

Encoding Redundancy for Task-Dependent Optimal Control: A Neural Network Model of Human Reaching

Inaugural-Dissertation
zur Erlangung der Doktorwürde der
Philosophischen Fakultät II
der
Julius-Maximilians-Universität Würzburg

Vorgelegt von
Oliver Herbort
aus Würzburg

Würzburg
2008

Erstgutachter: Prof. Dr. Joachim Hoffmann (Universität Würzburg)

Zweitgutachter: Prof. Dr. Ralf Möller (Universität Bielefeld)

Tag des Kolloquiums: 25.1.2008

Acknowledgment

This thesis would never have been possible without the outstanding support I enjoyed to receive from many. First, of course, I thank Julia for, well, everything. I also thank Martin Butz and Joachim Hoffmann for the many discussions, support, funding, and the many liberties I had. Final thanks goes to all the other people at the department for discussions, all the help, and rocking parties.

Contents

Abstract	6
Zusammenfassung	7
Introduction	8
The Inverse Problem	9
Control of the Human Motor System	11
The Musculoskeletal System	12
Spinal Neurons	12
The Brain	13
Short Summary	14
Motor Redundancy	14
Motor Synergies	15
Optimal Control	16
Short Summary	18
Motor Learning	18
Motor Learning and the Resolution of Redundancy	19
General Approaches to Model Motor Learning	20
Short Summary	23
Models of Motor Learning and Models of Flexible Behavior	24
Models of Motor Learning	24
Models of Flexible Behavior	28
Learning and Flexibility	30
Description of the Model	31
SURE_REACH Architecture	31
A Three Joint Planar Arm	33
Hand and Posture Space	34
Posture Memory	35
Motor Controller and Sensorimotor Model	36
Examples for Movement Preparation and Execution	43
Simulation of Human Behavior	48
General Properties	48
Effects of Extensive Training	50
Priming Effects on Movement Execution	53
Benefits of Encoding Kinematic Redundancy	54
Benefits of Encoding Sensorimotor Redundancy	61

Summary of Results	66
Visually Guided Reaching	67
Visual Feedback Controller	68
Evaluation	74
Discussion	75
Discussion	80
Modeling Capabilities and Potential Model Extensions	80
Relation to other Models	84
Validity	86
Summary	89
Conclusion	90
References	91
Appendix	102
Parameter Specifications	102
Simulation Code	103
Simulation Systems	103

Abstract

The human motor system is adaptive in two senses. It adapts to the properties of the body to enable effective control. It also adapts to different situational requirements and constraints. This thesis proposes a new neural network model of both kinds of adaptivity for the motor cortical control of human reaching movements, called SURE_REACH (sensorimotor unsupervised learning redundancy resolving control architecture). In this neural network approach, the kinematic and sensorimotor redundancy of a three-joint planar arm is encoded in task-independent internal models by an unsupervised learning scheme. Before a movement is executed, the neural networks prepare a movement plan from the task-independent internal models, which flexibly incorporates external, task-specific constraints. The movement plan is then implemented by proprioceptive or visual closed-loop control. This structure enables SURE_REACH to reach hand targets while incorporating task-specific constraints, for example adhering to kinematic constraints, anticipating the demands of subsequent movements, avoiding obstacles, or reducing the motion of impaired joints. Besides this functionality, the model accounts for temporal aspects of human reaching movements or for data from priming experiments. Additionally, the neural network structure reflects properties of motor cortical networks like interdependent population encoded body space representations, recurrent connectivity, or associative learning schemes. This thesis introduces and describes the new model, relates it to current computational models, evaluates its functionality, relates it to human behavior and neurophysiology, and finally discusses potential extensions as well as the validity of the model. In conclusion, the proposed model grounds highly flexible task-dependent behavior in a neural network framework and unsupervised sensorimotor learning.

Zusammenfassung

Das motorische System des Menschen ist in zweierlei Hinsicht anpassungsfähig. Es passt sich den Eigenschaften des Körpers an, um diesen effektiv zu kontrollieren. Es passt sich aber auch unterschiedlichen situationsabhängigen Erfordernissen und Beschränkungen an. Diese Dissertation stellt ein neues neuronales Netzwerk Modell der motor-kortikalen Steuerung von menschlichen Zeigebewegungen vor, das beide Arten von Anpassungsfähigkeit integriert (SURE_REACH, Sensusmotorische, unüberwacht lernende, redundanzauflösende Kontrollarchitektur). Das neuronale Netzwerk speichert kinematische und sensumotorische Redundanz eines planaren, dreigelenkigen Armes in aufgabenunabhängigen internen Modellen mittels unüberwachter Lernverfahren. Vor der Ausführung einer Bewegung bereitet das neuronale Netzwerk einen Bewegungsplan vor. Dieser basiert auf den aufgabenunabhängigen internen Modellen und passt sich flexibel äußeren, aufgabenabhängigen Erfordernissen an. Der Bewegungsplan wird dann durch propriozeptive oder visuelle Regelung umgesetzt. Auf diese Weise erklärt SURE_REACH Bewegungen zu Handzielen die aufgabenabhängige Erfordernisse berücksichtigen, zum Beispiel werden kinematische Beschränkungen miteinbezogen, Erfordernisse nachfolgender Aufgaben antizipiert, Hindernisse vermieden oder Bewegungen verletzter Gelenke reduziert. Desweiteren werden zeitliche Eigenschaften menschlicher Bewegungen oder die Ergebnisse von Primingexperimenten erklärt. Die neuronalen Netzwerke bilden zudem Eigenschaften motor-kortikaler Netzwerke ab, zum Beispiel wechselseitig abhängige Raumrepräsentationen, rekurrente Verbindungen oder assoziative Lernverfahren. Diese Dissertation beschreibt das neue Modell, vergleicht es mit anderen Modellen, untersucht seine Funktionalität, stellt Verbindungen zu menschlichem Verhalten und menschlicher Neurophysiologie her und erörtert schließlich mögliche Erweiterungen und die Validität des Modells. Zusammenfassend stellt das vorgeschlagene Modell eine Erklärung für flexibles aufgabenbezogenes Verhalten auf ein Fundament aus neuronalen Netzwerken und unüberwachten sensumotorischen Lernen.

Introduction

Virtually all human capabilities of both every day life and great virtuosity are mediated by the motor apparatus. The plan to open a door requires to reach the doorknob and press it down. The violinist has to move the bow and play the strings with her arms and fingers to delight the audience. And even the most distinguished philosophers had to use their vocal tracts to speak about sophisticated insights or had to make arm and finger movements to write them down. Thus, the ability to control the own body is one of the most basic requisites to implement higher cognitive or social competencies. Even more, it has been suggested that higher cognitive functions or even consciousness have only emerged on the basis of the complex representations and processes that are required to control the human body (Barsalou 1999; Cruse 2003; Grush 2004; Körding, Tenenbaum & Shadmehr 2007; Schubotz 2007; Wolpert, Doya & Kawato 2003). Besides, the control of the human body poses a superb challenge on its own.

This challenge is mastered by all humans (and of course many animals) with astonishing ease. In fact, most people do not think or even know about the daunting problem that brain and body have to solve all the time: how can a goal in the mind be transformed into a movement of the body? To solve this problem, some kind of adaptive mapping from goal representations to efferent neural activations has to exist in the brain. Since movement goals may be represented at some stage as a desired sensory perception or an abstraction thereof, this mapping is a sensorimotor mapping (Elsner & Hommel 2001; Greenwald 1970; James 1890; J. Hoffmann 1993, 2003; Powers 1973). The acquisition of this mapping may be called *motor learning*, and the use of it to guide movements may be called *motor control*.

The sensorimotor mapping is adaptive in two different senses. First, it is adaptive in the sense that it is learned and not innate. This is necessary because the exact body properties are unforeseeable by any innate mechanism and because the body changes continuously. Limbs grow, fatigue reduces muscle strength, tools extend our body, and injuries reduce its movability. In humans, the capability to make accurate goal-directed movements only develops during infancy (von Hofsten 2003).

Second, the mapping is adaptive in the sense that the mapping has to adapt quickly to different situations. This situational adaptation is possible because virtually all goals may be implemented by different movements, and it is necessary to adjust a movement to the specific requirements of each new task. For example, you might type your password slowly and carefully to log in to your computer because you won't be able to detect and correct an error. However, you might write a text less carefully and much faster because most errors are highlighted by your word processor and can be quickly corrected. Likewise, if your goal is to grasp a wine glass, you might grasp it in very different ways dependent if you intend to cheer to your friends or put it in the dish-washer.

In this thesis, I will present a novel computational model of motor learning and the control of human reaching movements that focuses on adaptivity in both senses¹. The

¹Most of the results of this dissertation have been reported in Butz, Herbort und Hoffmann (2007) and Herbort und Butz (2007).

next section gives a more formal description of the problem of turning a goal into a movement. Then I will highlight computational and representational problems which make the acquisition and use of the mapping difficult. First, the human motor system will be briefly described, emphasizing on the central requirements that enable behavioral control. Second, the problem of motor redundancy and accounts for its solution are discussed. Third, different approaches to motor learning are outlined and the problem of detecting relationships between the own efferent signals and body motions will be scrutinized. Starting from these problems, current computational models of motor learning or motor control are reviewed. It will become apparent, that models of motor learning on the one side and models that account for flexible adaptive control on the other side are well developed but that unifying approaches that combine both kinds of adaptivity are lagging behind. As the key contribution of this thesis, a novel computational model for motor learning and control of goal-directed arm movement is presented that aims to fill this gap. The subsequent evaluation and application of the model will show that it matches behavioral human and primate data and that it accounts for both, movement preparation and movement execution. A final discussion about potential extensions as well as psychological and neurophysiological validity concludes the thesis.

The Inverse Problem

A goal can be seen as a desired consequence of some actions or movements. The ability to move goal-directedly bases on the ability to transform such a desired consequence into a movement. This requires the internal inversion of the flow from cause to effect, from movements to their consequences. The problem of finding such a transformation may therefore be termed the *inverse problem* (Kalveram 2004).

The inverse problem was already discussed in the early days of psychology (Herbart 1825; Lotze 1852). Of course, the solution of this problem is crucial for motor control and has been referenced in early theories (Bernstein 1967). Later, Adam's (1971) "memory trace" and Schmidt's (1975) "recall schema" have been proposed for the acquisition of a solution to the inverse problem. However, these accounts did not pay tribute to the immense flexibility of human movement production and the capability to even exert precise and sophisticated movements that have never been executed before. In recent years, the term *internal inverse model*² has been assigned to neural structures that solve the inverse problem (Kawato 1999; Sabel 2000; Scott & Norman 2003). It emphasizes that motor control is not based on a simple, fixed relationship between goals and efferent signals but that a general, detailed representation of the body and how it moves underlies motor behavior (Conditt, Gandolfo & Mussa-Ivaldi 1997; Cruse, Steinkühler & Burkamp 1998; Kawato 1999).

After these general comments, it should be noted that the brain doesn't solve a

²Note, that the word "model" is used in two different meanings in this thesis. First, "model" might refer to some kind of understanding that the brain has from the motor system (e.g. "internal model"). Second, "model" might refer to some kind of understanding that scientists have from the brain (e.g. "a model of motor control"). If a sentence requires mentioning both kinds of models, the second kind is termed "approach" or "theory" to reduce confusion.

single inverse problem as such. It is most likely that a goal, like wanting to switch on the lights in a room, is not directly mapped onto motor commands but that a number of intermediate representations exists. Thus, the big inverse problem of choosing efferent signals to switching on the lights may be divided in many smaller inverse problems: for example, the selection of an effector, the selection of a movement trajectory of that effector, and finally a series of efferent neural impulses that initiate a movement.

However, the exact nature of the computations and its intermediate states are debated. Figure 1 shows a number of possible pathways from a goal for a reaching movement, a desired hand location, to motor commands. Some assume that a targeted hand

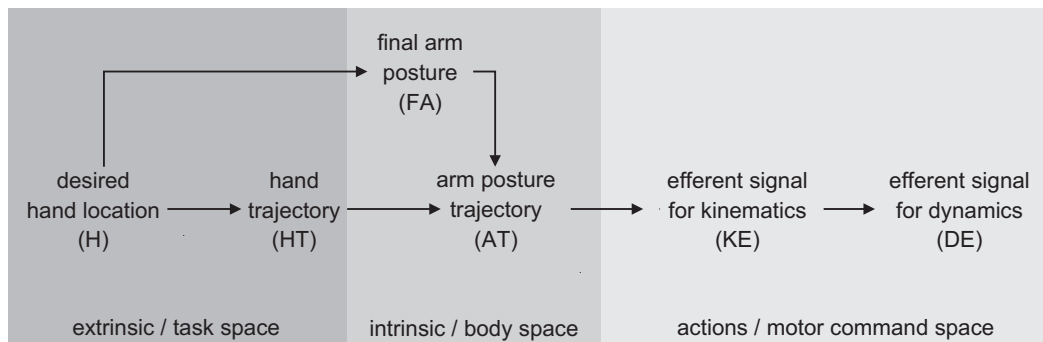


Figure 1. The transformation of an extrinsically encoded movement goal into efferent signals could follow different paths of intermediate representations. Not all or even no intermediate stages have to be explicitly represented.

coordinate is first explicitly transformed into a specific arm posture and then, based on initial and final posture, a trajectory of arm postures is traced (e.g. Rosenbaum, Loukopoulos, Meulenbroek, Vaughan & Engelbrecht 1995). Others propose that the trajectory is not planned in posture space but in visually encoded hand space (Flanagan & Rao 1995; Flash & Hogan 1985). Again others state that neither the final posture nor the arm posture trajectory or hand trajectory of a movement is explicitly planned (e.g. Bullock, Grossberg & Guenther 1993; Cisek & Kalaska 2005; Morasso, Sanguineti & Spada 1997). Likewise, it is debated if the efferent signals issued by the motor system directly control muscle force or related dynamic properties like joint torques (Barto, Fagg, Sitkoff & Houk 1999; Gottlieb 2000; Wolpert & Kawato 1998; Todorov 2000) or if only kinematic features of the movement are controlled (Feldman 1966; Gribble, Ostry, Sanguineti & Laboissiere 1998). Table 1 lists some models and the computations they assume or account for. Finally, the view that the different intermediate representations are computed one after another, resulting in the output of a prespecified movement has been challenged. Recent experiments suggest that movement initiation might even precede target selection (Cisek 2005; Cisek & Kalaska 2005; Song & Nakayama 2006).

The computational model presented in this thesis proposes that the inverse problem is solved by first determining possible end-postures and then approaching these without

planning a trajectory. However, although this is backed by empirical findings I do not want to make a strong claim about the modalities or properties of any intermediate representation but want to show that an intermediate representation of redundant solutions of an inverse problem is the key to understand behavioral flexibility.

Regardless of what the specific computations in the brain are, at least some if not all of the inverse motor problems are not easy to solve. The next sections will highlight three issues related to encoding a solution of the inverse problem. First, the solution of at least some of the inverse problems has to be a time-varying efferent signal. Second, the problem of redundancy requires the selection of a specific solution to the inverse problem among many seemingly equivalent possibilities. Third, possible accounts for the acquisition of inverse models by motor learning are scrutinized.

Table 1: Computational Models of Motor Control

Stages	References
H \rightarrow FA \rightarrow AT	Rosenbaum, Engelbrecht, Bushe & Loukopoulos 1993a, 1993b; Rosenbaum, Loukopoulos, Meulenbroek, Vaughan & Engelbrecht 1995; Rosenbaum, Meulenbroek & Vaughan 2001; Rosenbaum, Meulenbroek, Vaughan & Jansen 1999, 2001
H \rightarrow HT,AT, FA	Cruse & Steinkühler 1993; Cruse, Steinkühler & Burkamp 1998
H \rightarrow KE	Baraduc, Guigon & Burnod 2001; Berthier 1996; Berthier, Singh, Barto & Houk 1992, 1993; Bullock, Grossberg & Guenther 1993; Cisek 2006; Morasso, Sanguinetti & Spada 1997; Srinivasa & Grossberg 2007
H \rightarrow DE	Berthier, Rosenstein & Barto 2005; Rokni, Richardson, Bizzi & Seung 2007
AT \rightarrow DE	Gribble & Ostry 2000; Hirayama, Kawato & Jordan 1993
AT,FA \rightarrow DE	Barto, Fagg, Sitkoff & Houk 1999; Haruno, Wolpert & Kawato 2001, 2003; Herbort, Butz & Hoffmann 2005a; Kawato 1990; Kositsky & Barto 2002; Wolpert & Kawato 1998
Abbreviations: AT: arm posture trajectory, DE: efferent signals (dynamics), FA: final arm posture, H: final hand location, HT: hand trajectory, KE: efferent signals (kinematic).	

Control of the Human Motor System

In the following, three aspects of the human motor system and their relevance for the control of goal-directed movements are described: the musculoskeletal system, the spinal neural circuitry, and the brain. The sections about the musculoskeletal system and the spinal circuitry will focus on how these systems support central control of goal-directed movements and on the structure of required control signals. The section about

the brain will roughly outline the roles of different motor areas and interconnections between them and other areas. Additionally, it will position the computational model in the motor system.

The Musculoskeletal System

At first, of course, the setup of the skeleton and the muscles shapes the motor system. This system seems highly complex but it evolved to enable the CNS to interact with the environment as easily as possible. First, the geometrical configuration of the human body and the joint functionality facilitate important behavior, like walking or grasping objects and manipulating them under visual control (Shadmehr & Wise 2005). Second, the actuators of the body, the muscles, enhance control stability. The mechanics of a single muscle are comparable to a damped spring and each limb is actuated by at least two antagonistic muscles (Birbaumer & Schmidt 1996).

Both features enable to compensate on a purely mechanical basis for perturbations, resulting in a delay-free feedback loop. Even more, it has been hypothesized that by centrally setting specific α -motoneuron activations in antagonistic muscles attached to a limb, the limb might come to rest at a defined equilibrium position, regardless of initial limb posture or perturbations (Bizzi, Polit & Morasso 1976; Polit & Bizzi 1979). Thus, all what the brain would have to do to exert goal-directed movements is to generate a specific constant *alpha*-motoneuron activation, thus rendering motor control a rather simple task. However, this α - equilibrium point hypothesis could not be maintained in its simplicity. Nevertheless, the mechanics and arrangement of muscles clearly facilitate stable control (Bizzi, Accornero, Chapple & Hogan 1984; Buehrmann & Di Paolo 2006).

Spinal Neurons

The spinal neural circuitry can be considered the second part of the motor system. It provides neural circuitry for simple repetitive behavior (central pattern generators) and comprises spinal reflex arcs (Dietz 2003; Scott 2004). Whereas the ability to produce certain forms of rhythmic behavior like walking or swimming is not of paramount importance to voluntary movement control, especially reaching, reflex arcs stabilize the motor apparatus in a way that cannot be provided by the musculoskeletal system alone. For example, the equilibrium position of a limb clamped between a pair of muscles does not only depend on the contractional force exerted by the muscles but also on gravitational or inertial forces.

The monosynaptic reflex arcs can compensate for limb displacements due to external loads. It relays the perceived muscle length to α -motoneurons. By increasing muscle force when the muscle is involuntarily stretched, a certain muscle length and hence limb position can be maintained even if external loads apply.

Also, spinal circuitry connects different muscles, for example by inhibitory connections between antagonists. These findings have inspired the lambda version of the equilibrium point theory, which states that the final limb position of voluntary movements is centrally controlled by modifying reflex gains of the spinal circuitry (Adamovich, Levin & Feldman 1997; Feldman 1966; Feldman & Levin 1995; Mussa-Ivaldi & Bizzi 2000; Mussa-Ivaldi,

Giszter & Bizzi 1994). It has been hotly debated if the adjustment of reflex gains is sufficient to account for voluntary movement (Bizzi, Hogan, Mussa-Ivaldi & Giszter 1992; Gomi & Kawato 1997; Gottlieb 2000; Ostry & Feldman 2003). Nevertheless, it seems that central reflex arcs help to stabilize body postures without the need to react centrally to perturbing torques or forces (Loeb, Brown & Cheng 1999; Hof 2003).

However, equilibrium point theories only account for the control of the final position of a movement but not for control of the trajectory (McIntyre & Bizzi 1993). It has been noted that these models can exert control over trajectories by shifting the reflex gains along a more or less complex (virtual) trajectory (Gribble et al. 1998; Gribble & Ostry 2000). This reduces the original simplicity of these models because they now require the central generation of a complex time varying control signal. Hence, regardless if central motor control is mediated by reflex gains or if muscle forces are directly controlled (Gomi & Kawato 1997; Gottlieb 2000; Gribble & Ostry 1999; Todorov 2000) it seems that the brain has to provide a complex, time varying efference pattern to realize arbitrary voluntary movement.

The Brain

The brain completes the human motor system. The most prominent motor area is the primary motor cortex which connects (along with premotor areas) to many spinal interneurons or directly to motoneurons by the cortico-spinal-tract. It has been suggested that it encodes muscle force (Kakei, Hoffman & Strick 1999; Todorov 2000), target posture (Aflalo & Graziano 2006), actual arm posture (Ajemian, Bullock & Grossberg 2000) as well as more abstract movement properties like hand movement direction (Cisek 2005; Georgopoulos, Caminiti, Kalaska & Massey 1983). The primary motor cortex is closely connected to the primary somatosensory cortex, which provides proprioceptive information, as well as premotor areas and the supplementary motor area. The former seems to play an important part in representing movement goals and in planning and executing movements whereas the latter is involved in retrieving motoric information from memory (Afifi & Bergman 2005; Battaglia-Mayer, Caminiti, Lacquaniti & Zago 2003; Cisek & Kalaska 2005; Scott 2004). Both areas receive proprioceptive information (from area 5) as well as exteroceptive visual input (from area 7). This enables premotor areas to couple visual feedback to motor behavior. Additionally, they are connected to the prefrontal cortex which is generally associated to higher cognitive functions. For example, it encodes abstract movement categories (Shima, Isoda, Mushiake & Tanji 2007).

Additionally, many subcortical areas contribute to the generation of movements. First, the basal ganglia interact with the primary motor cortex via thalamic connections and seem to play a major role in context switching and action selection or inhibition (Redgrave, Prescott & Gurney 1999). Second, the cerebellum plays a major role in the acquisition and execution of motor behavior (Barlow 2002; Hikosaka, Nakamura, Sakai & Nakahara 2002). It is massively linked to the primary motor cortex by the cortico-pontocerebellar tract and affects spinal neurons mediated via the red nucleus, however mainly affecting shoulder and arm movements (Afifi & Bergman 2005). The cerebellum is

frequently seen as a side loop of the corticospinal system that adjusts or replaces cortical motor commands to enhance control, especially in face of complex dynamic interactions of different limbs (Barto et al. 1999; Berthier et al. 1993; Contreras-Vidal, Grossberg & Bullock 1997; Doya 1999; Houk, Buckingham & Barto 1996; Kawato & Gomi 1992; Schweighofer, Arbib & Kawato 1998; Schweighofer, Spolstra, Arbib & Kawato 1998). Also, its role as predictive or forward model has been emphasized (Desmurget & Grafton 2000; Miall, Weir, Wolpert & Stein 1993). Additionally, areas in the brain stem help to balance the body or regulate spinal central pattern generators (Scott 2004; Shadmehr & Wise 2005).

Short Summary

The last section gave an overview over the requirements to central control signals and which brain areas contribute to their production. Even though the muscle physiology and the spinal circuitry facilitates stable control of the body, a central controller must be able to produce complex, time-varying control signals to implement goal-directed movements.

In the following, computational problems of the acquisition and control of goal-directed behavior are examined from the perspective of motor cortical areas. This thesis will contribute to the question of how these areas can generate efferent signals that ultimately cause the body to reach a proprioceptively or exteroceptively encoded goal and how this competency is acquired. In other words, this thesis addresses the question: how does the brain solve the *inverse problem*.

Motor Redundancy

The motor system offers redundant possibilities to reach a specific goal because goals are usually specified in a low dimensional task space but efferent signals have to be specified in a higher dimensional space. To solve the inverse problem, it is not only necessary to link goal representations to the efferent signals that bring the goal about but also to select a specific efferent pattern among many alternatives. At last, only one movement may be carried out at a time.

The motor system is redundant on many levels. At first, there is redundancy on the neural level. Starting from the motor cortex, many efferent signals project to the spinal cord. The corticospinal tract alone is comprised of roughly a million neurons (Afifi & Bergman 2005). In the spinal cord these neurons are relayed on motor units, each of which actuate a fraction of the fibers that comprise a single muscle. Thus, a certain muscle contraction might be caused by many motoneuron activation patterns.

Second, there is kinematic and dynamic motor redundancy. The desired final hand position can be realized in many different ways. This kinematic redundancy applies to the human arm. For example, movements terminating at identical hand locations assume different arm configurations, depending on the starting posture (Cruse, Brüwer & Dean 1993; Soechting, Buneo, Herrmann & Flanders 1995). Moreover, the final posture of a pointing movement (M. H. Fischer, Rosenbau & Vaughan 1997) or the way an object is grasped (Weigelt, Cohen & Rosenbaum 2006; Weigelt, Kunde & Prinz 2006;

Rosenbaum et al. 1990) may be adjusted to facilitate subsequent movements. Likewise, the transition between two arm postures might be shaped in different ways. The most compelling example of this is that different trajectories to the same goal are used if obstacles block a direct transition from a start-point to the goal (Dean & Brüwer 1994) or did so recently (Jax & Rosenbaum 2007). Also, if only the visually perceived trajectory of a movement is distorted but not its start- and end-point, humans produce alternative arm trajectories (Flanagan & Rao 1995, cf. Lackner & DiZio 1998). Recent experiments revealed that especially task irrelevant properties of a movement show considerable trial-to-trial variability (Latash, Scholz & Schöner 2002; Scholz, Schöner & Latash 2000).

To summarize, a vast number of cortical neurons are used to control the only seven mechanical degrees of freedom of the human arm. Likewise, the mechanical degrees of freedom of the arm enable to reach most goals with an infinite number of possible muscle activation sequences, follow an infinite number of arm or hand trajectories and can assume an infinite number of final arm postures. Of this abundance of possibilities, only a single one can be applied at a time. The next sections describe how *motor synergies* and *optimal control* reduce and resolve motor redundancy.

Motor Synergies

Not all neural and muscular degrees of freedom have to be controlled because many neurons, muscles and joints are not independent but are organized in motor synergies (Bernstein 1967). Motor synergies can be found on many levels. On the neural level, a muscle, which is innervated by several motoneurons, could in principle be actuated by any combinations of motoneuron activations, resulting in one degree of freedom for each motoneuron. However, motoneurons are recruited in certain patterns, depending on cell size and excitatory threshold. A low efferent signal causes the activation of only the smallest motor units. If the efferent signal increases, medium-sized and later on larger motor units are additionally activated (de Luca, LeFever, McCue & Xenakis 1982; Henneman, Somjen & Carpenter 1965). The CNS clearly doesn't have to deal with the selection of specific motoneurons because a single central signal is automatically relayed to the different motor units by means of the described recruitment scheme. Thus, the number of functional degrees of freedom of a muscle is not determined by the number of motoneurons that innervate the muscle but the CNS has to control only few degrees of freedom per muscle, for example the contractile force of intra- and extrafusal muscles.

Likewise, many joints and muscles are organized in motor synergies (Mussa-Ivaldi et al. 1994). A prominent example for this are the fingers of the human hand which have in sum 23 degrees of freedom (Napier 1980). Fortunately, these are not functional degrees of freedom. For example, it is for most people impossible to flex the most distal joint of a finger while not flexing more proximal ones. Likewise, when a finger is required to produce force, e.g. for pressing on a button, other fingers exert at least some force, too (Latash, Scholz, Danion & Schöner 2001; Zhang, Zatsiorsky & Latash 2007).

In sum, by organizing degrees of freedom into motor synergies, the control of the body is dramatically simplified because much fewer variables have to be controlled (Mussa-Ivaldi et al. 1994; Mussa-Ivaldi 1999). Additionally, it has been hypothesized that

the reduction of motor complexity by forcing degrees of freedom into motor synergies facilitates initial motor learning because it reduces body complexity (Bernstein 1967; Berthier et al. 2005; von Hofsten 2003). However, even if motor synergies boil down neuronal, muscular and mechanical degrees of freedom by a large amount, they do not completely resolve the redundancy problem because some excessive functional degrees of freedom still remain.

Optimal Control

Virtually all organisms have to resolve excessive functional degrees of freedom, at least from time to time, simply because not all tasks require the complete maneuverability of the body or its appendages. For example, grasping a cup requires the control of six degrees of freedom because the hand has to be moved to the cup's handle and the hand has to be aligned in a way that enables approaching the handle from a reasonable direction. Thus, in this task, six out of seven degrees of freedom of the hand are determined by the task and only one excess degree of freedom remains. However, most of us have to press keys on a computer keyboard even more frequently than grasping cups. This task is less constrained because its basic requirement is that a finger is located at a certain point in space but no specific hand angle is required, thus leaving four excess degrees of freedom. Hence, simply the fact that some of the tasks we can accomplish include more constraints than others requires us to solve motor redundancy from time to time. One way to solve the problem would be to simply pick a random action among multiple equivalent alternatives. However, empiric findings suggest that the CNS selects actions in a more sophisticated way.

This becomes apparent during motor development. Esther Thelen and her colleagues (1993) observed the development of reaching movements in four infants from the 3rd to the 52th week of their lives. After three month, just about when the infants began to reach goal-directed, they showed huge interindividual differences. Two infants made forceful movements whereas the other two made slow movements. During the following month, the movement pattern of both groups converged. Likewise, it has been found that reaching movements of infants get not only more accurate but also more stereotyped during the first two years until they finally resemble those of adults (Konczak, Borutta & Dichgans 1997; Konczak & Dichgans 1997).

Also, in adults who learn a new motor skill, the variability of electrophysiological and kinematic movement parameters decreases (Flament, Shapiro, Kempf & Corcos 1999; Jaric & Latash 1999; Ludwig 1982; Mosier, Scheidt, Acosta & Mussa-Ivaldi 2005). These findings strongly support the notion that among the set of appropriate behaviors not a random choice is made but that actions are selected that optimize additional criteria.

Additional criteria that are not crucial for the task but can be used to explicitly select among possible solutions are termed *optimality criteria* (for a review see Engelbrecht 2001). Different optimality criteria have been proposed for the solution of the inverse kinematics problem of selecting an appropriate end-posture for a movement to a specific hand location. A straight forward criterion for end-posture selection is *end-state comfort*

(Cruse, Wischmeyer, Brüwer, Brockfeld & Dress 1990; Rosenbaum et al. 1990). The comfort criterion is supported by different lines of experiments. For example, participants grasp objects in that way among multiple alternative ways that feels comfortable and does not require extreme joint angles or awkward postures (Rosenbaum et al. 1990). Reaching movements with an arm restricted to three degrees of freedom terminated in joint configurations that were predicted to be optimally comfortable from participant ratings (Cruse et al. 1990, 1993). However, end-state comfort cannot solely account for the resolution of motor redundancy because also the start-posture of a movement has a considerable effect on the final arm state (Cruse et al. 1990, 1993; M. H. Fischer et al. 1997; Soechting et al. 1995). Additionally, it has been hypothesized that the inverse kinematics problem is resolved without the explicit application of an optimality criterion but by continuously moving the arm toward postures that realize hand positions with decreasing distance to the target, thus resolving kinematic redundancy implicitly (Bullock et al. 1993; Cruse & Steinkühler 1993; Cruse et al. 1998; Morasso et al. 1997; Torres & Zipser 2002). While these models account elegantly for the start-point dependency of end-postures of movements, they can hardly account for other end-posture selection criteria. For example the arm postures at the end of a movement depend on the subsequent tasks (M. H. Fischer et al. 1997; Rosenbaum et al. 1990; Weigelt, Kunde & Prinz 2006). Finally, it has been proposed that the resolution of kinematic redundancy adheres to a conjoint optimality criterion that includes movement accuracy, movement time, comfort or other criteria dependent on the requirements of the specific task (Rosenbaum et al. 1995; Rosenbaum, Meulenbroek, Vaughan & Jansen 2001). In conclusion, it seems unlikely that a single optimality criterion is applied in all situations and tasks but that the optimality criterion adapts to the situational constraints and task demands.

Even more than the inverse kinematics problem, the inverse dynamics problem of generating a specific sequence of motor commands requires the resolution of redundancy. Early theories of optimal trajectory formation predicated that a sequence of motor commands is executed that minimizes the overall jerk (change of acceleration) of the hand (Flash & Hogan 1985) or that minimizes the overall torque change of the joints during a movement (Uno, Kawato & Suzuki 1989). The theories account for behavioral data but have some conceptual problems. First, models that predicate the optimization of a motor command sequence to such criteria require an unrealistically precise and detailed knowledge of the body physics and highly elaborate and accurate planning mechanisms (Kawato, Maeda, Uno & Suzuki 1990). Second, while these theories offer a good description of movement trajectories they do not explain why trajectories are shaped the way they are and how the brain could compute such high-level movement statistics. Subsequently, an ecologically more valid optimality criterion has been suggested: the spatial accuracy of movement outcomes (Harris & Wolpert 1998). Based on the finding that motor errors increase with the amplitude of motor commands (signal-dependent noise) it is possible to demonstrate that minimum torque change trajectories also minimize the end-point variance of a movement, thus offering a ecologically plausible optimality criterion.

The theories outlined above mainly focus on preprogrammed open loop movement

execution but movements are regulated online by visual or proprioceptive feedback to a big extent (Desmurget & Grafton 2000). To account for this, theories of *optimal feedback control* have integrated closed-loop control and the optimality criterion of maximal movement accuracy (Todorov & Jordan 2002). Optimal feedback control is based on the premise of signal-dependent noise but emphasizes online correction of movements and that correcting a deviation in one movement parameter may cause in itself deviations in the remaining ones. Hence, correcting signals are kept as low as possible to reduce noise and are only exerted if critical movement parameters deviate from desired values (minimum intervention principle). As an example, consider grasping a broom that leans against a wall. In this task, the control of the vertical hand location is not very important because it doesn't matter where exactly the broom is grasped. However, the horizontal position of the hand has to match that of the shaft accurately. If the hand is perturbed and moves up or down, an optimal feedback controller would not correct for this irrelevant deviation because it does not hinder the success of the movement. Moreover, a correction of the vertical hand position could result in a critical deviation of the highly relevant horizontal hand position. Indeed, experiments have shown that the movement variability in the task irrelevant movement parameters is higher than in task relevant movement parameters (Latash et al. 2002; Scholz et al. 2000).

