UNIVERSITY OF CALIFORNIA, SAN DIEGO

Neural Network Models of the Brain Mechanisms of Bilateral Coordination

A dissertation submitted in partial satisfaction of the
requirements for the degree Doctor of Philosophy in

Cognitive Science

by

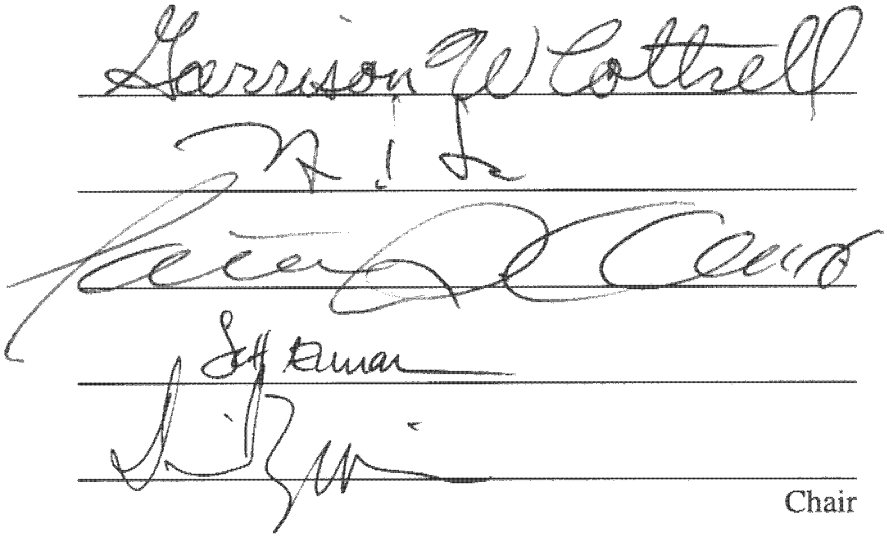
David Scott Farrar

Committee in charge:

Professor David Zipser, Chair
Professor Patricia Churchland
Professor Garrison Cottrell
Professor Jeffrey Elman
Professor Martin Sereno

1999

The dissertation of David Scott Farrar is approved, and it is acceptable in quality and form for publication on microfilm:



Chair

University of California, San Diego

1999

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I thank Jude Mitchell, Elizabeth Torres and all the members of my committee who were always willing to listen and give feedback. I thank L. Peter Deutsch, Mark Scott Johnson, Jim Weinrich, Don Barrett, and David Richey for being my “bear committee”. I thank Stefan Heck for being one of my role models. I thank my family for their love and encouragement. And I thank Cameron Robb, without whose love, support, and understanding this work would not have been completed.

VITA

February 18, 1970	Born, Stockton, California
1992	B.S., Stanford University
1992-1995	Research Fellow, University of California, San Diego
1994	M.S., University of California, San Diego
1993, 1995-1996, 1999	Teaching Assistant, University of California, San Diego
1995-1998	Research Assistant, University of California, San Diego
1999	Ph.D., University of California, San Diego

PUBLICATIONS

Farrar DS, Zipser D. (1999) Neural network models of bilateral coordination. *Biological Cybernetics*. (in press).

ABSTRACT OF THE DISSERTATION

Neural Network Models of the Brain Mechanisms of Bilateral Coordination

by

David Scott Farrar

Doctor of Philosophy in Cognitive Science

University of California, San Diego, 1999

Professor David Zipser, Chair

This thesis couples analytical techniques from theoretical robotics with neural network modeling to investigate possible brain mechanisms underlying the coordination of bilateral movements. It presents several neural network models that are biologically plausible candidate mechanisms for the neural computations underlying coordination. The models provide a connectionist account of how the brain might accomplish coordinated bilateral movements.

The thesis begins by describing several mechanisms for controlling the movement of a pair of arms. The first is an engineered motion planner that finds solutions to the ill-posed problem of making non-colliding, goal-directed movements. The planner exploits a representational tool that does not require a forced decomposition of coordination problems. The second uses feedforward neural networks that learn to emulate the coordinated behaviors of the motion planner using considerably less computational resources. Knowledge about the structure of the body necessary to solve the coordination problems is learned by the neural networks, and comes to be represented in the internal weights. The third uses biologically inspired recurrent network architectures to more closely approximate the kinds of neural phenomena seen in the brain. Analysis of the networks shows in general terms how they work, and allows us to make testable predictions about some of the response properties that might be observed in the brain systems serving bilateral coordination.

Chapter 1 : Introduction

Coordinated movements like scratching an itch on one arm with the opposite hand require the brain to solve many complicated computational problems. These behaviors are hard to plan because the arms must avoid hitting each other and the body even though their locations are constantly changing. The computational problems are ill-posed because there is generally no unique way to accomplish the movements. Coordinated movements have been extensively studied by the robotics community, but have received much less attention in neurobiology. Although the elegant concepts developed for the coordination of robots are not directly transferable to the nervous system, they can serve as a starting point for generating biologically plausible models. The approach described here is to first borrow techniques from robotics to implement motion planners that simulate two coordinated behaviors. The first behavior is a relatively simple posture-to-posture task; the second is a more challenging self-touching task analogous to scratching an itch. These planners are then used to generate the data needed to train neural networks to simulate the same behaviors. This allows us to characterize the computational problem using a robotic formalism, and find possible neural implementations by analyzing the networks. The rationale for this approach is that neural networks trained to do computations analogous to those performed by the brain often find solutions that are strikingly similar to those actually observed in neurophysiological experiments (Zipser, 1992).

