next section, I describe a few other types of multiple control schemes that share some functional attributes with the one I presented in this thesis.

6.1 Multiple Controllers

In the Introduction, I discussed how the decomposition of a complicated task into discrete subtasks aids in learning to perform that task. As an example of another type of decomposition, Haruno and Wolpert (2001) discuss how performance of a task in a large environment is aided by combining multiple controllers, each of which is trained to generate effective control signals in only part of the environment. Rather forcing one controller to learn a complicated environmental structure, a combination of controllers that learned simpler structures is used. In general, a modular approach, where a complicated functions are approximated by a combination of simpler ones, has many advantages.

Top down schemes

Several models use a gross architecture similar to mine: a *general*-purpose controller is used to help train one *specific* for the current task. The model of Daw et al. (2005), described in the discussion section of Chapter 4, showed how such a scheme may explain behavior seen in instrumental conditioning tasks with goal devaluation. Below are several others which focused on motor control.

In *Feedback-error-learning* (Kawato, 1990; Kawato et al., 1987; Kawato and Gomi, 1992), it is assumed that the system (e.g., your body) must execute control signals τ so as to achieve a desired trajectory of states $\mathbf{x}_d$. The general controller is a feedback controller which generates a control signal $\mathbf{u}$ based on the discrepancy between the current trajectory ($\mathbf{x}$) and the desired trajectory. The delay in receiving feedback information is significant, so the corrective motor commands it generates is based on delayed information — it works well for slow movements only. An inverse model can learn to produce motor commands ($\mathbf{u}^*$) which result in the desired trajectory, and therefore does not rely on delayed feedback. The inverse model uses the motor commands generated by the feedback controller as training signals. Overall control is a sum of the two controllers: $\tau = \mathbf{u}^* + \mathbf{u}$. Early in learning, the inverse model is not well trained, so the signals generated by the feedback controller are large; later in training, because the inverse model produces motor commands which result in an accurate trajectory, the trajectory error is small and the feedback controller contributes little to the overall control signal.

Supervised actor-critic RL is a scheme used by Rosenstein (Rosenstein and Barto, 2004; Rosenstein, 2003) in which the control signal, τ, is a weighted sum of the signals prescribed by an RL agent ($\mathbf{a}_e$) and a supervisor ($\mathbf{a}_s$): $\tau = k\mathbf{a}_e + (k-1)\mathbf{a}_s$. The signals generated depend on the state, s, and the supervisor prescribes a route directly to a goal state. However, the signal and route are suboptimal in certain environments. The RL agent is able to try out alternative signals and modifies its output based on a reward, given by the environment, and the signals generated by the supervisor. The weighting factor, k, is state-dependent and increases as the RL agent gains more experience in a particular state. Early in learning, the supervisor dominates, but later in learning, as the RL agent learns the task, its control dominates and performance exceeds that of the supervisor.

Different types of controllers have also been used in sequence to allow for exploration and accomplishment of the task. Randlov et al. (2000) defined an error-correcting controller that would take control when the agent was near the target state. An RL agent would try out actions, and when it reached a state under control of the error-correcting controller, the error-correcting controller would take over. This effectively increased the size of the target region for the RL agent, which cannot search a very large state-action space through exploratory actions alone in a reasonable amount of time. In a *hybrid RL/SL* scheme (Fagg et al., 1997a,b, 1998), an agent, which used a combination of RL and SL to modify control signals, suggested an initial motor command, $\mathbf{a}_e$, that changed the state s, of the system. If the goal state, g, was not reached, a "teacher" generated a sequence of crude corrective movements, $\mathbf{a}_s$, that eventually achieved the goal of the task. In the case of the hybrid RL/SL scheme, the overall control signal, τ, was either $\mathbf{a}_e$ or $\mathbf{a}_s$, not a summation of both. The actions of the RL/SL agent were updated according to reward signals and signals generated by the teacher.

A model of how the brain learns a sequence of movements (Hikosaka et al., 1999; Nakahara et al., 2001), built on behavioral studies using non-human primates (Hikosaka et al., 1995), suggests that the brain uses two parallel control mechanisms. One is in an abstract representation and learns quickly, but executes movements slowly, and the other uses a representation more specific to the actual task, learns slowly, but executes movements quickly. With experience, the latter dominates control.

Bottom up schemes

The focus of the research described above was to show how constructing a controller to accomplish a specific task is facilitated with the cooperation of a general purpose controller. The work presented in this thesis also displayed such a utility. In addition, in Chapter 4, I showed how the existence of a controller that accomplished a specific task (or portion of a task) can facilitate the learning of another task. For example, when playing a game of tennis, it is easier to consider the motor skill of "hit a forehand" as a single unit than to contemplate each associated movement. The *options* (Precup, 2000; Sutton et al., 1999) and *task decomposition* (Dietterich, 2000) frameworks, described in the discussion sections of Chapters 3 and 4, also illustrate such utility.

Similarly, techniques used in robotics show how breaking down complicated tasks with controllers designed to achieve specific goals greatly facilitates learning. The *Control Basis* framework (Huber et al., 1996; Coelho and Grupen, 1997; Grupen and Huber, 2005; Hart et al., 2008b) designed a set of low-level controllers that each generated control signals to achieve some objective, e.g., minimize the net moment about an object to be grasped or instability of the robot itself. These specific controllers are analogous to reflexes and their control signals can be combined according to some priority, e.g., projecting the control signals of a subordinate controller onto the null space of the superordinate controller. If a particular combination of controllers produced some useful behavior, that combination can be designated a controller as well, to be used to accomplish still higher level tasks. The *Subsumption Architecture* (Brooks, 1991) also layers controllers in a hierarchical manner, where a higher-level controller can use a lower-level controller in accomplishing some task. For example, the high level controller of "walk to the corner" will recruit the low level controllers of "avoid obstacles" and "walk." Because the low level controllers can handle all the details of their tasks, the high level controller can devote its resources elsewhere and also plan more efficiently.

The use of specific controllers essentially prevents one from "reinventing the wheel:" if some task is useful and has already been accomplished, an intelligent control scheme will use that information to accomplish more complicated tasks if applicable. Piaget (1952) suggests that this is exactly how we develop complex sensorimotor skills. Human infants begin life with a small set of basic skills, often referred to as reflexes, such as grasping whatever is put in their hands or sucking whatever is put in their mouths. As they develop, those skills are used a building blocks for more complex skills. This concept has been applied to the development of complex skills in the robotics domain (Hart et al., 2006, 2008a).

What constitutes a useful skill? In Chapter 4, "chunks" were developed along trajectories useful for several goals; the discussion of Chapter 4 describes theoretical work in which useful skills were identified by similar methods. However, identification of a useful skill may be accomplished by some inherent *intrinsic reward*, in which a part of the agent's design is to recognize some generic useful outcome and construct a skill to achieve it. For example, an unexpected change in sensory information is a surprising event; thus, a skill might be developed to achieve the state that led to that sensory information. Once the skill is learned, that sensory information is no longer surprising, so the agent will not be "motivated" to continue learning it. Research in RL (Barto et al., 2004; Stout et al., 2005; Şimşek and Barto, 2006) and robotics (Hart et al., 2008b, 2006) show how intrinsic motivation leads to the development of useful skills.

Single-layer schemes

The multiple controller schemes described above, and in my thesis, used general controllers to help train specific ones, and specific controllers to facilitate learning and performance in other tasks. However, different brain areas, each specialized to implement some computational mechanisms, may also work together on the same hierarchical level in solving a task. Here I describe two models that depart from the hierarchical schemes outlined above.

There is evidence that, like cortico-ganglio-thalamo loops, areas of cortex, cerebellum, and deep cerebellar nuclei are interconnected in segregated loops (Houk and Wise, 1995; Houk et al., 1993) Thus, the BG, cerebellum, and cortex may work together to control movement. Houk and Wise (1995) present a conceptual model in which the pattern recognition properties of the BG allow them to detect a sensory context. Striatal neurons are transiently activated and disinhibit thalamus neurons. The thalamus thus initiates activity in a thalamo-cortical self-sustaining positive feedback loop, which executes movement. The cerebellum, using error-related information and synaptic plasticity at the parallel fiber to Purkinje cell synapses, learns to modify movement. The cerebellum also has pattern recognition capabilities, allowing it to detect when the goal of the movement is nearly achieved and initiate the termination of movement. Houk et al. (2007) discusses how this scheme may account for sequence learning and the execution of corrective movements if the goal is not achieved (see also Houk 2005).

In Chapter 2 I discussed evidence that, as a motor skill is learned, control was transferred from cortical areas to the BG. In other types of tasks, where a solution or even the purpose of the task itself is not known a priori, activity in the BG has been shown to *precede* that of frontal cortices (Pasupathy and Miller, 2005; Seger and Cincotta, 2006), suggesting that cortical planning areas (such as the PFC) use information provided by the BG rather than the other way around. Modeling work described in O'Reilly and Frank (2006) (see also O'Reilly et al. 2007) suggests that, in order to plan effectively, the PFC maintains representations of relevant sensory information. The BG, through reward-related mechanisms, learns to determine which representations are useful for solving the current task and thus affect which representations are maintained by the PFC. In this model, the task of creating and maintaining representations is performed by the PFC, while the task of selecting relevant representations is performed by the BG.

6.2 Future directions

This thesis described a general theoretical framework that explains several aspects of motor skill acquisition. One advantage of this framework is its modularity — different learning and control mechanisms are segregated to a large degree. Such modularity enables us to improve upon different areas without reworking the entire structure. Motor control and optimal use of sensory information and processing have motivated much research in psychology and neuroscience; the results of such research can be used to refine the models presented in this thesis. Perhaps the most important direction for future work, though, focuses not on a particular control method, but rather the communication between controllers and the circumstances under which they are used.