Short Summary

The previous section showed that solving the inverse problem of a redundant system, such as the human motor apparatus, requires the resolution of motor redundancy. Redundancy is partially reduced by coupling degrees of freedom in motor synergies. However, at least some excess degrees of freedom remain in most tasks. Human motor control seems to resolve this remaining redundancy by optimal control. However, the neural mechanisms that achieve optimal control and the computations they carry out remain largely unknown.

Closely related to the problem of redundancy is the acquisition of inverse models during motor learning. The next section will elaborate on this problem. First, the question is raised whether motor redundancy is resolved before or after learning and what are the implications of both approaches. Second, the computational demands of different models of motor learning are discussed.

Motor Learning

Inverse models, which are the neural basis of solving the inverse motor problem for goal-directed movements, are acquired during infancy. On the one hand, neonates are not yet able to exert pointing and reaching movements. On the other hand, the inverse models have to be adjusted to the (changing) properties of the human perceptual and motor system. The competency to reach goal-directedly only starts to develop after the first few month of life. Two year olds already perform reaching movements that are very similar to those of adults (Konczak et al. 1997; Konczak & Dichgans 1997; Thelen et al. 1993; von Hofsten 2003). However, also adults show a remarkable capability to adjust movement control to novel circumstances, for example, novel body properties

(Conditt et al. 1997; Lackner & DiZio 1998; Robertson & Miall 1997), novel sensorimotor relationships (Mosier et al. 2005), or visuomotor distortions (Abeele & Bock 2001; Butz, Lenhard & Herbot 2007; Krakauer, Pine, Ghilardi & Ghez 2000).

Motor Learning and the Resolution of Redundancy

In principle, there are two basic ways to address learning an inverse model for a redundant body (see Figure 2): either an *optimized inverse model* or a *task-independent inverse model* is acquired. An optimized inverse model provides a mapping from each possible movement goal to specific efferent signals which are optimized for a specific context. A task-independent inverse model, on the other hand, maps each possible movement goal to all the efferent signals that might be used to pursue the goal, regardless if and in which sense they are optimal. Each approach yields different requirements for a neural controller and offers different capabilities.

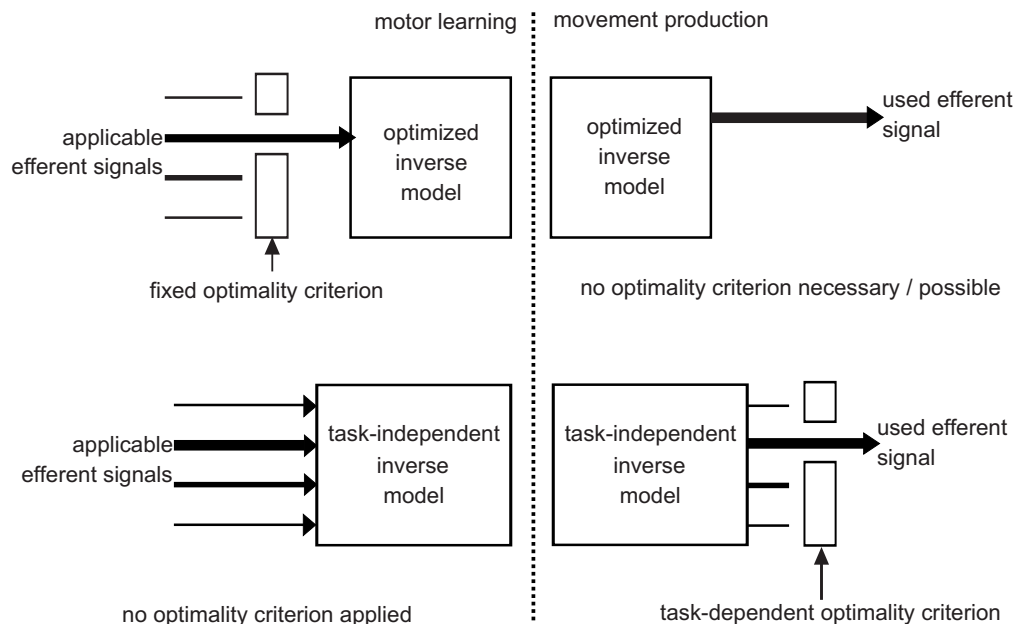


Figure 2. Motor redundancy may be resolved before learning (top) or during movement production (bottom).

Learning optimized inverse models

By far the most computational models resolve motor redundancy before learning and thus acquire an optimized inverse model (Figure 2, top). This has many computational advantages. First, during movement execution the problem of selecting optimal efferent

signals virtually does not exist because an optimized inverse model provides only a single solution of the inverse problem. Second, the inverse model can be represented very compact because only one efferent pattern has to be encoded for each situation and movement goal.

The drawback of this approach is that it lacks the flexibility to quickly adapt to new optimality criteria or to changes of the range of possible movements. Optimality criteria may change for internal (e.g. growth, injury, fatigue) or external (e.g. novel task requirements) reasons. If this happens, the inverse model cannot provide alternative efferent signals because it only encodes efferent signals that have been optimal in the motor learning context but are most likely suboptimal in any other context. Even worse, the range of possible actions may change because, for example, certain movements cannot be carried out in narrow places, due to obstacles, or due to injuries. If the formerly optimal action suddenly lies outside the range of applicable movements, some goals cannot be pursued at all, even if suboptimal actions would exist.

Learning task-independent inverse model

If redundancy is resolved only before movement production, the inverse model has to provide many redundant actions and a selection mechanism has to choose one (Figure 2, bottom). Thus, the inverse model may provide multiple, even an infinite variety of possible efferent signals for each movement goal. The approach is computationally more demanding than the optimized inverse model approach. First, not a single but many efferent signals have to be encoded, requiring larger and more sophisticated representations. The encoding of redundant actions is the basis to overcome the inflexibility of optimized inverse model approaches. However, to fully exploit the possibilities offered by the task-independent inverse model, a movement production mechanism is required that can select optimal actions dependent on current internal and external constraints. The advantage of such a more complex system would be the possibility to quickly adapt to novel or changing task-dependent optimality criteria and thus maintain optimal behavioral performance in a broad variety of situations and tasks.

General Approaches to Model Motor Learning

The previous paragraphs took a bird's eye perspective on the process of motor learning. Now, more detailed computational constraints of the acquisition of inverse models are examined. From a computational point of view, internal model learning is the acquisition of an input-output mapping. In an inverse model, the input is a desired state and the output should be an efferent signal that is suitable to pursue that goal. Hence, inverse model learning can be considered as the association of goal states (input) to certain efferent signals (output).

Thereby, a fundamental problem is the lack of external information about which efferent signals could be used to pursue a goal. Whereas supervision by a teacher might be able to support the acquisition of high level motor skills, for example in sports or music, the overwhelming part of motor learning cannot profit from external guidance. The problem at this level is, that the activity of efferent signals cannot be perceived

directly and we are also not able to consciously activate certain efferent neurons. We are restricted to perceive only the consequences of efferent signals, for example the tension or length of a muscle or the movement of a limb. For this reasons, no one can tell an infant which efferent signals to activate to pursue a goal, and, of course, the infant couldn't use this information anyway. In sum, to be able to associate the correct efferent signal (output of the inverse model) to a goal (input), a representation of both is necessary. Whereas the representation of the goal seems to be readily available, the representation of the correct efferent signal cannot be provided externally and has hence to be generated by the learning organism itself. Three different approaches of modeling motor learning exist: *error-based learning*, *reinforcement learning* and *ideomotor approaches*³.

Error-based Learning

Theories of error-based learning have been very influential in motor learning research (e.g. Jordan & Rumelhart 1992; Kawato 1990; R. A. Schmidt 1975, 1988). In error-based learning, a goal is specified and represented before movement onset. Then, an inverse model determines efferent signals that cause a certain movement and consequently specific sensory inputs. After the movement is completed, the desired goal state is compared to the movement outcome (knowledge of result). If the efferent signals caused the intended movements the inverse model is obviously suitable and is not changed. If not, the discrepancy between movement outcome and goal is used to adjust the inverse model.

As an example, consider an experimental participant, making reaching movements with visual feedback rotated clockwise by 30°. Initially, reaching movements will fail to acquire goals, because the movements that were successful under normal conditions now cause a directional error of about 30°. According to theories of error-based learning the participants motor learning system would notice a discrepancy in desired and actual movement outcome in the clockwise direction. Thus, the inverse model would be adjusted to cause a movement that is directed counterclockwise to the initial one and thus closer to the desired one.

However, adjusting the output of the inverse model requires that the error in perceptual space is converted into an error in motor space. Considering the non-linear arm geometry and the complex arrangement of muscles, even the adaptation to a simple visual rotation seems to require extensive computations. Furthermore, as perceptual space is usually low dimensional compared to motor command space, the conversion is also hit by the problem of redundancy. For example, many different adjustments of the inverse model's input-output mapping may result in higher movement accuracy. Indeed, a capable error-conversion mechanism can be already considered an inverse model because it would map from perceptual space to motor command space. It thus constitutes at least a partial solution of the inverse problem.

In conclusion, error-based learning schemes cannot solve the motor learning problem without a mechanism that maps errors from goal space to action space. However, such

³In recent publications, the term ideomotor principle has referred to the automatic priming or initiation of observed actions (Jansson, Wilson, Williams & Mon-Williams 2007). Here, I refer to the original notion that postulates links between sensory goal representations and actions (Stock & Stock 2004).

a mechanism could be considered an inverse model by itself, even if this inverse model only processed errors and not absolute values. Also, due to the iterative application of small adjustments, this mechanisms wouldn't have to be as accurate as an inverse model that is used for actual movement production.

Reinforcement Learning

The problem of error conversion is avoided in reinforcement learning (RL) schemes (Sutton & Barto 1998). Unlike error-based learning schemes, for RL it is sufficient to provide a “reward” value that indicates the degree of goal completion. The reward can usually be easily computed from the movement outcome and does not require a conversion in efferent signal space. The inverse model is trained to store those efferent signals that yield the highest rewards. However, the reward offers no guidance in which direction to adjust the output of the inverse model. Hence, different efferent signal patterns have to be randomly applied to search for rewarding actions, resulting in a less efficient learning process.

Both, error-based learning and RL are well suited to acquire an optimized inverse model because for each goal a single (hopefully optimal) solution is encoded. This solution is either acquired by an error guided adjustment of the input-output mapping or by encoding the efferent signals that yield highest rewards. However, these learning schemes cannot account for the acquisition of task-independent models, which map goals to many possible efferent signals.

Ideomotor Learning

Ideomotor approaches or *direct inverse modeling (DIM)* offer a surprisingly simple concept of motor learning without the necessity of an error or reward signal (Greenwald 1970; Herbart 1825; J. Hoffmann 1993, 2003; James 1890). According to these approaches, motor learning starts with the activation of random efferent patterns resulting in random movements (sometimes called “motor babbling”). During these movements, the sensory input is monitored and associated to preceding efferent patterns, which presumably elicited the current sensory effects. Afterwards, if the realization of a specific sensory state is desired (i.e. a goal is set), the efferent signals linked to this state are executed. These efferent signals are exactly those signals that usually precede and presumably cause the desired sensory signal, thus achieving the goal.

A main difference to error-based learning or RL lies in the role of the goal representation during motor learning. In the former, a goal is explicitly represented and failure or success of goal achievement is used to guide learning. In contrast, the latter doesn't make use of such a goal representation. Note, that this doesn't imply that early infant movements are not goal-directed but only that this goal representation is not used for motor learning. Thus, ideomotor learning does not collide with the notion that even the most clumsy infant movements are goal-directed (von Hofsten 2004). An conceptual advantage of ideomotor learning is that behavioral goals do not need to be specified before they have been encountered for the first time. Whereas error-based or RL can only be successful for those goal that are explicitly intended during motor learning, ideomotor

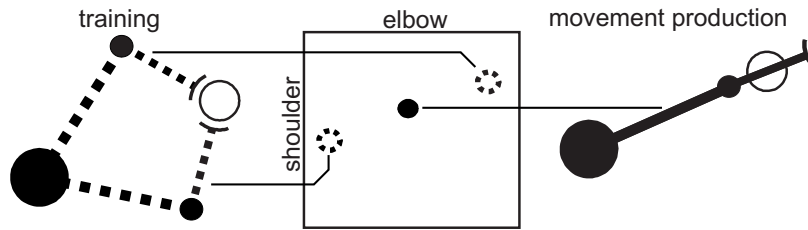


Figure 3. If the goal set is nonconvex (e.g., two separate points in joint angle space), DIM might store a non-solution as the solution of the inverse problem (black dot).

learning enables an unconstrained exploration of the space of goals without the need of prior knowledge of potential goal states.

However, ideomotor learning also has some considerable conceptual problems. First, it requires a mechanism that enables to determine what is an action and what is an effect. Whereas this seems straight forward in more symbolic contexts (e.g. Stock & Hoffmann 2002), it is not so simple in the domain of motor learning. During random movements, a continuous flow of efferent outputs has to be related to a continuous flow of sensory inputs. Furthermore in the dynamic context of movement, a specific sensory state is not only caused by the efferent signals before but also by previous efferent signals. Likewise, a single efferent signal does not only (partially) determine the subsequent sensory input but has effects that reach further into the future. The problem that there is no simple temporal relationship between cause and effect in motor control is termed *temporal alignment problem* in the remainder of the thesis.

Second, ideomotor learning is usually implemented by DIM. Unfortunately, DIM is an unreliable learning method for the acquisition of compact optimized inverse models because it might be unable to converge to a solution of the inverse problem in redundant contexts. It is destined to fail if the set of redundant efferent signals that are useful to pursue a specific goal is *nonconvex* (Jordan & Rumelhart 1992). Consider the task of determining a joint configuration that locates the hand of a two-joint planar arm at a desired position. During learning, each hand location is realized by different joint angle configurations, one with the elbow pointing clockwise and one with the elbow pointing counterclockwise. Hence, no consistent bias toward one of the joint configurations exists and the inverse model is likely to evoke a mixture of both postures (Figure 3). Direct inverse models may learn to stretch the arm, consequently missing the target.

Short Summary

To summarize, the internal inverse models that humans use to control reaching movements are acquired by motor learning. Motor learning might be error-based, done by reinforcement learning (RL), or by ideomotor learning. A hard conceptual problem in error-based motor learning is, that it requires the conversion of errors from perceptual space to efferent signal space. A mechanism that enables such a conversion is by itself at least a coarse inverse model and thus represents at least a partial solution of the

problem that should be solved by learning. Thus, error-based learning is not suited to model unsupervised motor learning. RL methods do not suffer this drawback but to my knowledge, no computational model exists that implements the acquisition of task-independent inverse models with this technique. However, only task-independent inverse models seem to account for the flexibility of human motor behavior. Direct inverse modeling or ideomotor learning might overcome some of the drawbacks of the former approaches as models of human motor learning. They do not require a priori goal representations and could potentially be used to acquire task-independent inverse models. However, to do so, two shortcomings of DIM have to be overcome. First, the problem of extracting learning information from a continuous flow of efferent signals and sensory inflow has to be solved. Second, the problem of learning to control a redundant body has to be solved.

Models of Motor Learning and Models of Flexible Behavior

This section reviews various computational models of motor learning and control. Models that explicitly investigate supervised learning (e.g. Chapeau-Blondeau & Chauvet 1991; Stringer, Rolls, Trappenberg & de Araujo 2003; Stringer, Rolls & Taylor 2007) or that include a learning mechanism but do not claim that this learning mechanism is biologically relevant (e.g. Stroeve 1996, 1997) are not reviewed. The review starts with the discussion of models that account for motor learning. All of them resolve motor redundancy before learning and encode optimized inverse models. Thus, they do not account for the flexibility with which humans adapt to new situational constraints. After that, approaches that use task-independent inverse model are discussed. These approaches show a remarkable adaptivity to situational constraints but are mostly unable to account for motor learning.

Models of Motor Learning

Models of motor learning that resolve redundancy before learning are usually based on goal-directed learning mechanisms. They strive to encode a single optimal efferent signal for each possible goal. Common goal-directed approaches to motor learning are *direct inverse modeling* (Jordan & Rumelhart 1992) and *feedback error learning* (Kawato, Furukawa & Suzuki 1987). Additionally, *reinforcement learning* methods have shown some success in this context (Berthier 1996; Berthier et al. 2005).

In the following, these approaches are discussed. Thereby, I focus on the three problems that became apparent in the previous sections. First, the model should be able to learn even if there is no clear temporal relationship between efferent signals and sensory consequences (temporal alignment problem). Second, the model should be able to account for learning to control a redundant body. Third, motor learning should be unsupervised.

Direct Inverse Modeling Approaches

Direct inverse modeling (DIM) has been applied to model motor learning within a range of computational models (Baraduc, Guigon & Burnod 1999; Baraduc et al. 2001;

Bullock et al. 1993; Butz, Lenhard & Herbort 2007; Kuperstein 1988, 1991; Ognibene, Mannella, Pezzulo & Baldassarre 2006; Ognibene, Rega & Baldassarre 2006; Srinivasa & Grossberg 2007). The inverse model learns by observing the effects of random efferent signals. It permanently predicts the efferent signals that might have caused the current sensory input. If differences between predicted and actually issued motor commands are detected, the inverse model is adjusted accordingly. As soon as it is sufficiently accurate, it can be used to exert goal-directed behavior.

It was already expounded that DIM cannot readily account for learning if no simple temporal relationship between efferent signals and their sensory consequences exists and if the body is redundant (c.f. Herbort 2005). However, models that provide mechanisms to cope with these problems might account for motor learning. The problem of redundancy is frequently sidestepped. Some DIM approaches use arm models with only two joints and a limited set of possible joint angles avoiding the necessity of handling redundancy (e.g. Baraduc et al. 1999, 2001; Butz, Lenhard & Herbort 2007; Ognibene, Mannella et al. 2006). Others are able to control bodies with redundant degrees of freedom by imposing additional constraints during learning, which eliminate the redundancy (Bullock et al. 1993).

The temporal alignment problem may be avoided by taking an equilibrium point control stance and by looking on entire movements episodes, not just fractions of it (Ognibene, Mannella et al. 2006). The assumption of equilibrium point control eliminates the necessity to provide a time-varying control signal. By only considering entire movement episodes and not looking at movement dynamics or kinematics, a clear relationship between an efferent signal (i.e. the signal before the movement) and sensory inputs (i.e. the sensory input after the movement) can be established. These simplifications may be justified dependent on the scope of the computational model and the targeted processes but models that build upon these simplifications are far from capturing the complexity of motor learning and control.

The motor system might avoid the temporal alignment problem by associating hand movement directions to efferent signals (Baraduc et al. 1999, 2001; Bullock et al. 1993). At least in the case of a kinematic arm, it is sufficient to relate efferent signals to the hand movement directions a short time afterwards. The desired hand movement direction may be inferred from the visually encoded discrepancy between hand and target. However, this requires that all movements in the same direction are always produced by the same motor commands, independent of the movement amplitude. This approach still does not solve the redundancy problem because movements in the same direction may well be caused by different motor commands but by additionally constraining the learning experience, it is possible to unsupervisedly learn an inverse model (Bullock et al. 1993).

With respect to learning, DIM is an unsupervised learning approach. As discussed previously, no training signals need to be provided from the outside. Thus, DIM is able to account for the extraction of an inverse model from mere random movements.

To summarize, DIM is an unsupervised learning approach. However, learning of complex body interactions that unfold in time is impossible without additional mechanisms. Most importantly, redundancy needs to be resolved before learning because DIM

is a goal-directed learning scheme. Redundancy is usually resolved by constraining the learning experience or the body characteristics.

Feedback Error Learning Approaches

Another goal-directed learning scheme for training inverse models is *feedback error learning* (FEL, Kawato et al. 1987; Kawato 1990). In this learning scheme, the to-be-trained inverse model works in parallel to a pre-existing, linear closed-loop controller. Initially, the inverse model does not contribute to the formation of effective efferent signals and does not enhance control. As a consequence, a difference between desired and perceived state will arise. The “innate” feedback controller triggers suitable efferent signals for adjustment, which are also used as training signals for the inverse model. This training signal enables the inverse model not only to replicate the feedback controller but also to acquire a non-linear mapping that levels out deficiencies of the linear feedback controller. As the inverse model gets trained, its control signal gets more efficient. Finally, it replaces the linear feedback controller as the main contributor to the efferent signal because the inverse model is so efficient that no larger discrepancies between desired and actual body position arise.

This principle was sometimes applied to motor learning in general (Karniel & Inbar 1997) but it is mainly used to model cerebellar motor learning (Barto et al. 1999; Berthier et al. 1992, 1993; Haruno et al. 2001; Kawato et al. 1987; Kawato & Gomi 1992; Schweighofer, Arbib & Kawato 1998; Schweighofer, Spelstra et al. 1998; Wolpert & Kawato 1998). In this context, the cerebellum is hypothesized to be an inverse dynamics model, which is trained by corrective cerebral motor commands. Thus, it helps to understand the role of the cerebellum in smoothing the movement dynamics (Barlow 2002).

Despite its success in modeling cerebellar learning, FEL cannot easily account for the resolution of redundancy and unsupervised learning. The redundancy resolving problem persists because an additional mechanism within the feedback controller is needed to resolve redundancy consistently. A feedback controller may not be able to compute a consistent, high dimensional training signal from a low dimensional error value. If different actions are applicable to reach a goal and the feedback controller randomly chooses between those actions, the learning mechanism is destined to fail if the set of alternatives is nonconvex (Jordan & Rumelhart 1992). The feedback controller has to be consistent in its choice of efferent signals to ensure successful learning. Hence, it is the preprogrammed feedback controller that implicitly solves the redundancy problem in FEL. However, a simple linear feedback controller as is usually used in FEL cannot always achieve this.

FEL somewhat sidesteps the problem of unsupervised learning because it assumes the existence of a linear closed-loop controller, which exerts corrective motor commands based on position error. The controller is used to transform errors in hand space to errors in efference space, serving as the missing teaching signal (Kawato 1990). Thus, the mapping from hand locations to efferent signals, which should be acquired unsupervisedly during learning, is wired into the system in form of the feedback controller. In conclusion,

FEL does not address the acquisition of an inverse model from scratch but accounts for additionally improving the performance of a pre-existing, maybe rather sparse, inverse model.

FEL models mainly address the problem of controlling dynamic and complex bodies. The temporal alignment problem is solved because potential goals do not have to be extracted from a sequence of efferent signals and sensory inputs like in DIM but are self generated and persist over a comparatively long time.

To summarize, FEL does not explicitly tackle the redundancy problem, rather, FEL requires its resolution by a pre-existing feedback controller. In this sense, FEL addresses the problem of acquiring a sophisticated inverse model based on a very basic feedback controller but not on learning such a model from scratch. Thus, FEL significantly refines a simple control strategy to be able to handle time-extended sensorimotor contingencies. It is thus a valid approach to model cerebellar motor learning but the learning mechanism should not be transferred to motor cortical models.

Reinforcement Learning Approaches

Unlike the above goal-directed learning schemes, which require the availability of only one “correct” solution during learning, reinforcement learning (RL) approaches shape behavior by means of reward values. The reward values indicate to which degree an efferent signal optimized an optimality criterion (Wolpert, Ghahramani & Flanagan 2001). The efferent signals with the highest reward is stored (Berthier et al. 2005; Kositsky & Barto 2002) or invariably selected for action execution (Berthier 1996). Similar to FEL and DIM approaches, redundancy is generally resolved before learning in RL approaches by imposing a particular reward scheme or optimality criterion.

The compelling advantage of RL is that it does not require a mechanism that maps errors in hand space to errors in efferent signal space because it doesn’t need a corrective training signal. A reward value is sufficient to learn an optimal control scheme and this value may be easily derived from movement outcomes because it does not require the conversion of a low dimensional input value to a high dimensional output. Thus, RL accounts for unsupervised learning of a redundant body. However, as the reward value conveys less information than an error signal, learning takes considerably longer as compared to FEL or DIM.

RL approaches are able to cope with dynamic bodies and the temporal alignment problem because reward values can be back-propagated to past efferent signal choices. Thus, actions can be associated with goals that require the execution of long and complex action sequences.

To summarize, RL methods provide an interesting alternative to the above-mentioned mechanisms. Unlike DIM, the RL approach is also applicable in redundant, dynamic bodies. And unlike FEL, the RL approach does not require a pre-existing inverse model that provides consistent learning signals.

Short Summary

This section showed how different learning schemes solve the redundancy problem before learning. Direct inverse modeling (DIM) requires a non-redundant body with a one-to-one mapping between actions and effects, or it requires the imposition of additional mechanisms that resolve redundancy before learning. Feedback error learning (FEL) requires the pre-existence of a feedback controller, which is used to provide the required teaching signals for the supervised learning approach. Finally, reinforcement learning (RL) methods enable learning in dynamic environments as long as an optimality criterion is provided.

In all of the discussed learning schemes, only one single solution is stored for a given control problem. While this solves the redundancy problem, the approaches lack the flexibility to quickly adapt to novel situations or tasks, in which executable actions may change, action preferences may change, or novel spatial or other constraints may be imposed. For example, if the usually chosen action is not available any longer due to, for example, obstacles or a broken limb, then the discussed approaches do not offer alternative behavior. Similarly, if a usually chosen joint movement or joint position is suddenly highly uncomfortable due to injury, the models are not able to rapidly compensate.

Instead of storing only a single solution of an inverse problem, it can be advantageous to represent redundant actions in a control structure and delay action selection until it is really necessary. If obstacles or new optimality criteria reduce the appropriateness of a formerly suitable action, alternatives would be immediately available. In the following section I review computational models that resolve redundancy only during action preparation, storing multiple actions for each possible task.

Models of Flexible Behavior

By now, some approaches have been proposed that use task-independent body models. These task-independent body models offer the motor control system a broad repertoire of possible efferent signals for each goal. Thus, these models are not bound to pursuing a goal with a single efferent signal but can select among many possibilities to optimally adapt to situational constraints. In the following, I review two different theories that follow this approach. The *mean of multiple computations* network (Cruse & Steinkühler 1993) is a neural network model that encodes the complete kinematics of a body. The influential *posture based motion planning theory* (Rosenbaum et al. 1993b) selects targets according to task-dependent constraints. Both approaches resolve redundancy for inverse kinematic problems.

Mean of Multiple Computations Network

The *mean of multiple computation* recurrent neural network (*MMC*) models the complete kinematics of the body in question. It solves the inverse kinematics problem for arbitrary combinations of desired output values. It thus does not only solve the inverse kinematics problem but can also solve the direct kinematics problem or any mixed problem (Cruse & Steinkühler 1993; Cruse et al. 1998).

The neural network dynamics inherent in MMC resolve redundancy iteratively. By fixing some of the network inputs to currently desired values, the network dynamics cause the network activation to shift into one of the states that represent a solution to the posed inverse problem. Thus, MMC implicitly resolves redundancy by means of its system dynamics. By tracking the shift in network weights, it is also possible to determine a trajectory to the target. The structure allows the imposition of novel constraints on the controlled arm, such as a desired joint angle or a desired position. Thus, it is possible to adapt motor control to the demands of changing tasks.

The MMC model also has some considerable problems. First, even if the MMC network itself is highly compact, it requires considerable pre- and post-processing of the input and output values. For example, to enable the processing of joint angles, the input and output values have to be normalized with trigonometric functions (Cruse et al. 1998). Thus, a big part of the computational burden of representing a non-linear body has to be shouldered by signal normalization. Second, it is unknown if the MMC can be trained unsupervisedly. In recent work the neural network weights and connectivity have been derived analytically (Cruse & Steinkühler 1993; Cruse et al. 1998; Kindermann & Cruse 2002; Steinkühler, Beyn & Cruse 1995). Despite some first successes to enable learning in MMC networks (Kühn & Cruse 2005), enabling MMC to unsupervisedly learn to control a redundant, nonlinear body remains an unsolved problem.

In conclusion, the MMC clearly shows that neural network models that encode all solutions for an inverse problem may account for highly flexible behavior. However, learning is not yet possible within the MMC framework.

Posture Based Motion Planning

Whereas the MMC is a neural network based approach, the *posture based motion planning theory (PB theory)* is formulated rather abstract (Rosenbaum et al. 1993b, 1995; Rosenbaum, Meulenbroek, Vaughan & Jansen 2001). The PB theory solves the inverse kinematics problem of selecting an appropriate end-posture for a movement to a hand target. Additionally, it provides a mechanism to plan a trajectory in posture space from the starting posture to the end-posture, which may circumvent obstacles.

In contrast to the approaches that resolve redundancy before learning, in PB theory, an optimality criterion is only defined before movement onset. This rather complex optimality criterion usually integrates different aspects of a movement, like accuracy or movement time, in a weighting scheme (Rosenbaum et al. 1995) or in a constraint hierarchy (Rosenbaum, Meulenbroek, Vaughan & Jansen 2001). End-posture selection is based on two consecutive mechanisms. First a limited number of frequently used postures are stored. All of these postures are evaluated according to the conjoint optimality criterion and the highest ranking stored posture is selected for further processing. Second, as it is likely that none of the stored postures is optimal for the current task, the selected end-posture is further optimized by local linear adjustments (“feedforward correction” in Rosenbaum et al. 1995) or breadth first search in posture space (“diffusion ’til a deadline” in Rosenbaum et al. 1999; Rosenbaum, Meulenbroek & Vaughan 2001; Rosenbaum, Meulenbroek, Vaughan & Jansen 2001). When a good end-posture

of a movement is determined the movement trajectory is planned by imposing a sinusoidal shift between start- and end-posture. If this trajectory happens to collide with an obstacle, it is altered by superimposing a back and forth movement. Once a suitable superimposed movement is found, the combined movement, which effectively moves the arm around the obstacle, is executed. The PB theory accounts for a wide range of empirical findings and excels at providing a framework for modeling the flexibility of human reaching, grasping, or tool-use.

The PB theory emphasizes the resolution of redundancy but not the motor learning problem. No properties of the stored postures, which are relevant for the evaluation process, such as the coinciding hand location in space or transition times from one posture to another, are learned. What is learned are postures, which serve as starting points for the search of a posture that fulfills the current constraints and optimizes the optimality criteria. Thus the model requires that knowledge about arm kinematics, movement speeds, transition costs, or other potential selection criteria is already acquired and readily accessible to the motor system. The acquisition of this information is not subject of the model.

In conclusion, the PB theory shows how the motor system may maintain a high performance, even if it has to adapt to changing task demands from one moment to the next. This capability builds on the availability of a task-independent body models. However, the PB theory does not account for their acquisition.

Learning and Flexibility

Both of the models discussed in the last section account for the flexible resolution of motor redundancy. The MMC allows the invocation of posture constraints, which are resolved by the inherent network dynamics but cannot be trained unsupervised. The PB theory selects movement end-postures based on conjoint optimality criteria but cannot account for the acquisition of the underlying task-independent body model. In contrast, the earlier discussed models of motor learning do not account for behavioral flexibility as do the MMC and PB theory.

By now, few approaches attempt to combine both, motor learning and flexible control. Such approaches might have several obvious advantages. First, to be able to learn, they do not require any kind of teaching or reward signals that enable the model to store one specific solution for the inverse kinematics problem. This eliminates the necessity to provide a reward signal from the outside, pre-wire a basic inverse model into the system as in FEL, or generate a reward signal as in RL approaches. Thus, it might enable unsupervised learning. Second, since all solutions for a problem are stored, the organism is still able to act in a goal-directed manner, if the preferred solution cannot be applied. Additionally, if optimality criteria or reward values change after learning, a complete remapping of the inverse model is not necessary. Instead, only constraints need to be adjusted. Also, current findings suggest that the motor system does not strive to encode stereotypic actions but rather to encode and exploit motor redundancy to enhance control (Latash et al. 2002; Todorov & Jordan 2002).

Description of the Model

The survey of related approaches in the previous section has shown that motor redundancy can either be resolved *before* learning or *after* learning immediately during movement preparation and execution. On the one side, direct inverse modeling (DIM), feedback error learning (FEL), and reinforcement learning (RL) approaches enable motor learning, but lack the flexibility to adjust their motor policy online to account for variations in task-related constraints. On the other side, the MMC and the posture-based motion planning theory are much more flexible, but offer no or only restricted learning capabilities.

In this section, I will introduce a novel computational model that adds to the flexibility of the latter a neural-based, unsupervised learning architecture that grounds distance measures in experienced sensorimotor contingencies. As the model can be described as a *sensorimotor, unsupervised learning, redundancy resolving control architecture* it is called *SURE_REACH*. The architecture is implemented as a biologically plausible neural network model. It differs from its predecessors which accounted for the control of a one- or two-joint dynamic arm in two ways (Herbort 2005; Herbort et al. 2005a; Herbort, Butz & Hoffmann 2005b). First, it introduces a mechanism for movement preparation which helps to solve the temporal alignment problem and enables behavioral flexibility. Second, whereas the previous models showed that unsupervised ideomotor learning may be used to control a dynamic body, this thesis focusses on the unsupervised acquisition of task-independent body models and their use for highly flexible control. The model and some extensions of it have been described in (Butz, Herbort & Hoffmann 2007; Herbort & Butz 2007; Herbort, Ognibene, Butz & Baldassarre 2007).

SURE_REACH Architecture

SURE_REACH is hierarchically structured (Figure 4), consisting of population encoded extrinsic hand and intrinsic posture space representations, a *posture memory*, *sensorimotor model*, and a *motor controller*. The posture memory is a complete, task-independent inverse kinematics model that transforms goals in hand space (XY coordinates) into redundant goal representations in posture space (shoulder, elbow and wrist angle combinations). It thereby co-activates all coinciding postures for a given hand target to keep action selection maximally flexible. The sensorimotor model is a task-independent model of how motor commands affect body postures. It thus encodes basic sensorimotor contingencies. The motor controller participates in two partially overlapping phases of the movement process. First, a movement is prepared by means of neurally implemented dynamic programming⁴ based on the sensorimotor model, which

⁴Dynamic programming is a method to determine optimal choices in a multi-stage decision process (Bellman 1957), which may be applied to the control of behavior (Sutton 1990; Toussaint 2006). To find the optimal sequence of choices, the method iteratively generates a *policy*, which, for each possible situation, determines which decision optimizes a given criterion. For example, to exit a maze with known layout as quickly as possible, it is necessary to make turns to the optimal direction at each intersection. It is easy to turn to the right direction at the last intersection because one might already see the exit down one way or the other. In similar fashion, the optimal decisions at intersections adjacent to that last

generates a sensory-to-motor mapping, given a goal in posture space. The sensory-to-motor mapping provides motor commands dependent on the current posture. Second, during movement execution, this mapping is used to approach the goal by closed-loop control. By default, these motor commands lead directly to the closest goal posture. The closeness of two postures is represented in the weights of the sensorimotor model. It depends on the motor command dependent temporal adjacency of posture as well as on the frequency with which they are assumed. Closeness can be considered the default optimality criterion. If no other constraints are imposed, the motor controller generates a movement to that posture among many possible goal postures that can be reached by the shortest movement trajectory.