Like several other top down multiple controller schemes discussed earlier, the models I present do not offer a method to immediately disengage the controller trained for the specific task and revert control back to the general controller, as would be beneficial when the task suddenly changed. Though such a reversion is not explicitly discussed in Daw et al. (2005), their scheme will revert control from their specific controller (the *Cached-values* controller) to their general (*Tree-search*) controller once the confidence of each controller reflects their true inaccuracies. However, such confidence is determined by evidence; if the task changed, but the confidence measure is based on a long history of previously accurate predictions, it may require some experience for reversion to occur. Nevertheless, only their model, and the feedback error-learning model of Kawato (1990), enables the general controller to assume control without significant retraining of the specific controller. A simple modification to the network dynamics I describe in Chapter 4 is to require the *Action* neurons to be in an "up" state for the *Value*-based controller as well as for the *Automatic* controller, i.e., limit the weights of the projections representing the *Value*-based controller. Thus, if consequences of an action taken close to the expected goal was suddenly largely negative, as would be the case if the task suddenly changed, the *Action* neurons could be set to a "down" state and control reverted to the *Planner*.

Continuing with the topic of controller selection, the different models described under the top down multiple controller scheme use different forms of arbitration. Some (Kawato, 1990; Fagg et al., 1997a) recruit the general controller only if it is needed;

others (Nakahara et al., 2001; Rosenstein and Barto, 2004) increase the contribution of the controller trained for the task as it gains experience; the model presented in Daw et al. (2005) recruits the controller with the most confidence. Dickinson (1985) suggests that action selection is transferred from a controller that explicitly takes the consequence of the action into account to one that does not when the rate of reward no longer increases in response to an increase in the rate of behavior. With the models of Chapters 4 and 5 of thesis, I assume that a controller with higher computational requirements requires more time to make a decision. Thus, the controllers were designed so that simpler controllers make decisions earlier if they are sufficiently trained, resulting in an arbitration scheme that is essentially based on experience. However, the design does not include any explicit advantage to using a simpler controller to make the same decision as a more complicated one. In some cases, it may be easier to use a simpler controller, even if the actions it selects are not optimal for the task at hand. For example, a poor typist might develop motor skills so that he uses only his index fingers. If the typist only typed when he needed to perform well, the cost of using those suboptimal skills is less than the cost of using planning mechanisms to help develop better skills.

Finally, in the first part of this chapter I mentioned how the functional mechanisms described in this thesis were separated absolutely: the *Planner*, *Value*-based controller, and *Automatic* controller were distinct. The segregation enabled us to clearly see how each controller contributed to behavior, but it is more likely that a continuum of control strategies is used by our nervous systems. Such a continuum can be approximated by implementing controllers whose horizons recede according to experience, performance, and predictions. In addition, each of the three controllers I used excited *Action* neurons independently, e.g., the *Automatic* controller first excited *Action* neurons, then the *Value*-based controller did. However, it is likely that the influence of lower controllers would contribute action selection even if they alone cannot. Such a case is easily implemented in the models presented in this thesis and may provide greater insights as to how the behavior arises from the different control mechanisms.

The work presented in this thesis ultimately focuses on decision-making in motor skill acquisition. Thus, a high level of abstraction was used in that both the systems to be controlled the the environments in which they must perform tasks were relatively simple. Such abstraction was intentional as I showed how the functional mechanisms I hypothesize lead to behaviors characteristic of motor skills even with simple systems in simple environments. More realistic systems and environments only further demonstrates the advantages of the functional mechanisms used.

For example, one relatively minor modification to the models presented in this thesis could further demonstrate the utility of redundancy in sensory information. In Chapter 5, I show how different sensory modalities are best used in different parts of the state space. The utility of different modalities is greater if some can only be directed to a subset of the environment at any given time. For example, when typing on a keyboard, we tend (if we're practiced) to look at the computer screen. This isn't because the use of non-visual information is necessarily better for typing. Rather, it is because it is worth it to risk errors and practice typing without looking at our fingers so that we can observe directly the actual output of our typing on the computer screen. The task in Chapter 5 can be augmented so that the agent must control positions on *two* environments, but can only direct the analog of a visual modality to one at a time.

Finally, I acknowledge here an important consideration thus far ignored: dynamics. In order for the techniques presented in this thesis to apply to a wider range of motor skills, they must be adapted to handle dynamics. In many types of motor skills, such as swinging a forehand in tennis or throwing a ball, humans exploit the dynamics of their bodies and environment. The inclusion of dynamics introduces instabilities that make control difficult, so much so that many control systems attempt to minimize their effects. However, much like how the degrees of freedom problem (Bernstein, 1967) presents an opportunity to maximize other objectives (discussed in Chapter 3), the dynamics affords us another dimension in which to increase proficiency and even achieve goals that are impossible achieve without dynamics. One (relatively) recent thesis from my lab, *Learning to Exploit Dynamics for Robot Motor Coordination* (Rosenstein, 2003), focused on some problems similar to the ones discussed in this thesis. As the title indicates, though, dynamics provided another dimension to exploit (see also VanEmmerik et al. 2004). In particular, in the third chapter of Rosenstein (2003), a simulated three-link dynamic arm was charged with the task of lifting a weight. Through the use of proportional-derivative controllers, which significantly reduce complications associated with dynamics, and a search algorithm similar to the undirected exploration I used in Chapter 3, the agent was able to devise ways to lift the weight by exploiting the dynamics of its system. In addition, functional mechanisms attributable to areas of the cerebellum, greatly minimized in my thesis, can be leveraged to provide stability and control in dynamical systems. Such mechanisms can be implemented in ways similar to techniques from the Control Basis framework (discussed earlier, cf. Hart et al. 2008b for a recent paper).

6.3 Concluding Remarks

The work presented in this thesis contributes to answering the questions of *how* and *why* motor skills are acquired. Movement is a relatively direct way to measure behavior and infer decision-making processes. Its study is attractive as it elucidates the strategies used by the nervous system to solve problems. In tackling the problem of how behavior described as automatization is developed, I provided a computational analog of “thought” and “consciousness” (discussed in Chapter 4 of this thesis and also in Daw et al. 2005), which are difficult concepts to formally describe. Because of its abstract nature, the concept behind the computational analog can be applied to non-motor behaviors as well. Consider, for example, the cognitive task of multiplying two integers. A generic *planning* mechanism can be employed to multiply any two integers, x and y, with the following algorithm:

$$x \times y = \sum_{i=1}^{y} x,$$

e.g., $2 \times 3 = 2 + 2 + 2$. Such an algorithm has advantages in that any two integers can be multiplied. However, it has disadvantages in that it takes time and effort to multiply the two integers. However, early in grade school, we are taught a task-specific short-cut: memorize multiplication tables. Rather than repeat the same algorithm every time we must multiply two integers, we memorize common products. Such a strategy (discussed in greater detail in Logan 1988) is analogous to using simpler controllers to select actions frequently selected by a more sophisticated controller.

Other researchers have noted the parallels between cognitive skills and motor skills (VanLegn, 1996; Rosenbaum et al., 2001). The similarities have been discussed on experimental and theoretical levels. Through cortico-ganglio-thalamic loops described in Chapter 2, the BG may contribute to the control of cognitive behaviors (Brown et al., 1997). Tourette's syndrome, which is characterized by uncontrolled motor tics, is often accompanied by obsessive compulsive disorder, which can be characterized by uncontrolled cognitive habits (Leckman and Riddle, 2000; Graybiel and Rauch, 2000). BG dysfunction has been associated with obsessive compulsive disorder (Rapoport and Wise, 1988; Graybiel and Rauch, 2000), and Saka and Graybiel (2003) discuss how Tourette's syndrome may also be related to BG dysfunction (in a sense, the models I present in this thesis are obsessive compulsive due to their inability to revert control to the *Planner*). Graybiel (1997) relates uncontrolled cognitive habits with behavior characterizing schizophrenia, which is associated with an overactivity of dopamine systems. Similarly, Smith et al. (2007) discuss how computational accounts of dopamine function can explain some aspects of psychotic behaviors; Redish et al. (2008) discuss how addiction can be explained by decision-making processes similar to those used in this thesis; and Houk et al. (2007) discuss the implications of their model to schizophrenia.

In this thesis, I investigated the process of motor skill acquisition. The phenomenon of motor skills is well-studied by psychologists and neuroscientists. Thus, a wealth of biological and behavioral data was available from which to construct a computational theory. I based the functional mechanisms employed on those attributable to brain areas and model behavior was related to behavior seen in humans and animals. By using the methods of theoretical neuroscience, this thesis contributes not only to the study of motor skill acquisition, but also to discovery and characterization of the general computational strategies employed by our nervous systems.

BIBLIOGRAPHY

Abbs, J., Gracco, V., and Cole, K. (1984). Control of multimovement coordination: sensorimotor mechanisms in speech motor programming. *Journal of motor behavior*, 16:195–231.

Agostino, R., Berardeli, A., Formica, A., Accornero, N., and Manfredi, M. (1992). Sequential arm movements in patients with parkinson's disease, huntington's diesease and dystonia. *Brain*, 115:1481–1495.

Albus, J. (1971). A theory of cerebellar function. *Mathematical Biosciences*, 10:25–61.

Aldridge, J. W. and Berridge, K. C. (1998). Coding of serial order by neostriatal neurons: a "natural action" approach to movement sequence. *The Journal of Neuroscience*, 18:2777–2787.

Aldridge, J. W., Berridge, K. C., Herman, M., and Zimmer, L. (1993). Neuronal coding of serial order: syntax of grooming in the neostriatum. *Psychological Science*, 4:391–395.

Aldridge, J. W., Thompson, J. F., and Gilman, S. (1997). Unilateral striatal lesions in the cat disrupt well-learned motor plans in a go/no-go reaching task. *Experimental Brain Research*, 113:379–393.

Alexander, G. E. and Crutcher, M. (1990). Functional architecture of basal ganglia circuits: neural substrates of parallel processing. *Trends in Neuroscience*, 13:266–271.

Alexander, G. E., Delong, M. R., and Strick, P. L. (1986). Parallel organization of functionally segregated circuits linking basal ganglia and cortex. *Annual Review of Neuroscience*, 9:357–381.

Aosaki, T., Graybiel, A. M., and Kimura, M. (1994a). Effect of the nigrostriatal dopamine system on acquired neural responses in the striatum of behaving monkeys. *Science*, 265:412–415.

Aosaki, T., Tsubokawa, H., Ishida, A., Watanabe, K., Graybiel, A. M., and Kimura, M. (1994b). Responses of tonically active neurons in the primate's striatum underog systematic changes during behavioral sensorimotor conditioning. *The Journal of Neuroscience*, 14:3969–3984.

Arbib, M. (2002). Schema theory. In Arbib, M. A., editor, *The Handbook of Brain Theory and Neural Networks*, pages 993–998. MIT Press, Cambridge, MA.

Aron, A., Schlaghecken, F., Fletcher, P., Bullmore, E., Eimer, M., Barker, R., Sahakian, B., and Robbins, T. (2003). Inhibition of subliminally primed responses is mediated by the caudate and thalamus: evidence from functional mri and huntington's disease. *Brain*, 126:713–723.

Ashby, F., Ennis, J., and Spiering, B. (2007). A neurobiological theory of automaticity in perceptual categorization. *Psychological Review*, 114:632–656.

Aubert, I., Ghorayeb, I., Normand, E., and Bloch, B. (2000). Phenotypical characterization of the neurons expressing the d1 and d2 dopamine receptors in the monkey striatum. *Journal of Comparative Neurology*, 418:22–32.

Baader, A., Kasennikov, O., and Wiesendanger, M. (2005). Coordination of bowing and fingering in violin playing. *Cognitive Brain Research*, 23:436–443.

Balleine, B. and Ostlund, S. (2007). Still at the choice-point: action selection and initiation in instrumental conditioning. *Annals of the New York Academy of Sciences*, 1104:147–171.

Bar-Gad, I., Morris, G., and Bergman, H. (2003). Information processing, dimensionality reduction, and reinforcement learning in the basal ganglia. *Progress in Neurobiology*, 71:439–473.

Barbas, H. and Pandya, D. N. (1989). Architecture and intrinsic connections of the prefrontal cortex in the rhesus monkey. *Journal of Comparative Neurology*, 286:353–375.

Barto, A. and Dietterich, T. (2004). Reinforcement learning and its relationship to supervised learning. In Si, J., Barto, A., Powell, W., and Wunsch, D., editors, *Handbook of Learning and Approximate Dynamic Programming*, IEEE Press Series on Computational Intelligence, chapter 2, pages 47–64. Wiley-IEEE Press, Piscataway, NJ.

Barto, A. and Mahadevan, S. (2003). Recent advances in hierachircal reinforcement learning. *Discrete Event Dynamic Systems*, 13:341–379.

Barto, A., Singh, S., and Chentanez, N. (2004). Intrinsically motivated learning of hierarchical collections of skills. In *International Conference on Developmental Learning (ICDL)*, LaJolla, CA, USA.

Barto, A. G. (1995). Adaptive critics and the basal ganglia. In Houk, J. C., Davis, J. L., and Beiser, D. G., editors, *Models of Information Processing in the Basal Ganlgia*, pages 215–232. MIT Press, Cambridge, MA.

Battaglia, P. and Schrater, P. (2007). Humans trade off viewing time and movement duration to improve visuomotor accuracy in a fast reaching task. *The Journal of Neuroscience*, 27:6984–6994.

Battaglia-Mayer, A., Caminiti, R., Lacquaniti, F., and Zago, M. (2003). Multiple levels of representation of reaching in the parieto-frontal network. *Cerebral Cortex*, 13:1009–1022.

Bays, P. and Wolpert, D. M. (2007). Computational principles of sensorimotor control that minimise uncertainty and variability. *Journal of Physiology*, 578.2:387–396.

Beiser, D. G. and Houk, J. C. (1998). Model of cortical-basal ganglionic processing: encoding the serial order of sensory events. *Journal of Neurophysiology*, 79:3168–3188.

Benecke, R., Rothwell, J. C., Dick, J. P. R., Day, B. L., and Marsden, C. D. (1986). Performance of simultaneous movements in patients with parkingon's disease. *Brain*, 109:739–757.

Benecke, R., Rothwell, J. C., Dick, J. P. R., Day, B. L., and Marsden, C. D. (1987). Disturbance of sequential movements in patients with parkinson's disease. *Brain*, 110:361–379.

Berns, G. S. and Sejnowski, T. J. (1998). A computational model of how the basal ganglia produce sequences. *Journal of Cognitive Neuroscience*, 10:108–21.

Bernstein, N. A. (1967). *The Coordination and regulation of movements*. Pergamon Press, Oxford, UK.

Berridge, K. C. (1989a). Progressive degradation of serial grooming chains by descending decerebration. *Behavioral Brain Research*, 33:241–253.

Berridge, K. C. (1989b). Substantia nigra 6-ohda lesions mimic striatopallidal disruption of syntactic grooming chains: a neural systems analysis of sequence control. *Psychobiology*, 17:377–385.

Berridge, K. C. and Aldridge, J. W. (2000a). Super-stereotypy i: enhancement of a complex movement sequence by sustemic dopamine d1 agonists. *Synapse*, 37:194–204.

Berridge, K. C. and Aldridge, J. W. (2000b). Super-stereotypy ii: enhancement of a complex movement sequence by intraventricular dopamine d1 agonists. *Synapse*, 37:205–215.

Berridge, K. C. and Fentress, J. C. (1987). Disruption of natural grooming chains after striatopallidal lesions. *Psychobiology*, 15:336–342.

Berridge, K. C., Fentress, J. C., and Parr, H. (1987). Natrual syntax rules control action sequences of rats. *Behavioral Brain Research*, 23:59–68.

Berridge, K. C. and Whishaw, I. Q. (1992). Cortex, striatum, and cerebellum: control of serial order in a frooming sequence. *Experimental Brain Research*, 90:275–290.

Berthier, N., Clifton, R., McCall, D., and Robin, D. (1999). Proximodistal structure of early reaching in human infants. *Experimental Brain Research*, 127:259–269.

Berthier, N. and Keen, R. (2006). Development of reaching in infancy. *Experimental Brain Research*, 169:507–518.

Berthier, N., Rosenstein, M., and Barto, A. (2005). Approximate optimal control as a model for motor learning. *Psychological Review*, 112:329–346.

Berthier, N. E., Singh, S. P., and Barto, A. G. (1993). Distributed representation of limb motor programs in arrays of adjustable pattern generators. *Journal of Cognitive Neuroscience*, 5:56–78.

Bizzi, E., Mussa-Ivaldi, F. A., and Giszter, S. (1991). Computations underlying the execution of movement: A biological perspective. *Science*, 253:287–291.

Boecker, H., Dagher, A., Ceballos-Baumann, A. O., Passingham, R. E., Samuel, M., Friston, K. J., Poline, J. B., Dettmers, C., Conrad, B., and Brooks, D. J. (1998). Role of the human rostral supplementary motor area and the basal ganglia in motor sequence control: investigations with h_2 ^{15}o pet. *Journal of Neurophysiology*, 79:1070–1080.

Bogacz, R. and Gurney, K. (2007). The basal ganglia and cortex implement optimal decision making between alternative actions. *Neural Computation*, 19:442–477.

Bolam, J., Hanley, J., Booth, P., and Bevan, M. (2000). Synaptic organisation of the basal ganglia. *Journal of Anatomy*, 196:527–542.

Bolam, J., Smith, Y., von Krosigk, C., and Smith, A. (1993). Convergence of synaptic terminals from the striatum and the globus pallidus onto single neurones in the substantia nigra and the entopeduncular nucleus. *Progress in Brain Research*, 99:73–88.

Bolivar, V. J., Danilchuk, W., and Fentress, J. C. (1996). Separation of activation and pattern in grooming development of weaver mice. *Behavioural Brain Research*, 75:49–58.

Boraud, T., Bezard, E., Bioulac, B., and Gross, C. (2002). From single extracellular unit recording in experimental and human parkinsonism to the development of a functional concept of the role played by the basal ganglia in motor control. *Progress in Neurobiology*, 66:265–283.

Botvinick, M. (2008). Hierarchical models of behavior and prefrontal function. *Trends in Cognitive Sciences*, 12:201–208.

Botvinick, M. and Plaut, D. (2002). Representing task context: proposals based on a connectionist model of action. *Psychological Research*, 66:298–311.

Botvinick, M. and Plaut, D. (2004). Doing without schema hierarchies: a recurrent connectionist approach to normal and impaired routine sequential action. *Psychological Review*, 111:395–429.

Botvinick, M. and Plaut, D. (2006). Short-term memory for serial order: a recurrent neural network model. *Psychological Review*, 113:210–233.

Breteler, M. K., Hondzinski, J., and Flanders, M. (2003). Drawing sequences of segments in 3d: kinetic influences on arm configuration. *Journal of Neurophysiology*, 89:3253–3263.

Britten, K., Shadlen, M., Newsome, W., and Movshon, J. (1992). The analysis of visual motion: a comparison of neuronal and psychophysical performance. *The Journal of Neurophysiology*, 12:4745–4765.

Brooks, R. (1991). New approaches to robotics. *Science*, 253:1227–1232.

Brotchie, P., Iansek, R., and Horne, M. K. (1991). Motor function of the monkey globus pallidus. 2. cognitive aspects of movement and phasic neural activity. *Brain*, 114:1685–1702.

Brown, L., Schneider, J., and Lidsky, T. (1997). Sensory and cognitive functions of the basal ganglia. *Current Opinion in Neurobiology*, 7:157–163.

Canales, J. J. and Graybiel, A. M. (2000). A measure of striatal function predicts motor stereotypy. *Nature Neuroscience*, 3:377–383.

Carson, R. and Kelso, J. (2004). Governing coordination: behavioral principles and neural correlates. *Experimental Brain Research*, 154:267–274.

Centonze, D., Picconi, B., Gubellini, P., Bernardi, G., and Calabresi, P. (2001). Dopaminergic control of synaptic plasticity in the dorsal striatum. *European Journal of Neuroscience*, 13:1071–1077.