The modular, hierarchical encoding enables the flexible imposition of additional constraints on hand space or on posture space. On the one side, the explicit representation of redundant goal postures enables to change the range of acceptable end-postures (left side of Figure 4). The goal representation, which is provided by the posture memory, could be expanded or narrowed down thus altering the range of acceptable goal postures processed by the motor controller. For example, this enables generating a target representation for a movement to a certain hand position with additional constraints on joint angles. On the other side, the encoding of multiple sensorimotor contingencies provides a possibility to alter movement trajectories (right side of Figure 4). For example, inhibiting areas of posture space that coincide with obstacles during movement preparation results in the generation of sensory-to-motor mappings that yield alternative movement trajectories. The mapping from externally represented obstacles into corresponding posture space representations is achieved by the posture memory.

Both, the motor controller and the posture memory are acquired in an initial motor learning phase, during which random movements are executed. During this phase, the sensorimotor model encodes sensorimotor contingencies and the posture memory acquires its kinematic mapping. This does not imply that infant motor behavior is based merely on random motoneuron excitations. The ideomotor learning approach, which underlies the model, just does not *require* that movements are intentionally directed at certain goals during learning. Random movements are only used as a model for early infant movements because SURE_REACH is not intended to account for infant reaching or exploration strategies (e.g. Berthier 1996) but for the extraction of neural body models from those movements.

In the following sections, the body and the neural implementation of SURE_REACH are described in detail. First, the body that is used to evaluate the model is explained. Next, the implementation is detailed specifying the body space representations, posture memory, sensorimotor model and motor controller. Finally, examples illustrate how the activation of goal(s) and potentially additional constraints trigger appropriate motor activity.

intersection can now be easily determined. Thus, starting from the goal, one might readily arrive at a policy that tells the best direction to go for each point in the maze. By following this policy a sequence of turns is made that minimizes the way out, or in general terms, a sequence of decisions is generated that optimizes a certain criterion. In this way, dynamic programming solves complex problems by dividing them into small and tractable ones.

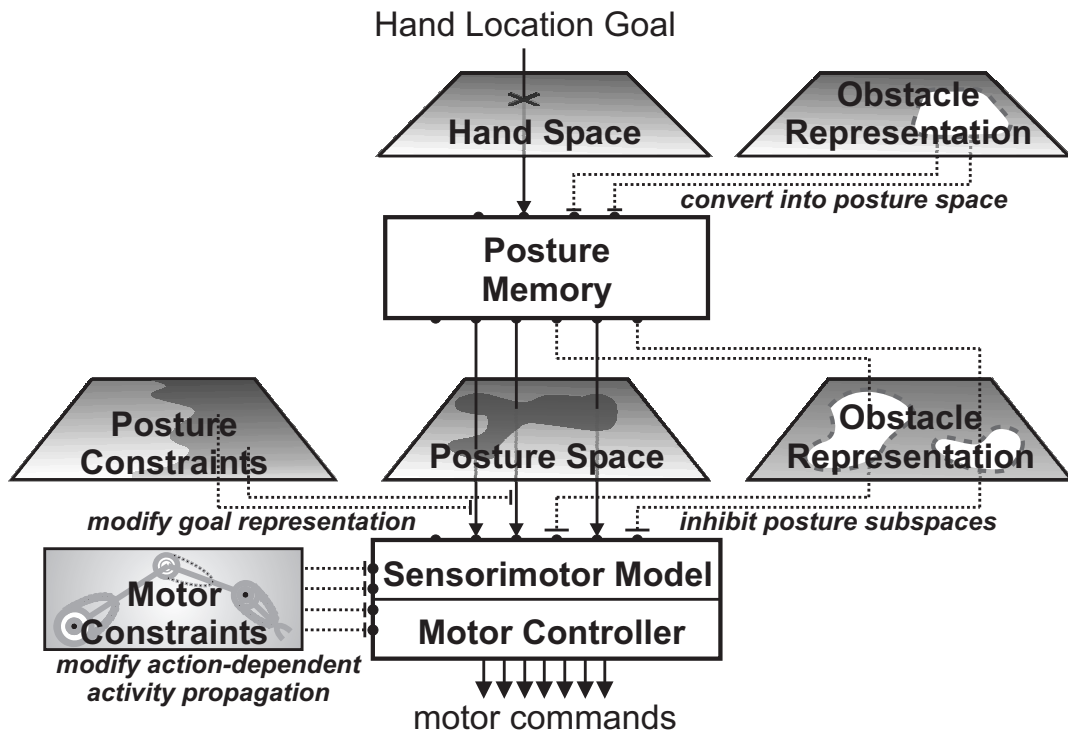


Figure 4. The SURE_REACH architecture is hierarchically structured associating extrinsic *hand space* with intrinsic *posture space* by an associative posture memory. Both spaces are population encoded. The motor controller processes arbitrary goal activations in posture space to invoke motor commands based on the sensorimotor model, depending on the actual arm state. Additional constraints can be imposed on the goal representation on posture space as well as on the movement preparation process realized in the motor controller.

A Three Joint Planar Arm

The body that is controlled by the current SURE_REACH implementation is a three joint planar arm (Figure 5). The three limbs have the lengths $l_1 = 1.0$, $l_2 = 0.8$ and $l_3 = 0.6$. The shoulder and elbow joint were allowed to rotate within $\pm 180^\circ$, whereas the wrist joint was restricted to values between 0° and 180° . Note that shoulder and elbow joints cannot circle, that is, they cannot jump from -180° to 180° or vice versa.

Each joint is controlled by two actuators—causing a clockwise or counterclockwise rotation step. Motoneurons y_i^{fin} , which yield excitation levels between zero and one, gradually activate the actuators. The actual joint displacement per time step during the simulation is calculated by subtracting the activation level of two antagonistic actuators and multiplying the result with a gain factor g . The gain factor determines the maximal joint angle displacement in one activation step. If not stated differently, the gain factor is set to $g = 15^\circ$. The total number of encoded actuators in the body is seven: two for

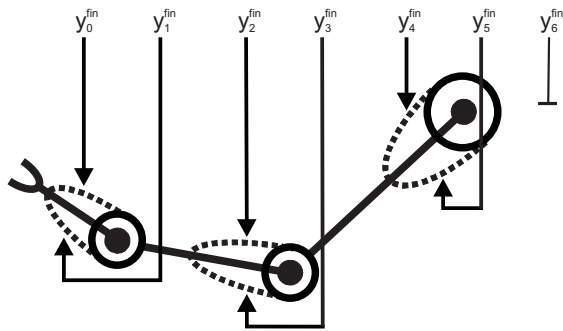


Figure 5. The three joint arm is controlled by three pairs of antagonistic actuators. The actuators are gradually controlled by motoneurons y_i^{fin} . Additionally, a seventh motor command exists that has no effect on the arm.

each of the three joints and one null actuator. Any weighted combination of actuators is possible. The null actuator has no effect on the arm. It is used to include the possibility to observe motor commands during learning that do not cause movements in the relevant limbs.

Of course, this arm model lacks the complexity of a human arm, but it captures two important features. First, the arm has redundant degrees of freedom. It is kinematically redundant because each desired hand location can be realized by multiple arm postures. It is redundant on the sensorimotor level because each arm posture can be reached by many different motor command sequences.

Second, the temporal alignment problem has to be solved during learning. Most goal states cannot be reached by issuing a single motor command but require the execution of a motor command sequence. Thus, although joint torques and velocities are not modeled, the arm model captures the challenging property that, in general, goals cannot be reached by issuing a single motor command. Additionally, during learning the arm jiggles around randomly so that no clear relationship between motor commands and their outcomes can be easily established.

Hand and Posture Space

Hand space and posture space are represented by two population codes. The population codes cover the spaces with a uniform field of partially overlapping receptive fields. Due to the overlap, a particular position is encoded by a unique activity distribution in the neurons whose receptive fields overlap with the position.

Hand space is encoded by a vector \vec{h} of size $h = 21 \times 21 = 441$ with local activation patterns for each neuron (see Figure 6). Each neuron h_i of \vec{h} fires if the current hand coordinates (x_{hand}, y_{hand}) are sufficiently close to the neuron's preferred hand location (h_i^x, h_i^y) :

$$h_i = \max\left(1.0 - \frac{|x_{hand} - h_i^x|}{.24}; 0\right) \times \max\left(1.0 - \frac{|y_{hand} - h_i^y|}{.24}; 0\right) \quad (1)$$

This sets the activation of the neuron to 1.0 if an actual hand location coincides with

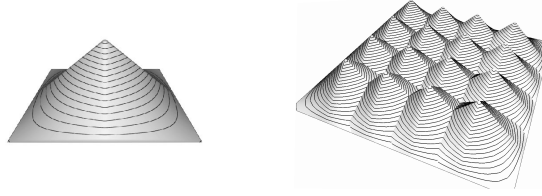


Figure 6. The chart shows the pyramidal activation function of a single hand space neuron (left) and the alignment of the activation functions of several neurons (right).

the neuron’s preferred hand location. The receptive fields of the neurons are distributed equidistantly in the coordinate space centered on the shoulder joint and covering a square with side length twice the length of the extended arm. The receptive fields of adjacent neurons overlap, reaching an activity of zero at the center of the eight neighboring receptive fields, as illustrated in Figure 6. The resulting distance between adjacent preferred hand locations is 10% ($d = .24$) of the length of the stretched arm. Due to the overlap of receptive fields and the shape of the activation function, any location of the hand in the coordinate space has a unique neural activity encoding, which is expressed by the four closest neurons surrounding a hand coordinate.

The posture space is represented likewise by a vector \vec{p} of size $p = 9 \times 9 \times 5 = 405$. The neuron’s receptive fields cover the entire posture space ($360^\circ \times 360^\circ \times 180^\circ$). The consequent distance between adjacent preferred postures is 45° . Dependent on the current arm posture, each neuron has the following activity:

$$p_i = \prod_{j=1}^3 \max\left(1.0 - \frac{|\phi_j - p_i^{\phi_j}|}{45^\circ}; 0\right), \quad (2)$$

where $p_i^{\phi_j}$ are the preferred joint angles of each neuron p_i . Thus, all neurons are broadly tuned to 90° wide, overlapping receptive fields in each dimension. Just like hand space, the posture space as well has a unique encoding for any possible posture.

Posture Memory

The posture memory encodes the complete inverse kinematics of the arm. It learns to associate hand space activations with corresponding, redundant posture space activations. The upper left of Figure 7 shows a population encoded hand target and the population encoded redundant posture representation generated by the posture memory.

The posture memory is implemented as a fully connected single layer neural network, with vectors of neurons encoding hand position (\vec{h} of size $h = 21 \times 21 = 441$) and posture (\vec{p} of size $p = 9 \times 9 \times 5 = 405$) forming the input and output layer, respectively. The network weights encode the degree of correlation between postures and hand positions. The weights are stored in the $p \times h$ -matrix W_{PM} . To compute a set of appropriate postures \vec{p}_g for a desired hand position \vec{h}_g , the activation of the hand space neurons is

propagated through the neural network as follows:

$$\vec{p}_g = W_{PM} \times \vec{h}_g \quad (3)$$

To train the neural network a Hebbian learning rule is applied (Hebb 1949):

$$W_{PM}(t) = W_{PM}(t-1) + \epsilon \vec{p} \vec{h}^T, \quad (4)$$

where \vec{h}^T denotes the transpose of column vector \vec{h} . The learning rate parameter $\epsilon = 0.001$ modulates the speed of learning. This unsupervised learning rule increases the synaptic weights between neurons that are active at the same time. This learning rule causes weights to grow very large if applied over a long period of time. In the following simulation, this has no effect because learning is limited to 1.000.000 time steps and the output of the posture memory is always normalized. The problem of infinite weight growth could be addressed by including a threshold term or reducing the learn rate throughout learning (Milner 2003).

Due to the population encoding of hand space and posture space, joint angles of different matching postures are not intermixed. Thus, the nonconvexity problem, which constraints the applicability of direct inverse modeling (DIM), does not limit the posture memory. Moreover, the overlapping receptive fields of the population codes result in an implicit generalization. The neural encoding of any hand coordinate is associated with the corresponding neural encoding in posture space. Due to the redundancy of the arm and the associative learning rule, the posture memory will associate progressively larger areas of posture space with each hand coordinate.

Motor Controller and Sensorimotor Model

The motor controller is engaged in two partially overlapping processes: the preparation of a movement and its execution by closed-loop control. During movement preparation, a target representation is transformed into a sensory-to-motor mapping that provides for each possible posture a set of motor command that enables pursuing the goal. The movement preparation is based on the sensorimotor model. Once a movement is prepared, motor commands, which are provided by the sensorimotor mapping, are repeatedly forwarded to the arm, resulting in effective close loop control. Note, that movement preparation and movement control may happen in parallel, generating a mapping for a new goal while another movement is still executed.

The motor controller and the sensorimotor model are organized in $n = 7$ *action columns*. Each action column is associated to a specific motor command and consists of a synaptic weight matrix which stores a part of the sensorimotor model and neural circuitry for learning, movement preparation, and movement execution, which is a part of the motor controller. During motor learning, an action column extracts the sensorimotor contingencies related to the associated motor command. Thus, after learning, the synaptic weights of each action column encode the posture transitions that occur if the associated motor command is executed. Together, the synaptic weight representations of all columns form the sensorimotor model.

During movement preparation, neural activity is distributed in a representation of the posture space by dynamic programming. Activity spreads out from a goal representation, resulting in a pattern of neural activity that depends on the learned synaptic connectivity and on crosstalk between action columns. Finally, each neuron of the posture space representation has a certain activity. Figure 7 (right) shows the activation maps resulting from the movement preparation process. Together, the activation pattern of the seven action columns represent the sensory-to-motor mapping, which is used for movement execution. A neuron’s activity reflects the degree to which the motor command associated to the action column is suited to pursue the current goal from the posture encoded by the neuron. By comparing the activities in those neurons of the different action columns that encode the current arm posture, a motor command can be shaped and executed (Figure 7, bottom). This changes the arm posture and causes the subsequent generation of new motor commands, resulting in proprioceptive closed-loop control.

Movement Preparation

Figure 8 (top) shows the neural circuitry of an action column that is used for movement preparation. For movement preparation, each column i makes use of neurons that encode the posture space as well as the synaptic weights that encode sensorimotor contingencies. The posture space is covered by a population code, which is represented by an activation map \vec{a}_i of size p . The sensorimotor contingencies are represented by the synaptic weight matrix W_i of size $p \times p$. The synaptic weight between two neurons has been increased during learning, if the posture represented by one neuron often preceded the posture represented by the other.

The activation pattern, which is necessary to direct movement execution, may be initiated in hand space and is then converted to a posture space representation by the posture memory. Then, this activation pattern is diffused by means of the synaptic weights that form the sensorimotor model (W_i) using dynamic programming. The activation propagation process in one action column is illustrated in the top part of Figure 8. In the figure, the goal is to move to the posture represented by the most rightward goal neuron (activation is indicated by a black circle). Let’s assume that the action column is associated to a motor command that usually moves the arm from the posture encoded by the leftmost neuron to the posture encoded by the rightmost neuron. Hence, synaptic weights were established during learning, that connect the rightmost neuron to its left neighbors. The goal activation spreads through these weights to the leftward postures, which are exactly those postures, from which the goal is reached if the motor command associated to the columns is executed. As the synaptic weights are not symmetrically and differ in the action columns, a unique activation pattern emerges in each column. Finally, those neurons that encode identical postures in different action columns are interconnected. These lateral connections enable the propagation of activation to postures from which the goal can only be reached by a sequence of different motor commands.

To generate activation patterns in action columns, the following computations are carried out. Given the current goal-based posture space activation, which is represented

in the neural activity of the posture space, the activation map of the i -th action column \vec{a}_i is updated by the following equations:

$$\vec{a}_i^* \leftarrow \max \left\{ \beta \left(\gamma \frac{\sum_{j \neq i} \vec{a}_j}{n-1} + (1-\gamma) \vec{a}_i \right), \vec{p}_g \right\}, \quad (5)$$

$$\vec{a}_i \leftarrow \vec{a}_i^* + W_i \times \vec{a}_i^*, \quad (6)$$

where $n = 7$ is the number of action columns, \max is an operator that computes the entry-wise maximum of two vectors, β is a scalar that scales down network activation, and the coefficient γ is a scalar that determines the contribution of other action columns versus the own action column during activation propagation. The settings of these parameters are specified in the appendix. The goal activation pattern (\vec{p}_g) is normalized in each iteration, so that the component values in each vector add up to 1.0. The action column activation vectors (\vec{a}_i) are restricted to values between 0.0 and 1.0.

The iterative equation propagates activity backwards within the posture space starting from the goal activation pattern (\vec{p}_g). Parameter γ balances the activation propagation within one action column (\vec{a}_i) with the activation propagation stemming from other action columns ($\vec{a}_{j \neq i}$). Parameter β discounts activation propagation and assures that activation ceases when no goal activation is applied. In effect, the activation pattern in each action column \vec{a}_i encodes distance in posture space to the currently activated goals, given motor command i is excited first. Larger values indicate smaller distances.

Movement Execution

To trigger motor activity, first, the activation maps generated during movement preparation need to be read out. As mentioned, the activations in the columnar posture space neurons \vec{a}_i encode the suitability of the i -th motor command for pursuing the goal with respect to all postures. To access the activity of the different activation maps for a particular posture, the activity of the maps is multiplied by a representation of the current posture (see Figure 7). Thus, the inner product between current posture space activation representing the actual arm posture \vec{p} and the columnar posture space activity \vec{a}_i is computed:

$$y_i = \vec{p}^T \vec{a}_i, \quad (7)$$

which effectively determines the suitability of the i -th motor command to reach the currently activated goal given current arm posture activity \vec{p} . The lower part of Figure 8 illustrates the inner product computation dependent on the state of the current posture in one action column. In the figure, the leftmost neuron of the encoding of the current posture is active. Thus, only the activity in the leftmost neuron of the activation map is used to compute the motor command.

The motor commands y_i are normalized to the squared activation before they are forwarded to the arm:

$$y_i^{net} = \frac{y_i^2}{\sum_{j=0}^6 y_j^2} \quad (8)$$

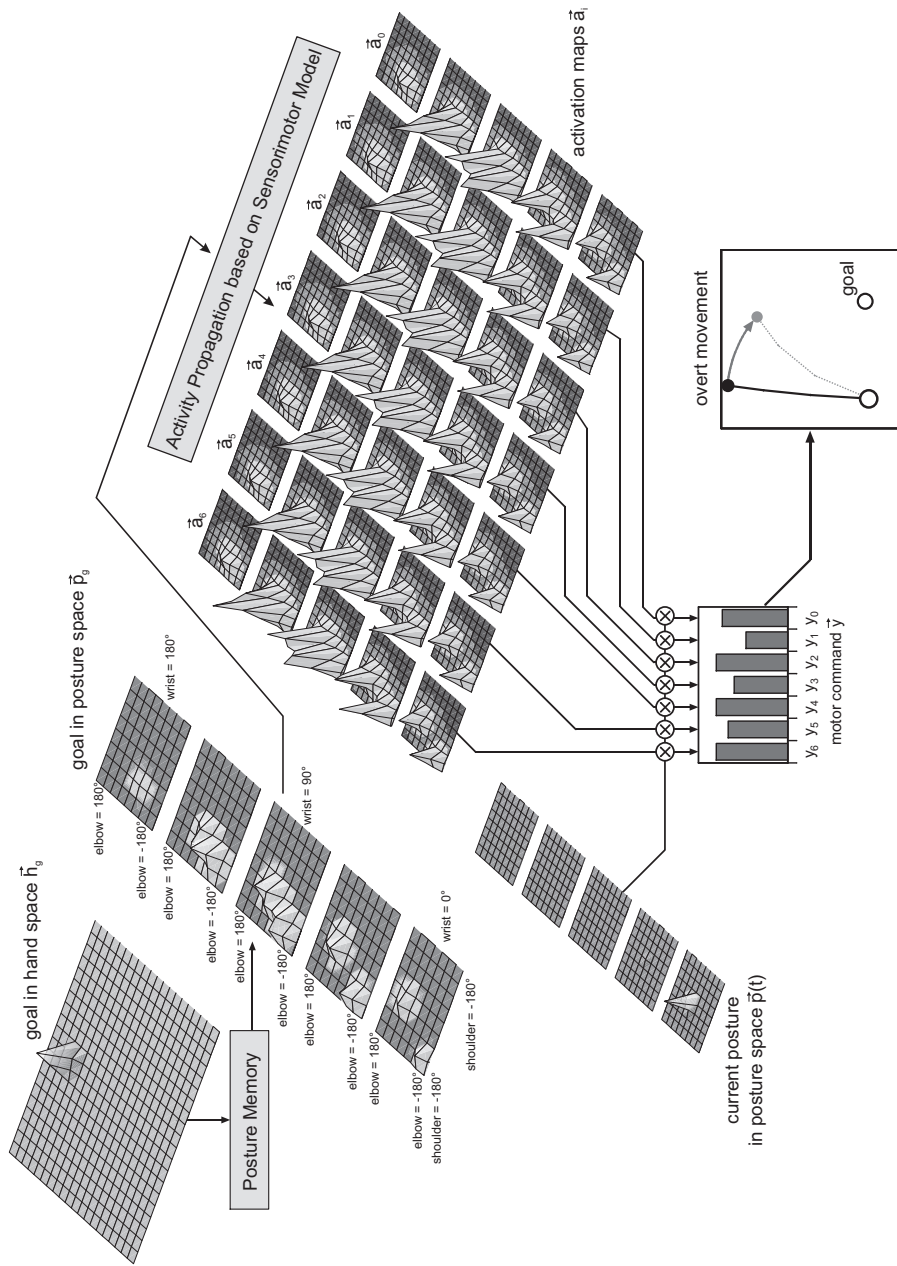


Figure 7. The chart depicts the neuronal activation within SURE_REACH during movement preparation and movement execution phases. During movement preparation, a goal that is first encoded in a population of neurons in hand space (\vec{h}_g) is transformed into a representation of all corresponding postures (\vec{p}_g) by the posture memory. This goal activity induces a dynamic programming process in the motor controller, which results in the generation of slightly different activation patterns in the activation maps \vec{a}_i of the sensory-to-motor mapping. To control a movement, the activities in the activation maps are read out by combining them with a representation of the current posture. This results in different excitation levels in the motor commands y_i causing a movement toward the goal.

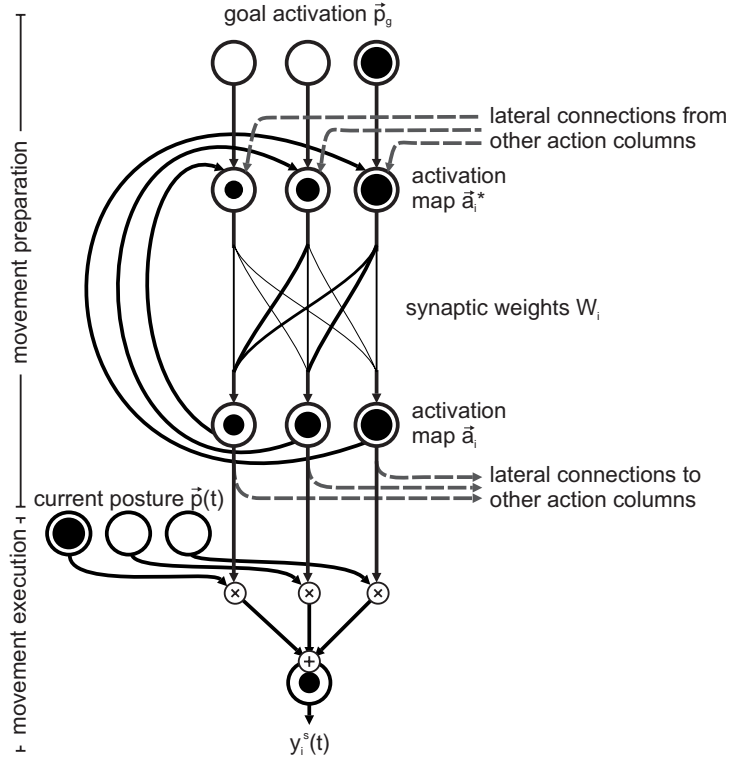


Figure 8. During movement preparation, an action column propagates goal activations using its learned weight matrix W_i , incoming and outgoing connections to other columns (lateral connections). The so generated activation map \vec{a}_i can be read out during movement execution by multiplicative units that determine the suitability of a motor command dependent on the current posture state (top). The graph represents the activation propagation process specified in Equations 5 and 6. The black dots indicate exemplar activations, spreading out from the goal encoding neuron (\vec{p}_g), bottom right).

The activations of opposing actuators (those that cause antagonistic movements) then cancel each other out. That is,

$$y_i^{net*} = \begin{cases} y_i^{net} - y_j^{net}, & \text{if } y_i^{net} > y_j^{net}; \\ 0, & \text{otherwise} \end{cases}, \quad (9)$$

given i and j are antagonistic motor command pairs. Finally, the resulting absolute activities are normalized to 1.0 and multiplied by gain factor g .

$$y_i^{fin} = g \frac{y_i^{net*}}{\sum_{j=0}^6 y_j^{net*}} \quad (10)$$

In this way, movement velocity becomes independent from absolute activation levels and could in principle be further modulated (Bullock, Cisek & Grossberg 1998; Bullock et al.

1993). Currently, during learning y_i^* is purely determined by an endogenous excitation generator that causes random motor commands and hence random movements.

Motor Learning

In order to prepare and execute movements, the synaptic weights of the sensorimotor model have to be adapted during initial random movements. Figure 9 shows the circuitry in the action columns that is necessary for motor learning. Leaky integrator neurons \vec{r}_i encode traces of previous posture activations. If the motor command represented by an action column i is executed during motor learning, neurons that represent subsequent postures are associated, storing the encountered sensorimotor contingencies in W_i .

Learning is again realized by a Hebbian learning rule that updates the weights in the synaptic weight matrix of column i . The rule strengthens the weights between simultaneously firing posture space activation neurons $\vec{p}(t)$ and columnar recurrent connection neurons \vec{a}_i , dependent of the currently issued motor command.

$$\vec{r}_i(t) = y_i(t-1)\vec{p}(t-1) + \rho\vec{r}_i(t-1), \quad (11)$$

$$w_i^{jk}(t) = w_i^{jk}(t-1) + \delta r_i^j(t)p^k(t)(\theta - w_i^{jk}(t-1)), \quad (12)$$

where lower case letters with upper indices indicate particular values in the weight matrix W_i , the columnar leaky integrator neurons \vec{r}_i , or the current posture space activation \vec{p} . δ is the learning rate, ρ is the decay coefficient of the leaky columnar activity r_i , and y_i^* is the activation of the i -th motoneuron during learning. The upper threshold θ prevents the network weights from growing infinitely high after long training phases. The settings of the parameters are specified in the appendix.

After learning, the weights represent motor command dependent associations between successively firing posture encodings resulting in the desired sensorimotor model. The leaky integrator mechanism in neurons \vec{r}_i links each state not only to the directly preceding states, but to a trace of many preceding states in which the motor command associated to the action column has been carried out. This mechanism enables the motor command dependent association of temporally remote postures. Thus, each (potential goal) state gets linked to those states from which the potential goal can be reached by carrying out the motor command associated to the respective action column, regardless if this is sufficient to acquire a goal or if the execution of subsequent motor commands is required. Note that the connections between potential goals and temporally remote states are weaker due to the discounting factor of the leaky integrator neurons. This implies that motor commands that move the arm faster from a given posture to a goal are executed preferably. This mechanism helps SURE_REACH to cope with the temporal alignment problem.

Remote Goals in Learning and Movement Preparation

The last sections described the movement preparation by dynamic programming and motor learning. In this paragraph, I want to discuss why the apparent complexity of these processes is necessary to enable learning and enhance behavioral flexibility. At

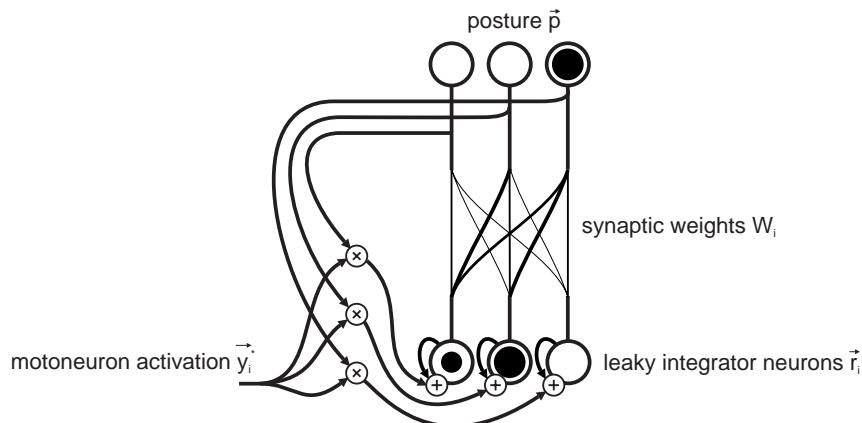


Figure 9. During learning, each action column develops a unique weight matrix (W_i) between successively firing neurons. The graph visualizes the inverse model learning approach, specified in Equations 12 and 11. The columnar leaky integrator neurons \vec{r}_i remember not only the most recent posture activity but a motor command dependent, exponentially decreasing trace of recent activities enabling the motor command dependent association of more distant postures.

the first glance, the synaptic weights of the action columns seem to be well suited to store associations between any two reachable postures. However, even if the neural networks enable in principle to represent association between any two postures, it is hard for the unsupervised learning mechanism to acquire such associations. Consider a simple unsupervised learning mechanism, much like direct inverse modeling (DIM), that associates the posture and motor command in time step $t - 1$ to the posture in time step t to encode a mapping from current and desired postures to motor commands. If a movement from posture A to a distant posture B takes a considerable amount of time, the learning mechanism would never encounter a motor command that, in one time step, moves the arm from A to B . Hence, if B is desired and the current state is A , no motor command can be provided. However, it might be possible to move from A to be B by executing a sequence of motor commands. This problem is tackled by two independent mechanisms.

First, the learning rule itself does not only associate postures to the immediately preceding ones but also to the postures encountered before, however only with discounted associative strength and dependent on the then executed motor commands. This enhancement to basic DIM enlarges the maximal distance between postures that can be associated by the learning mechanism but it still implies that only movements can be executed that have been executed during motor learning. Also, this mechanism alone would result in a very inflexible mapping from goals to motor commands.

The second mechanisms, the movement preparation by dynamic programming, finally solves these problems. On the one side, associations between remote postures can be established because the iterative use of local synaptic connections spreads activity from

the goal encoding neurons to all neurons in the activation maps. On the other side, as the association between the current posture, the desired posture(s) and motor commands are not fixed but are newly generated for each goal, the movement preparation process can be adjusted to the requirements of the current situation or task. For example, the contribution of specific joints to a movement can be minimized by reducing activation propagation in certain action columns or obstacles can be avoided by inhibiting certain neuron populations. Thus, the complexity of the learning mechanism and the movement preparation phase adds two important functionalities. First, they enable unsupervised motor learning of temporally far-reaching sensorimotor contingencies thus solving the temporal alignment problem. Second, they enable the model to account for the use of multiple trajectories to reach goals and thus enhance behavioral flexibility.

Examples for Movement Preparation and Execution

The following examples illustrate movement preparation and execution by the motor controller and the sensorimotor model. The activation propagation within the actual simulation, conducted on the three-joint arm, is shown. To facilitate visualization, examples are chosen in which wrist movements are not crucial. Of course, the evaluation in the subsequent sections is not restricted in such a way.

Example: Approaching a Particular Posture

Figure 11A shows a typical activity propagation process in the *SURE_REACH* implementation, when one particular goal posture is activated. Only in this illustrative example, the wrist angle does not change from 0° to enable the two-dimensional representation of the posture space. Figure 11A shows cross sections (wrist angle is 0°) of the activation maps. Rows show the differing maps of four action columns. Columns show the maps at different moments during movement preparation. White areas are not activated at all, dark areas are highly activated. The data stems from a well trained controller (see next section for training procedure).

Initially, only one particular goal posture is activated in this example. Next, the activity is propagated in the four rows of Figure 11A. This can be clearly seen in column $t = 1$ where, for example, a counterclockwise movement of the shoulder (top row) co-activates the clockwise space more strongly whereas a clockwise movement (second row from top) co-activates the counterclockwise space more strongly. Due to the leaky-integrator property of neurons \vec{a}_i during learning, not only the immediate surrounding is activated but also more distant areas of posture space. These far-reaching connections facilitate the movement preparation process and enable increasingly faster movement onset, as will be shown later.

The motor commands, which are ultimately sent to the actuators, depend on the activation of the neurons representing the current state in the different activation maps. Figure 11B shows the corresponding sensory-to-motor mapping, which is generated from the activation maps in the seven action columns by applying the normalization procedure described in the previous section (Equations 7 - 10). The action columns of the motor controller trigger motoneuron activity for many locations in posture space. However,

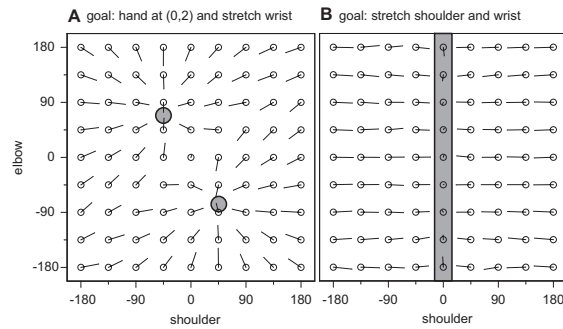


Figure 10. A) The goal (marked in gray) consists of two distinct joint angle configurations. B) The goal is the infinite set of postures with a stretched shoulder and wrist. Both charts shows intersections (for $\phi_{wrist} = 0$) of sensory-to-motor mappings.

between some state neurons no synaptic connectivity may have emerged because they represent very distant postures and the transition time between those postures is high. Thus, the activation maps for $t = 1$ do not cover the entire posture space by far. To enable the action column to establish a connection between highly remote postures, activation is propagated further. The activation maps for $t = 2$ cover a substantial larger part of the posture space. After some iterations ($t = 10$ in the example), suitable sets of motor commands are available for the entire posture space.

Movement can be initialized as soon as the activation pattern reaches the current arm posture. Figures 11C, D show the resulting movement trajectories in hand space and posture space, respectively. As the arm posture changes, the closed-loop control process reads out the activation of the state neurons and issues motor commands accordingly.