Coelho, J. and Grupen, R. (1997). A control basis for learning multifingered grasps. *Journal of Robotic Systems*, 14:545–557.

Cohen, P. (1995). *Empirical Methods for Computer Science*, chapter 5. MIT Press, Cambridge, MA, USA.

Cohen, R. and Rosenbaum, D. (2004). Where grasps are made reveals how grasps are planned: generation and recall of motor plans. *Experimental Brain Research*, 157:486–495.

Collins, J. J. (1995). The redundant nature of locomotor optimization laws. *Journal of Biomechanics*, 28:251–267.

Cools, A. R. (1980). Role of the neostriatal dopaminergic activity in sequencing and selecting behavioural strategies: facilitation of processes involved in selecting the best strategy in a stressful situation. *Behavioural Brain Research*, 1:361–378.

Cooper, R. and Shallice, T. (2006). Hierarchical schemas and goals in the control of sequential behavior. *Psychological Review*, 113:887–916.

Craig, J. (2004). *Introduction to Robotics: Mechanics and Control*. Prentice Hall, Upper Saddle River, New Jersey, USA, 3 edition.

Cromwell, H. C. and Berridge, K. C. (1996). Implementation of action sequences by a neostriatal site: a lesion mapping study of grooming syntax. *The Journal of Neuroscience*, 16:3444–3458.

Cromwell, H. C., Berridge, K. C., Drago, J., and Levine, M. S. (1998). Action sequencing is impaired in d1a-deficient mutant mice. *European Journal of Neuroscience*, 10:2426–2432.

Daw, N., Niv, Y., and Dayan, P. (2005). Uncertainty-based competition between prefrontal and dorsolateral striatal systems for behavioral control. *Nature Neuroscience*, 8:1704–1711.

Dayan, P. and Abbott, L. (2001). *Theoretical Neuroscience*. MIT Press, Cambridge, MA.

Dearden, R., Friedman, N., and Russell, S. (1998). Bayesian q-learning. In *Proceedings of the Fifteenth National Conference on Aritificial intelligence (AAAI-98)*, pages 761–768.

Debaere, F., Wenderoth, N., Sunaert, S., Hecke, P. V., and Swinner, S. (2003). Internal vs external generation of movements: differential neural pathways involved in bimanual coordination performed in the presence or absence of augmented visual feedback. *Neuroimage*, 19:764–776.

Dickinson, A. (1985). Actions and habits: the development of behavioral autonomy. *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences*, 308:67–78.

Dickson, P. R., Lang, C. G., Hinton, S. C., and Kelley, A. E. (1994). Oral stereotypy induced by amphetamine microinjection into striatum: an anatomical mapping study. *Neuroscience*, 61:81–91.

Dietterich, T. (2000). Hierarchical reinforcement learning with the maxq value function decomposition. *Journal of Artificial Intelligence Research*, 13:227–303.

Dominey, P. (2002). Sequence learning. In Arbib, M. A., editor, *The Handbook of Brain Theory and Neural Networks*, pages 1027–1030. MIT Press, Cambridge, MA.

Dominey, P. F. (1995). Complex sensory-motor sequence learning based on recurrent state representation and reinforcement learning. *Biological Cybernetics*, 73:265–274.

Doya, K. (1999). What are the computations of the cerebellum, the basal ganglia, and the cerebral cortex. *Neural Networks*, 12:961–974.

Doya, K. (2002). Recurrent networks: learning algortihms. In Arbib, M. A., editor, *The Handbook of Brain Theory and Neural Networks*, pages 955–960. MIT Press, Cambridge, MA.

Doya, K. (2007). Reinforcement learning: computational theory and biological mechanisms. *HFSP Journal*, 1:30–40.

Doyon, J. and Benali, H. (2005). Reorganization and plasticity in the adult brain during learning of motor skills. *Current Opinion in Neurobiology*, 15:161–167.

Dum, R. and Strick, P. L. (2002). Motor areas in the frontal lobe of the primate. *Physiology and Behavior*, 77:677–682.

Duncan, J. (2001). An adaptive coding model of neural function in the prefrontal cortex. *Nature Reviews Neuroscience*, 2:820–829.

Engel, K. C., Flanders, M., and Soechting, J. F. (1997). Anticipatory and sequenctial motor control in piano playing. *Experimental Brain Research*, 113:189–199.

Engelbrecht, S. (2001). Minimum principles in motor control. *Journal of Mathematical Psychology*, 45:497–542.

Engelbrecht, S., Berthier, N., and O'Sullivan, L. (2003). The undershoot bias: learning to act optimally under uncertainty. *Psychological Sciences*, 14:257–261.

Ernst, M. and Bulthoff, H. (2004). Merging the senses into a robust precept. *Trends in Cognitive Sciences*, 8:162–169.

Fagg, A., Barto, A. G., and Houk, J. C. (1998). Learning to reach via corrective movements. In *Proceedings of the Tenth Yale Workshop on Adaptive and Learning Systems*, pages 179–185, New Haven, CT.

Fagg, A., Shah, A., and Barto, A. (2002). A computational model of muscle recruitment for wrist movements. *Journal of Neurophysiology*, 88:3348–3358.

Fagg, A., Zelevinsky, L., Barto, A., and Houk, J. C. (1997a). Using crude corrective movements to learn accurate motor programs for reaching. In *presented at NIPS Workshop on Can Artificial Cerebellar Models Compete to Control Robots*, Breckenridge, CO.

Fagg, A., Zelevinsky, L., Barto, A. G., and Houk, J. C. (1997b). Cerebellar learning for control of a two-link arm in muscle space. In *Proceedings of the IEEE Conference on Robotics and Automation*, pages 2638–2644.

Fitts, P. (1954). The information capacity of the human motor system in controlling the amplitude of movements. *Journal of Experimental Psychology*, 47:381–391.

Fitts, P. and Posner, M. (1967). *Human Performance*. Greenwood Press, Westport, CT, USA.

Flaherty, A. W. and Graybiel, A. M. (1991). Corticostriatal transformations in the primate somoatosensory system. projections from physiologically mapped body-part representations. *Journal of Neurophysiology*, 66:1249–1263.

Flash, T. and Hogan, N. (1985). The coordination of arm movements: an experimentally confirmed mathematical model. *Journal of Neuroscience*, 7:1688–1703.

Flash, T. and Sejnowski, T. (2001). Computational approaches to motor control. *Current Opinion in Neurobiology*, 11:655–662.

Fukai, T. and Tanaka, S. (1997). A simple neural network exhibiting selective activation of neuronal ensembles: from winner-take-all to winners-share-all. *Neural Computation*, 9:77–98.

Fuster, J. M. (1997). *The prefrontal cortex: anatomy, physiology, and neuropsychology of the frontal lobe.* Lippincott-Raven, Philadelphia, PA, 3 edition.

Fuster, J. M. (2000). Executive frontal functions. *Experimental Brain Research*, 133:66–70.

Gardiner, T. W. and Kitai, S. T. (1992). Single-unit activity in the globus pallidus and neostriatum of the rat during performance of a trained hard movement. *Experimental Brain Research*, 88:517–530.

Garwicz, M. (2002). Spinal reflexes provide motor error signals to cerebellar modules — relevance for motor coordination. *Brain Research Reviews*, 40:152–165.

Gdowski, M., Miller, L., Parrish, T., Nenonene, E., and Houk, J. (2001). Context dependency in the globus pallidus internal segement during targeted arm movements. *Journal of Neurophysiology*, 85:998–1004.

Gerfen, C. (2004). Dopamine: A neurotransmitter famous enough to have its own movie? Amherst College Neuroscience Seminar Series.

Gerfen, C., Engber, T., Mahan, L., Susel, Z., Chase, T., Monsma, F., and Sibley, D. (1990). D1 and d2 dopamine receptor-regulated gene expression of striatonigral and striatopallidal neurons. *Science*, 250:1429–1432.

Gerfen, C., Miyachi, S., Paletzki, R., and Brown, P. (2002). D1 dopamine receptor supersensitivity in the dopamine-depleted striatum results from a switch in the regulation of erk1/2/map kinase. *Journal of Neuroscience*, 22:5042–5054.

Glimcher, P. (2002). Decisions, decisions, decisions: Choosing a biological science of choice. *Neuron*, 36:323–332.

Goldman-Rakic, P. S. (1995). Celluar basis of working memory. *Neuron*, 14:477–485.

Goldman-Rakic, P. S. and Selemon, L. D. (1997). Functional and anatomical aspects of prefrontal pathology in schizophrenia. *Schizophrenia Bulletin*, 23:437–458.

Grafton, S. T., Hazeltine, E., and Ivry, R. (1995). Funtional mapping of sequence learning in normal humans. *Journal of Cognitive Neuroscience*, 7:497–510.

Grafton, S. T., mazziotta, J. C., Presty, S., Friston, K. J., Frackowiak, R. S. J., and Phelps, M. E. (1992). Functional anatomy of human procedural learning determined with regional cerebral blood flow and pet. *The Journal of Neuroscience*, 12:2542–2548.

Grass, J. and Zilberstein, S. (1997). Value-driven information gathering. In *AAAI Workshop on Building Resource-Bounded Reasoning Systems*, Providence, Rhode Island,USA.

Graybiel, A. (2005). The basal gangli: learning new tricks and loving it. *Current Opinion in Neurobiology*, 15:638–644.

Graybiel, A. M. (1997). The basal ganglia and cognitive pattern generators. *Schizophrenia Bulletin*, 23:459–469.

Graybiel, A. M. (1998). The basal ganglia and chunking of action repertoires. *Neurobiology of Learning and Memory*, 70:119–136.

Graybiel, A. M., Aosaki, T., Flaherty, A. W., and Kimura, M. (1994). The basal ganglia and adaptive motor control. *Science*, 265:1826–1831.

Graybiel, A. M. and Kimura, M. (1995). Adaptive neural networks in the basal ganlgia. In Houk, J. C., Davis, J. L., and Beiser, D. G., editors, *Models of Information Processing in the Basal Ganlgia*, pages 103–116. MIT Press, Cambridge, MA.

Graybiel, A. M. and Rauch, S. L. (2000). Toward a neurobiology of obsessive-compulsive disorder. *Neuron*, 28:343–347.

Gruber, A., Solla, S., Surmeier, D., and Houk, J. (2003). Modulation of striatal single units by expected reward: a spiny neuron model displaying dopamine-induced bistability. *Journal of Neurophysiology*, 90:1095–1114.

Grupen, R. and Huber, M. (2005). A framework for the development of robot behavior. In *2005 AAAI Spring Symposium Series: Developmental Robotics (at Stanford University)*, Stanford, CA, USA.

Guenther, F. (1995). Speech sound acquisition, coarticulation, and rate effects in a neurral network model of speech production. *Psychological Review*, 102:594–621.

Guigon, E., Koechlin, E., and Burnod, Y. (2002). Short-term memory. In Arbib, M. A., editor, *The Handbook of Brain Theory and Neural Networks*, pages 1030–1034. MIT Press, Cambridge, MA.

Gurney, K., Prescott, T., and Redgrave, R. (2001). A computational model of action selection in the basal ganglia. i. a new functional anatomy. *Biological Cybernetics*, 84:401–410.

Gurney, K., Prescott, T., Wickens, J., and Redgrave, P. (2004). Computational models of the basal ganglia: from robots to membranes. *Trends in Neurosciences*, 27:453–459.

Haber, S. (2003). The primate basal ganglia: parallel and integrative networks. *Journal of Chemical Neuroanatomy*, 26:217–330.

Haber, S., Fudge, J., and McFarland, N. (2000). Striatonigrostriatal pathways in primates form an ascending spiral from the shell to the dorsolateral striatum. *The Journal of Neuroscience*, 20:2369–2382.

Hampton, A. N., Bossaerts, P., and O'Doherty, J. (2006). The role fo the ventromedial prefrontal cortex in abstract state-based inference during decicion-making in humans. *The Journal of Neuroscience*, 26:8260–8367.

Hansen, E., Barto, A., and Zilberstein, S. (1996). Reinforcement learning for mixed open-loop and closed-loop control. In *Proceedings of Neural Information Processing Systems Conference (NIPS)*, Denver, Colorado, USA.

Hanson, J. and Jaeger, D. (2002). Short-term plasticity shapes the response to simulated normal and parkinsonian input patterns in the globus pallidus. *The Journal of Neuroscience*, 22:5164–5172.

Harris, C. M. and Wolpert, D. M. (1998). Signal dependent noise determines motor planning. *Nature*, 394:780–784.

Hart, P., Nilsson, N., and Raphael, B. (1968). A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, SSC-4:100–107.

Hart, S., Ou, S., Sweeney, J., and Grupen, R. (2006). A framework for learning declarative structure,. In *Robotics: Science and Systems (RSS) - Workshop on Manipulation for Human Environments.*, Philadelphia, PA, USA.

Hart, S., Sen, S., and Grupen, R. (2008a). Generalization and transfer in robot control. In *Prodeedings of the Eigth International Conference on Epigenetic Robotics,*, University of Sussex, Brighton, UK.

Hart, S., Sen, S., and Grupen, R. (2008b). Intrinsically motivated hierachical manipulation. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Pasadena, CA, USA.

Haruno, M. and Wolpert, D. (2001). Mosaic model for sensorimotor learning and control. *Neural Computation*, 13:2201–2220.

Hatsopoulos, N., Paninski, L., and Donoghue, J. (2003). Sequential movement representations based on correlated neuronal activity. *Experimental Brain Research*, 149:478–486.

Hazeltine, E., Grafton, S. T., and Ivry, R. (1997). Attention and stimulus characteristics determine the locus of motor-sequence encoding. *Brain*, 120:123–140.

Hebb, D. O. (1949). *The Organization of Behavior*. Wiley.

Henderson, J., Carpenter, K., Cartwright, H., and Halliday, G. (2000). Degeneration of the centre median-parafascicular complex in parkinson's disease. *Annals of Neurology*, 47:345–352.

Hikosaka, O., Nakahara, H., Rand, M. K., Sakai, K., Lu, X., Nakamura, K., Miyachi, S., and Doya, K. (1999). Parallel neural networks for learning sequential procedures. *Trends in Neuroscience*, 22:464–471.

Hikosaka, O., Rand, M. K., Miyachi, S., and Miyashita, K. (1995). Learning of sequential movements in the monkey: process of learning and retention of memory. *Journal of Neurophysiology*, 74:1652–1661.

Hikosaka, O., Sakamoto, M., and Sadanari, U. (1989). Functional properties of monkey caudate neurons. ii. visual and auditory responses. *Journal of Neurophysiology*, 61:799–813.

Hikosaka, O., Takikawa, Y., and Kawagoe, R. (2000). Role of the basal ganglia in the control of purposive saccadic eye movements. *Physiological Reviews*, 80:953–978.

Hoff, B. and Arbib, M. (1993). Models of trajectory formation and temporal interaction of reach and grasp. *Journal of Motor Behvavior*, 25:175–192.

Hoover, J. E. and Strick, P. L. (1993). Multiple output channels in the basal ganglia. *Science*, 259:819–821.

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proc. Natl. Acad. Sci. USA*, 79:2554–2558.

Horvitz, J. (2000). Mesolimbiocortical and nigrostriatal dopamine responses to salient non-reward events. *Neuroscience*, 96:651–656.

Horvitz, J. (2002). Dopamine gating of glutamatergic sensorimotor and incentive motivational input signals to the striatum. *Behavioural Brain Resarch*, 137:65–74.

Houghton, G. and Hartley, T. (1995). Parallel models of serial behavior: Lashley revisited. web page: http://psyche.cs.monash.edu.au/v2/psyche-2-25-houghton.html.

Houk, J. (2005). Agents of the mind. *Biological Cybernetics*, 92:427–437.

Houk, J., Bastianen, C., Fansler, D., Fishbach, A., Fraser, D., Reber, P., Roy, S., and Simo, K. (2007). Action selection and refinement in subcortical loops through basal ganglia and cerebellum. *Philosophical Transactions of the Royal Society-B*, 362:1573–1583.

Houk, J. C., Adams, J., and Barto, A. G. (1995). A model of how the basal ganglia generate and use neural signals that predict reinforcement. In Houk, J. C., Davis, J. L., and Beiser, D. G., editors, *Models of Information Processing in the Basal Ganlgia*, pages 249–270. MIT Press, Cambridge, MA.

Houk, J. C., Buckingham, J., and Barto, A. (1996). Models of the cerebellum and motor learning. *Behavioral and Brain Sciences*, 19:368–383.

Houk, J. C., Keifer, J., and Barto, A. G. (1993). Distributed motor commands in the limb premotor network. *Trends in Neuroscience*, 16:27–33.

Houk, J. C. and Wise, S. P. (1995). Distributed modular architectures linking basal ganglia, cerebellum, and cerebral cortex: their role in planning and controlling actions. *Cerebral Cortex*, 5:95–110.

Huber, M., MacDonald, W., and Grupen, R. (1996). A control basis for multilegged walking. In *Proceedings of the 1996 IEEE Conference on Robotics and Automation*, Minneapolis, MN, USA.

Hudson, T., Maloney, L., and Landy, M. (2007). Movement planning with probabilistic target information. *The Journal of Neurophysiology*, 98:3034–3046.

Ingham, C., Hood, S., Mijnster, M., Baldock, R., and Arbuthnott, G. (1997). Plasticity of striatopallidal terminals following unilateral lesion of the dopaminergic nigrostriatal pathway: a morphological study. *Experimental Brain Research*, 116:39–49.

Ito, M. (2000). Mechanisms of motor learning in the cerebellum. *Brain Research*, 886:237–245.

Jackson, G. M., Jackson, S. R., Harrison, J., Henderson, L., and Kennard, C. (1995). Serial reaction time learning and parkinson's disease: evidence for a procedural learning deficit. *Neuropsychologia*, 33:577–593.

James, W. (1890). *The Principles of Psychology*. Christopher D. Green, York University, Toronto, Ontario. online digitized text from webpage Classics in the History of Psychology: `http://psychclassics.yorku.ca/James/Principles/index.htm`.

Jeannerod, M. (1981). Intersegmental coordination during reaching at natural visual objects. In Long, J. and Baddeley, A., editors, *Attention and performance IX*. Erlbaum, Hillsdale, NJ.

Jenkins, I. H., Brooks, D. J., Nixon, P. D., Frackowiak, R. S. J., and Passingham, R. E. (1994). Motor sequence learning: a study with positron emission tomography. *The Journal of Neuroscience*, 14:3775–3790.

Jerde, T., Soechting, J., and Flanders, M. (2003). Coarticulation in fluent finger spelling. *The Journal of Neuroscience*, 23:2383–2393.

Joel, D., Niv, Y., and Ruppin, E. (2002). Actor-critic models of the basla ganglia: new anatomical and computational perspectives. *Neural Networks*, 15:535–547.

Joel, D. and Weiner, I. (1994). The organization of the basal ganglia-thalamocortical circuits: open interconnected rather than closed segregated. *Neuroscience*, 63:363–379.

Joel, D. and Weiner, I. (2000). The connections of the dopaminergic system with the striatum in rats and primates: an analysis with respect to the functional and compartmental organization of the striatum. *Neuroscience*, 96:451–474.

Jog, M. S., Kubota, Y., Connolly, C. I., Hillegaart, V., and Graybiel, A. M. (1999). Building neural representations of habits. *Science*, 286:1745–1749.

Johnson, S. and Grafton, S. (2003). From 'acting on' to 'acting with': the functional anatomy of object-oriented action schemata. *Progress in Brain Research*, 142:127–139.

Jordan, M. (1988). Supervised learning and systems with excess degrees of freedom. Technical report, Massachusetts Institute of Technology, Cambridge, MA, USA.