Example: Activation Maps with Multiple Goals

The last section described the activation of a single posture. However, the posture memory usually provides a goal representation which includes not only a single goal posture but many of them. The activation propagation process can also be initialized when more than one goal neuron is activated. In this case, the sensory-to-motor mapping does not lead toward a single posture but toward a set of goal postures. Thereby, each of the postures in the goal set represents an acceptable final state. In Figure 10A, the goal set is the solution of the inverse kinematic problem of moving the hand to a specific location and thereby maintaining a stretched wrist joint. The sensory-to-motor mapping directs the shortest path to the goal for all possible arm postures. Note that this problem cannot be solved by DIM. Furthermore, goal sets with an infinite number of solutions are possible. For example, Figure 10B charts the sensory-to-motor mapping to any posture with stretched wrist and shoulder joints.

Example: Obstacle Avoidance

So far, examples without trajectory constraints were given. However, SURE_REACH is also able to avoid arbitrary areas of posture space, consequently realizing, for example,

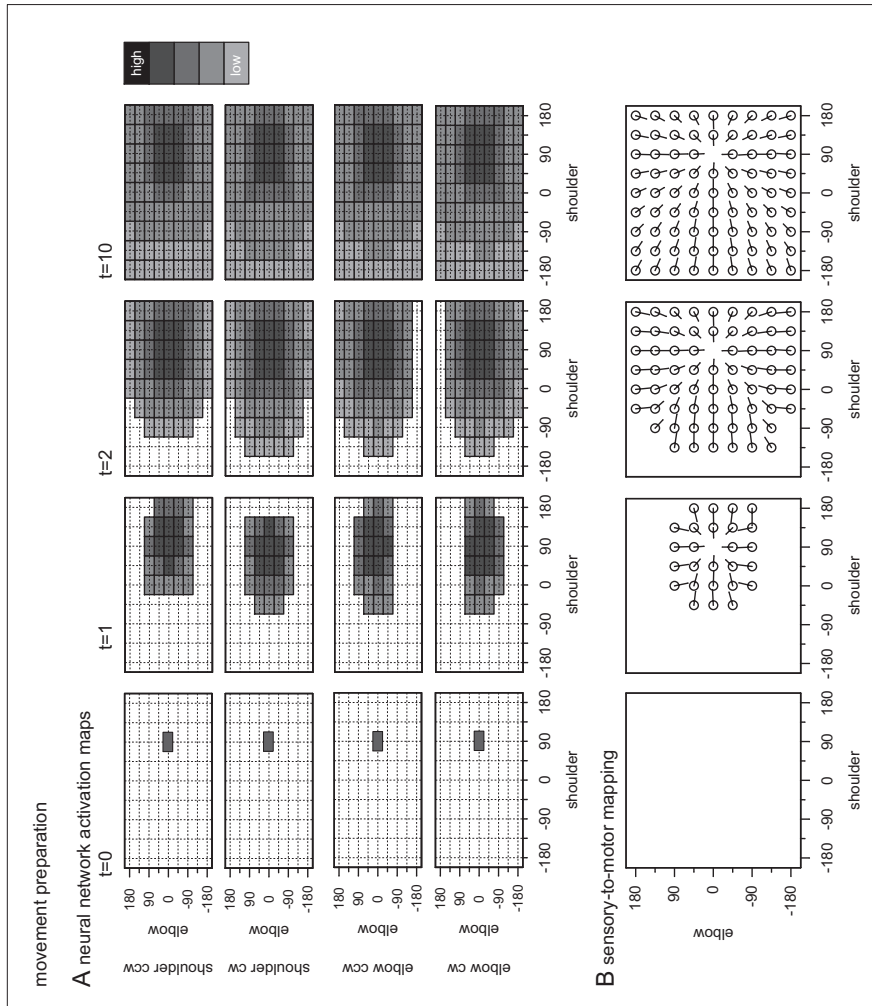
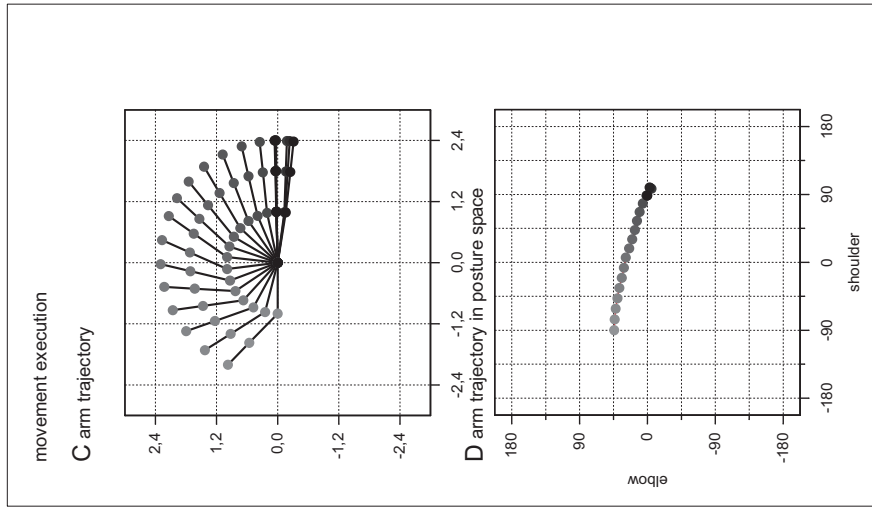


Figure 11. A) The maps display the activation level of those neurons that encode postures with $\phi_{wrist} = 0^\circ$ at different time steps and for different motor commands (cw = clockwise, ccw = counterclockwise). White areas are not activated at all, dark areas are highly activated. B) The sensory-to-motor mappings are derived from weighting the activation levels in the different neurons in the action columns. They display the effects that the motor commands generated by the motor controller would have, depending on the actual arm posture. C) The chart shows the movement that the motor controller exerts after a sensory-to-motor mapping is available for the initial joint configuration. D) The trajectory in posture space ($\phi_{wrist} = 0^\circ$) is nearly linear.

obstacle avoidance. In the following example the neural activity for some postures is inhibited, effectively disabling the arm to pass through the associated areas in posture spaces. In the subsequent evaluations, areas specified in hand space are avoided by inhibiting neurons in hand space and propagating this inhibition through to posture space using the learned posture memory. Note, however, that currently the arm only encodes its end-point in coordinate space so that it is only possible to prevent collision of the hand, not that of other parts of the arm. However, it is imaginable that in later simulations the full arm may be represented in coordinate space and associated with posture space, which will make more advanced obstacle avoidance possible.

Figure 12A shows an example in which the posture space is constrained disallowing postures in which the elbow is flexed when the shoulder angle is between -90° and 90° . The movement starts with the upper limb pointing downward ($\phi_{shoulder} = 135^\circ$) and a 90° -flexed elbow. The goal posture requires a counterclockwise movement of the shoulder until the upper limb points straight down. The elbow and wrist angles in start and goal posture are identical, but the additional constraint requires an extension of the elbow during the movement.

The constraint is realized by inhibiting all postures that collide with the virtual obstacle. The constraint requires the activation to spread from neurons in the clockwise elbow rotation action column, to neurons in the counterclockwise shoulder rotation column, as can be inferred from the activation maps in Figure 12A. The resulting sensory-to-motor mapping (Figure 12B) causes substantial changes to the motor commands during the movement. Figures 12C, D show that the trajectory contains changes in the elbow motor commands to avoid the obstacle, first stretching it and then flexing it again.

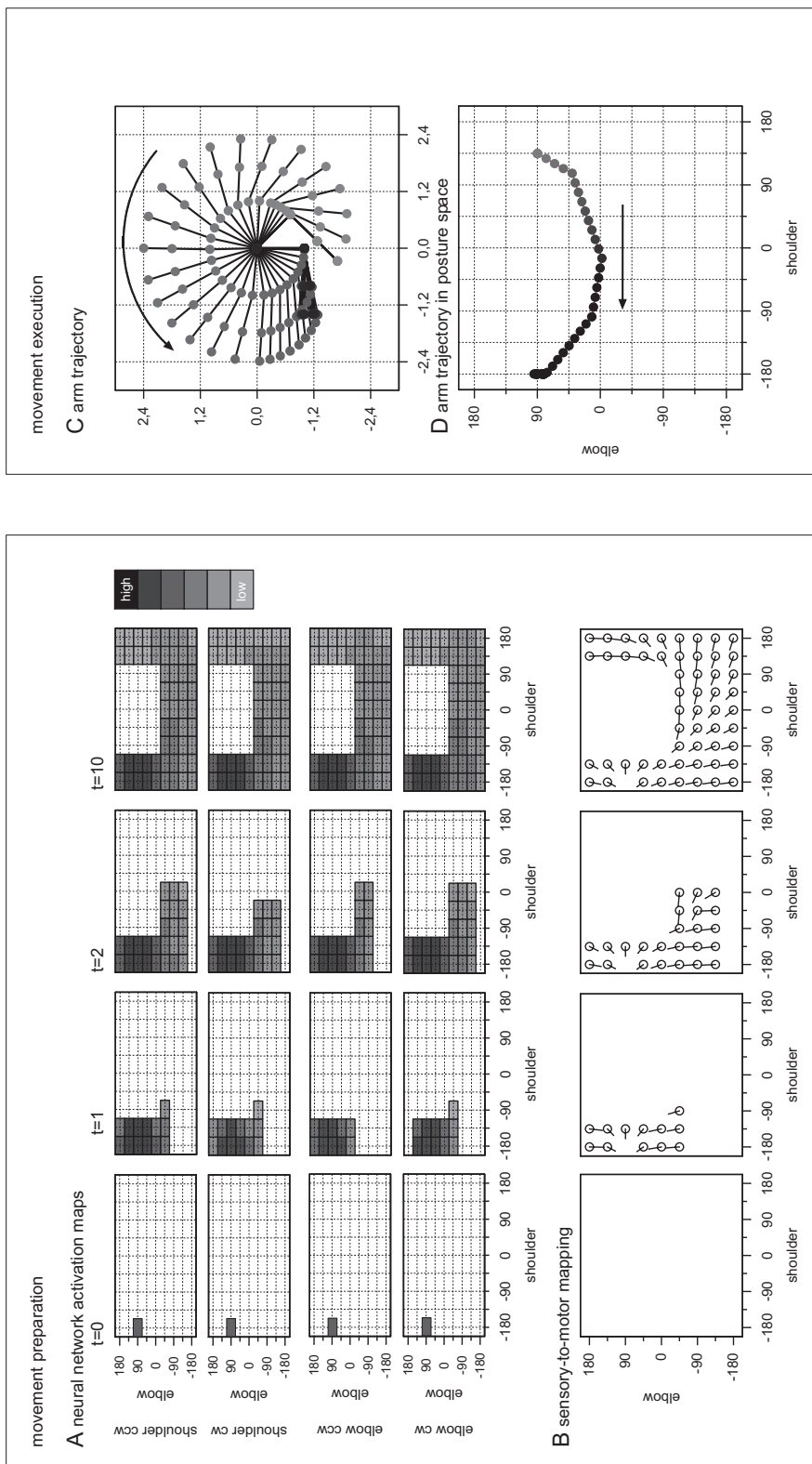


Figure 12. A) The maps display the activation level of the neurons that encode postures with $\phi_{wrist} = 0^\circ$ at different time steps and for different motor commands (cw = clockwise, ccw = counterclockwise). White areas are not activated at all, dark areas are highly activated. State neurons that represent an obstacle remain unactivated (white rectangle) because they are inhibited. B) The sensory-to-motor mappings derived from activation levels in the action columns show that the motor controller is able to avoid the obstacle. They display the effects that the motor commands generated by the motor controller would have, depending on the arm posture. C,D) Chart C and D show how the movement bypasses the obstacle in hand (C) and posture space (D). To avoid the obstacle, the elbow is first rotated counterclockwise and then clockwise. Charts A, B and D show cross sections of maps for $\phi_{wrist} = 0$.

Simulation of Human Behavior

After this exemplification of SURE_REACH’s capabilities, I now evaluate learning and behavior in the architecture systematically and relate it to human motor behavior. The evaluation starts with an assessment of the general learning accuracy of the model and its parameter dependency. Next, the behavior of the model is characterized and compared to various behavioral findings in humans. First, the effects of extensive training on reaction and movement times are considered. Second, I show that the dynamics of the movement preparation in SURE_REACH produce priming effects that are comparable to psychological experiments. Third, it is confirmed that SURE_REACH benefits from the representation of motor redundancy on the end-posture level as well as on the trajectory level, exhibiting behavior in accordance with experimental data.

General Properties

A model that accounts for motor learning and control should improve during learning and acquire at least some dexterity. To show this, the overall improvement of movement accuracy during learning in SURE_REACH is evaluated. First, the sensorimotor model and the motor controller alone are evaluated for movements toward specific arm postures. Second, the complete SURE_REACH model, including its posture memory, is evaluated for movements toward specific hand targets. Finally, to assure that the model is robust and does not rely on a fine-tuned set of parameters, the sensitivity of the model’s performance is tested with respect to its most important parameters.

Motor Controller

To evaluate the overall properties of the model, ten controllers were trained individually for 1,000,000 time steps each. The ten controllers differed in two properties. First, their learning experiences differed because each controller executed a different sequence of random motor commands during motor learning. Second, the goals generated during evaluation of each controller differed. During performance tests, each controller had to perform a unique set of test movements, starting from certain randomly selected postures, pursuing particular goals. Each controller had to perform the set of test movements after various iterations of training. Thus, variances in the performances of the controllers reflect both, variances due to different learning experiences and variances due to different test movements.

Training began by setting the arm to a random posture. Then, a random motor command set was generated and changed every one to four time steps. Each motoneuron was set to 1.0 with a probability of 0.3 and was otherwise 0.0. If none of the seven motoneurons were activated, this procedure was repeated. In each time step, the neural network weights of the posture memory and the sensorimotor model were updated. To evaluate learning progress, the neural network weights were frozen and the controller was tested. After a test trial, training was continued from a new random posture.

To assess the accuracy of the sensorimotor model and the capabilities of the motor controller alone, each individual controller was tested in 16 trials to move from randomly selected starting postures to randomly selected target postures ($\phi_{shoulder}, \phi_{elbow} \in$

$[-135^\circ; 135^\circ]$, $\phi_{wrist} \in [45^\circ; 135^\circ]$). Target posture activity (\vec{p}_g as in Equation 3) was directly generated, bypassing the posture memory (using Equation 2). The controller was allowed to take up to 80 time steps to reach the activated goal posture.

The average error of the last 10 time steps of each movement was used to determine the accuracy of a single movement. The error was computed by averaging the absolute differences between the target and actual shoulder, elbow, and wrist angles. To assess the accuracy of a controller, the average movement accuracy and the worst movement accuracy during a test phase were computed. Figure 13A shows the average and worst case accuracy of the 10 individual controllers. The error before training was on average 82.1° ($SD = 5.84^\circ$). After 1,000,000 steps of training, the average error dropped to 3.52° ($SD = .114^\circ$) and the average error of the least accurate movement of each controller dropped to 4.43° ($SD = .314^\circ$). The results confirm that the architecture is capable of reliably moving the arm with reasonable accuracy to desired postures, considering the sparse distribution of the receptive fields (45° between adjacent centers of receptive fields).

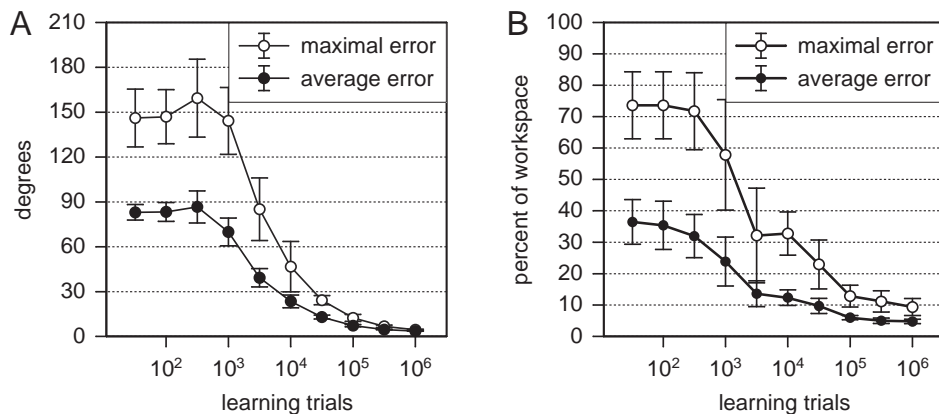


Figure 13. A) The average and worst end-posture error in mean absolute degrees continuously improves during training. B) The average and maximal Euclidean distance from the desired goal location continuously decreases during learning and consistently reaches a low level. Error bars show standard deviations.

Posture Memory and Motor Controller

Similar to the evaluation of reaching goal postures, the accuracy of the whole architecture was evaluated. The error is now defined by the Euclidean distance between the goal hand position and the final hand position in percent of the workspace size (twice the length of the arm). Each controller had to perform 16 movements from random start-postures to random hand targets. Hand targets (\vec{h}_g) were generated by computing the hand position of random arm postures to assure that the target was within reach of the arm. Figure 13B depicts the average and maximal error during learning. The average error dropped from 36.5% ($SD = 7.07\%$) of the workspace size to 4.73% ($SD = .715\%$)

after 1,000,000 time steps of learning. The average error of the least accurate movements of each controller was 9.32% ($SD = 2.70\%$).

The results confirm that the SURE_REACH implementation can account for accurate goal-directed hand movements. As in the case of the posture space evaluation above, the remaining error in final hand location is due to the low resolution of posture and hand space. Potential solutions to this problem are discussed below.

Parameter and Training Sensitivity

To assure that the performance is not bound to a specific set of parameters or a specific training setup, three important factors were systematically varied. First, the random activation generation procedure applied in the training phase was varied. Either exactly one random motoneuron was activated or each motoneuron was activated with a probability of 0.3 or 0.5 (repeating the motor command activation process, if no motor command was excited). Second, the joint angle gain g was set to 11.25° , 15.0° , or 22.5° during learning. Third, parameter ρ , which modulates the leak of the leaky integrator neurons, was set to 0.0, 0.1 or 0.5. Ten individual controllers were trained for each combination of values and the performance after 1,000,000 time steps of learning was assessed as described above, except that only the final arm state of a movement was used to calculate error values.

When the sensorimotor model plus motor controller were tested alone by activating random, single goal postures, little impact from the parameter variations could be observed. A 3x3x3 ANOVA revealed only a significant influence from the random activation generation procedure, yielding better results if more than one motor command could be active at a time, $F(2, 243) = 7.46$, $p < .01$. However, the variation had little impact on absolute performance: All average end-point errors ranged between 3.25° ($SD = .370^\circ$) and 4.05° ($SD = .199^\circ$).

When the complete architecture was tested by the activation of goal coordinates and the successive posture space activation by the posture memory, a 3x3x3 ANOVA also revealed a significant influence from the random activation generation procedure, $F(2, 243) = 18.5$, $p < .01$. In this case, better results were obtained when only one motor command could be applied at a time. For the co-activation of multiple goal postures, the results indicate that it is slightly better to learn the sensorimotor model by executing single motoneurons, since the inverse model becomes more uniquely dependent on each motor command. Nonetheless, the results ranged between 3.85% ($SD = .686\%$) and 5.10% ($SD = .727\%$), so that the absolute difference in error remained small.

In conclusion, the small performance differences suggest that parameter and training variations only slightly affect model performance. Thus, the architecture does not need a fine-tuned set of parameters to work but seems generally robust.

Effects of Extensive Training

It is a common finding that training does not only affect the accuracy of human movements, but also that movement times and reaction times decrease during learning (Gottlieb, Corcos, Jaric & Agarwal 1988; Flament et al. 1999; Lavrysen et al. 2003;

Ludwig 1982). The model contributes to both findings. First, in SURE_REACH the time it takes to initiate a movement decreases by encoding temporally far-reaching sensori-motor contingencies. The further reaching the contingencies, though, the longer it takes to encode them during training. Second, the movement time decreases because sensori-motor model and posture memory encode progressively more, and thus more efficient, movement trajectories and postures, respectively.

Reduced Reaction Times

For movements to remote goals, the dynamic programming process requires some time to prepare a movement, because movement onset relies on a sufficient spread of activation through the activation maps (\vec{a}_i). The time it takes from presenting a target to SURE_REACH until the activation is spread far enough to initiate a movement can be considered the *latency* or *reaction time*. This time decreases during motor learning.

The leaky integrator neurons enable the controller to establish direct connections between remote situations and goals, thus being able to replace, or at least enhance, the activation propagation process. If the parameter ρ , which specifies the leak of the leaky integrators, is set to zero, only associations of state neurons that are activated in subsequent time steps are learned. In this case, it is impossible to learn far-reaching connections. The higher ρ , the more far reaching connections will be learned and the faster the movement is initiated.

To confirm this, three groups of ten individual controllers were trained for 1,000,000 time steps with the leaky integrator set to $\rho = 0.0$, $\rho = 0.5$ and $\rho = 0.8$, respectively. The test procedure for movements to different postures was applied. Only the final arm position was used to compute the error value. After 10,000 time steps of training all controllers were trained well enough to be able to initialize movements to all given goals and movement latencies were thus comparable⁵. Figure 14A and Table 2 show that a high value of ρ causes on average a significantly decreased movement latency after 1,000,000 time steps of learning, $F(2, 27) = 36.9$, $p < .01$.

Table 2: Movement latencies for different values of parameter ρ

ρ	after 10^6 time steps		from $10^{5.5}$ to 10^6 time steps		
	latency M (SD)	error M (SD)	Δ latency M (SD)	$t(9)$	p
0.00	.663 (.145)	3.71° (.417°)	.000 (.000)		
0.50	.356 (.179)	3.54° (.227°)	.0438 (.0593)	2.33	< .05
0.80	.113 (.0922)	3.85° (.471°)	.0250 (.0323)	2.45	< .05

⁵The minimal amount of learning required to initiate a movement and the movement latency are likely to be correlated. To avoid a bias in the movement latency that depends on the selection of evaluated movements, only data from evaluation phases, in which all movements could be included into the analysis, are analyzed.

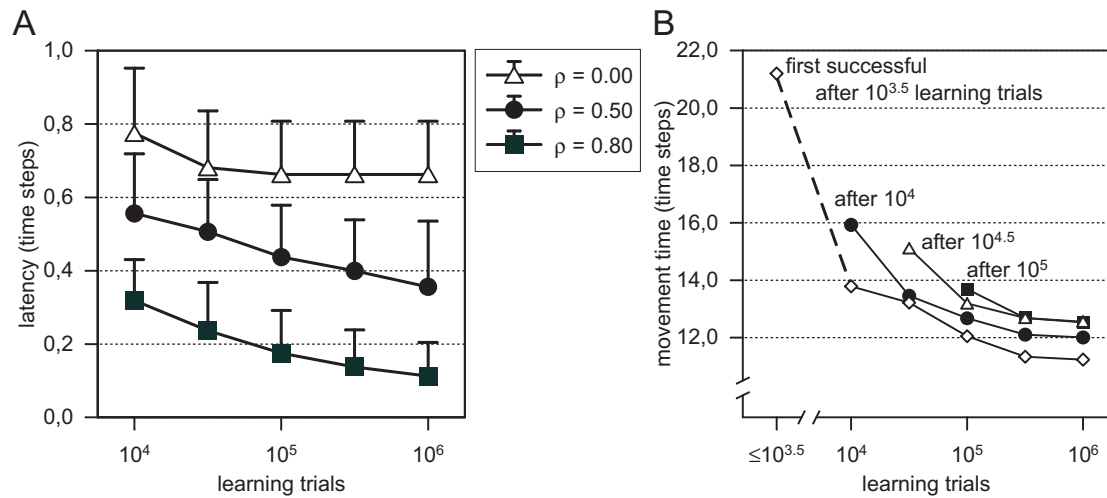


Figure 14. A) Movement latencies are lower and continue to decrease during extensive training if the sensorimotor model is able to learn far-reaching posture associations ($\rho = 0.8$ and $\rho = 0.5$). The decrease ceases if the leaky integrator neurons do not carry the activation of past states ($\rho = 0$). B) The chart shows the movement times for movements to goals that could first be reached after $10^{3.5}$, after 10^4 , after $10^{4.5}$, or after 10^5 time steps of learning. Movement times decrease monotonously.

Furthermore, if ρ is set to 0.5 or 0.8, the movement latency progressively decreases during learning. Pairwise t-tests revealed that a significant decrease is even detectable between the final two test periods (see Table 2). On the other hand, if ρ is set to zero, the decrease of movement onset time ceases completely after 100,000 time steps of learning.

Despite the variations of parameter ρ and the consequent differences in the connectivity within the inverse models, no significant impact on the average end-posture accuracy was detectable, $F(2, 27) = 1.64$, $p > .05$. The results confirm that the leaky integrator neurons enable the inverse model to associate far-reaching arm postures. Movement latency is significantly reduced with extensive training without affecting the accuracy of the reaching movements. Hence, SURE_REACH accounts for the empiric finding that movement latencies or reaction times decrease during motor learning.

Improved Movement Times

Besides a reduction of the reaction time, training also reduces the time needed to move to a goal. In the model, two factors are responsible for this effect. First, during training, the inverse kinematics model learns more and more postures that coincide with particular hand locations. A well-trained posture memory provides a broader goal posture activity (\vec{p}_g). It is likely, that at least some of the postures activated in the broader goal activity are closer to the starting position than any posture in a smaller subset. Hence, movement transitions get faster on average. Second, the representation of sensorimotor contingencies gets more reliable during training and covers bigger parts of

posture space. Since the determination of the motor commands depends on the learned inverse sensorimotor model, more suitable motor commands are generated after longer learning periods.

To show the influence of training on movement time, the data of the previous section was reanalyzed. Each controller had to perform identical movement tasks after varying amounts of learning. Only movement times from movements that did arrive at the target (distance from hand to target lower than 15% of workspace size) were compared. For each run, the targets were divided into four groups: those targets that were reached after $10^{3.5}$ learning iterations (16.3% movements total), those that were reached after 10^4 (19.2% movements total), those that were reached after $10^{4.5}$ (29.6% movements total), and those that were reached after 10^5 learning iterations (29.4% movements total). The remaining 5.6% movements required more than 10^5 learning iterations and were not included in the analysis.

Since parameter ρ had no significant impact on movement times, all controllers were evaluated independent of this factor. Figure 14B depicts the development of average movement times for the four groups of movements. On average, when a movement is successful for the first time, it takes longer, $m = 15.7$, $SD = 2.21$, than after complete training, $m = 12.4$, $SD = 1.50$, $t(29) = 10.8$, $p < 0.001$. Both, the development of posture memory and sensorimotor model contribute to the decrease of movement time. On the one hand, the average distances of the movement in joint space (2-norm) decrease from 186° , $SD = 19.2$ to 180° , $SD = 20.6$; $t(29) = 8.06$, $p < 0.001$. This shows that the posture memory activates end-postures that are closer to the goal after complete training, compared to not fully trained controllers. On the other hand, a similar analysis of the movements to goal postures, which are controlled by the motor controller and the sensorimotor model but not the posture memory, reveals that movement times also decrease, comparing first successful movements, $m = 11.0$, $SD = 0.868$, with completely trained movements, $m = 10.0$, $SD = 0.752$; $t(9) = 6.11$, $p < 0.001$. This shows that due to the increase of both, the accuracy of posture memory and sensorimotor model, movement times decrease during training.

Priming Effects on Movement Execution

Movement preparation and execution often depend on environmental stimuli. Reactions to external stimuli can be facilitated to some degree if primes precede the stimulus that affords a reaction. The influence of these primes on action execution has been extensively studied, mainly with choice reaction time tasks (Dehaene et al. 1998; Kunde, Kiesel & Hoffmann 2003; Vorberg, Mattler, Heinecke, Schmidt & Schwarzbach 2003).

Continuous movements have also been studied in this context. Participants in an experiment by T. Schmidt (2002) were shown a red and a green target at opposing directions from their resting index finger. As soon as these targets appeared, they had to point as quickly as possible to the target with a specific color. For example, a participant could be instructed to point to the red target. However, 10ms to 60ms before the actual targets appeared either congruent or incongruent primes were displayed for 10ms. In the congruent case, the primes appeared at the same locations and had the

same color as the actual targets. In the incongruent case, the primes appeared at the same locations as the actual targets but with switched colors.

The movement trajectories of incongruently primed targets clearly showed a short motion in the wrong (i.e. primed) direction before this error was corrected and the trajectories approached the actual target. The extent of the motion in the wrong direction was significantly larger if the incongruent primes were shown longer before the targets appeared (longer stimulus onset asynchrony, SOA; see Figure 2 in T. Schmidt, 2002). The primes were replaced by the targets for at least approximately 200ms before movement onset. The results were explained by assuming that, as soon as the primes were visible, they contributed to the generation of a response until the target appeared, thus causing initially misguided movements in the incongruent case.

SURE_REACH accounts for these effects. Priming of different durations was simulated by pre-activating the activation maps for 1 to 5 time steps with a congruent or incongruent target posture, without actually executing the movement (targets: stretched arm, $\phi_{elbow, wrist} = 0$, with shoulder either $\phi_{shoulder} = -90^\circ$ or $\phi_{shoulder} = 90^\circ$, movements started from $\phi_{shoulder, elbow, wrist} = 0^\circ$). Then the primes were replaced by the targets and the movement was initialized. To measure the extent of movements in the wrong direction, the maximal Euclidean distances between the hand coordinates associated with the goal posture ($\pm 2.4, 0$) and the hand coordinates during each movement were assessed. The movements were executed by 10 controllers that were individually trained for 1,000,000 time steps. To enhance the effect, the inertia of the dynamic generation of the sensory-to-motor mapping was increased by adjusting the parameter that scales down network activity to $\beta = .48$ (Equation 5), and the movement gain was reduced to 6° (Equation 10).

Figure 15A shows the distance between the hand and the actual target during the movement. The chart displays the average performances of ten controllers in both movement directions. If the prime is congruent, the arm moves monotonously toward the target. However, if the prime is incongruent, an initial movement in the wrong direction is made. As in T. Schmidt’s (2002) experiment, the extent of this movement depends on the duration between prime onset and the onset of the actual target (SOA). Figure 15B plots the average maximal distance of the movements against different SOAs for the incongruent case. A one-way ANOVA revealed a significant main effect for the SOA, $F(4, 45) = 42.2$, $p < 0.01$.

To summarize, the population encoded space representation is suitable to model preparatory effects of movements, once a goal is selected. Moreover, priming effects that alter the encoded goal representations can be simulated due to the neural network dynamics.

Benefits of Encoding Kinematic Redundancy

One of the claimed advantages of storing multiple solutions for a single goal is enhanced behavioral flexibility. To test this flexibility, the posture goal representation is now further constrained to examine if SURE_REACH benefits from storing redundancy at the kinematic level. Also, it will be shown that, as observable in humans, the fi-

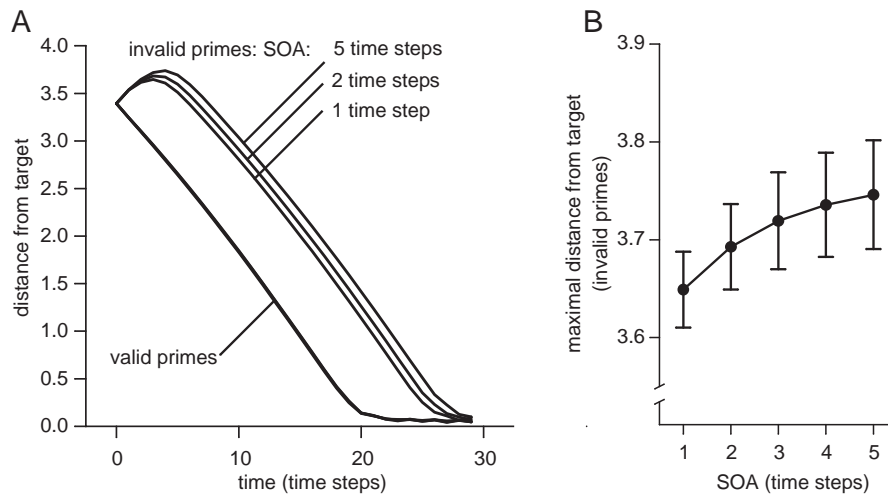


Figure 15. A) The average distance to the actual target of 10 different controllers after a congruent (lower lines) or an incongruent (top three lines) prime. The effect of the incongruent prime is altered by the duration between the onsets of the prime and the actual target (stimulus onset asynchrony, SOA). B) The maximal Euclidean distances from the target for movements with incongruent primes increases with the SOA. Error bars show standard deviations.

nal posture of a movement depends on the starting posture. Finally, the anticipatory selection of a final posture dependent on a consecutive goal is simulated.

Constrained Joint-Angle Space and Posture Variability

The posture memory activates sets of postures that all realize a desired hand position. If this redundancy is beneficial, the controller should exert the shortest possible path in joint angle space to reach a goal location. Less direct but still accurate movements should be triggered if additional constraints apply.

To test this, ten unconstrained individual controllers were trained with the procedure described above and each controller was tested in five different conditions. In the first one, there were no constraints except the targeted hand location. In the second and third conditions, besides the targeted hand location, the desired shoulder angle was set to 0° and 45° , respectively. In the fourth and the fifth conditions, besides a desired hand position, the desired elbow angle was set to 0° and 45° , respectively. During each test phase, 16 movements were made toward random goals, but only movements toward goals that could be theoretically reached with the given constraints were included in the evaluation. The additional constraints were imposed by inhibiting all neurons in posture memory output (\vec{p}_g) that did not satisfy the joint angle constraints.