Jordan, M. I. (1990). Motor learning and the degrees of freedom problem. *Attention and Performance*, 8:796–836.

Jordan, M. I. (1992). Constrained supervised learning. *Journal of Mathematical Psychology*, 36:396–425.

Jordan, M. I. and Rumelhart, D. E. (1992). Forward models: Supervised learning with a distal teacher. *Cognitive Science*, 16:307–354.

Jueptner, M., Frith, C. D., Brooks, D. J., Frackowiak, R. S. J., and Passingham, R. E. (1997). Anatomy of motor learning. ii. subcortical structures and learning by trial and error. *Journal of Neurophysiology*, 77:1325–1337.

Kaelbling, L., Littman, M., and Cassandra, A. (1998). Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 101:99–134.

Kakei, S., Hoffman, D. S., and Strick, P. L. (1999). Muscle and movement representations in the primary motor cortex. *Science*, 285:2136–2139.

Kalman, R. (1960). A new approach to linear filtering and prediction problems. *Transation of the ASME — Journal of Basic Engineering*, 82:35–45.

Kasanetz, F., Riquelme, L., O'Donnell, O., and Murer, M. (2006). Turning off cortical ensembles stops striatal up states and elicits phase perturbations in cortical and striatal slow oscillations in rat in vivo. *Journal of Physiology*, 577:97–113.

Kawato, M. (1990). Feedback-error-learning neural network for supervised motor learning. In Eckmiller, R., editor, *Advanced Neural Computers*, pages 365–372. Elsevier, North-Holland.

Kawato, M. (1999). Internal models for motor control and trajectory planning. *Current Opinion in Neurobiology*, 9:718–727.

Kawato, M., Furukawa, K., and Suzuki, R. (1987). A hierarchical neural-network model for control and learning of voluntary movement. *Biological Cybernetics*, 57:169–185.

Kawato, M. and Gomi, H. (1992). The cerebellum and vor/okr learning models. *Trends in Neuroscience*, 15:445–453.

Keele, S., Mayr, U., Ivry, R., Hazeltine, E., and Heuer, H. (2003). The cognitive and neural architecture of sequence representation. *Psychological Review*, 110:316–339.

Keele, S. W., Jennings, P., Jones, S., Caulton, D., and Cohen, A. (1995). On the modularity of sequence representation. *Journal of Motor Behavior*, 27:17–30.

Kelly, R. and Strick, P. (2004). Macro-architecture of basal ganglia loops with the cerebral cortex: use of rabies virus to reveal multisynaptic circuits. *Progress in Brain Research*, 143:449–459.

Kelso, J. (1982). *Human Motor Behavior: An Introduction*. Lawrence Erblaum, Hillsdale, NJ, USA.

Kemardi, I. and Joseph, J. P. (1995). Activity in the caudate nucleus of monkey during spatial sequencing. *Journal of Neurophysiology*, 74:911–933.

Kent, R. D. and Minifie, F. D. (1977). Coarticulation in recent speech production models. *Journal of Phonetics*, 5:115–117.

Kermadi, I., Jurquet, Y., Arzi, M., and Joseph, J. P. (1993). Neural activity in the caudate nucleus of monkeys during spatial sequencing. *Experimental Brain Research*, 94:352–356.

Kimura, M. (1986). The role of primate putamen neurons in the association of sensory stimulus with movement. *Neuroscience Research*, 3:436–443.

Kimura, M. (1990). Behaviorally contingent property of movement-related activity of the primate putamen. *Journal of Neurophysiology*, 63:1277–1296.

Kimura, M., Aosaki, T., Ishida, H. A., and Watanabe, K. (1992). Activity of primate putamen neurons is selective to the mode of voluntary movement: visually guided, self-initiated, or memory-guided. *Experimental Brain Research*, 89:473–477.

Kimura, M., Kato, M., Shimazaki, H., Watanabe, K., and Matsumoto, N. (1996). Neural information transferred from the putamen to the globus pallidus during learned movement in the monkey. *Journal of Neurophysiology*, 76:3771–3786.

Kitazawa, S., Kimura, T., and Yin, P. (1998). Cerebellar complex spikes encode both destinations and errors in arm movements. *Nature*, 392:494–497.

Knopman, D. and Nissen, M. J. (1991). Procedural learning is imparied in huntington's disease: evidence from the serial reaction time task. *Neuropsychologia*, 29:245–254.

Koch, I. and Hoffman, J. (2000). Patterns, chunk, and hierarchies in serial reaction-time tasks. *Psychological Research*, 63:22–35.

Kording, K. and Wolpert, D. (2006). Bayesian decision theory in sensorimotor control. *Trends in Cognitive Sciences*, 10:319–326.

Lackner, J. and DiZio, P. (1998). Adaptation in a rotating artificial gravity environment. *Brain Reseach Reviews*, 28:194–202.

Lackner, J. and DiZio, P. (2002). Adaptation to coriolis force perturbation of movement trajectory: role of prioprioceptive and cutaneous somatosensory feedback. *Advances in Experimental Medical Biology*, 508:69–78.

Lashley, K. (1951). The problem of serial order in behavior. In Jeffress, L., editor, *Cerebral Mechanisms in Behavior: The Hixon Symposium*, chapter 26, pages 112–136. John Wiley.

Leckman, J. F. and Riddle, M. A. (2000). Tourette's syndrome: when habit-forming systems form habits of their own? *Neuron*, 28:349–354.

Ledoux, J. (1998). *The Emotional Brain*. Simon and Schuster, New York, NY.

Li, Z., Latash, M., and Zatsiorsky, V. (1998). Force sharing among fingers as a model of the redundancy problem. *Experimental Brain Research*, 119:276–286.

Littman, M., Cassandra, A., and Kaelbling, L. (1995). Learning policies for partially observable environments: Scaling up. In *Proceedings of the Twelfth International Conference on Machine Learning*, pages 362–370, San Francisco, CA. Morgan Kaufmann.

Logan, G. (1988). Toward an instance theory of automatization. *Psychological Review*, 95:492–527.

Logan, G., Taylor, S., and Etherton, J. (1999). Attention and automaticity: towards a theoretical integration. *Psychological Research*, 62:165–181.

Mannor, S., Simester, D., Sun, P., and Tsitsiklis, J. (2004). Bias and variance in value function estimation. In *Proceedings of the Twenty-First International Conference on Machine Learning*, pages 308–322.

Marr, D. (1969). A theory of cerebellar cortex. *Journal of Physiology*, 202:437–470.

Marr, D. (1982). *Vision: a computational investigation into the human representation and processing of visual information*. W.H. Freeman, New York, NY.

Martin, K. E., Phillips, J. G., Iansek, R., and Bradshaw, J. L. (1994). Inaccuracy and instability of sequential movements in parkinson's disease. *Experimental Brain Research*, 102:131–140.

Matsumoto, N., Hanakawa, T., Maki, S., Graybiel, A. M., and Kimura, M. (1999). Nigrostriatal dopamine system in learning to perform sequential motor tasks in a predictive manner. *Journal of Neurophysiology*, 82:978–998.

Matsumoto, N., Minamimoto, T., Graybiel, A. M., and Kimura, M. (2001). Neurons in the thalamic cm-pf complex supply striatal neurons with information about behaviorally significant sensory events. *Journal of Neurophysiology*, 85:960–976.

Matsuzaka, Y., Picard, N., and Strick, P. (2007). Skill representation in the primary motor cortex after long-term practice. *Journal of Neurophysiology*, 97:1819–1832.

McFarland, N. and Haber, S. (2002). Thalamic relay nuclei of the basal ganglia form both reciprocal and nonreciprocal cortical connections, linking multiple frontal cortical areas. *The Journal of Neuroscience*, 22:8117–8132.

McGovern, A. (2002). *Autonomous Discovery of Temporal Abstractions from Interaction with an Environment*. PhD thesis, University of Massachusetts Amherst.

McGovern, A. and Barto, A. (2001). Automatic discovery of subgoals in reinforcement learning using diverse density. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 361–368, West Lafayette, IN, USA.

Messier, J., Adamovich, S., Berkinblit, M., Tunik, E., and Poizner, H. (2003). Influecne of movement speef on accuracy and coordination of reaching movements to memorized targets in three-dimensional space in a deafferented subject. *Experimental Brain Research*, 150:399–419.

Middleton, F. A. and Strick, P. L. (2000). Basal ganglia and cerebellar loops: motor and cognitive circuits. *Brain Research Reviews*, 31:236–250.

Miller, E. K. and Cohen, J. D. (2001). An integrative theory of prefrontal cortex function. *Annual Review of Neuroscience*, 24:167–202.

Mink, J. (1996). The basal ganglia: focused selection and inhibition of competing motor programs. *Progress in Neurobiology*, 50:381–425.

Muchiake, H., Saito, N., Sakamoto, K., Sato, Y., and Tanji, J. (2001). Visually based path-planning by japanese monkeys. *Cognitive Brain Research*, 11:165–169.

Mussa-Ivaldi, F. A., Morasso, P., and Zaccaria, R. (1988). Kinematic networks. *Biological Cybernetics*, 60:1–16.

Nakahara, H., Doya, K., and Hikosaka, O. (2001). Parallel cortico-basal ganlgia mechanisms for acquisition and execution of visuomotor sequences - a computational approach. *Journal of Cognitive Neuroscience*, 13:626–647.

Nambu, A., Kaneda, K., Tokuno, H., and Takada, M. (2002). Organization of corticostriatal motor inputs in monkey putamen. *Journal of Neurophysiology*, 88:1830–1842.

Newell, K. (1991). Motor skill acquisition. *Annual Reviews Psychology*, 42:213–237.

Nicola, S., Surmeier, D. J., and Malenka, R. (2000). Dopaminergic modulation of neuronal excitability in the striatum and nucleus accumbens. *Annual Review of Neuroscience*, 23:185–215.

Nissen, M. and Bullemer, P. (1987). Attentional requirements of learning: Evidence from performance measures. *Cognitive Psychology*, 19:1–32.

Niv, Y., Diff, M., and Dayan, P. (2005). Dopamine, uncertainty, and td learning. *Behavioral and Brain Fucntions*, 1. open-access online journal: doi:10.1186/1744-9081-1-6.

Obeso, J., Rodriguez-Oroz, M., Rodriguez, M., Arbizu, J., and Gimenez-Amaya, J. (2002). The basal ganglia and disorders of movement: pathophysiological mechanisms. *News in Physiological Sciences*, 17:51–55.

Opris, I. and Bruce, C. (2005). Neural circuitry of judgement and decision mechanisms. *Brain Research Reviews*, 48:509–526.

O'Reilly, R. and Frank, M. (2006). Making working memory work: a computational model of learning in the prefrontal cortex and basal ganglia. *Neural Computation*, 18:283–328.

O'Reilly, R., Frank, M., Hazy, T., and Watz, B. (2007). Pvlv: the primary value and leanred value pavlovian learning algorithm. *Behavioral Neuroscience*, 121:31–49.

Parent, A. and Hazrati, L. (1995). Functional anatomy of the basal ganglia ii. the place of the subthalamic nucleus and external pallidum in basal ganglia circuitry. *Brain Research Reviews*, 20:128–154.

Passingham, R. E., Toni, I., and Rushworth, M. F. S. (2000). Specialisation within the prefrontal cortex: the ventral prefrontal cortex and associative learning. *Experimental Brain Research*, 133:103–113.

Pasupathy, A. and Miller, E. K. (2005). Different time courses of learning-related activity in the prefrontal cortex and striatum. *Nature*, 433:873–876.

Pedotti, A., Krishnan, V. V., and Stark, L. (1978). Optimization of muscle-force sequencing in human locomotion. *Mathematical Biosciences*, 38:57–76.

Pellis, S. M., neda, E. C., McKenna, M. M., Tran-Bguren, L. T. L., and Whishaw, I. Q. (1993). The role of the striatum in organizing sequences of play fighting in neonatally dopamine-depleted rats. *Neuroscience Letters*, 158:13–15.

Perkins, T. (2002). *Lyapunov methods for safe intelligent agent design*. PhD thesis, Department of Computer Science, University of Massachusetts Amherst.

Perkins, T. and Barto, A. (2001). Lyapunov-constrained action sets for reinforcement learning. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 409–416. International Conference on Machine Learning.

Piaget, J. (1952). *The Origins of Intelligence in Childhood*. International Universities Press.

Platt, R., Fagg, A., and Grupen, R. (2002). Nullspace composition of control laws for grasping. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*.

Precup, D. (2000). *Temporal Abstraction in Reinforcement Learning*. PhD thesis, University of Massachusetts Amherst.

Precup, D., Sutton, R., and Singh, S. (1998). Theoretical results on reinforcement learning with temporally abstract behaviors. In *Proceedings of the 10th European Conference on Machine Learning*, pages 382–393, Chemnitz, Germany. Springer Verlag.

Precup, D., Sutton, R., and Singh, S. (2000). Eligibility traces for off-policy policy evaluation. In *Proceedings of the Seventeenth Conference on Machine Learning*, pages 759–766, Stanford, California, USA. Morgan Kaufman.

Puttemans, V., Wenderoth, N., and Swinnen, S. (2005). Changes in brain activation during the acquisition of a multifrequency bimanual coordination task: from the cognitive stage to advanced levels of automaticity. *The Journal of Neuroscience*, 25:4270–4278.

Ramanathan, S., Hanley, J., Deniau, J., and Bolam, J. (2002). Synaptic convergence of motor and somatosensory cortical afferents onto gabaergic interneurons in the rat striatum. *The Journal of Neuroscience*, 22:8158–8169.

Rand, M. K., Hikosaka, O., Miyachi, S., Lu, X., and Miyashita, K. (1998). Characteristics of a long-term procedural skill in the monkey. *Experimental Brain Research*, 118:293–297.

Rand, M. Y., Hikosaka, O., Miyachi, S., Lu, X., Nakamura, K., Kitaguchi, K., and Shimo, Y. (2000). Characterstics of sequential movements during early learning period in monkeys. *Experimental Brain Research*, 131:293–304.

Randlov, J., Barto, A., and Rosenstein, M. (2000). Combining reinforcement learning with a local control algorithm. In *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 775–782.

Rangel, A., Camerer, C., and Monague, P. (2008). A framework for studying the neurobiology of value-based decision making. *Nature Reviews Neuroscience*, 9:545–556.

Rao, A. and Gordon, A. (2001). Contribution of tactile information to accuracy in pointing movements. *Experimental Brain Research*, 138:438–445.

Rapoport, J. L. and Wise, S. P. (1988). Obsessive-compulsice disorder: evidence for basal ganlgia disorder. *Psychopharmacology Bulletin*, 24:380–384.

Rauch, S. L., Whalen, P. J., Curran, T., McInerney, S., Heckers, S., and Savage, C. R. (1998). Thalamic deactivation during early implicit sequence learning: a functional mri study. *NeuroReport*, 9:865–870.

Reading, P. J., Dunnett, S. B., and Robbins, T. W. (1991). Dissociable roles of the ventral, medial, and lateral striatum on the acquisition and performance of a complex visual stimulus-response habit. *Behavioural Brain Research*, 45:147–161.

Redish, A., Jensen, S., and Johnson, A. (2008). A unified framework for addiction: vulnerabilities in the decision process. *Behavioral and Brain Sciences*, 31:415–437.

Rohanimanesh, K., Platt, R., Mahadevan, S., and Grupen, R. (2004). Coarticulation in markov decision processes. In *18th Annual Conference on Neural Information Processing Systems*, Vancouver, BC, Canada.

Romo, R., Scarnati, E., and Schultz, W. (1992). Role of primate basal ganglia and frontal cortex in the internal generation of movements ii. movement-related activity in the anterior striatum. *Experimental Brain Research*, 91:385–395.

Rosenbaum, D. (1991). *Human Motor Control*. Academic Press, New York, NY, USA.

Rosenbaum, D., Carlson, R., and Gilmore, R. (2001). Acquisition of intellectual and perceptual-motor skills. *Annual Reviews Psychology*, 52:453–470.

Rosenbaum, D., Engelbrecht, S., Bushe, M., and Loukopoulos, L. (1993). A model for reaching control. *Acta Psychologica*, 82:237–250.

Rosenbaum, D., Loukopoulos, L., Meulenbroek, R., Vaughan, J., and Engelbrecht, S. (1995). Planning reaches by evaluating stored postures. *Psychological Review*, 102:28–67.

Rosenbaum, D., Meulenbroek, R., Loukopoulos, L., and Jansen, C. (1999). Coordination of reaching and grasping by capitalizing on obstacle avoidance and other constraints. *Experimental Brain Research*, 128:92–100.

Rosenstein, M. (2003). *Learning To Exploit Dynamics For Robot Motor Coordination*. PhD thesis, University of Massachusetts Amherst.

Rosenstein, M. and Barto, A. (2001). Robot weightlifting by direct policy search. In *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence*, volume 2, pages 839–844.

Rosenstein, M. and Barto, A. (2004). Supervised actor-critic reinforcement learning. In Si, J., Barto, A., Powell, W., and Wunsch, D., editors, *Handbook of Learning and Approximate Dynamic Programming*, IEEE Press Series on Computational Intelligence, chapter 14, pages 359–380. Wiley-IEEE Press, Piscataway, NJ.

Rummery, G. and Niranjan, M. (1994). On-line q-learning using connectionist systems. Technical Report CUED/F-INFENG/TR 166, Engineering Department, Cambridge University, Cambridge, England.

Rushworth, M., Walton, M., Kennerley, S., and Bannerman, D. (2004). Action sets and decisions in the medial frontal cortex. *Trends in Cognitive Science*, 8:410–417.

Sabol, K., Neill, D. B., Wages, S. A., Church, W. H., and Justice, J. B. (1985). Dopamine depletion in a striatal subregion disrupts performance of a skilled motor task in the rat. *Brain Research*, 335:33–43.

Saito, N., Mushiake, H., Sakamoto, K., Itoyama, Y., and Tanji, J. (2005). Representation of immediate and final behavioral goals in the monkey prefrontal cortex during an instructed delay period. *Cerebral Cortex*, 15:1535–1546.

Saka, E. and Graybiel, A. (2003). Pathophysiology of tourette's syndrome: striatal pathways revisited. *Brain and Development*, 25. supplement 1, pages S15–S19.

Samejima, K., Ueda, Y., Doya, K., and Kimura, M. (2005). Representation of action-specific reward values in the striatum. *Science*, 310:1337–1340.

Sanes, J. (2003). Nerocortical mechanisms in motor learning. *Current Opinion in Neurobiology*, 13:225–231.

Schall, J. (2001). Neural basis of deciding, choosing, and acting. *Nature Reviews Neuroscience*, 2:33–42.

Schlegel, T. and Schuster, S. (2008). Small circuits for large tasks: high-speed decision-making in archerfish. *Science*, 319:104–106.

Schmidt, R. (1982). *Motor Control and Learning: A Behavioral Emphasis*. Human Kinetics, Champaign, IL, USA.

Scholz, J. and Schöner, G. (1999). The uncontroller manifold concept: identifying control variables for a functional task. *Experimental Brain Research*, 126:289–306.

Schultz, W. (1998). Predictive reward signal of dopamine neurons. *Journal of Neurophysiology*, 80:1–27.

Schultz, W. and Romo, R. (1992). Role of primate basal ganglia and frontal cortex in the internal generation of movements i. preparatory activity in the anterior striatum. *Experimental Brain Research*, 91:363–384.

Schweighofer, N., Doya, K., and Kuroda, S. (2004). Cerebellar aminergic neuromodulation: towards a functional understanding. *Brain Research Reviews*, 44:103–116.

Seger, C. and Cincotta, C. (2006). Dynamics of frontal, striatal, and hippocampal systems during rule learning. *Cerebral Cortex*, 16:1546–1555.