Figure 16 shows the distribution of shoulder, elbow, and wrist angles when the goal was reached. The desired joint angle constraint is met selectively for each condition (with a standard deviation of $\pm 45^\circ$ for each constraint). Furthermore, the controller uses a broad variety of final joint angles in the unconstrained joints to reach the activated goal

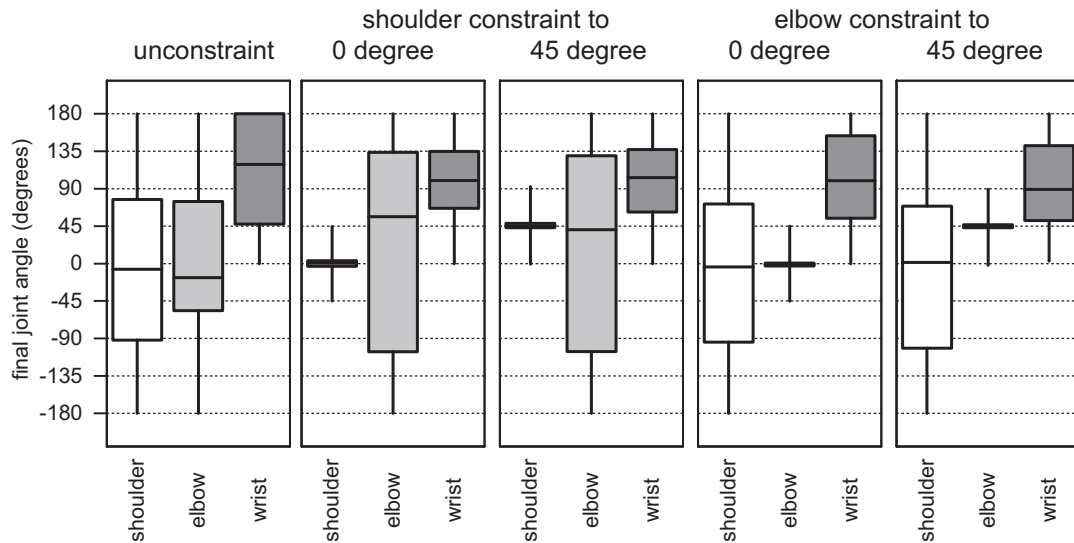


Figure 16. The chart shows box-plots for the distribution of the final joint angles in five constraint conditions. The different constraints for elbow and shoulder joint are met selectively, albeit not completely accurately, due to the broad tuning of the receptive fields in joint space (90°). A box ranges from the lower to the upper quartile, the lines in the boxes indicate medians, the whiskers show the range from the smallest to the largest observed joint angle.

location, indicating that the architecture effectively exploits redundancy.

To check if fulfilling constraints comes at a price, the hand position accuracy and movement duration for constrained and unconstrained movements were compared. Figure 17 shows the development of movement duration and error after various amounts of learning. Movement duration is only charted from 100,000 trials of learning because this is the first time that all controllers were able to finish all required movements. After 1,000,000 time steps, there is no difference in average hand position accuracy between constrained ($M = 4.77\%$ of workspace size, $SD = .835\%$) and unconstrained ($M = 4.56\%$, $SD = .791\%$) movements, $t(18) = .591$, $p > .05$. On the other hand, movement durations differed significantly. The movement duration is measured as the number of time steps used from the first change of the posture until the hand moves closer than 15% of the workspace size to the goal. After 1,000,000 time steps of learning, the average movement time of an unconstrained movement is $M = 6.44$ steps ($SD = 1.79$). Constrained movements are significantly slower, $M = 16.6$, $SD = 6.51$, $t(10.4) = 4.75$, $p < .01$.

In human hand movements, the final arm posture that places the hand at the target location is dependent on the starting posture (Cruse et al. 1993; Jaric, Corcos & Latash 1992; Soechting et al. 1995). This exploitation of kinematic redundancy seems to minimize movement costs (M. H. Fischer et al. 1997). To allow an analysis of start-posture dependency, each hand target in the previous evaluation was approached from two different random starting postures. For movements with the same hand target but different

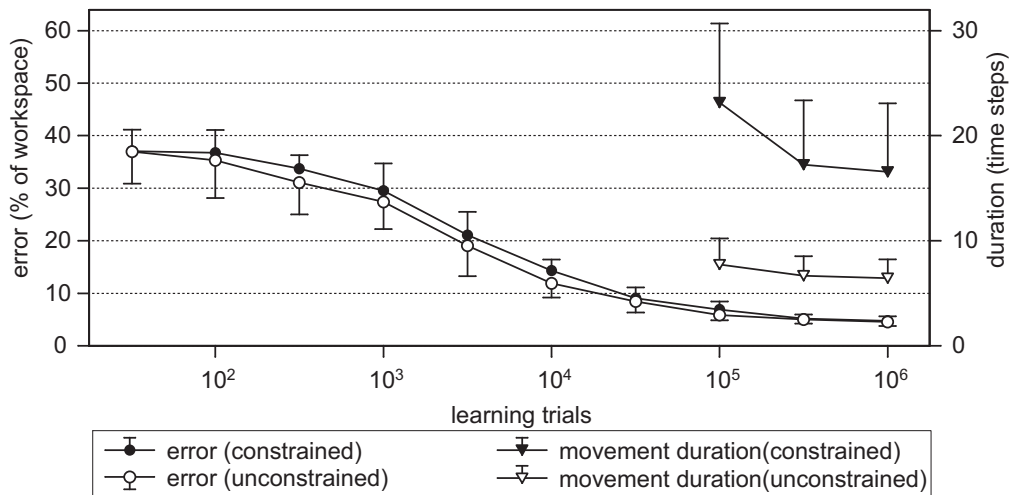


Figure 17. While the average hand position errors of constrained and unconstrained movements do not differ significantly, unconstrained movements are much faster than constrained movements. Error bars show standard deviations.

starting postures, the final postures of the movement were compared (2-norm). On average, the end-postures of unconstrained movements differed by 111° ($SD = 56.3^\circ$). For the constrained movements, the average posture difference was 70.9° ($SD = 49.2^\circ$). Constrained movements could still express a high amount of start-posture dependency, because some targets could be realized by moving one joint to one extreme or the other, and because not all constraints were exactly met. Both values differ significantly from zero and from each other: from zero for unconstrained movements, $t(9) = 6.22$, $p < 0.001$; from zero for constrained movements, $t(9) = 4.56$, $p < 0.001$; from each other (pairwise t-test), $t(9) = 3.32$, $p < 0.01$. In conclusion, this shows that the model generates short trajectories in unconstrained conditions but is also able to produce longer movements to match additional constraints.

Anticipatory Posture Adjustment

Most movements in every day life are part of a larger sequence. For example, grasping a cup is often followed by moving the cup to the mouth. Hence, in movement sequences, motor redundancy could be exploited to make the outcome of one movement a good starting point for the subsequent one. Indeed, data from humans that had to sequentially reach different hand targets revealed that the arm posture at an intermediate target location depends on the subsequent target (M. H. Fischer et al. 1997).

Computational models of motor learning and control that do not encode redundant solutions for the inverse kinematics problem are unlikely to account for this finding. In SURE_REACH, the redundant postures that are represented for each hand target can be weighted, dependent on their utility to reach the next goal. This was simulated⁶ by a two

⁶Different parameter settings, which are specified in the appendix, were used for this simulation.

step process involving both posture memory and motor controller. First, activation maps for moving to the second of two targets were generated for 25 time steps without actually moving the arm. Second, this activation was combined with a goal representation for the first target (\vec{p}_g) to generate a target representation (\vec{p}_g^*) for the first movement, that incorporates demands for the second movement.

$$\vec{p}_g^{*j} = \vec{p}_g^j \times (10^{-\alpha} + \max(a_{i,0 \leq i \leq n}^j)) \quad (13)$$

where p_g^{*j} are the components of \vec{p}_g^* and p_g^j are the components of \vec{p}_g , and α is a weighting parameter that determines the amount to which the compound target representation is influenced by the subsequent target. The larger α , the higher the influence of the second target. The a^j are the components of the activation maps \vec{a}_i , which indicate the closeness of the associated posture to the subsequent target, assuming that the i -th motor command is activated. Thus, the largest a_i indicates closeness, assuming that the optimal motor command is activated.

The anticipatory capabilities of the controller were tested with the ten controllers, which were independently trained for 1,000,000 time steps. Each controller had to perform 50 sets of movements. A set consisted of four movements to random locations *Via*, T_1 , and T_2 : (1) a movement from a starting posture *S* to a via target *Via* anticipating a subsequent target T_1 , (2) a subsequent movement to T_1 , (3) a movement from *S* to *Via* anticipating T_2 , and (4) the subsequent movement to T_2 . Thereby the goal representation for movements to the via target (1,3) were determined by Equation 13. Figure 18 shows example movements ($\alpha = 6.0$). Each of the movement sets was simulated with four different settings of α ($\alpha = 3, 4, 5, 6$) and a control setting, in which movements to the via target were carried out independently of the subsequent goal. The hand location at *S* and the targets T_1 , T_2 , and *Via* were separated by at least 20cm from each other.

For each controller and each of the four settings of α , the average difference between end-postures of movements to the same via location but with different anticipated subsequent targets (1,3) was computed (2-norm) to determine how α affects the dependency of a movement's end-posture on a subsequent task. In the control setting the postures at the via location did not depend on a subsequent target. Figure 19A shows that the increasing impact of the anticipated goal increases the posture difference at the via location as well. A one-way ANOVA revealed a significant effect of α , $F(3, 36) = 70.2, p < 0.001$.

To assess if this effect was accompanied by an increase in efficiency, I further analyzed two performance measures for movements (2) and (4): the joint angle transitions (2-norm) made during the movement and its duration⁷. The performance of movements in the anticipatory conditions is contrasted to those in the control condition by subtracting the former from the latter. Positive values indicate lower joint angle transitions and

Mainly, the arm geometry was adapted to resemble a human arm more closely, hand- and posture space encoding was more fine-grained, and only six motor commands were used.

⁷The movement duration was considered the time between the onset of a target and the number of time steps required to move to an area within 5cm of the target. To obtain valid results, movement sets were excluded from the computation of the movement duration if at least one movement didn't reach the 5cm criteria (6.0%).

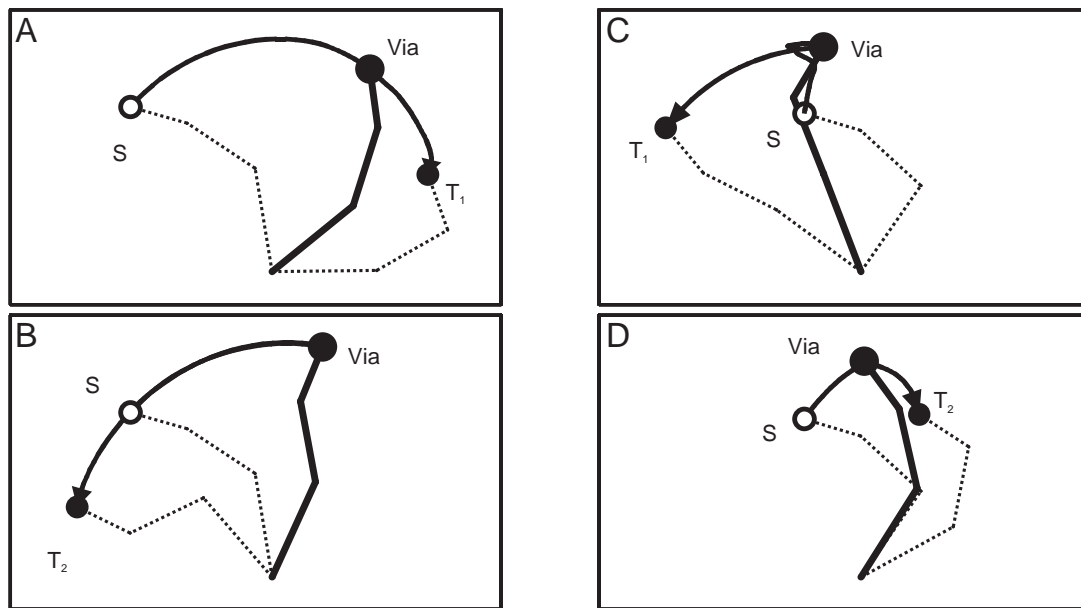


Figure 18. The end-postures of movements from identical start-postures (S) to identical targets (Via) can be adjusted so that good starting positions for movements to subsequent targets (T_1, T_2) are assumed.

faster movement times compared to control movements. Figures 19B and C show that the efficiency of the subsequent movements (2) and (4) increases if the goals of these movements are more strongly incorporated in the goal representations of the preceding movements. One-way ANOVAs revealed a significant impact of α on the remaining joint angle transition ($F[3, 36] = 108, p < 0.001$) and on movement duration ($F[3, 36] = 127, p < 0.001$).

The simulated experiments show that SURE_REACH exploits kinematic redundancy to incorporate demands of the subsequent task in its goal representation. By doing so, the subsequent movement can be carried out faster because it starts from an advantageous posture. The suitability of a posture to serve as starting posture for a movement to a particular hand target is provided by the sensorimotor grounded distance measures in the sensorimotor model. Similar behavior in humans has been found in reaching tasks (M. H. Fischer et al. 1997) but also in other domains like bimanual object manipulation (Weigelt, Kunde & Prinz 2006) or speech production (Dell, Chang & Griffin 1999). Additionally, the more complex movement preparation process is in line with experimental findings, which show an increase in preparation time for the initiation of the first movement of a sequence of aiming movements (Lavrysen et al. 2003). In conclusion, the availability of redundant postures provides the flexibility to align movements to the demands of future tasks.

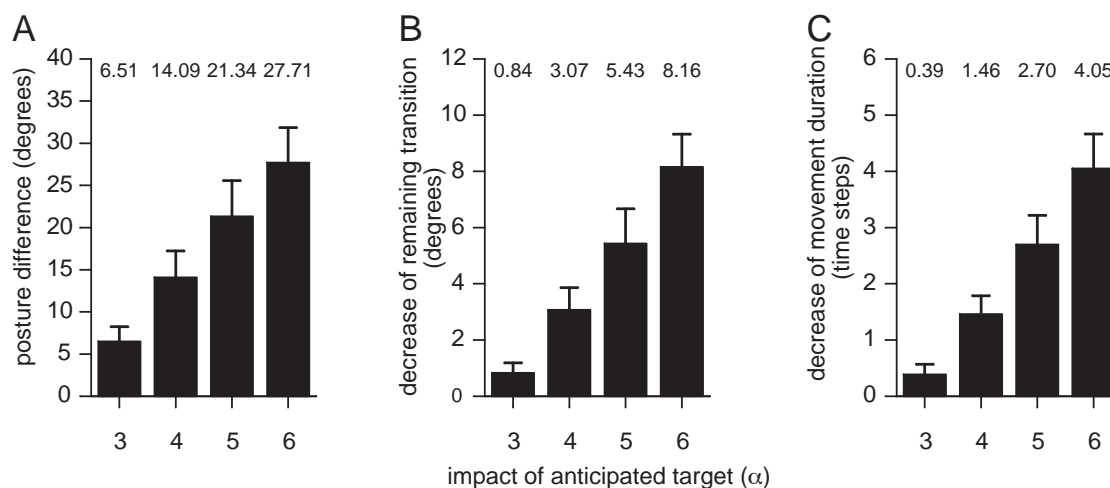


Figure 19. A) A goal representation in SURE_REACH can be adjusted to facilitate an anticipated subsequent movement. The more the goal of a subsequent movement is incorporated (α) into a goal representation, the more differ the end-postures of movements to the same location, given different subsequent targets. B) The differences between the required transition in posture space of movements that independently followed another movement and movements that have been anticipated in the preceding movement, increase with α . C) The same effect is observable for movement times. Error bars show standard deviations of the average performances of ten individually trained controllers.

Short Summary

The data confirms two important claims. First, the capability to store and process many possible arm postures for single goal coordinates enables SURE_REACH to flexibly incorporate new task-dependent constraints. For example, a goal that requires the hand to reach a certain position while maintaining a specific elbow angle can be easily pursued by the model—even if this task has never been explicitly trained. This capability may be useful if the hand rotation has to be aligned in order to grasp an object or press a button. Second, the significant difference in movement times reveals that the sensory-to-motor mapping induces more efficient movements if it is activated by a larger set of acceptable goal postures. Additionally, the anticipatory posture alignment found in human reaching movements could only be modeled because redundant goal postures are encoded. SURE_REACH exploits the redundancy provided by a larger goal set reaching the desired goal locations faster, because the posture within the goal set that is closest to the starting posture is approached. These findings parallel current behavioral data from humans (Cruse et al. 1993; M. H. Fischer et al. 1997; Jaric et al. 1992; Soechting et al. 1995).

Benefits of Encoding Sensorimotor Redundancy

In the last section, it was demonstrated that the representation of kinematic redundancy significantly enhances the flexibility of the controller. In this section, sensorimotor redundancy is exploited to adapt to obstacles, different movement costs, and immobilized joints.

Obstacle Avoidance

The previous section showed that the end-posture of a movement can be influenced by additional constraints. It was shown that the redundancy of postures associated to a hand position can be exploited to choose amongst alternative goal postures, considering additional constraints.

Another form of motor redundancy resolution lies in the trajectory generation by means of dynamic programming. If an obstacle blocks a certain area in posture space, the activity in the associated neural subspace may be inhibited consecutively. Thus, the activation diffusion by means of dynamic programming generates an alternative trajectory that circumvents the obstacle. However, obstacles are usually seen before movement onset and are represented in an extrinsic coordinate frame. In SURE_REACH, an extrinsically represented obstacle can be transformed into an obstacle representation in posture space by the posture memory (see right-hand side of Figure 4). Since the posture memory activates all postures that realize certain hand positions, feeding the complete extrinsic obstacle representation into the posture memory results in a representation of all those postures for which the hand would collide with the obstacle. For now, the posture memory only activates those postures that coincide with certain hand locations, so that it is only possible to avoid hand collisions but not collisions of other body parts. A more general posture memory that does not only map from certain hand locations to postures but that could determine arm postures that coincide with any other point on the arm could be used to extend obstacle avoidance to the entire arm.

To evaluate the current obstacle avoidance capability, 10 individual controllers were trained for 1,000,000 time steps and each of them was tested in two different tasks (see Figure 20). In each task, obstacles had to be avoided. Obstacles were defined in hand space. A hand space obstacle representation was generated by inhibiting neurons whose preferred values laid within the obstacle. This inhibition was passed through to posture space by means of the posture memory, consequently inhibiting those neurons in posture space that collide with the obstacle. All neurons in the activation maps (\vec{a}_i) were inhibited (set to 0.0) that had an activation level of at least .01 stemming from the inhibition passed through posture memory.

In the first task, the arm had to move the hand to the lowest position in the workspace (empty circle in Figures 20A, B) from an upward pointing posture. This goal can be pursued by two different movements. Either the arm can be rotated clockwise or counterclockwise. In two different settings, either the clockwise or counterclockwise movement was blocked by a square obstacle placed next to the shoulder (see appendix for details). Figures 20A, B show the movement trajectories of ten individual controllers.

Each controller avoided the obstacle by rotating the shoulder joint in the unblocked direction.

In the second task, the controllers had to move the arm to a stretched posture with a shoulder angle of -225° . The initial posture was a stretched arm, however with a shoulder angle of 225° . If no obstacle was in the way, the controller rotated only the shoulder joint (Figure 20C), following the shortest trajectory in joint angle space. In a second condition, a ceiling obstacle was introduced that would cause a collision if the hand location was too close to the ceiling. In this case, the controller bent the other joints in order to reduce the height of the extended arm and thus moved the hand beneath the obstacle (Figure 20D). After the obstacle was passed, the arm re-stretched the joints to reach the desired posture. The highest hand locations within the trajectories, $M = 1.40$, $SD = .0563$, were significantly lower when an obstacle was present, compared to unconstrained movements, $M = 2.39$, $SD = .00626$, pairwise t-test, $t(9) = -58.7$, $p < .01$. The figure shows that all trajectories slightly moved through areas of hand space that were part of the obstacle. These collisions result from the broad tuning of the posture space neurons' receptive fields and might be avoided by a larger set of neurons covering the hand- and posture spaces, or by a further reaching inhibition, which would trigger even stronger obstacle avoidance.

To summarize, obstacle avoidance can be incorporated into SURE_REACH by inhibiting neurons during movement preparation. This capability is due to the fact that the sensorimotor model of the motor controller implicitly stores all possible trajectories to a particular goal position and that the posture memory can be readily used to convert an obstacle representation from a hand-based to posture-based representation. The sensorimotor model triggers the most direct trajectory in posture space if no obstacle is present. If such a trajectory is blocked, a sensory-to-motor mapping can be generated that causes movements around the obstacle. As obstacle avoidance is based on the sparsely encoded posture representation, the current model can only account for the avoidance of larger obstacles and goals that are sufficiently distant from obstacles. However, a more fine-grained neural encoding might also enable the avoidance of smaller objects and closer goal and obstacle locations. Finally, note that obstacle avoidance in SURE_REACH is based on proprioceptive obstacle representations and proprioceptive feedback. Thus, the models capabilities might be enhanced by including visual feedback of the arm into motor control.

Reduced Joint Mobility

During life, the costs of moving certain limbs might suddenly change. For example, arthralgic patients suffer sever pain from moving a specific joint and hence have to achieve certain behavioral goals while trying to move one joint or the other as little as possible. If a joint is in a cast, some motions might not only be costly but suddenly impossible. Despite this impairment, patients can usually control their arm accurately and effectively with the remaining mobility. SURE_REACH can account for this flexibility limiting the extent to which some motor commands are applied by reducing the impact of those motor commands on the activation propagation process. This adjustment might be considered

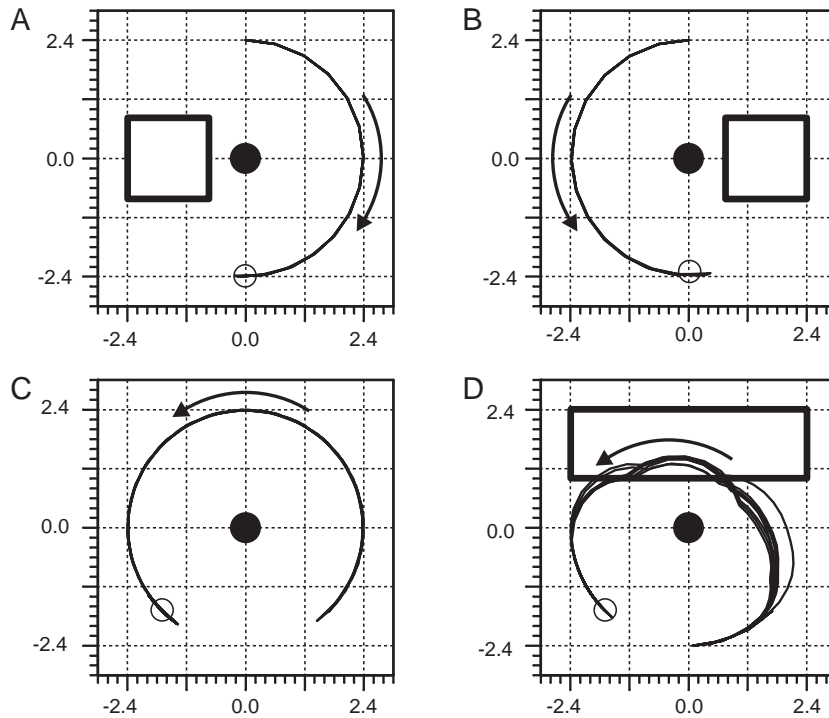


Figure 20. A, B) The hand paths of movements differ significantly, if an obstacle (rectangle) is presented to the left (A) or to the right (B). All movements bypass the obstacle successfully. C) The chart shows a movement of a stretched arm without any obstacles. D) The chart shows the same movement if a ceiling obstacle constrains possible hand locations. The empty circles indicate desired hand targets, the filled circles indicate the position of the shoulder joint.

a neural implementation of the adjustment of movement cost functions proposed by Rosenbaum et al. (1995). It was simulated by enhancing Equation 5 as follows:

$$\vec{a}_i^* \leftarrow v_i \max \left\{ \beta \left(\gamma \frac{\sum_{j \neq i} \vec{a}_j}{y-1} + (1-\gamma) \vec{a}_i \right), \vec{p}_g \right\}. \quad (14)$$

The difference between Equations 5 and 14 is that in the former all motor commands contribute equally to the activation propagation process, whereas in the latter, the contribution of each motor command is weighted according to a weighting coefficient, v_i . By adjusting v_i , the extent to which certain motor commands and hence certain joint motions are executed during a movement can be regulated. A larger coefficient v_i means relying more on the associated motor command.

An Arthralgic Joint

Figure 21 shows examples of movements that result from the modified activation propagation process. Whereas Figure 21A displays a normal unconstrained movement,

Figures 21B, C show movements in which the contribution of the motor commands affecting the elbow and wrist joint have been reduced to 1% of the contribution of the remaining motor commands. It is apparent that all joints are used in the unconstrained movement but that the motions of the elbow or wrist joint, respectively, are highly reduced in the other examples.

To check if this process also holds in the general case, ten controllers were trained individually for 1,000,000 steps. Each controller had to perform movements from 16 different start-postures to different goals provided in hand space. For each of the 16 start-goal pairs, one normal movement, and three movements with a reduced desired contribution of the shoulder, elbow, or wrist were executed. For normal movements, all motoneurons contributed equally to the activation propagation process. For movements in which a reduced motion of a specific joint was desired, the v_i s for motor commands associated to that joint were set to 1% of the v_i s of the remaining motor commands. For example, if the wrist joint should move as little as possible, v_i s were set to $v_0 = v_1 = 0.01$ and $v_2 = \dots = v_6 = 1.00$. The contribution of each joint to a movement was operationalized as the absolute difference between its initial angle and final angle. The contributions were averaged for each controller, each of the four movement conditions, and each joint. Start-goal pairs yielding movements that did not move as close as 15% of the workspace size to the goal within 160 time steps in at least one of the four conditions were removed from the evaluation (48.8%). Figure 21 and Table 3 summarize the results. It can be seen, that the contribution of each joint angle is reduced selectively for the desired joint. Pairwise t-tests confirm this finding (see Table 3).

Table 3: Transitions of restricted and unrestricted joints

joint	normal transition (SD)	constraint transition (SD)	$t(9)$	p
shoulder	69.1° (49.1°)	32.5° (28.2°)	15.7	< 0.01
elbow	66.5° (48.7°)	26.5° (25.2°)	17.2	< 0.01
wrist	60.2° (44.5°)	24.0° (22.5°)	16.4	< 0.01

Thereby, the final position error is only slightly larger for movements with reduced joint motions: for normal movements, $m = 4.00\%$ of workspace size, $SD = 0.427$; for constraint movements, $m = 4.67\%$, $SD = 0.443$; $t(9) = 5.216$, $p < 0.01$. Motion of the impaired joint is not reduced to zero, because many movements require a transition of all joints in order to move to the goal. Note that the reduction of the contribution of a motor command to the activation propagation process does not inhibit the motoneuron per se. Only when other motor commands can be used to fulfill the given goal, the inhibited motoneuron will not be excited.

A Broken Arm

Sometimes it might be necessary to move the arm without the capability of relying on the complete movement repertoire that was available during motor learning. This

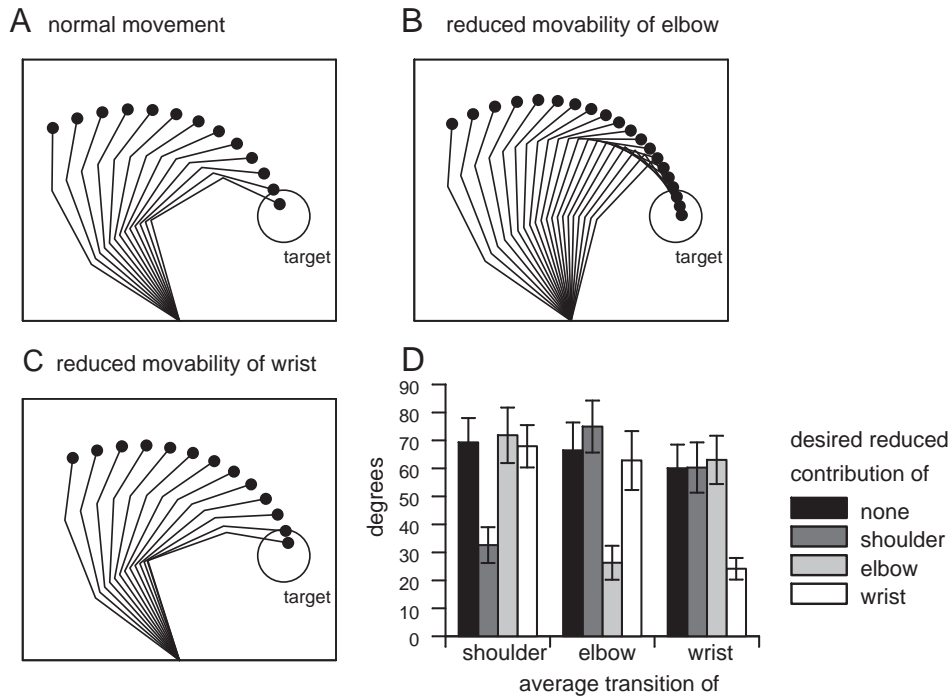


Figure 21. A) An example of an unconstrained movement. B, C) The charts show movements with reduced rotation of the elbow (B) and wrist (C) joint to the same target. D) A systematic evaluation reveals that the contribution of each joint to a movement (operationalized as the change in joint angle between movement onset and the final position) can be selectively reduced. Error bars show standard deviations.

may be the case if the arm is broken and in a cast. Usually used motor command sequences will then be fruitless if they rely on motor commands that now have no effect. Experiments show that constraining one limb hardly affects the capability for accurate movements and does not require exhaustive relearning (Robertson & Miall 1997).

To verify that the reduction of the contribution of not anymore useful motor commands to the activation process can be used to control an arm with a joint in a cast, 10 controllers were individually trained for 1,000,000 steps and had to perform reaching movements with either normal mobility or with the shoulder, elbow, or wrist joint angle set to $\phi_i = 0$, regardless of the motor commands that are executed. The v_i values for motor commands associated to joints in a cast were set to $v_{joint\ in\ cast} = 0$, whereas all others were set to $v_{free} = 1.0$. To compare the different conditions, the end-point error of all reaching movements with random goals that could be theoretically reached in all movement conditions (on average 12.6 per controller) were averaged and compared by pairwise T-tests. Movements with the shoulder or wrist in a cast were only somewhat less accurate but fixing the elbow did not affect average accuracy (see Table 4).

To summarize, the activation propagation process can be modified by weighting the

Table 4: Accuracy of movements with a joint in a cast

condition	end-point error (SD)	compared to normal condition	
		$t(9)$	p
normal	3.54% (0.659%)		
shoulder in a cast	8.08% (2.40%)	5.72	< 0.01
elbow in a cast	3.24% (0.724%)	-1.118	> 0.05
wrist in a cast	6.70% (0.861%)	11.22	< 0.01

contribution of sensorimotor contingencies associated to certain joint motions. On the one hand, joint movements that cause pain or are otherwise costly can be replaced by movements of the other joints (as long as this is possible). On the other hand, the modification can be used to control the arm if certain joint movements are suddenly impossible. In either case, it is not necessary to relearn the sensorimotor mapping of the arm.

Summary of Results

The evaluations of this implementation of SURE_REACH confirmed several interesting model features. The overall reliability and accuracy of the model was high despite the rather sparse representation of postures and end-point coordinates. Model learning was stable under several different learning and parameter conditions.

More important, SURE_REACH accounts for empirical findings related to human motor learning and control. First, during human motor learning, accuracy increases and movement times and reaction times decrease. The model does not only account for increasingly accurate movements, but also exhibits that training decreases movement preparation and movement execution times. Second, representing goals by population codes is not only in line with neurophysiological data but also with psychological findings and theories (e.g. (Erlhagen & Schöner 2002; Flash & Sejnowski 2001)). In contrast to many other models of motor learning and control, which can only process a single target posture or hand location, SURE_REACH can account for more complex target representations, such as the target activation of multiple alternative postures. Likewise, humans and primates are able to partially prepare movements towards subsets of movement directions or distances. Third, a priming experiment was replicated by means of the space representation and network dynamics. Fourth, it was demonstrated that representing kinematic redundancy enables the simulation of some features of the flexibility of human motor control. In humans, the final arm posture of a movement depends on the starting posture. Also in SURE_REACH, movements to the same hand position differ, depending on the starting posture. Additionally, anticipatory effects of posture selection were modeled. This was only possible because multiple end-postures were explicitly associated to each hand position. To my knowledge, this effect has not been simulated before. Finally, due to the encoded sensorimotor redundancy in the sensorimotor model, SURE_REACH is able to adapt quickly to novel trajectory constraints. The architecture is able to avoid obstacles and recruit alternative motor commands,

given that previously optimal motor commands are suddenly costly or even impossible. Together, the evaluations show that the encoding of motor redundancy on many levels enables the simulation of flexible human motor behavior that cannot be accounted for by models that strive to resolve redundancy before learning. Moreover, the online generation of the sensory-to-motor mapping enables to model the dynamics of movement preparation.

Visually Guided Reaching

The computational model presented so far uses a visually encoded goal representation to prepare a movement. The control of the movement itself is exclusively guided by proprioceptive feedback. However, it has long been known that human reaching relies extensively on vision of the hand and the target throughout a movement (Woodworth 1899). In this section, the model will be extended to enable visually guided movements. This extension enhances the model's ecological validity and additionally, it sketches how a hierarchy of control loops coordinates movements.

The control of accurate human reaching movements relies on the availability of visual feedback. Movement accuracy decreases considerably if vision is occluded (Spijkers & Spellerberg 1995; Ma-Wyatt & McKee 2007). Extensive training under visual control even increases the impact of withdrawing visual feedback (Khan, Franks & Goodman 1998; Proteau, Marteniuk, Girouard & Dugas 1987). Likewise, errors in arm positioning tasks result mostly from inaccurate vision (van den Dobbelen, Brenner & Smeets 2001). Furthermore, even if visual feedback is not available, visual cortical areas are active during a movements (Darling, Seitz, Peltier & Butler 2007).

However, the computational model described so far only accounts in a limited way for the exploitation of visual feedback. Even if a hand target may be encoded in vision-based hand space, this target representation is converted into a postural target representation before it is used to control the movement. Hence, the generation of motor commands solely depends on a posture space representation of the goal and the proprioceptively perceived arm posture. The model accounts already for the ability to update target locations during movements because the posture goal representation may be updated during an ongoing movement based on a novel visual goal representation. However, there is no mechanism that is able to influence motor command generation based on the *difference* between the actual hand position and the target location. Thus, movement errors cannot be corrected based on visual information. This might be necessary to level out inaccuracies of the posture memory, which might arise due to insufficient learning experience, limited spatial resolution, or sudden kinematic distortions. In the following section, the model is enhanced by a mechanism that enables movement adjustment based on visual feedback.

In principle, two different approaches could account for corrective movements based on visual error perception. First, the perceived error could directly be converted into motor command space. Neural network models that implement this approach acquire a mapping from the direction of the vector between hand and target, and motor commands (e.g. Baraduc et al. 1999, 2001; Bullock et al. 1993; Srinivasa & Grossberg 2007).

According to these models, all movements are specified at some point in terms of a vision-based error signal⁸. Also, these models are in line with experiments that suggest that movements are planned in terms of hand displacements (e.g. Krakauer et al. 2000; Wang & Sainburg 2005). The benefits and limitations of these models have already been discussed in the introduction. In short, these models do not represent postural redundancy and thus cannot account for most of human movement flexibility.

Second, the perceived error could indirectly elicit changes in the motor commands by altering the target representations. I took this approach to include visual feedback in the SURE_REACH model. Here, the perceived error causes an adjustment of the visual goal representation. The altered visual goal representation in turn causes an adjustment in the postural goal representation, subsequently an adjustment of the sensory-to-motor mapping, and finally causes a change in the motor commands. An indirect motor command adjustment is computational advantageous. First, altering the visual target representation is unproblematic because both the perceived error and the consequent adjustment have to be made in the same sensory modality. In contrast, a direct conversion of visually represented errors into changes in motor commands would require the demanding mapping from a low dimensional visual space into a high dimensional motor command space. Second, altering the visual target representation builds on the lower levels of the motor control system, conserving redundant postural target representations. Thus, the model is inline with the notion that motor control is hierarchically organized (Cisek 2006; Jordan & Wolpert 1999; Todorov 2004; Wolpert et al. 2003). Finally, this approach complies with behavioral and neurophysiological studies that favor end-point based movement control (Aflalo & Graziano 2006; van den Dobbelen et al. 2001).

Visual Feedback Controller

The following section describes the neural structure that enables visually guided reaching in the SURE_REACH model. Figure 22 outlines the mechanism. Once a (visual) hand target is provided, two different target representations are formed. A *static* target representation encodes the actual goal of the movement. It is used to compute an error signal by contrasting it to the representation of the perceived hand position (comparator). The resulting error signal controls the *dynamic* target representation. The dynamic target representation serves as input to the lower levels of the control hierarchy. Initially, it may encode the same information as the static target representation but it may change as soon as a discrepancy between actual target and hand position is perceived. Thus, the dynamic target representation can be thought of as encoding a *virtual* target that is not necessarily identical to the desired hand target but helps the underlying control structures to acquire the actual target more accurately. It is continuously forwarded to the posture memory whose output in turn causes a continuous adjustment of the sensory-to-motor mapping by the motor controller. If an error is perceived, the virtual target is shifted in the direction opposite to that of the error. For example, if the hand is to the left of the actual target, the virtual target is shifted slowly

⁸The vision based error signal may also be derived from internal feedback (Bullock et al. 1993; Desmurget & Grafton 2000).

to the right, resulting in an arm movement to the right. A similar mechanism has been used in a version of the posture based motion planning model (Rosenbaum et al. 1995). The next section describes the realization of target representations, comparator, and the control mechanisms for the dynamic target representation by neural networks.

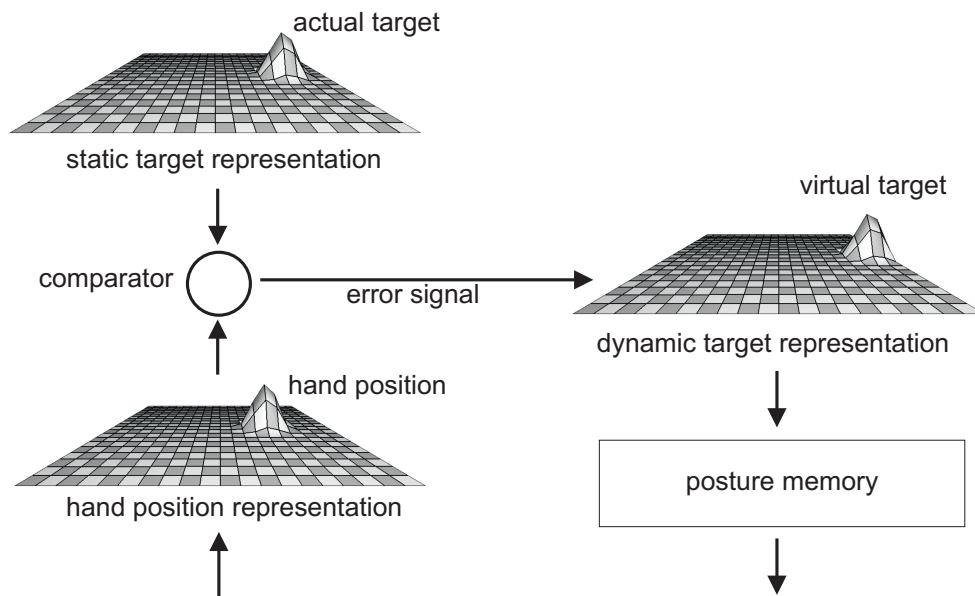


Figure 22. The comparator computes an error signal from the difference between the actual target, encoded by the static target representation, and the hand position. The error signal induces shifts of the virtual target, encoded in the dynamic target representation. The dynamic target representation is detached from the actual target perception during the movement and serves as input to the posture memory and subsequent networks.

Adaptive Neural Fields

The dynamic target representation has to meet three requirements. First, the virtual target encoded in the dynamic target representation has to be detached from the actual target representation to enable a shift in the virtual target. Second, despite this detachment, the dynamic target representation has to be able to maintain a neural activity distribution that accurately encodes the virtual target throughout the movement. Third, the neural activity in the dynamic target representation has to be controllable in order to induce corrective movements.

A neural network structure that fulfills the first two criteria is the lateral inhibition neural field (Amari 1977; Erlhagen & Schöner 2002). Such a neural field consists of many neurons, each of which is associated with certain spatial coordinates, similar to

the hand and posture space encodings. These coordinates may, for example, be the hand position, to which the neuron responds maximally. Distances between two neurons are calculated based on these coordinates. Adjacent neurons are connected by excitatory connections and distant neurons are connected by inhibitory connections. Once some activity is injected into such a network, it is able to maintain a focused “hill” of neural activity (see hills in Figure 22). The activity hill can be maintained without further activity injection due to the positive feedback loop formed by the excitatory connections between adjacent neurons. Additionally, the inhibitory connections between distant neurons keep the activity in the neural network local and focused.

The lateral inhibition neural field is suitable to stabilize and maintain activity but it cannot be used to control shifts of the activity hill. Hence, additional adaptive neuronal connections are introduced. They can impose a bias in the lateral connectivity, pushing the activity hill in a specific direction (Stringer et al. 2007, 2003). Thus, the neural field is extended to an *adaptive neural field*. Figure 23 sketches a simplified one-dimensional adaptive neural field. The adaptive neural connections are shaped during motor learning together with the posture memory and the sensorimotor model. In the following implementation, the adaptive connections are organized in several layers, comparable to the different neural networks encoding the sensorimotor model. The synaptic weights of the layers differ from each other. Thus, the activity distribution can be controlled by modulating the contribution of each layer to the overall activity propagation. In the model, the differences between layers develop because layers are only trained when movements in specific directions are executed. Thus, activating specific layers of the adaptive neural connectivity enables the imposition of a specific bias on the neural field activity propagation. During corrective movements, a population coded error representation is computed from the static target representation and the perceived hand location, which in turn biases the adaptive connections. In the current architecture, the error representation is derived solely from the perceived hand position and the neural network structure that encodes the adaptive connections.

The adaptive neural field is a hand space representation modeled as a vector \vec{h}_{dyn} consisting of $h = 21 \times 21 = 441$ neurons. The adaptive connections are realized by $c = 8$ neural networks with $h = 441$ input neurons and $h = 441$ output neurons, which are connected by $h \times h$ weight matrices W_{ACj} . Finally, a comparator mechanism computes an error signal \vec{e} with $c = 8$ components.

In the extended model, the hand target representation \vec{h}_g (also consisting of $h = 21 \times 21 = 441$ neurons), as used in Equation 3, is not directly forwarded to the posture memory but serves as the static target representation. In the following, I refer to \vec{h}_g as \vec{h}_{stat} to distinguish it from the dynamic target representation \vec{h}_{dyn} . When a novel target is presented, \vec{h}_{dyn} is set to \vec{h}_{stat} . Then, static and dynamic target representations are detached. The static target representation is conserved during the movement. The dynamic target representation is updated each time step according to the following

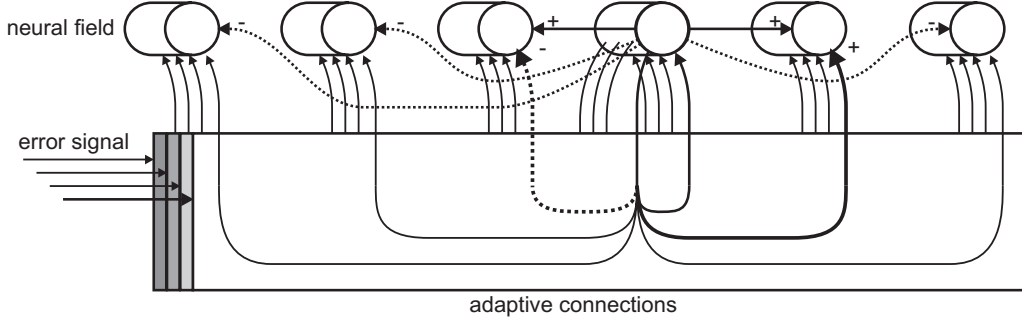


Figure 23. The dynamic target representation is realized by an adaptive neural field. It contains lateral excitatory (black lines) and inhibitory (dotted lines) connections, which maintain an accurate target representation. Several layers of adaptive connections, which are modulated by the error signal, induce shifts in the neural activity. In the figure, the drawn layer of adaptive connections induces an activity shift to the right. Lateral connections are only drawn for one neuron for clarity, but in the network all neurons are interconnected.

equation:

$$\Delta h_{dyn}^i = -\alpha_{dyn} h_{dyn}^i - \beta_{dyn} \sum_{j=0}^{h-1} f(h_{dyn}^j) + \gamma_{dyn} \sum_{j=0}^{h-1} f(h_{dyn}^j) d(i, j) + \delta_{dyn} c^i(\vec{h}_{dyn}, \vec{e}) \quad (15)$$

where Δh_{dyn}^i is the change in the i -th component of \vec{h}_{dyn} , α_{dyn} scales the reduction of the activity of the neuron proportional to its activity, β_{dyn} weighs a general inhibitory term that scales down overall network activity, γ_{dyn} weighs the influence of the lateral inhibitory and excitatory connections, and δ_{dyn} weighs the influence of the adaptive connections. Table 5 lists the values of the parameters. The activity of a single neuron is restricted to the range from 0.0 to 2.0. The steep nonlinear sigmoidal function $f(t)$ bounds the propagated neural network activity:

$$f(t) = \frac{1}{1 + e^{-20(t-0.8)}} \quad (16)$$

The weight of lateral connections is determined by the function $d(i, j)$

$$d'(i, j) = \frac{\pi}{d_{inh}} \sqrt{(i_x - j_x)^2 + (i_y - j_y)^2} \quad (17)$$

$$d(i, j) = \begin{cases} \cos(d'(i, j)) & d'(i, j) < \pi \\ -1 & d'(i, j) \geq \pi \end{cases} \quad (18)$$

where $d_{inh} = 0.36$ is the smallest distance between preferred values of neurons that are connected by maximally inhibitory lateral connections and $i_x, i_y, j_x,$ and j_y are the preferred values of the i -th and j -th neuron, respectively. This results in some self-excitation for each neuron, low impact on adjacent neurons and inhibitory impact on distant neurons.

Finally, the adaptive connections impose a bias on the propagation of neural activity dependent on the error signal. In Equation 15, function $c^i(\vec{h}_{dyn}, \vec{e})$ introduces the adaptive connectivity, which returns the i -th component of the vectorial function

$$c(\vec{h}_{dyn}, \vec{e}) = \sum_{j=0}^{c-1} e^j (W_{ACj} \times f_v(\vec{h}_{dyn})) \quad (19)$$

where $f_v(\vec{t})$ is a vectorial function that computes $f(t)$ for each component, e^j is the j -th component of the error signal, and W_{ACj} is the j -th weight matrix for the adaptive connections. In sum, this function determines the connectivity between all neurons of the neural field by weighing the impact of the different layers of adaptive connectivity with the error signal. This weighted adaptive connectivity propagates the activity of the neural field with the desired bias. Figure 24 shows two examples of the different propagation biases of each of the eight layers of adaptive connections. White areas

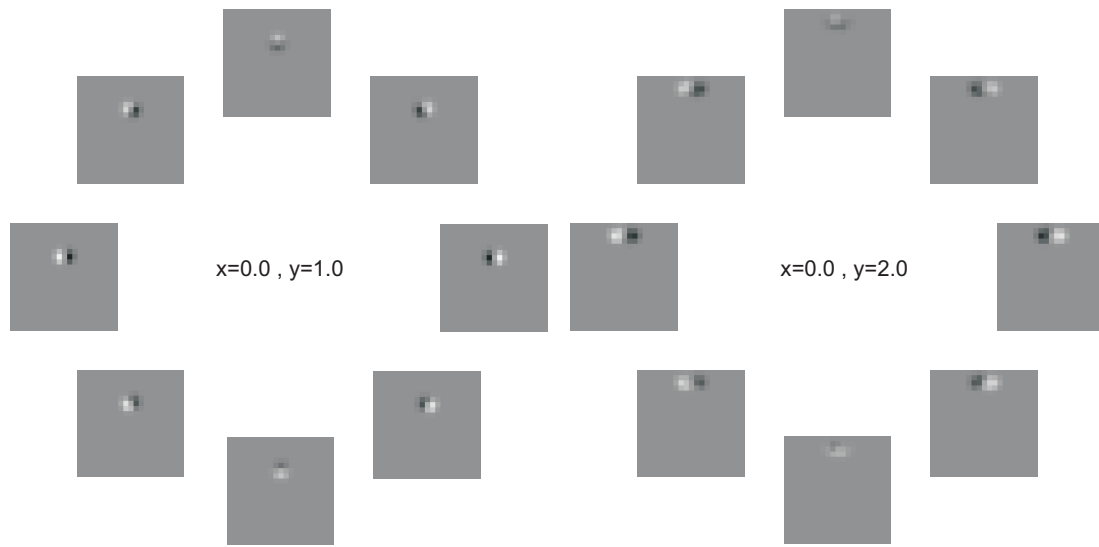


Figure 24. The figures show neural activity propagation in the eight different adaptive connections networks for activity injection at two different positions. Each network propagates activity in a specific direction.

indicate excitatory connections whereas dark areas indicate inhibitory connections. The examples show, that each layer of adaptive connections locally propagates activity in a specific direction.

Comparator and Error Signal

In order to bias activity propagation in the dynamic target representation, an error signal has to be computed from the perceived hand location and the actual target location. Many models do not consider how such an error signal may be computed (e.g.

Table 5: Parameters

parameter	value	description
α_{dyn}	0.2	activity leak
β_{dyn}	0.05	weight of general inhibitory term
γ_{dyn}	0.1	weight of lateral connections
δ_{dyn}	3	weight of adaptive connections
d_{inh}	0.36	minimal distance for full lateral inhibition
ϵ_{AC}	0.001	learn rate for adaptive connections

Baraduc et al. 2001; Bullock et al. 1993; Srinivasa & Grossberg 2007) but the error signal is provided from an external source. In contrast, the mechanism provided here explores the possibility to deduce an *internal* error signal solely from a vision-based representation of hand and target, building upon already encoded neural networks and representations.

The comparator processes population codes representing the static hand target and the actual hand representation. Each layer of adaptive connections is used to predict the activity distribution that would arise from applying the specific layers. The overlaps of these predictions with the static target representation are used to determine the error signal. This is modeled by the equation

$$e^i = f_e((W_{ACi} \times \vec{h}_{act})^T \vec{h}_{stat}) \quad (20)$$

in which e^i is the i -th component of the error signal \vec{e} , W_{ACi} is the weight matrix of the i -th layer of adaptive connections, \vec{h}_{act} is a $h = 441$ component vector that encodes the actual hand location similar to the target representations \vec{h}_{stat} and \vec{h}_{dyn} , and \vec{h}_{stat} is the static target representation. The sigmoidal function

$$f_e(t) = \frac{1}{1 + e^{-3t}} \quad (21)$$

limits the maximal activities in the error signal while pronouncing smaller differences.

The logic behind this process is closely related to the mechanisms for motor command generation in the motor controller. Both rely on independent neural networks that are used to establish the degree to which certain efferent signals, motor commands in the motor controller or in this case the error signal, are useful to pursue a given target.

Learning

Before the adaptive connections can effectively control the neural activity of the dynamic target representation they are shaped during motor learning. As in all neural networks of the SURE_REACH model, the weights of the adaptive connections are acquired unsupervisedly during random movements. Thereby, it depends on the direction of the ongoing movement if connections are strengthened or weakened in a specific layer. This direction dependency is later used to bias activity propagation in the dynamic hand representation. Note, that synaptic connections may get inhibitory (negative weights).

During learning, each layer i of adaptive connections is associated to a specific angle α_i . In the simulation these angles are set to $\alpha_0 = 0^\circ, \alpha_1 = 45^\circ, \alpha_2 = 90^\circ, \dots, \alpha_7 = 315^\circ$. In each time step t , the i -th layer of adaptive connections is updated according to the following equation

$$W_{ACi}(t) = W_{ACi}(t-1) + \epsilon_{AC} \vec{h}_{act}(t) \vec{h}_{act}^T(t-1) \frac{d_x \cos(\alpha_i) + d_y \sin(\alpha_i)}{\sqrt{d_x^2 + d_y^2}} \quad (22)$$

where $W_{ACi}(t)$ is the weight matrix of the i -th layer of adaptive connections at time step t , $\epsilon_{AC} = 0.001$ is a learning rate, $\vec{h}_{act}(t)$ is a vector that encodes the current hand location, $\vec{h}_{act}^T(t-1)$ is the transpose of a vector that encodes the hand location at time step $t-1$, and the fraction term causes a strengthening or weakening of the lateral connection weights dependent of the direction of the actual hand movement (d_x, d_y) in euclidean space. Note that during learning some explicit knowledge about movement direction is used. This information could be retrieved, for example, from the motor commands that are used to track the hand with saccades. However, this information only enables the layers of adaptive connections to develop distinct activation propagation biases and is not used during movement execution.

Evaluation

To test the advantage that visually controlled movements yields over exclusively proprioceptively controlled movements, ten individual neural networks, including sensorimotor model, posture memory, and adaptive connections were independently trained for 1,000,000 time steps. Each controller had to exert 16 movements from randomly chosen start-postures to randomly chosen hand targets in two conditions. First, in the *visually guided* condition, the SURE_REACH model and the described visual feedback neural network were used. Second, in the *proprioception only* condition, the SURE_REACH model without the visual feedback neural network was used. Each movement was executed for 320 time steps. The gain g , as used in Equation 10, was set to $g = 2.86^\circ (0.05rad)^9$. Figure 25 details the trajectory of the hand and the virtual target of an exemplar movement. It can be seen that a shift of the virtual target enables the controller to compensate for an initial overshoot of the movement. Figure 26 compares the trajectories of hand-target distances for three visually guided and three proprioceptively controlled movements. All three charts show a reduction of the error for the visually guided movements, which sets in as soon as the hand moves closer to the target.

For each controller and both modes of control, the distances from the hand to the target at the end of the movement were averaged. Average error of visually guided movements was only 2.50% of workspace size ($SD = 0.965\%$) whereas the average error of proprioceptively controlled movements was 3.82% ($SD = 0.676\%$). Pairwise t-tests confirm the significance of the decrease, $t(9) = 8.11, p < 0.01$.

⁹During learning, the gain was doubled. A higher gain was used during learning to ensure a complete exploration of the arm's workspace whereas a smaller gain was used during testing to enhance overall precision of both proprioceptive and visually guided movements.

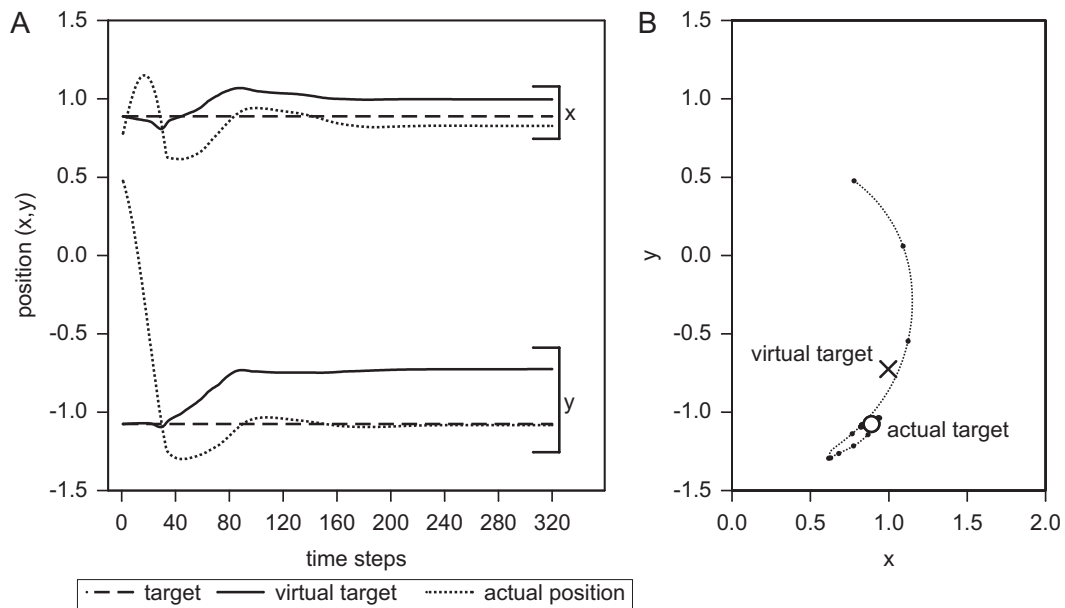


Figure 25. A) The left chart shows the x- (top three lines) and y-coordinates (lower three lines) of the actual target (dashed), the virtual target (black), and the hand location (dotted) of an exemplar movement. B) The right chart displays the hand trajectory, the actual target (circle), and the final position of the virtual target (cross). The virtual target shifts considerably to compensate for the initial overshoot. The charted location of the virtual target is the mean of the preferred values of the dynamic target representation weighted by their activations.

Discussion

The previous section introduced a neural network for incorporating visual feedback in SURE_REACH's control process. The mechanism is able to detect discrepancies between the (static) visual goal representation and a similar representation of the hand. The discrepancy influences motor command generation indirectly by altering intermediate goal representations. In this case, a dynamic hand target representation, which is processed by the posture memory, is adjusted. A systematic evaluation reveals that additional visual guidance reduces the final error on average by 35%.

A qualitative look on movement trajectories reveals that the hand initially approaches the target rather fast and then homes into the target with a slower, more precise movement. This compares to human visually guided arm movements, which are also comprised of an initial fast approach and then slower corrective movements (Elliott, Helsen & Chua 2001; Woodworth 1899). In the model, error correction movements are clearly distinct from the initial approaching movement because only the former are based on, and limited by, the adjustment of the dynamic target representation.

The dynamic target representation, which is used for motor command generation, is

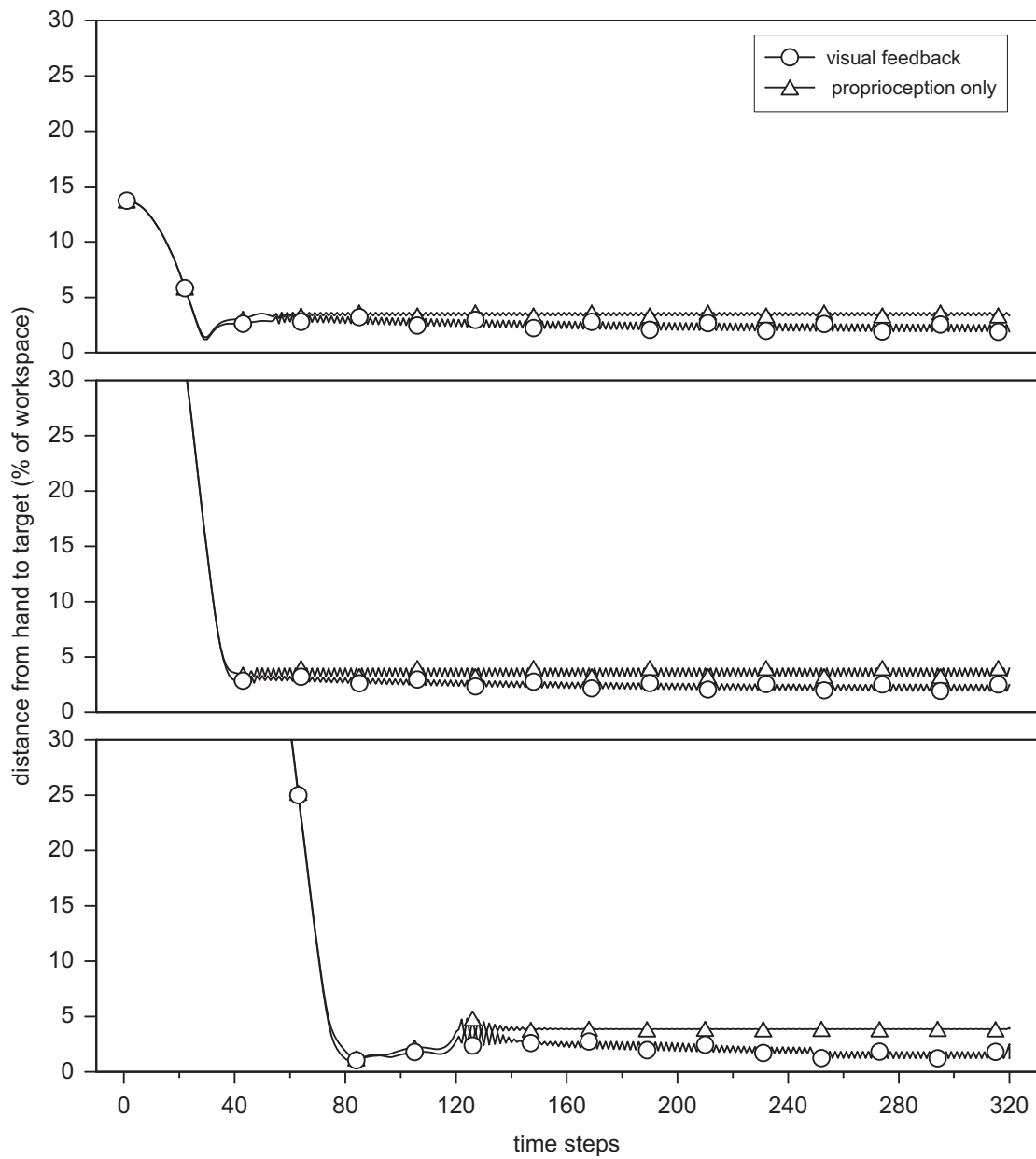


Figure 26. The charts show trajectories of the distance between hand and target for three exemplar movements under visual (circles) and solely proprioceptive (triangles) control. Initially, both control processes generate nearly identical trajectories. As the hand moves closer to the target, proprioceptively controlled movements come to rest at some distance to the target whereas visually guided movements slowly approach the target.

detached from visual representations of the target. Hence, displacements of the target during the movement cannot be directly incorporated in the dynamic target representation from which the sensory-to-motor mapping is ultimately generated. Nevertheless, the proposed architecture can still react to target displacements in two different ways. If a significant change in the target location is detected, the dynamic target representation could be reset and a movement could be prepared independent from the preceding movement plan. In this case, the corrections induced so far would get lost, but it seems unlikely that they would be appropriate in a completely different area of the workspace anyhow. If a smaller target change is detected, the adjustments could be made indirectly by shifting the target representation based on an error signal. Indeed, experiments show that prepared movements are reprogrammed by continuous target shifts (van Sonderen & Dernier van der Gon 1991; van Sonderen, Denier van der Gon & Gielen 1988).

Finally, the neural network is able to compute an error signal based only on visually encoded hand and target representations. Thus, given that sufficiently distinct biases in the layers of adaptive connections have developed, the architecture does not require an externally provided error signal to correct smaller deficiencies of executed movements. The next sections discuss specific properties of the neural network implementation of the visual feedback controller and relates it to hierarchical motor control.

Discussion of the Neural Network Implementation

The generation of the error signal is partially based on the actual hand location. However, the perception of the hand not yet having reached the target does not imply that the hand goal representation requires an adjustment. It is well possible that, after some time, the hand acquires the target perfectly. The mechanism outlined above will almost certainly adjust the dynamic goal representation as the hand approaches the target because it cannot distinguish between the meaningless discrepancies that persist during the execution of the movement and the meaningful discrepancies that arise due to an inappropriate target representation. However, adjustments are limited to the final phase of the movement because the comparatively slow movements during the learning phase result in a highly local neural network connectivity (see Figure 24). Due to this local connectivity of the adaptive connections an error signal that imposes a clear bias on the activity of the dynamic target representation is only generated as the hand approximates the target. On the other hand, this also implies that large discrepancies between hand and target cannot be corrected. Anyhow, it is questionable if the proposed mechanism can compensate for very large disturbances in the posture memory due to the non-linearity of the arm kinematics. In this case, the movement plan should be completely re-prepared, including a reset of the dynamic target representation. In future work, the error signal might not depend on the actual hand location but on a prediction of the final hand location to avoid adjusting motor commands based on a not yet finished movement (Desmurget & Grafton 2000).

Finally, even though the proposed neural network reduces the average error of movements considerably, visual guidance does not eliminate the error completely in all movements. Unsystematic experiments with an analytic implementation of the described

mechanism showed that the end-point error can be almost reduced to zero. Additionally, a similar (analytic) mechanism was successfully applied elsewhere (Rosenbaum et al. 1995). The inaccuracy of the neural network implementation might be mainly caused by the low number of neurons covering visual space. First, a small number of neurons implies that hand and target location are encoded with a low resolution. Thus, also deviations are harder to detect and to process. Second, the small number of neurons also reduces the stability of the neural field. This could result in a deterioration of the dynamic target representation, for example, caused by a total cessation of neural activity or by over-activation of larger populations of neurons. In other neural network models, dynamic neural fields are usually implemented with 10 times as many neurons for each spatial dimension (Erlhagen & Schöner 2002; Stringer et al. 2003). However, the memory requirements for the adaptive connections currently prohibit larger dynamic fields. In future work, high-resolution adaptive neural fields could be implemented by only sparsely interconnecting the neurons of the neural field thus combining high spatial resolution and compact neural networks (see Baraduc et al. 2001 for neural fields with sparse inputs).

Outlook on Hierarchical Motor Control

Motor control is generally understood as a hierarchy or cascade of nested control processes (Cisek 2006; Haruno et al. 2003; J. Hoffmann, Butz, Herbolt, Kiesel & Lenhard 2007; Jordan & Wolpert 1999; Mussa-Ivaldi 1999; Mussa-Ivaldi & Bizzi 2000; Powers 1973; Todorov 2004; Wolpert et al. 2003). In such a hierarchy, a rather abstract goal — such as wanting to have the hand at a certain location — is decoded into more concrete representations, for example in proprioceptive terms, until motor commands are finally generated. Thereby, intermediate representations are not only the byproduct of a stepwise transformation process but are also actively controlled to enhance movement accuracy and stability.

SURE_REACH and the presented visual feedback neural networks fit well into this framework. They form two nested control loops (Figure 27). The lower level loop, formed by the motor controller, controls the proprioceptive arm posture. It strives to match actual and desired posture. This loop is enclosed by the visual feedback neural network, which minimizes the distance between the exteroceptively perceived hand and target location. It does so indirectly by adjusting the posture representation of the motor controller via an intermediate hand representation and the kinematic mapping encoded in the posture memory. Thus the model differs from approaches that claim that rather higher level goals, for example, visually encoded errors, are directly relayed to motor commands or movement based representations (Baraduc et al. 1999; Bullock et al. 1993; Srinivasa & Grossberg 2007).

Conclusion

In conclusion, the described neural network incorporates visually guided reaching in the SURE_REACH framework. A vision-based representations of the hand and the target are used to adjust a virtual goal representation that serves as input to lower

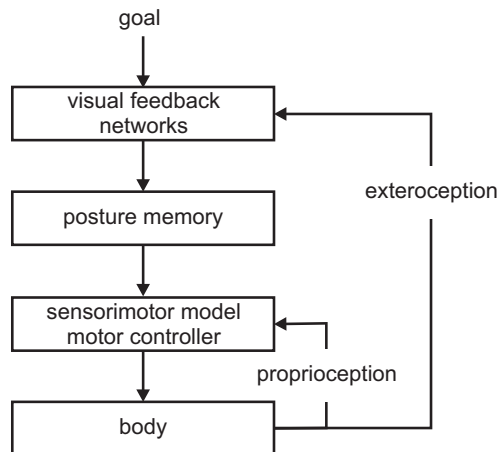


Figure 27. The visual feedback networks, posture memory, sensorimotor model, and motor controller form a cascade of control loops. The visual feedback processing neural networks control the input for the lower level, proprioceptive controller.

levels of the control hierarchy. Thus, the discrepancy between hand and target location indirectly induces corrective movements that enhance movement accuracy. Even if the implementation of the neural network could still be enhanced, it already realizes hierarchical motor control.

Discussion

In the previous sections, a new computational model of motor learning and control was introduced and evaluated. The basic philosophy behind the model is that motor learning is all about acquiring a very general understanding of the own body and its movements, about forming a general body model. This general body model, which is implemented in SURE_REACH as sensorimotor model and posture memory, cannot be directly used for movement control because it is not specific to a certain task, with its specific end-state goals and constraints. However, once the task is known a movement can be prepared based upon the general body model. Even more, tasks can be accomplished that have very different constraints or requirements, compared to those in the learning context. Note, that the random behavior during motor learning differs tremendously from the behavior required during the evaluation phases. This emphasize of a task-independent body model and a mechanism that enables to use this body model to prepare movement for concrete tasks is the main difference between this approach and many other neural network models of motor learning and control. It is also the basis of much of the model's functionality and enables a more profound comparison to behavioral and neurophysiological data.

In the following, three topics are discussed. First, I will point at the model's potential to simulate further aspects of human motor control and propose structural extensions. Second, the model is related to other computational modeling approaches and I highlight in which respect it extends them. Third, I will argue that SURE_REACH is a computationally, neurophysiologically, and psychologically valid model of motor cortical learning and control.

Modeling Capabilities and Potential Model Extensions

The previous sections presented the current capabilities of the SURE_REACH model. Future extensions of the model may aim in two directions. First, the current architecture may be challenged to model additional tasks and experiments. Second, the current architecture may be structurally extended to capture more of the functionality of the human motor control system.

Further Modeling Capabilities

In the recent sections, it was shown that the model accounts for a range of data gathered from experimental psychology. Even if the main contribution of the model is its account for the flexibility of human motor control, it also captures aspects of the dynamics of movement preparation, for example reactions time or priming effects. In the following section, I want to highlight some experimental findings that also could be modeled with the current architecture without substantially modifying it.