Seidler, R., Purushotham, A., Kim, S., Ugurbil, K., Willingham, D., and Ashe, J. (2002). Cerebellum activation associated with performance change but not motor learning. *Science*, 296:2043–2046.

Shadmehr, R. and Krakauer, J. (2008). A computational neuroanatomy of motor control. *Experimental Brain Research*, 185:359–381.

Shadmehr, R. and Wise, S. (2005). *The Computational Neurobiology of Reaching and Pointing*. MIT Press, Cambridge, MA.

Sharma, S. and Carew, T. J. (2004). The roles of mapk cascades in synaptic plasticity and memory in *aplysia*: facilitatory effects and inhibitory constraints. *Learna and Memory*, 11:373–378.

Shima, K., Isoda, M., Mushiake, H., and Tanji, J. (2007). Categorization of behavioral sequences in the prefrontal cortex. *Nature*, 445:315–318.

Şimşek, O. and Barto, A. (2006). An intrinsic reward mechanism for efficient exploration. In *Proceedings of the Twenty-Third International Conference on Machine Learning (ICML)*, Pittsburgh, PA, USA.

Smith, A., Li, M., Becker, S., and Kapur, S. (2007). Linking animal models of psychosis to computational models of dopamine function. *Neuropsychopharmacology*, 32:54–66.

Smith, G. (1999). Teaching a long sequence of behavior using whole task training, forward chaining, and backward chaining. *Perceptual and Motor Skills*, 89:951–965.

Smith, Y., Raji, D., Pare, J., and Sidibe, M. (2004). The thalamostriatal system: a highly specific netowkr of the basal ganglia circuitry. *Trends in Neuroscience*, 27:520–527.

Sober, S. and Sabes, P. (2003). Multisensory integration during motor planning. *The Journal of Neuroscience*, 23:6982–6992.

Soechting, J. F. and Flanders, M. (1992). Organization of sequential typing movements. *Journal of Neurophysiology*, 67:1275–1290.

Stout, A., Konidaris, G., and Barto, A. (2005). Intrinsically motivated reinforcement learning: A promising framework for developmental robot learning. In *Proceedings of the AAAI Spring Symposium on Developmental Robotics (Stanford University)*, Stanford, CA, USA.

Sutton, R. (1988). Learning to predict by the methods of temporal differences. *Machine Learning*, 3:9–44.

Sutton, R. and Barto, A. (1998). *Reinforcement Learning*. MIT Press, Cambridge, MA.

Sutton, R., Precup, D., and Singh, S. (1998). Intra-option learning about temporally abstract actions. In *Proceedings of the Fifteenth International Conference on Machine Learning*, pages 556–564, Madison, Wisconsin, USA. Morgan Kaufman.

Sutton, R., Precup, D., and Singh, S. (1999). Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence*, 112:181–211.

Takada, M., Tokuno, H., Hamada, I., Inase, M., Ito, Y., Imanishi, M., Hasegawa, N., Akazawa, T., Katanaka, N., and Nambu, A. (2001). Organization of inputs from cingulate motor areas to basal ganglia in macaque monkey. *European Journal of Neuroscience*, 14:1633–1650.

Takada, M., Tokuno, H., Nambu, A., and Inase, M. (1998). Corticostriatal projections from the somatic motor areas of the frontal cortex in the macaque monkey: segregation versus overlap of input zones from the primary motor cortex, the supplementary motor area, and the premotor cortex. *Experimental Brain Research*, 120:114–128.

Tanaka, S., Doya, K., Okada, G., Ueda, K., Okamoto, Y., and Yamawaki, S. (2004). Prediction of immediate and future rewards differentially recruits cortico-basal ganglia loops. *Nature Neuroscience*, 7:887–893.

Taniwaki, T., Okayama, A., Yoshiura, T., Nakamura, Y., Goto, Y., and Kira, J. (2003). Reappraisal of the motor role of basal ganglia: a function magentic resonance image study. *The Journal of Neuroscience*, 23:3432–3438.

Tanji, J. (2001). Sequential organization of multiple movements: involvement of cortical motor areas. *Annual Review of Neuroscience*, 24:631–651.

Tanji, J. and Hoshi, E. (2008). Role of the lateral prefrontal cortex in executive behavioral control. *Physiological Reviews*, 88:37–57.

Tassinari, H., Hudson, T., and Landy, M. (2006). Combining priors and noisy visual cues in a rapid pointing task. *The Journal of Neuroscience*, 26:10154–10163.

Thomas, G. M. and Huganir, R. L. (2004). Mapk cascade signalling and synpatic plasticity. *Nature Reviews Neuroscience*, 5:173–183.

Thorndike, E. L. (1911). *Animal Intelligence*. Hafner, Darien, CT.

Todorov, E. (2002). Cosine tuning minimizes errors. *Neural Computation*, 14:1233–1260.

Todorov, E. and Jordan, M. (2002). Optimal feedback control as a theory of motor coordination. *Nature Neuroscience*, 5:1226–1235.

Tokuno, H., Inase, M., Nambu, A., Akazawa, T., Miyachi, S., and Takada, M. (1999). Corticostriatal projections from distal and proximal forelimb representations of the monkey primary motor cortex. *Neuroscienc Letters*, 269:33–36.

Toni, I., Krams, M., Turner, R., and Passingham, R. E. (1998). The time course of changes during motor sequence learning: a whole-brain fmri study. *Neuroimage*, 8:50–61.

Torres, E. and Zipser, D. (2004). Simultaneous control of hand displacements and rotations in orientation-matching experiments. *Journal of Applied Physiology*, 96:1978–1987.

Tremblay, L. and Schultz, W. (1999). Relative reward preference in primate orbitofrontal cortex. *Nature*, 398:704–708.

Tunik, E., Poizner, H., Levin, M., Adamovich, S., Messier, J., Lamarre, Y., and Feldman, A. (2003). Arm-trunk coordination in the absence of proprioception. *Experimental Brain Research*, 153:343–355.

Tyrone, M., Kegl, J., and Poizner, H. (1999). Interarticulator co-ordination in deaf signers with parkinson's disease. *Neuropsychologia*, 37:1271–1283.

Uno, Y., Kawato, M., and Suzuki, R. (1989). Formation and control of optimal trajectory in human multijoint arm movement. *Biological Cybernetics*, 61:89–101.

van Beers, R., Haggard, P., and Wolpert, D. M. (2004). The role of execution noise in movement variability. *Journal of Neurophysiology*, 91:1050–1063.

VanEmmerik, R., Rosenstein, M., McDermott, W., and Hamil, J. (2004). Nonlinear dynamical approaches to human movement. *Journal of Applied Biomechanics*, 20:396–420.

VanLegn, K. (1996). Cognitive skill acquisition. *Annual Reviews Psychology*, 47:513–539.

Verschueren, S., Swinnen, S., Dom, R., and Weerdt, W. (1997). Interlimb coordination in patients with parkinson's disease: motor learning deficits and the importance of augmented information feedback. *Experimental Brain Research*, 113:497–508.

Vogels, T., Rajan, K., and Abbott, L. (2005). Neural network dynamics. *Annual Reviews Neuroscience*, 28:357–376.

Volkmann, J., Hefter, H., Lange, H. W., and Freund, J. (1992). Impairment of temporal organizations of speech in basal ganglia diseases. *Brain and Language*, 43:386–399.

Washburn, M. (1916). *Movement and mental imagery*. Houghton Mifflin, Boston.

Watkins, C. (1989). *Learning from delayed rewards*. PhD thesis, Cambridge University.

Watkins, C. and Dayan, P. (1992). Q-learning. *Machine Learning*, 8:279–292.

Watson, J. (1920). Is thinking merely the action of the language mechanisms? *British Journal of Psychology*, 11:86–104.

Whone, A., Moore, R., and Brooks, P. P. D. (2003). Plasticity of the nigropallidal pathway in parkinson's disease. *Annals of Neurology*, 53:206–213.

Wickens, J., Reynolds, J., and Hyland, B. (2003). Neural mechanisms of reward-related motor learning. *Current Opinion in Neurobiology*, 13:685–690.

Wickens, J. and Wilson, C. (1998). Regulation of action-potential firing in spiny neurons of the rat neostriatum in vivo. *Journal of Neurophysiology*, 79:2358–2364.

Wiesendanger, M. and Serrien, D. (2001). Toward a physiological understanding of human dexterity. *News in Physiological Science*, 15:228–233.

Wilson, C. (2008). Up and down states. *Scholarpedia*, 3:1410.

Wilson, C. and Groves, P. (1981). Spontaneous firing patterns of identified spiny neurons in the rat neostriatum. *Brain Research*, 220:67–90.

Wilson, C. and Oorschot, D. (2000). Neural dynamics and surround inhibition in the neostriatum: a possible connection. In Miller, R. and Wickens, J., editors, *Brain Dynamics and the Striatal Complex*, pages 141–149. Harwood, Amsterdam.

Wolpert, D. (2007). Probabilistic models in human sensorimotor control. *Human Movement Science*, 27:511–524.

Yin, H. and Knowlton, B. (2006). The role of the basal ganglia in habit formation. *Nature Reviews Neuroscience*, 7:464–476.

Yoshida, S., Nambu, A., and Jinnai, K. (1993). The distribution of the globus pallidus neurons with input from various cortical areas in the monkeys. *Brain Research*, 611:170–174.

Yoshida, W. and Ishii, S. (2006). Resolution of uncertainty in prefrontal cortex. *Neuron*, 50:781–789.

Zheng, T. and Wilson, C. (2002). The implications of corticostriatal axonal arborizations. *Journal of Neurophysiology*, 87:1007–1017.

Zilberstein, S. (1994). Meta-level control of approximate reasoning: A decision theoretic approach. In *Proceedings of the Eighth International Symposium on Methodologies for Intelligent Systems (ISMIS)*, Charlotte, North Carolina, USA.

Zilberstein, S. and Russell, S. (1993). Anytime sensing, planning and action: A practical model for robot control. In *Proceedings of the 13th International Joint Conference on Artificial Intelligence (IJCAI)*, Chambery, France.