Bayesian Information Processing. In general, humans rely strongly on visual feedback to control movements. However, if vision is blurry or occluded, humans tend to rely more on a priori knowledge, for example about generally likely finger positions or perturbations, to control their movements (Körding & Wolpert 2004; Körding, Ku & Wolpert

2004). Thus, multiple information sources are integrated, dependent on the significance and current reliability of each available information source. Population codes seem well suited to neurally implement such *Bayesian information processing* (Deneve & Pouget 2004; Knill & Pouget 2004; Ma, Beck, Latham & Pouget 2006). SURE_REACH seems to be able to account for Bayesian integration of current proprioceptive feedback with a priori knowledge of likely and unlikely arm postures. The arm posture is represented by a number of different neurons forming a population code. In situations of uncertainty, many neurons may be activated, forming a broader, more blurry representation of the current posture, whereas only few neurons may be active if the posture is accurately perceived. Additionally, frequently observed sensorimotor contingencies will be linked by stronger synaptic connections than others due to the associative learning scheme. Hence, if the arm state is well defined, a very narrow set of sensorimotor contingencies is included in the motor command generation process, resulting in precise movements. On the other hand, if the arm state is uncertain and the current posture is consequently represented by many neurons, many sensorimotor contingencies compete for their contribution to the motor commands. Due to the stronger synaptic connectivity associated to frequently experienced postures, these postures will have a stronger impact on motor command generation than others. In conclusion, it should be well possible to replicate behavioral data from related experiments (e.g. Körding & Wolpert 2004). Likewise, the neural network model for visual control of movements is also capable to account for Bayesian integration, as blurry representations of either hand or target would result in less pronounced error signals and thus a motor control process that relies more on proprioceptive input than on vision.

Memory Consolidation. When humans learn a new motor task over a period of several days, it is often observed that the performance improves considerably from the end of one training session to the beginning of the next one on a subsequent day (S. Fischer, Hallschmid, Elsner & Born 2002). This improvement is caused by the consolidation of the newly learned skill during a period of rest (Korman et al. 2007). The current structure of the sensorimotor model may at least partially account for memory consolidation. Synaptic connections in the sensorimotor model may be split in two groups. Local connections relate to small changes in arm posture and are easily acquired during motor learning. In contrast, far-reaching connections, which help to initiate longer movements faster, are only acquired after extensive training. The time to acquire sufficient far-reaching connections may be reduced by separating periods of online learning by offline learning periods. In these periods, neural activity could be propagated through posture space by local connections. The resulting activity distribution could then be used to update the connectivity of far-reaching connections. Thus, the sensorimotor model could be improved without engaging in actual movements.

A concept closely related to consolidation is interference. Experiments have shown that a newly acquired motor skill can hardly be consolidated if right after training a related but different motor skill is practiced (Shadmehr & Brashers-Krug 1997, see also Caithness et al. 2004). If the sensorimotor model's far-reaching synaptic connections, rather than local connections, are mostly responsible for the initiation of movements,

it could even be possible to account for retrograde interference. In this case, mostly synaptic changes that were caused by the second of two subsequently trained tasks will be consolidated to far-reaching synaptic connections and later on play a major role in motor control. Thus, training of one skill could interfere with the recent training of another, different skill.

Model Extensions

The SURE.REACH model focusses on the unsupervised acquisition of inverse models and how these inverse models may be used to control movements. However, it is clear from theoretical, physiological and psychological considerations that there is more to motor control than inverse models (Desmurget & Grafton 2000; Jordan & Wolpert 1999; Kawato 1999; Scott & Norman 2003). Hence, new structures should be added to the architecture to cover more of the human motor system's functionality and also to highlight computational problems that arise from the interaction of these structures with the current architecture. Additionally, components of the current architecture could be advanced to address more complex control tasks. In the following, potential extensions of the model are discussed.

Balancing Synaptic Connectivity. In the moment, the connectivity in the model is not well balanced. For example, due to the relative low maximal synaptic weight of $\theta = 0.1$ in the sensorimotor model, the activity of the sensory-to-motor mapping differs by some magnitudes between neurons that encode the target and neurons that encode remote areas in workspace. This results in an unrealistically wide spectrum of neural activity. Thus, the learning rules and activity propagation mechanism should be improved to provide a more coherent activity distribution. The introduction of inhibitory connectivity, as applied in the networks for visual feedback control, might help to realize this goal.

Next, the acquisition of far-reaching sensorimotor contingencies is necessary to model reaction time effects of the model but at the same time seem to hinder its capabilities to avoid obstacles. This contradiction needs to be resolved by further discounting the weight of far-reaching connections. This would result in the fast generation of a weakly represented movement plan based on the far-reaching connections, which is subsequently overridden by the slowly spreading activity propagated by local connections. Thus both features, decreasing reaction times and obstacle avoidance, could be preserved in the model. In conclusion, further improvement of the learning and movement preparation mechanisms seems necessary to enhance the models robustness and provide a sound foundation for future model extensions.

Adaptive Space Representation. By now, the representation of hand and posture space is prewired. Body spaces are covered by neuron populations of predefined size with uniformly distributed receptive fields, regardless of the relevance of different parts of the workspace. Currently, the memory requirements for space representation and neural networks grow exponentially with the number of input dimensions, resulting in neither biological plausible nor computational tractable models for more complex control

tasks. Thus, compact and adaptive space representations are crucial to address more complex control problems, for example three-dimensional bodies with more degrees of freedom or with dynamic properties. In the future, body space representations should develop unsupervised in order to cover body spaces adaptively. Body spaces could be based on self-organizing maps (Morasso et al. 1997; Toussaint 2006), neural-gas like structures (H. Hoffmann & Möller 2003; Martinez, Berkovich & Schulten 1993), or other machine learning techniques (Butz 2005, 2006). In these representations, receptive fields of neurons are not only distributed adaptively in each spatial dimension but it is also possible to cover entire subspaces with single neurons. This would result in a considerable reduction of the required number of neurons for each dimension of a body space and would enable to address more complex control problems.

Merging Acting and Learning. SURE_REACH's learning scheme does not require the existence of goals during motor learning, though it also doesn't deny that goals may be represented. Hence, random movements are chosen as a model of the movements during early motor learning. In the recent sections, phases of motor learning and purposive behavior were strictly separated to enable a sound evaluation. However, it is known that most behavior is goal-oriented even in the youngest infants and it seems obvious that infants learn from such behavior (von Hofsten 2004). Thus, it should also be possible to engage in goal-directed reaching and meanwhile adapt the neural connectivity of SURE_REACH. This would allow to compare the development of the model's movement trajectories to those of infants that develop their reaching skills. Additionally, learning during goal-directed behavior may lead to a preferred exploration of those regions of the body's workspace that are especially relevant. This might result in more efficient body space representations and motor learning.

Predictive Mechanisms and Internal Feedback. Not only inverse models but also forward models seem to play a crucial role in motor control (Bhushan & Shadmehr 1999; Karniel 2002; Kawato 1999). Forward models encode the direct input-output mapping of a body. Thus, they are relevant for many tasks that require some knowledge about how the body might potentially move. First, forward models might help to compensate for missing, unreliable, or delayed sensory feedback by predicting how the body reacts to issued motor commands (Bullock et al. 1993; Kalman 1960; Sabel 2000; Scott & Norman 2003). Integrating forward models into SURE_REACH might enable to simulate control of movements despite impaired sensory feedback. It would require the integration of two possibly contradicting sources of information for estimating the current hand position or arm posture. The neural population codes used throughout the model seem well suited to integrate multiple information sources (Deneve & Pouget 2004). Second, forward models may be used to predict the outcome of an ongoing movement. This capability seems crucial to further develop the neural networks for visual movement control or other potential, higher-level control loops, which require that lower-level goal representations are adjusted based on an expected movement outcome rather than on the current body state (Desmurget & Grafton 2000). Third, forward models are important for motor learning with a distal teacher (Jordan & Rumelhart 1992). Whereas the motor learning

approach favored in this thesis clearly does not require forward models, forward models might be necessary to model offline motor learning. In sum, striving to integrate forward models for state estimation and outcome prediction into the present architecture might considerably advance the model and might also highlight new interesting problems.

Target Evaluation. In many situations, multiple objects offer themselves as targets for reaching movements. Thereby, the choice is not only based on the value associated to one or the other object but also the costs of moving there (Trommershäuser, Landy & Maloney 2006). Neurophysiological studies suggest that information about target location and the value of target acquisitions are integrated in parietal sensorimotor areas (Musallam, Corneil, Greger, Scherberger & Andersen 2004). Additionally, it is well possible that the actual process of target selection happens not earlier than in primary motor areas (Bastian, Schöner & Riehle 2003; Cisek 2006; Cisek & Kalaska 2005). Thus, target selection and motor control seem to be deeply entangled. A recent integration of the SURE_REACH model with a neural network model for target selection accounted for such, more complex motor behavior (Herbort et al. 2007; Ognibene, Mannella et al. 2006). In this case, experimental data was replicated that shows that humans integrate knowledge about their own motor variability and potential targets to maximize the performance in an aiming task (Trommershäuser, Maloney & Landy 2003). However, in this approach the interaction between motor control structures and mechanism of target selection was unidirectional. In future work, it might prove helpful to enable crosstalk between both to enable integrating motor control with higher-level cognitive or motivational processes, such as decisions between multiple targets.

Relation to other Models

The previous sections outlined SURE_REACH's modeling potential and possible structural extensions. This section discusses its relationship to other computational models, which were reviewed earlier in more detail (page 24). It owes much to many of these models and many of them are more powerful than SURE_REACH in one aspect or the other. Anyhow, in the next paragraphs, this model's relations to other approaches are discussed, highlighting the advancement manifested in this approach.

Feedback Error Learning

The cerebellar models of feedback error learning (FEL) address the acquisition of fine-tuned, fast, cerebellar inverse models for reaching movements by learning from, and finally substituting, cerebral motor commands (Berthier et al. 1993; Haruno et al. 2001, 2003; Kawato et al. 1987). SURE_REACH, on the other hand, addresses a different part of the human motor system because it is a model of unsupervised motor cortical learning. However, it might serve as a model of the cerebral source of the motor signals, which are necessary to enable FEL. In the future, SURE_REACH might be integrated with cerebellar models of motor control to cover a broader range of the anatomy and computational mechanism that underly human motor control.

Direct Inverse Modeling

The presented model extends models that implement direct inverse modeling (DIM) in two ways. First, the temporally weighted association mechanism in the action columns and the activation propagation process results in effective links between potential goal states, and temporally remote actions and initial condition. This enables SURE_REACH to learn motor control even in a context in which the consequences of motor commands only unfold in time and no clear relationship between actions and perception exists, thus solving the temporal alignment problem. Second, the unsupervised learning method enables the representation and flexible online resolution of motor redundancy. Hence, unlike DIM approaches, SURE_REACH is able to effectively and flexibly learn to control a redundant body.

Posture-Based Motion Planning Theory

SURE_REACH is able to solve the redundancy problem online and under varying constraints, similar to the *posture-based motion planning theory* (PB theory, Rosenbaum et al. 1993b, 1995; Rosenbaum, Meulenbroek, Vaughan & Jansen 2001). Thus, it owes to the PB theory the emphasize on the task-dependent preparation of movements each time a novel target is provided or situational constraints change. It extends the PB theory in three ways.

First, unlike the PB theory, SURE_REACH accounts for the acquisition of a sensorimotor body model. Second, the PB theory is mainly a theory for movement preparation and details no mechanism for movement control. In contrast, SURE_REACH also models the execution of a movement, continuously integrating novel information. Third, whereas SURE_REACH is implemented in a neural network model that reflects aspects of the neurophysiology of the human motor control system, the PB theory is analytically formulated and, by now, offers no neural network implementation and doesn't relate to cortical representations. Nonetheless, the PB theory has much to offer to SURE_REACH as it is the model that embraces human behavioral flexibility to the largest extent, including for example grasping and tool use (Meulenbroek, Rosenbaum, Jansen, Vaughan & Vogt 2001).

Reinforcement Learning

The dynamic programming approach of the movement preparation process resembles model-based reinforcement learning (RL, Sutton 1990; Sutton & Barto 1998). Nevertheless, there is a strong difference to RL approaches discussed in the context of motor learning (Berthier 1996; Berthier et al. 2005; Kositsky & Barto 2002). In these models, task-specific inverse models are acquired and the motor control process does not enable the integration of additional constraints or novel tasks. In general, dynamic programming might enhance the flexibility of RL approaches in being able to approach arbitrary representable goals and adhere to additional constraints.

Validity

The recent paragraphs related the presented model to other current models of motor learning and control, and pin-pointed the specific contribution of SURE_REACH. In the following sections, I will discuss the validity of the model with respect to the computational demands of the task, neurophysiological findings, and behavioral data.

The Simulated Body

The computational demands for the presented model are mostly defined by the body that it has to control. In this case, the task was to control a planar, kinematic, three-joint arm by several pairs of antagonistic motoneurons, each of which controls the motion of a certain joint. Doubtless, this is a dramatic simplification of the musculoskeletal system of the human arm and the spinal circuitry that connects the motor cortex with the arm's motoneurons. Due to this simplifications, several problems that make motor control difficult were not directly addressed.

First, the modeled arm is merely kinematic and does not account for the velocities, torques, and forces that have to be considered when controlling a physical body (Gribble & Ostry 1999). This also means that the motor commands, which the motor controller produces, don't need to be properly scaled because only the direction of the movement is ultimately relevant but not the amplitude of the motor commands. Second, in the model, motor commands directly influenced the displacement of the arm rather than contracting muscles or altering the properties of spinal feedback loops. Additionally, the model's actuators only affect single joints, are arranged symmetrically, and their strength is well balanced. In contrast, human muscle often span several joints, are not symmetrically arranged and are of differing strengths. Third, efferent and afferent signals are not subject to delays. Thus, the model doesn't need to compensate for delayed perception and motor commands.

The purpose of the presented model is mainly to account for the unsupervised acquisition of flexible motor control in a redundant context. The arm model captures two critical properties that are relevant for understanding this task. First, the arm is redundant on the kinematic and the sensorimotor level; each hand position can be realized by many different arm postures, and transition between any two postures can be realized by different sequences of motor commands. This redundancy enables to avoid obstacles or to include demands of future tasks in movement planning, but it also requires body models to encode redundant solutions for the different motor problems. Second, each movement requires the execution of a sequence of motor commands and the neural networks have to learn from a continuous stream of perceptions and motor commands. This is especially important to understand the challenges of unsupervised motor learning because there is no clear temporal relationship between start-conditions, goals, and motor commands (temporal alignment problem). Motor redundancy and the temporal properties of movement execution are crucial to enable behavioral flexibility but also impose severe problems on unsupervised learning schemes. In conclusion, the simulated arm can be seen as one of the simplest bodies that imposes the targeted problems on a computational model.

The next step to improve the models functional validity would certainly be the introduction of a dynamic body. One of the key problems of applying unsupervised learning methods to dynamic bodies, the temporal alignment problem, is already addressed. Additionally, a fore-runner of this model has already shown success in controlling a dynamic arm (Herbort et al. 2005a; Herbort 2005). Thus, SURE_REACH should be able to control dynamic bodies.

Neural Body Spaces

The model was clearly not devised to closely mirror motor cortical anatomy or physiology. Nevertheless, the model is based on several known properties of cortical motor control structures. Representations of the body, be it hand or posture space, are implemented as neural population codes. Neuron recordings show that body spaces are similarly encoded in the primate brain's sensory, sensorimotor, and motor areas (Aflalo & Graziano 2006; Cisek & Kalaska 2005; Georgopoulos 1995; Georgopoulos et al. 1983; Kakei et al. 1999; Rizzolatti, Fadiga, Fogassi & Gallese 1997; Scherberger & Andersen 2007; Scherberger, Jarvis & Andersen 2005; Shadmehr & Wise 2005). In fact, representations of arm movements in ventral premotor areas and primary motor cortex differ if vision is artificially distorted (Schwartz, Moran & Reina 2004). Thus, different body spaces, whose activities depend predominantly on different sensory information, can also be found in the brain. The identified encodings parallel SURE_REACH's posture space and hand space, in which the former encodes proprioceptive inputs whereas the latter relies on vision.

Moreover, the motor cortical representations seem to encode arm postures. Cell responses during ongoing movements depend on arm postures (Scott & Kalaska 1997) and are most likely encoded in a posture-based coordinate frame (Ajemian et al. 2000). Neural activity and artificial neural stimulation of a neuron correlate most significantly with particular postures, either holding the arm in that posture or causing the arm to move to that posture (Graziano 2006). Additionally, motor cortical cells in freely behaving monkeys seem to mostly encode movement end-postures (Aflalo & Graziano 2006). Likewise, in the model movement control and goal representations are encoded posture-based.

Additionally, the finding that potential to-be-executed movements that differ only slightly from each other cannot be independently prepared as compared to rather distinct movements fits well in the context of neural population codes, which also have a limited spatial resolution (Favilla 1997; van Sonderen & Dernier van der Gon 1991; van Sonderen et al. 1988).

Motor Learning

From a macroscopic perspective, the model fits well into current theories that locate associative unsupervised learning mechanisms in the cerebral cortex and that emphasize the importance of recurrent neural connections in this area (Doya 1999, 2000). Furthermore, humans independently acquire inverse models for movement kinematics and dynamics (Krakauer, Ghilardi & Ghez 1999). This fact is reflected in SURE_REACH

because the posture memory, which is an inverse kinematics model, is trained independently from the sensorimotor model, which relates more to inverse dynamic models. On a synaptic level, the model relies entirely on simple Hebbian learning, which is principally known to modulate synaptic connectivity in the brain (Abbott & Nelson 2000; Jackson, Mavoori & Fetz 2006). Thus, the learning mechanisms in SURE_REACH seem biologically justified.

Representations of Redundancy

One of the distinct features of the SURE_REACH model is its ability to encode redundant goal representations. For example, in the motor controller the goal to move the hand to a certain location is represented by the activation of many neurons, which encode redundant postures. To my knowledge, by now no experiment has been conducted that might allow to decide if redundant goal representations are also used in the brain to control movements. However, multiple lines of neurophysiological and psychological experiments show that redundancy might be explicitly encoded.

Evidence from psychological and neurophysiological experiments shows that multiple goals are represented at least during the preparation of a movement. In these experiments, participants or monkeys were presented with a cue indicating multiple potential movement targets before the final target was specified. Humans profit from such prior information, which reduces the set of possible targets, even if some ambiguity is left (Bock & Arnold 1992; Rosenbaum 1980). Neurophysiological recordings show that monkeys can pre-activate ranges of multiple movements (Bastian et al. 2003) or even opposing movements (Cisek & Kalaska 2005). However, when these experiments are related to the SURE_REACH model, some limitations should be kept in mind. On the one side, they examine the possibility of representing multiple targets for the hand rather than multiple postures for a single hand target. On the other side, they focus on representations that are active before a movement is initiated or even before it is completely prepared, but in the model these representations persist throughout the movement. Nevertheless, it shows that multiple targets may be coactivated in other stages of the movement preparation process and motor cortical networks are principally capable of representing redundancy.

Furthermore, behavioral experiments suggest that redundancy is explicitly encoded during movement preparation and execution. The way humans carry out actions that are part of a sequence depends on the requirements of subsequent tasks (Dell et al. 1999; M. H. Fischer et al. 1997; Weigelt, Kunde & Prinz 2006). Thus, among the possible ways to carry out an action a way is selected that optimizes the performance of the subsequent action. As mentioned earlier, it seems likely that this capability requires the explicit representation of motor redundancy in some form.

Finally, the extent of task-irrelevant variability of movement trajectories depends on target certainty even if movement start- and end-points are identical (de Freitas, Scholz & Stehman 2007). Thus, the authors concluded that “the use of motor redundancy can be affected by movement planning” (de Freitas et al. 2007, p. 71). However, if the amount of redundancy allowed in a movement can be explicitly controlled, there should also

be some form of explicit representation of motor redundancy as in the SURE_REACH model.

In sum, neurophysiological studies show that motor cortical areas are capable and actually do encode redundant action possibilities at some stages of the movement preparation process. Even more so, behavioral experiments suggest that redundancy is explicitly encoded and processed during movement preparation and execution. In conclusion, it seems that motor control builds on, as does the model, explicit representations of motor redundancy.

Parallel Processes

In humans, movement preparation and movement execution seem to be parallel and comparatively independent processes, which can even be experimentally detached. On the one hand, movements may be prepared without being executed (Bastian et al. 2003), and on the other hand, movements may be executed without having been properly prepared (van Sonderen & Dernier van der Gon 1991; van Sonderen et al. 1988). Also, the model incorporates parallel mechanisms for movement preparation and movement execution because the sensory-to-motor mapping can gradually adapt to a new goal representation during an ongoing movement. For example, this was the case in the simulation of the priming experiment (page 53).

In this context it is interesting to note that humans who move a computer mouse react quicker to target displacements than to cursor displacements during an ongoing movement (Brenner & Smeets 2003). This asymmetry might relate to the parallel processes of movement preparation and movement execution in humans and in the model. Whereas a change of the cursor location only requires a readout of the movement plan for another position, similar to the combination of the sensory-to-motor mapping and the current posture representation, a target displacement requires a more time consuming update of the movement plan. Besides the empirical data that support the SURE_REACH model, this parallel processing captures an important and interesting aspect of the motor system, which has to be examined more thoroughly in future work.

Summary

As discussed earlier, recent models of motor learning fall short in accounting for the high flexibility with which human motor control adapts to novel situations. On the other hand, theories of motor control that emphasize this human ability offer no account for sensorimotor learning. The SURE_REACH model is a new framework for understanding both, motor learning and highly adaptive behavioral control. The advancement that this model has made for the understanding of motor learning is the combination of the following two properties. First, it proposed a mechanism that enables effective and unsupervised motor learning in a redundant context in which movements extend in time. Second, motor learning is based on encoding redundant kinematics and trajectories. In contrast to many other approaches, SURE_REACH emphasizes the necessity to prepare movements based on a general, task-independent body model. The adaptive preparation of movements is the basis for its functionality and has enabled the model to account

for a range of behavioral data, for example the capability to avoid obstacles or adjust movements to fit the requirements of subsequent behavior.

The model of the movement preparation process enables to compare the model's behavior to many experiments that are related to movement preparation and initiation. In this context, for example, priming effects have been already modeled successfully. Additionally, the interaction of motor learning, movement preparation, and motor performance may be examined with the model. The decrease of movement durations and reaction times with extensive training has already been shown in the model. Likewise, the model might also account for performance increases due to memory consolidation or bayesian integration.

Finally, the model builds on principles that underly human learning and motor control. Multiple interdependent body space representations are organized by populations of neurons and motor learning is realized by local learning rules. On a systemic levels, no neural signals are provided that the brain could not easily compute. The purpose of the neural networks presented is not to model closely the anatomy or microstructure of motor cortical tissue. Nevertheless, it may serve as a good starting point to develop models of cortical motor networks that detail neuroanatomical and neurophysiological features *and* account for the high functionality of human motor learning and behavioral control.

Conclusion

In conclusion, the model offers a comprehensive account of how humans adapt to their bodies on the one side and adapt motor control flexibly to ever-changing situational requirements and constraints on the other side. Whereas many details of the model may be debated and some of them might be oversimplifications, I think the basic approach that led to the development of the model is a considerable advancement in modeling human motor learning and behavioral control. On the one hand, SURE_REACH steps behind the highly constraint and somewhat simplistic notion of human behavior that is eminent in many current neural network models. On the other hand, it offers a foundation for grounding abstract theories of highly adaptive behavior in unsupervised sensorimotor learning and a neural network structure. Thus, the model opens the door to now integrate more cognitive aspects of motor control into neural network models. These might finally offer a real account for the intricate brain mechanisms that enable humans to learn to move their bodies seemingly effortless and nevertheless with unmatched sophistication.

References

- Abbott, L. F. & Nelson, S. B. (2000). Synaptic plasticity: taming the beast. *Nature Neuroscience*, 3, Suppl. 1178-1183.
- Abeele, S. & Bock, O. (2001). Sensorimotor adaptation to rotated visual input: different mechanisms for small versus large rotations. *Experimental Brain Research*, 140(4), 407-410.
- Adamovich, S. V., Levin, M. F. & Feldman, A. G. (1997). Central modifications of reflex parameters may underlie the fastest arm movements. *Neurophysiology*, 77(3), 1460-1469.
- Adams, J. A. (1971). A closed-loop theory of motor learning. *Journal of Motor Behavior*, 3(2), 111-149.
- Afifi, A. K. & Bergman, R. A. (2005). *Functional neuroanatomy: Text and atlas* (2nd Edition Aufl.; I. Nogueira, J. Foltin, J. Malley & L. A. Sheinis, Hg.). New York: McGraw-Hill.
- Aflalo, T. N. & Graziano, M. S. A. (2006). Partial tuning of motor cortex neurons to final posture in a free-moving paradigm. *Proceedings of the National Academy of Science*, 8, 2909-2914.
- Ajemian, R., Bullock, D. & Grossberg, S. (2000). Kinematic coordinates in which motor cortical cells encode movement direction. *Journal of Neurophysiology*, 84, 2191-2203.
- Amari, S. (1977). Dynamics of pattern formation in lateral-inhibition type neural fields. *Biological Cybernetics*, 27(2), 77-87.
- Baraduc, P., Guigon, E. & Burnod, Y. (1999). Where does the population vector of motor cortical cells point during reaching movements? In M. Kearns, S. Solla & D. Cohn (Hg.), *Advances in neural information processing systems* (Bd. 11, S. 83-89). Cambridge, MA: MIT Press.
- Baraduc, P., Guigon, E. & Burnod, Y. (2001). Recoding arm position to learn visuomotor transformations. *Cerebral Cortex*, 11, 906-917.
- Barlow, J. S. (2002). *The cerebellum and adaptive control*. New York, NY: Cambridge University Press.
- Barsalou, L. W. (1999). Perceptual symbol systems. *Behavioral and Brain Sciences*, 22(4), 577-600.
- Barto, A. G., Fagg, A. H., Sitkoff, N. & Houk, J. C. (1999). A cerebellar model of timing and prediction in the control of reaching. *Neural Computation*, 11, 565-594.
- Bastian, A., Schöner, G. & Riehle, A. (2003). Preshaping and continuous evolution of motor cortical representations during movement preparation. *European Journal of Neuroscience*, 18, 2047-2058.
- Battaglia-Mayer, A., Caminiti, R., Lacquaniti, F. & Zago, M. (2003). Multiple levels of representation of reaching in the parieto-frontal network. *Cerebral Cortex*, 13(10), 1009-1022.
- Bellman, R. E. (1957). *Dynamic programming*. Princeton, NJ: Princeton University Press.
- Bernstein, N. A. (1967). *The co-ordination and regulation of movements*. Oxford: Pergamon Press.
- Berthier, N. E. (1996). Learning to reach: A mathematical model. *Developmental Psychology*, 32(5), 811-823.
- Berthier, N. E., Rosenstein, M. T. & Barto, A. G. (2005). Approximate optimal control as a model for motor learning. *Psychological Review*, 112(2), 329 - 346.
- Berthier, N. E., Singh, S. P., Barto, A. G. & Houk, J. C. (1992). A cortico-cerebellar model that learns to generate distributed motor commands to control a kinematic arm. *Advances in Neural Information Processing Systems*, 4, 611-618.
- Berthier, N. E., Singh, S. P., Barto, A. G. & Houk, J. C. (1993). Distributed representation

- of limb motor programs in arrays of adjustable pattern generators. *Journal of Cognitive Neuroscience*, 5, 56-78.
- Bhushan, N. & Shadmehr, R. (1999). Computational nature of human adaptive control during learning of reaching movements in force fields. *Biological Cybernetics*, 81, 39-60.
- Birbaumer, N. & Schmidt, R. (1996). *Biologische Psychologie [Biological psychology]* (3rd Aufl.). Berlin Heidelberg, Germany: Springer.
- Bizzi, E., Accornero, N., Chapple, W. & Hogan, N. (1984). Posture control and trajectory formation during arm movement. *The Journal of Neuroscience*, 4(11), 2738-2744.
- Bizzi, E., Hogan, N., Mussa-Ivaldi, F. A. & Giszter, S. (1992). Does the nervous system use equilibrium-point control to guide single and multiple joint movements? *Behavioral and Brain Sciences*, 15, 603-613.
- Bizzi, E., Polit, A. & Morasso, P. (1976). Mechanisms underlying achievement of final head position. *Journal of Neurophysiology*, 39(2), 435-444.
- Bock, O. & Arnold, K. (1992). Motor control prior to movement onset: preparatory mechanisms for pointing at visual targets. *Experimental Brain Research*, 90, 209-216.
- Brenner, E. & Smeets, J. N. J. (2003). Fast corrections of movements with a computer mouse. *Spatial Vision*, 16(3-4), 365-376.
- Buehrmann, T. & Di Paolo, E. (2006). Biological actuators are not just springs: Investigating muscle dynamics and control signals. In S. Nolfi et al. (Hg.), *From Animals to Animals 9: The Ninth International Conference on the Simulation of Adaptive Behavior* (S. 89-100). Berlin, Heidelberg: Springer.
- Bullock, D., Cisek, P. & Grossberg, S. (1998). Cortical networks for control of voluntary arm movements under variable force conditions. *Cerebral Cortex*, 8, 48-62.
- Bullock, D., Grossberg, S. & Guenther, F. H. (1993). A self-organizing neural model of motor equivalent reaching and tool use by a multijoint arm. *Journal of Cognitive Neuroscience*, 5(4), 408-435.
- Butz, M. V. (2005). Kernel-based, ellipsoidal conditions in the real-valued XCS classifier system. *GECCO 2005: Genetic and Evolutionary Computation Conference*, 1835-1842.
- Butz, M. V. (2006). *Rule-based evolutionary online learning systems: A principled approach to LCS analysis and design*. Berlin Heidelberg: Springer-Verlag.
- Butz, M. V., Herbort, O. & Hoffmann, J. (2007). Exploiting redundancy for flexible behavior: Unsupervised learning in a modular sensorimotor control architecture. *Psychological Review*, 114(4), 1015-1046.
- Butz, M. V., Lenhard, A. & Herbort, O. (2007). Emergent effector-independent internal spaces: Adaptation and intermanual learning transfer in humans and neural networks. *Proceedings of the International Joint Conference on Neural Networks*, 20, 1970-1975.
- Caithness, G., Osu, R., Bays, P., Chase, H., Klassen, J., Kawato, M. et al. (2004). Failure to consolidate the consolidation theory of learning for sensorimotor adaptation tasks. *The Journal of Neuroscience*, 24(40), 8662-8671.
- Chapeau-Blondeau, F. & Chauvet, G. (1991). A neural network model of the cerebellar cortex performing dynamic associations. *Biological Cybernetics*, 65, 267-279.
- Cisek, P. (2005). Neural representations of motor plans, desired trajectories, and controlled objects. *Cognitive Processing*, 6(1), 15-24.
- Cisek, P. (2006). Integrated neural processes for defining potential actions and deciding between them: A computational model. *Journal of Neuroscience*, 26(38), 9761-9770.
- Cisek, P. & Kalaska, J. F. (2005). Neural correlates of reaching decisions in dorsal premotor cortex: Specification of multiple direction choices and final selection of action. *Neuron*, 45(5), 801-814.
- Conditt, M. A., Gandolfo, F. & Mussa-Ivaldi, F. A. (1997). The motor system does not learn the

- dynamics of the arm by rote memorization of past experience. *Journal of Neurophysiology*, 78, 554-560.
- Contreras-Vidal, J. L., Grossberg, S. & Bullock, D. (1997). A neural model of cerebellar learning for arm movement control: Cortico-spino-cerebellar dynamics. *Learning and Memory*, 3, 475-502.
- Cruse, H. (2003). The evolution of cognition - a hypothesis. *Cognitive Science*, 27, 135-155.
- Cruse, H., Brüwer, M. & Dean, J. (1993). Control of three- and four-joint arm movement: Strategies for a manipulator with redundant degrees of freedom. *Journal of Motor Behavior*, 25(3), 131-139.
- Cruse, H. & Steinkühler, U. (1993). Solution of the direct and inverse kinematic problems by a common algorithm based on the mean of multiple computations. *Biological Cybernetics*, 69, 341-351.
- Cruse, H., Steinkühler, U. & Burkamp, C. (1998). MMC - a recurrent neural network which can be used as manipulable body model. In R. Pfeifer, B. Blumberg, J.-A. Meyer & S. Wilson (Hg.), *From Animals to Animats 5: The Fifth International Conference on the Simulation of Adaptive Behavior* (S. 381-389). Cambridge, MA: MIT Press.
- Cruse, H., Wischmeyer, E., Brüwer, M., Brockfeld, P. & Dress, A. (1990). On the cost functions for the control of the human arm movement. *Biological Cybernetics*, 62, 519-528.
- Darling, W. G., Seitz, R. J., Peltier, S. & Butler, L. T. A. J. (2007). Visual cortex activation in kinesthetic guidance of reaching. *Experimental Brain Research*, 179(4), 607-619.
- de Freitas, S. M. S. F., Scholz, J. P. & Stehman, A. J. (2007). Effect of motor planning on use of motor abundance. *Neuroscience Letters*, 417(1), 66-71.
- de Luca, C. J., LeFever, R. S., McCue, M. P. & Xenakis, A. P. (1982). Behaviour of human motor units in different muscles during linearly varying contractions. *Journal of Physiology*, 329, 113-128.
- Dean, J. & Brüwer, M. (1994). Control of human arm movements in two dimensions: Paths and joint control in avoiding simple linear obstacles. *Experimental Brain Research*, 97, 497-514.
- Dehaene, S., Naccache, L., Le Clec'H, G., Koechlin, E., Mueller, M., Dehaene-Lambertz, G. et al. (1998). Imaging unconscious semantic priming. *Nature*, 395, 597-600.
- Dell, G. S., Chang, F. & Griffin, Z. M. (1999). Connectionist models of language production: Lexical access and grammatical encoding. *Cognitive Science*, 23(4), 123-147.
- Deneve, S. & Pouget, A. (2004). Bayesian multisensory integration and cross-modal spatial links. *Journal of Physiology - Paris*, 98(1-3), 249-258.
- Desmurget, M. & Grafton, S. (2000, November). Forward modeling allows feedback control for fast reaching movements. *Trends in Cognitive Sciences*, 4(11), 423-431.
- Dietz, V. (2003). Spinal cord pattern generators for locomotion. *Clinical Neurophysiology*, 114, 1379-1389.
- Doya, K. (1999). What are the computations of the cerebellum, the basal ganglia and the cerebral cortex? *Neural Networks*, 12(7-8), 961-974.
- Doya, K. (2000). Complementary roles of basal ganglia and cerebellum in learning and motor control. *Current Opinion in Neurobiology*, 10(6), 732-739.
- Elliott, D., Helsen, W. F. & Chua, R. (2001, May). A century later: Woodworth's (1899) two-component model of goal-directed aiming. *Psychological Bulletin*, 127(3), 342-57.
- Elsner, B. & Hommel, B. (2001). Effect anticipations and action control. *Journal of Experimental Psychology*, 27(1), 229-240.
- Engelbrecht, S. E. (2001). Minimum principles in motor control. *Journal of Mathematical Psychology*, 45, 497-542.

- Erlhagen, W. & Schöner, G. (2002). Dynamic field theory of movement preparation. *Psychological Review*, 109(3), 545-572.
- Favilla, M. (1997). Reaching movements: Concurrency of continuous and discrete programming. *NeuroReport*, 8, 3973-3977.
- Feldman, A. G. (1966). Functional tuning of nervous system with control of movement or maintenance of a steady posture. II. Controlable parameters of the muscle. *Biophysics*, 11, 565-578.
- Feldman, A. G. & Levin, M. F. (1995). Positional frames of reference in motor control: Origin and use. *Behavioral and Brain Sciences*, 18, 723-806.
- Fischer, M. H., Rosenbau, D. A. & Vaughan, J. (1997). Speed and sequential effects in reaching. *Journal of Experimental Psychology: Human Perception and Performance*, 23(2), 404-428.
- Fischer, S., Hallschmid, M., Elsner, A. L. & Born, J. (2002). Sleep forms memory for finger skills. *PNAS*, 99(18), 11987-11991.
- Flament, D., Shapiro, M. B., Kempf, T. & Corcos, D. M. (1999). Time course and temporal order of changes in movement kinematics during learning of fast and accurate elbow flexions. *Experimental Brain Research*, 129, 441-450.
- Flanagan, J. R. & Rao, A. K. (1995). Trajectory adaptation to a nonlinear visuomotor transformation: evidence of motion planning in visually perceived space. *Journal of Neurophysiology*, 74(5), 2174-2178.
- Flash, T. & Hogan, N. (1985). The coordination of arm movements: An experimentally confirmed mathematical model. *The Journal of Neuroscience*, 5(7), 1688-1703.
- Flash, T. & Sejnowski, T. J. (2001). Computational approaches to motor control. *Current Opinion in Neurobiology*, 11, 655-662.
- Georgopoulos, A. P. (1995). Current issues in directional motor control. *Trends in Neuroscience*, 18(11), 506-510.
- Georgopoulos, A. P., Caminiti, R., Kalaska, J. F. & Massey, J. T. (1983). Spatial coding of movement: A hypothesis concerning the coding of movement direction by motor cortical populations. *Experimental Brain Research*, 49(7), 327-336.
- Gomi, H. & Kawato, M. (1997). Human arm stiffness and equilibrium-point trajectory during multi-joint movement. *Biological Cybernetics*, 76, 163-171.
- Gottlieb, G. L. (2000). A test of torque-control and equilibrium-point models of motor control. *Human Movement Science*, 19(6), 925-931.
- Gottlieb, G. L., Corcos, D. M., Jaric, S. & Agarwal, G. C. (1988). Practice improves even the simplest movements. *Experimental Brain Research*, 73, 436-440.
- Graziano, M. (2006). The organization of behavioral repertoire in motor cortex. *Annual Review Neuroscience*, 29, 105-134.
- Greenwald, A. G. (1970). Sensory feedback mechanisms in performance control: With special reference to the ideo-motor mechanism. *Psychological Review*, 77(2), 73-99.
- Gribble, P. L., Ostry, D., Sanguineti, V. & Laboissiere, R. (1998). Are complex control signals required for human arm movement? *Journal of Neurophysiology*, 79, 1409-1424.
- Gribble, P. L. & Ostry, D. J. (1999). Compensation for interaction torques during single- and multijoint limb movement. *Journal of Neurophysiol*, 82, 2310-2326.
- Gribble, P. L. & Ostry, D. J. (2000). Compensation for loads during arm movements using equilibrium-point control. *Experimental Brain Research*, 135, 474-482.
- Grush, R. (2004). The emulation theory of representation: Motor control, imagery, and perception. *Behavioral and Brain Sciences*, 27, 377-396.
- Harris, C. M. & Wolpert, D. M. (1998). Signal-dependent noise determines motor planning. *Nature*, 394, 780-784.

- Haruno, M., Wolpert, D. M. & Kawato, M. (2001). Mosaic model for sensorimotor learning and control. *Neural Computation*, 13(10), 2201-2220.
- Haruno, M., Wolpert, D. M. & Kawato, M. (2003). Hierarchical mosaic for movement generation. In T. Ono et al. (Hg.), *Excepta Medica International Congress Series* (Bd. 1250, S. 575-590). Amsterdam, Netherlands: Elsevier Science B.V.
- Hebb, D. O. (1949). *The organization of behavior*. New York: John Wiley.
- Henneman, E., Somjen, G. & Carpenter, D. O. (1965). Excitability and inhibitability of motoneurons of different sizes. *Journal of Neurophysiology*, 28, 599-620.
- Herbart, J. F. (1825). *Psychologie als Wissenschaft neu gegründet auf Erfahrung, Metaphysik und Mathematik. Zweiter analytischer Teil [Psychology as a science newly founded on experience, metaphysics and mathematics: Second, analytical part]*. Königsberg, Germany: August Wilhelm Unzer.
- Herbort, O. (2005). *A computational model of hierarchical anticipatory control for reaching*. (Unpublished Diploma Thesis, University of Würzburg)
- Herbort, O. & Butz, M. V. (2007). Encoding complete body models enables task dependent optimal control. *Proceedings of the International Joint Conference on Neural Networks*, 20, 1639-1644.
- Herbort, O., Butz, M. V. & Hoffmann, J. (2005a). Towards an adaptive hierarchical anticipatory behavioral control system. In C. Castelfranchi, C. Balkenius, M. V. Butz & A. Ortony (Hg.), *From Reactive to Anticipatory Cognitive Embodied Systems: Papers from the AAAI Fall Symposium* (S. 83-90). Menlo Park, CA: AAAI Press.
- Herbort, O., Butz, M. V. & Hoffmann, J. (2005b). Towards the advantages of hierarchical anticipatory behavioral control. In K. Opwis & I.-K. Penner (Hg.), *Proceedings of the KogWis05. The German Cognitive Science Conference* (S. 77-82). Basel: Schwabe.
- Herbort, O., Ognibene, D., Butz, M. V. & Baldassarre, G. (2007). Learning to select targets within targets in reaching tasks. *Proceedings of the 6th International IEEE Conference on Development and Learning, 2007*, 6, 7-12.
- Hikosaka, O., Nakamura, K., Sakai, K. & Nakahara, H. (2002). Central mechanisms of motor skill learning. *Current Opinion in Neurobiology*, 12, 217-222.
- Hirayama, M., Kawato, M. & Jordan, M. I. (1993). The cascade neural network model and a speed-accuracy trade-off of arm movement. *Journal of Motor Behaviour*, 25(3), 162-174.
- Hof, A. L. (2003). Muscle mechanics and neuromuscular control. *Journal of Biomechanics*, 36, 1031-1038.
- Hoffmann, H. & Möller, R. (2003). Unsupervised learning of a kinematic arm model. In O. Kaynak, E. Alpaydin, E. Oja & L. Xu (Hg.), *Artificial Neural Networks and Neural Information Processing—ICANN/ICONIP 2003, LNCS* (Bd. 2714, S. 463-470). Berlin: Springer.
- Hoffmann, J. (1993). *Vorhersage und Erkenntnis: Die Funktion von Antizipationen in der menschlichen Verhaltenssteuerung und Wahrnehmung [Anticipation and cognition: The function of anticipations in human behavioral control and perception]*. Göttingen, Germany: Hogrefe.
- Hoffmann, J. (2003). Anticipatory behavior control. In M. Butz, O. Sigaud & P. Gérard (Hg.), *Anticipatory behavior in adaptive learning systems: Foundations, theories, and systems* (Bd. 2684, S. 44-65). Berlin Heidelberg, Germany: Springer.
- Hoffmann, J., Butz, M. V., Herbort, O., Kiesel, A. & Lenhard, A. (2007). Spekulationen zur Struktur ideomotorischer Beziehungen [Speculations about the structure of ideomotor relations]. *Zeitschrift für Sportpsychologie*, 14(3), 95-103.
- Houk, J. C., Buckingham, J. T. & Barto, A. G. (1996). Models of the cerebellum and motor learning. *Behavioral and Brain Sciences*, 19(3), 368-383.

- Jackson, A., Mavoori, J. & Fetz, E. E. (2006). Long-term motor cortex plasticity induced by an electronic neural implant. *Nature*, *444*, 56-60.
- James, W. (1890). *The principles of psychology* (Bd. 1). New York: Holt.
- Jansson, E., Wilson, A. D., Williams, J. H. G. & Mon-Williams, M. (2007). Methodological problems undermine tests of the ideo-motor conjecture. *Experimental Brain Research*, *182*(4), 549-558.
- Jaric, S., Corcos, D. M. & Latash, M. L. (1992). Effects of practice on final position reproduction. *Experimental Brain Research*, *91*, 129-134.
- Jaric, S. & Latash, M. L. (1999). Learning a pointing task with a kinematically redundant limb: Emerging synergies and patterns of final position variability. *Human Movement Science*, *18*, 819-838.
- Jax, S. A. & Rosenbaum, D. A. (2007). Hand path priming in manual obstacle avoidance: Evidence that the dorsal stream does not only control visually guided actions in real time. *Journal of Experimental Psychology: Human Perception and Performance*, *33*(2), 425-441.
- Jordan, M. I. & Rumelhart, D. E. (1992). Forward models: Supervised learning with a distal teacher. *Cognitive Science*, *16*, 307-354.
- Jordan, M. I. & Wolpert, D. M. (1999). Computational motor control. In Gazzaniga (Hg.), *The cognitive neuroscience* (S. 601-620). Cambridge, MA: MIT Press.
- Takei, S., Hoffman, D. S. & Strick, P. L. (1999, September). Muscle and movement representations in the primary motor cortex. *Science*, *285*, 2136-2139.
- Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Transactions of the ASME-Journal of Basic Engineering*, *82*(Series D), 35-45.
- Kalveram, K. T. (2004). The inverse problem in cognitive, perceptual and proprioceptive control of sensorimotor behaviour: Towards a biologically plausible model of the control of aiming movements. *International Journal of Sport and Exercise*, *2*, 255-273.
- Karniel, A. (2002). Three creatures named forward model. *Neural Networks*, *15*(3), 305-307.
- Karniel, A. & Inbar, G. F. (1997). A model for learning human reaching movements. *Biological Cybernetics*, *77*, 173-183.
- Kawato, M. (1990). Advanced neural computers. In R. Eckmiller (Hg.), (S. 365-372). Amsterdam: North-Holland.
- Kawato, M. (1999). Internal models for motor control and trajectory planning. *Current Opinion in Neurobiology*, *9*, 718-727.
- Kawato, M., Furukawa, K. & Suzuki, R. (1987). A hierarchical neural-network model for control and learning of voluntary movement. *Biological Cybernetics*, *57*, 169-185.
- Kawato, M. & Gomi, H. (1992). A computational model of four regions of the cerebellum based on feedback-error learning. *Biological Cybernetics*, *68*(2), 95-103.
- Kawato, M., Maeda, M., Uno, Y. & Suzuki, R. (1990). Trajectory formation of arm movement by cascade neural-network model based on minimum torque change criterion. *Biological Cybernetics*, *57*, 169-185.
- Khan, M. A., Franks, I. M. & Goodman, D. (1998). The effect of practice on the control of rapid aiming movements: Evidence for an interdependency between programming and feedback processing. *The Quarterly Journal of Experimental Psychology Section A*, *51*(2), 425-443.
- Kindermann, T. & Cruse, H. (2002). MMC - a new numerical approach to the kinematics of complex manipulators. *Mechanism and Machine Theory*, *37*(4), 375-394.
- Knill, D. C. & Pouget, A. (2004). The bayesian brain: The role of uncertainty in neural coding and computation. *Trends in Neurosciences*, *27*(12), 712-719.
- Konczak, J., Borutta, M. & Dichgans, J. (1997). The development of goal-directed reaching in infants II. Learning to produce task-adequate patterns of joint torque. *Experimental Brain Research*, *113*, 465-474.

- Konczak, J. & Dichgans, J. (1997). The development toward stereotypic arm kinematics during reaching in the first 3 years of life. *Experimental Brain Research*, *117*, 346-354.
- Korman, M., Doyon, J., Doljansky, J., Carrier, J., Dagan, Y. & Karni, A. (2007). Daytime sleep condenses the time course of motor memory consolidation. *Nature Neuroscience*, *10*, 1206-1213.
- Kositsky, M. & Barto, A. G. (2002). The emergence of movement units through learning with noisy effrent signals and delayed sensory feedback. *Neurocomputing*, *44-46*, 889 - 895.
- Krakauer, J. W., Ghilardi, M.-F. & Ghez, C. (1999). Independent learning of internal models for kinematic and dynamic control of reaching. *Nature Neuroscience*, *2*(11), 1026-1031.
- Krakauer, J. W., Pine, Z. M., Ghilardi, M.-F. & Ghez, C. (2000). Learning of visuomotor transformations for vectorial planning of reaching trajectories. *The Journal of Neuroscience*, *20*(23), 8916-8924.
- Kühn, S. & Cruse, H. (2005, December). Static mental representations in recurrent neural networks for the control of dynamic behavioural sequences. *Connection Science*, *17*(3-4), 343-360.
- Kunde, W., Kiesel, A. & Hoffmann, J. (2003). Conscious control over the content of unconscious cognition. *Cognition*, *88*, 223-242.
- Kuperstein, M. (1988). Neural model of adaptive hand-eye coordination for single postures. *Science*, *239*, 1308-1311.
- Kuperstein, M. (1991). Infant neural controller for adaptive sensory-motor coordination. *Neural Netw.*, *4*(2), 131-145.
- Körding, K. P., Ku, S. & Wolpert, D. M. (2004). Bayesian integration in force estimation. *Journal of Neurophysiology*, *92*, 3161-3165.
- Körding, K. P., Tenenbaum, J. B. & Shadmehr, R. (2007). The dynamics of memory as a consequence of optimal adaptation to a changing body. *Nature Neuroscience*, *10*(6), 779-786.
- Körding, K. P. & Wolpert, D. M. (2004). Bayesian integration in sensorimotor learning. *Nature*, *427*, 244-247.
- Lackner, J. R. & DiZio, P. (1998). Adaptation in a rotating artificial gravity environment. *Brain Research Reviews*, *28*, 194-202.
- Latash, M. L., Scholz, J. F., Danion, F. & Schöner, G. (2001). Structure of motor variability in marginally redundant multifinger force production tasks. *Experimental Brain Research*, *141*, 153-165.
- Latash, M. L., Scholz, J. P. & Schöner, G. (2002). Motor control strategies revealed in the structure of motor variability. *Exercise & Sport Sciences Reviews*, *30*(1), 26-31.
- Lavrysen, A., Helsen, W. F., Tremblay, L., Elliott, D., Adam, J. J., Feys, P. et al. (2003). The control of sequential aiming movements: The influence of practice and manual asymmetries on the one-target advantage. *Cortex*, *39*, 307-325.
- Loeb, G. E., Brown, I. & Cheng, E. (1999). A hierarchical foundation for models of sensorimotor control. *Experimental Brain Research*, *126*, 1-18.
- Lotze, H. R. (1852). *Medicinische Psychologie oder Physiologie der Seele [Medical psychology or physiology of the soul]*. Leipzig: Weidmannsche Buchhandlung.
- Ludwig, D. A. (1982). Emg changes during the acquisition of a motor skill. *American Journal of Physical Medicine*, *61*(5), 229-43.
- Ma, W. J., Beck, J. M., Latham, P. E. & Pouget, A. (2006). Bayesian inference with probabilistic population codes. *Nature Neuroscience*, *9*, 11.
- Martinez, T. M., Berkovich, S. G. & Schulten, K. J. (1993). Neural-gas network for vector quantization and its application in time-series prediction. *IEEE Transactions on Neural Networks*, *4*(4), 558-569.

- Ma-Wyatt, A. & McKee, S. P. (2007). Visual information throughout a reach determines endpoint precision. *Experimental Brain Research*, 179(1), 55-64.
- McIntyre, J. & Bizzi, E. (1993). Servo hypotheses for the biological control of movement. *Journal of Motor Behavior*, 25(3), 193-202.
- Meulenbroek, R. G. J., Rosenbaum, D. A., Jansen, C., Vaughan, J. & Vogt, S. (2001). Multijoint grasping movements. Simulated and observed effects of object location, object size, and initial aperture. *Experimental Brain Research*, 138, 219-34.
- Miall, R. C., Weir, D. J., Wolpert, D. M. & Stein, J. F. (1993). Is the cerebellum a smith predictor? *Journal of Motor Behavior*, 25(3), 203-216.
- Milner, P. (2003). A brief history of the hebbian learning rule. *Canadian Psychology*, 44(1), 5-9.
- Morasso, P., Sanguineti, V. & Spada, G. (1997). A computational theory of targeting movements based on force fields and topology representing networks. *Neurocomputing*, 15(3-4), 411-434.
- Mosier, K. M., Scheidt, R. A., Acosta, S. & Mussa-Ivaldi, F. A. (2005). Remapping hand movements in a novel geometrical environment. *Journal of Neurophysiology*, 94, 4362-4372.
- Musallam, S., Corneil, B. D., Greger, B., Scherberger, H. & Andersen, R. A. (2004). Cognitive control signals for neural prosthetics. *Science*, 305(5681), 258 - 262.
- Mussa-Ivaldi, F. A. (1999). Modular features of motor control and learning. *Current Opinion in Neurobiology*, 9, 713-717.
- Mussa-Ivaldi, F. A. & Bizzi, E. (2000). Motor learning through the combination of primitives. *Philosophical Transactions of the Royal Society: Biological Sciences*, 355, 1755-1769.
- Mussa-Ivaldi, F. A., Giszter, S. F. & Bizzi, E. (1994). Linear combinations of primitives in vertebrate motor control. *Proceedings of the National Academy of Sciences*, 91, 7534-7538.
- Napier, J. (1980). *Hands*. New York: Pantheon Books.
- Ognibene, D., Mannella, F., Pezzulo, G. & Baldassarre, G. (2006). Integrating reinforcement-learning, accumulator models, and motor-primitives to study action selection and reaching in monkeys. In D. Fum, F. Del Missier & A. Stocco (Hg.), *Proceedings of the Seventh International Conference on Cognitive Modeling (ICCM2006)* (S. 214-219). Trieste: Edizioni Goliardiche.
- Ognibene, D., Rega, A. & Baldassarre, G. (2006). A model of reaching integrating continuous reinforcement learning, accumulator models, and direct inverse modelling. In S. Nolfi et al. (Hg.), *From Animals to Animats 9: The Ninth International Conference on the Simulation of Adaptive Behavior* (S. 381-393). Berlin: Springer Verlag.
- Ostry, D. J. & Feldman, A. G. (2003). A critical evaluation of the force control hypothesis in motor control. *Experimental Brain Research*, 153, 275-288.
- Polit, A. & Bizzi, E. (1979). Characteristics of motor programs underlying arm movements in monkeys. *Journal of Neurophysiol*, 42, 183-194.
- Powers, W. T. (1973). *Behavior: The control of perception*. New York: Aldine de Gruyter.
- Proteau, L., Marteniuk, R. G., Girouard, Y. & Dugas, C. (1987). On the type of information used to control and learn an aiming movement after moderate and extensive training. *Human Movement Science*, 6, 181-199.
- Redgrave, P., Prescott, T. J. & Gurney, K. (1999). The basal ganglia: a vertebrate solution to the selection problem? *Neuroscience*, 89, 1009-1021.
- Rizzolatti, G., Fadiga, L., Fogassi, L. & Gallese, V. (1997). Enhanced: The space around us. *Science*, 277, 190-191.
- Robertson, E. M. & Miall, R. C. (1997). Multi-joint limbs permit a flexible response to unpredictable events. *Experimental Brain Research*, 117, 148-152.
- Rokni, U., Richardson, A. G., Bizzi, E. & Seung, H. S. (2007). Motor learning with unstable neural representations. *Neuron*, 54(4), 653-666.

- Rosenbaum, D. A. (1980). Human movement initiation: Specification of arm, direction and extent. *Journal of Experimental Psychology: General*, *109*, 444-474.
- Rosenbaum, D. A., Engelbrecht, S. E., Bushe, M. M. & Loukopoulos, L. D. (1993a). Knowledge model for selecting and producing reaching movements. *Journal of Motor Behavior*, *25*, 217-227.
- Rosenbaum, D. A., Engelbrecht, S. E., Bushe, M. M. & Loukopoulos, L. D. (1993b). A model for reaching control. *Acta Psychologica*, *82*(1-3), 237-50.
- Rosenbaum, D. A., Loukopoulos, L. D., Meulenbroek, R. G. J., Vaughan, J. & Engelbrecht, S. E. (1995). Planning reaches by evaluating stored postures. *Psychological Review*, *102*(1), 28-67.
- Rosenbaum, D. A., Marchak, F., Barnes, H. J., Vaughan, J. than, Siotta, J. D. & Jorgensen, M. J. (1990). Constraints for action selection: Overhand versus underhand grips. In M. Jeannerod (Hg.), *Attention and performance* (Bd. XIII, S. 321-345). Hillsdale, New Jersey, Hove and London: LAWRENCE ERLBAUM ASSOCIATES.
- Rosenbaum, D. A., Meulenbroek, R. G. J. & Vaughan, J. (2001). Planning reaching and grasping movements: Theoretical outlines and practical implications. *Motor Control*, *5*(2), 99-115.
- Rosenbaum, D. A., Meulenbroek, R. G. J., Vaughan, J. & Jansen, C. (1999). Coordination of reaching and grasping by capitalizing on obstacle avoidance and other constraints. *Experimental Brain Research*, *128*, 92-100.
- Rosenbaum, D. A., Meulenbroek, R. G. J., Vaughan, J. & Jansen, C. (2001). Posture-based motion planning: Applications to grasping. *Psychological Review*, *108*(4), 709-734.
- Sabes, P. N. (2000). The planning and control of reaching movements. *Current Opinion in Neurobiology*, *10*(6), 740-746.
- Scherberger, H. & Andersen, R. A. (2007). Target selection signals for arm reaching in the posterior parietal cortex. *The Journal of Neuroscience*, *27*(8), 2001-2012.
- Scherberger, H., Jarvis, M. R. & Andersen, R. A. (2005). Cortical local field potential encodes movement intentions in the posterior parietal cortex. *Neuron*, *46*, 347-354.
- Schmidt, R. A. (1975). A schema theory of discrete motor skill-learning. *Psychological Review*, *82*(4), 229-261.
- Schmidt, R. A. (1988). *Motor control and learning - a behavioral emphasize* (2nd Aufl.). Champaign, IL: Human Kinetics Publishers.
- Schmidt, T. (2002). The finger in flight: Real-time motor control by visually masked color stimuli. *Psychological Science*, *13*(2), 112-118.
- Scholz, J. P., Schöner, G. & Latash, M. L. (2000). Identifying the control structure of multijoint coordination during pistol shooting. *Experimental Brain Research*, *135*, 382-404.
- Schubotz, R. I. (2007). Prediction of external events with our motor system: Towards a new framework. *Trends in Cognitive Sciences*, *11*(5), 211-218.
- Schwartz, A. B., Moran, D. W. & Reina, G. A. (2004). Differential representation of perception and action in the frontal cortex. *Science*, *303*, 380-383.
- Schweighofer, N., Arbib, M. A. & Kawato, M. (1998). Role of the cerebellum in reaching movements in humans. I. Distributed inverse dynamics control. *European Journal of Neuroscience*, *10*, 86-94.
- Schweighofer, N., Spolstra, J., Arbib, M. A. & Kawato, M. (1998). Role of the cerebellum in reaching movements in humans. II. A neural model of the intermediate cerebellum. *European Journal of Neuroscience*, *10*(1), 95-105.
- Scott, S. H. (2004). Optimal feedback control and the neural basis of volitional motor control. *Nature Reviews Neuroscience*, *5*, 532-544.
- Scott, S. H. & Kalaska, J. F. (1997). Reaching movements with similar hand paths but different

- arm orientations. I. Activity of individual cells in motor cortex. *Journal of Neurophysiology*, 77, 826-852.
- Scott, S. H. & Norman, K. E. (2003). Computational approaches to motor control and their potential role for interpreting motor dysfunction. *Current Opinion in Neurology*, 16(6), 693-698.
- Shadmehr, R. & Brashers-Krug, T. (1997, Jan). Functional stages in the formation of human long term motor memory. *Journal of Neuroscience*, 17(1), 409-419.
- Shadmehr, R. & Wise, S. P. (2005). *The computational neurobiology of reaching and pointing: A foundation for motor learning*. Cambridge, MA: MIT Press.
- Shima, K., Isoda, M., Mushiake, H. & Tanji, J. (2007). Categorization of behavioural sequences in the prefrontal cortex. *Nature*, 445, 315-318.
- Soechting, J. F., Buneo, C. A., Herrmann, U. & Flanders, M. (1995). Moving effortlessly in three dimensions: Does Donders' law apply to arm movement? *Journal of Neuroscience*, 15, 6271-6280.
- Song, J.-H. & Nakayama, K. (2006). Role of focal attention on latencies and trajectories of visually guided manual pointing. *Journal of Vision*, 6(9), 982-995.
- Spijkers, W. & Spellerberg, S. (1995). On-line visual control of aiming movements? *Acta Psychologica*, 90(1-3), 333-348.
- Srinivasa, N. & Grossberg, S. (2007). A self-organizing neural model for fault-tolerant control of redundant robots. *Proceedings of the International Joint Conference on Neural Networks*, 20, 483-488.
- Steinkühler, U., Beyn, W.-J. & Cruse, H. (1995). A simplified MMC model for the control of an arm with redundant degrees of freedom. *Neural Processing Letters*, 2(4), 11-15.
- Stock, A. & Hoffmann, J. (2002). Intentional fixation of behavioral learning or how R-E learning blocks S-R learning. *European Journal of Cognitive Psychology*, 14(1), 127-153.
- Stock, A. & Stock, C. (2004). A short history of ideo-motor action. *Psychological Research*, 68(2/3), 176-188.
- Stringer, S. M., Rolls, E. & Taylor, P. (2007). Learning movement sequences with a delayed reward signal in a hierarchical model of motor function. *Neural Networks*, 20(2), 172-181.
- Stringer, S. M., Rolls, E. T., Trappenberg, T. P. & de Araujo, I. E. T. (2003). Self-organizing continuous attractor networks and motor function. *Neural Networks*, 16, 161-182.
- Stroeve, S. (1996). Linear combined feedback and feedforward control of a musculoskeletal system. *Biological Cybernetics*, 75, 73-83.
- Stroeve, S. (1997). A learning feedback and feedforward neuromuscular control model for two degrees of freedom human arm movements. *Human Movement Science*, 16, 621-651.
- Sutton, R. S. (1990). Integrated architectures for learning, planning, and reacting based on approximating dynamic programming. *Proceedings of the Seventh International Conference on Machine Learning*, 216-224.
- Sutton, R. S. & Barto, A. G. (1998). *Reinforcement learning: an introduction*. Cambridge, MA: MIT Press.
- Thelen, E., Corbetta, D., Kamm, K., Spencer, J. P., Schneider, K. & Zernicke, R. F. (1993). The transition to reaching: Mapping intention and intrinsic dynamics. *Child Development*, 64(4), 1058-1098.
- Todorov, E. (2000). Direct cortical control of muscle activation in voluntary arm movements: A model. *Nature*, 3(4), 391-398.
- Todorov, E. (2004). Optimality principles in sensorimotor control. *Nature Review Neuroscience*, 7(9), 907-915.
- Todorov, E. & Jordan, M. I. (2002). Optimal feedback control as a theory of motor coordination. *Nature Neuroscience*, 5(11), 1226-1235.

- Torres, E. B. & Zipser, D. (2002). Reaching to grasp with a multi-jointed arm. I. Computational model. *Journal of Neurophysiology*, *88*, 2355-2367.
- Toussaint, M. (2006). A sensorimotor map: Modulating lateral interactions for anticipation and planning. *Neural Computation*, *18*, 1132-1155.
- Trommershäuser, J., Maloney, L. T. & Landy, M. S. (2003). Statistical decision theory and trade-offs in the control of motor response. *Spatial Vision*, *16*(3-4), 255-275.
- Trommershäuser, J., Landy, M. S. & Maloney, L. T. (2006). Humans rapidly estimate expected gain in movement planning. *Psychological Science*, *17*, 981-988.
- Uno, Y., Kawato, M. & Suzuki, R. (1989). Formation and control of optimal trajectory in human multijoint arm movement: Minimum torque-change model. *Biological Cybernetics*, *61*(2), 89-101.
- van den Dobbelaert, J. J., Brenner, E. & Smeets, J. B. J. (2001). Endpoints of arm movements to visual targets. *Experimental Brain Research*, *138*(3), 279-287.
- van Sinderen, J. F., Denier van der Gon, J. J. & Gielen, C. C. A. M. (1988). Conditions determining early modification of motor programmes in response to changes in target location. *Experimental Brain Research*, *71*(2), 320-328.
- van Sinderen, J. F. & Denier van der Gon, J. J. (1991). Reaction-time-dependent differences in the initial movement direction of fast goal-directed arm movements. *Human Movement Science*, *10*(6), 713-726.
- von Hofsten, C. (2003). On the development of perception and action. In J. Valsiner & K. J. Connolly (Hg.), *Handbook of developmental psychology* (S. 114-140). London: Sage.
- von Hofsten, C. (2004). An action perspective on motor development. *Trends in Cognitive Science*, *8*(6), 266-272.
- Vorberg, D., Mattler, U., Heinecke, A., Schmidt, T. & Schwarzbach, J. (2003). Different time courses for visual perception and action priming. *Proceedings of the National Academy of Sciences of the United States of America*, *100*, 6275-6280.
- Wang, J. & Sainburg, R. L. (2005). Adaptation to visuomotor rotations remaps movement vectors, not final positions. *The Journal of Neuroscience*, *25*(16), 4024-4030.
- Weigelt, M., Cohen, R. & Rosenbaum, D. A. (2006). Returning home: Location memory versus posture memory in object manipulation. *Experimental Brain Research*.
- Weigelt, M., Kunde, W. & Prinz, W. (2006). End-state comfort in bimanual object manipulation. *Experimental Psychology*, *53*(2), 143-148.
- Wolpert, D. M., Doya, K. & Kawato, M. (2003). A unifying computational framework for motor control and social interaction. *Philosophical Transactions of the Royal Society of London*, *358*, 593-602.
- Wolpert, D. M., Ghahramani, Z. & Flanagan, J. R. (2001). Perspectives and problems in motor learning. *Trends in Cognitive Sciences*, *5*, 487-494.
- Wolpert, D. M. & Kawato, M. (1998). Multiple paired forward and inverse models for motor control. *Neural Networks*, *11*(7-8), 1317-1329.
- Woodworth, R. S. (1899). *The accuracy of voluntary movement*. Lancaster, PA: New Era Printing Company.
- Zhang, W., Zatsiorsky, V. M. & Latash, M. L. (2007). Finger synergies during multi-finger cyclic production of moment of force. *Experimental Brain Research*, *177*(2), 243-254.

Appendix

Parameter Specifications

General Parameters

In section *Simulation of Human Behavior* (page 48), the following parameters were used to evaluate the model, if not mentioned otherwise in the text: $\theta = 0.10$; $\rho = 0.10$; In the motor controller, δ decayed exponentially from 0.1 in the first time step to 0.01 in the 1,000,000th time step; The learning rate in posture memory was set to $\epsilon = 0.001$; For the generation of the sensory-to-motor mappings the following factors were used: $\gamma = .434$, $\beta = .172$.

Anticipatory Posture Adjustment

In section *Anticipatory Posture Adjustment* (page 57) a different model setup was used. The three joint planar arm approximates the kinematic features of a human arm that is restricted to the transverse plane. The lengths of the upper arm, forearm, and hand are $l_1 = 32\text{cm}$, $l_2 = 25\text{cm}$ and $l_3 = 18\text{cm}$, respectively. The shoulder, elbow and wrist joints are allowed to move within $-60^\circ \leq \phi_1 \leq 120^\circ$, $-160^\circ \leq \phi_2 \leq 0^\circ$, and $-80^\circ \leq \phi_3 \leq 60^\circ$, respectively.

Each neuron h_i of the hand space representation \vec{h} fires if the hand coordinates (x, y) are close enough to the neuron's preferred hand location (h_i^x, h_i^y) :

$$h_i = \max(1.0 - \frac{|x - h_i^x|}{3.0}; 0) \cdot \max(1.0 - \frac{|y - h_i^y|}{3.0}; 0)$$

The preferred hand locations are arranged in a $51 \times 26 = 1326$ grid with 3cm distance, covering a $150\text{cm} \times 75\text{cm}$ rectangle, which covers the upper half of the arm's work space. The shoulder joint is centered on the lower line of this rectangle. Arm postures are encoded in a similar population of neurons \vec{p} , where each neuron p_i is activated according to the following equation:

$$p_i = \prod_{j=1}^3 \max(1.0 - \frac{|\phi_j - p_i^{\phi_j}|}{20.0^\circ}; 0),$$

where $p_i^{\phi_j}$ are the preferred joint angles of each neuron p_i , which are arranged in a $10 \times 9 \times 8 = 720$ grid covering the entire posture space. The distance between two adjacent neurons is 20° .

Only $n = 6$ action columns are used, two actuating each joint. Motor command activation is normalized (1-norm) so that the arm moves with a gain of $g = 2.25^\circ$ in posture space, in each time step. During motor learning, a new set of motor commands was generated by setting each motor command to 1.0 with a probability of $p = 0.3$ and to 0.0 otherwise, in random intervals of 1 to 8 time steps.

Obstacle Avoidance

In the first obstacle avoidance task (page 61), neurons with preferred values within the rectangle with top left corner at $(-2.4, 0.8)$ and bottom right corner at $(-0.8, -0.8)$ were considered part of the left obstacle and inhibitory activations of these neurons were set to 1.0. The corners of the rectangle representing the right obstacle were $(0.8, 0.8)$ and $(2.4, -0.8)$. In the second task, state neurons with preferred values that had a y-component of at least 1.0 were considered part of the ceiling obstacle. The obstacle avoidance tasks were performed with the same 10 controllers that were used in the Constrained Joint-Angle SpaceSection.

Simulation Code

A demo program, the simulation tools, the running code and the source code of both, documentation and syntax files for SPSS (14) can be found on the web at:

http://www.psychologie.uni-wuerzburg.de/i3pages/SURE_REACH/

Simulation Systems

The model is implemented in JAVA 1.6. The experiments were run on a standard office computers (Pentium 4 HT 3.2GHz and Intel Core 2 Duo). Training and evaluating a single controller took approximately 25 minutes.