A variety of strategies have been used for planning the coordinated motion of two robotic arms in a shared workspace. The planner described in this thesis is based on a variation of the gradient descent method, defined over a multidimensional configuration

space (Barraquand and Latombe, 1991; Latombe, 1991). In this elegant framework, the complete posture of the robot is represented by a single point in a high-dimensional configuration space composed of variables such as joint angles. The elegance of this method lies in the fact that the movement of a single point in configuration space represents the simultaneous movement of both arms in their workspace. Motion planning is reduced to finding a continuous path through this space from a starting point to a goal point; the path must avoid configuration space obstacles, which represent colliding postures. Motion planners based on this representation have been applied to a variety of complicated coordination problems, including ill-posed problems with many redundant degrees of freedom (Kavraki, et. al. 1996), tasks requiring navigation through cluttered workspaces (Barraquand and Latombe, 1991), and tasks requiring multi-arm manipulation of workspace objects (Koga and Latombe, 1994).

Ill-posed problems can be solved only if additional constraints are used. One sufficient constraint is to follow the path traced out by gradient descent in an appropriately chosen potential function in configuration space. The potential function must have a gradient that flows along a unique, non-colliding path to the goal. The planner described in Chapter 2 uses such a potential function to navigate through the complex shape of the space and resolve the ill-posedness of the posture-to-posture and self-touching tasks.

1.1 Previous Modeling Efforts

The psychophysical and neurobehavioral study of coordination has often focused on rhythmic movements. Turvey (1990), for example, suggests that the evaluation of rhythmic movements should be the starting point of any investigation into the

coordination of the limbs. Rhythmic movements are an important kind of coordination, but many voluntary coordinated movements, such as bilateral reaching, are not rhythmic.

Few biological models of the mechanisms of non-rhythmic coordination exist. (Nussbaum, et. al. 1997) propose a neural network model of torso muscle coordination that relies on known muscle equilibrium constraints for optimization. The connections in their network are hand-tuned to match recorded electromyographic data. Other previous plausible neural models of coordination have been restricted to rhythmic bilateral movements where the limbs do not spatially interact. For example, a neural network that acts as a central pattern generator for the coordination of rhythmic bilateral movements has been described previously (Grossberg, et. al., 1997). A biologically inspired, distributed neural network controller that coordinates the limbs of a six-legged robot for a variety of stable gaits has also been constructed (Beer, et. al., 1992). Although such systems are often adaptable to a wide range of repetitive movements, they do not address the coordination of arbitrary, goal-directed, spatially interacting movements.

Several engineered solutions to the problem of bilateral coordination and collision avoidance already exist. Typically, the coordination problem is decomposed into two separate arm control problems, where the trajectory for one arm is planned before the other. A higher level planner adjusts the trajectories to avoid collisions by using task-specific knowledge – represented by a table of rules (Cui and Shin, 1996) or a prioritized list of processes (Li and Latombe, 1997). The planner uses information about the structure of the body and the workspace to constrain the choice of possible movements.

1.2 Subsumption Architectures

The organizational principles apparent in the nervous system and those used in nontraditional robotics are strikingly parallel. In the subsumption architecture (Brooks,

1986), a mobile robot's control system is decomposed into a set of task-achieving behaviors that operate in parallel. Behaviors are layered on "top" of one another to increase the complexity of tasks that can be accomplished. When conflicts between behaviors arise, some layers will suppress or subsume the output of others. A surprising range of complex, goal-directed behaviors can emerge from the architecture (Brooks, 1986; Brooks, 1990).

The subsumption architecture echoes the layering of functionality that is evident through the evolutionary history of most of the nervous system (Brooks, 1991). Phylogenetically younger structures, such as the cortex and the cerebellum, work in parallel with yet subsume behaviors accomplished by older structures such as the basal ganglia and the spinal cord. These older structures provide a set of elementary behaviors that can then be adapted and exploited by the evolutionary process to create increasing levels of behavioral competency. The organizational principles apparent in the nervous system reappear in this engineering methodology.

1.3 Neurophysiological Background

The organization of the spinal cord reflects a functional representation of the musculoskeletal architecture (Nichols, 1994). Motoneuronal regions representing muscles that participate in common joint action are organized into synergistic groups with mutual excitatory connections. Regions representing muscles with opposing postural actions are linked with mutually inhibitory connections. Similar excitatory and inhibitory connections are observed between spinal regions that control distinct joints. This internal connectivity reduces the number of degrees of freedom in the system, facilitating interjoint coordination for behaviors such as locomotion or forelimb reaching. The spinal cord thus reflects a behavior-based organization that embodies "knowledge"

about what muscle groups must cooperate, and how they must cooperate, to achieve action.

Bilateral rhythm generating networks of neurons have been identified in several species. The turtle spinal cord contains central pattern generators for a variety of behaviors, including fictive rostral scratching (Stein, et. al. 1995). The normal performance of this behavior is dependent on both contralateral and ipsilateral spinal circuits, although components of the behaviors can be obtained from stimulation of the ipsilateral side only. The circuits are capable of a limited behavioral repertoire, including 1:1 and 2:1 phase-locked interlimb coordination (Stein and McCullough, 1988).

In rats (Kjaelruff and Kiehn, 1996), rhythmogenic neurons are distributed both transversely and longitudinally down the spinal cord. Fiber pathways serving left/right coordination appear to be localized exclusively to the ventral commissure of the spinal cord. These neurons and fiber pathways form the basis for multiple central pattern generators used in locomotion and other automatic behaviors.

Spinalized cats are capable of a wide variety of rhythmic interlimb movements, including a variety of gaits with unequal frequencies between the limbs (Forssberg, et. al. 1980). Although there may be separate spinal neural control mechanisms for each limb, neural activity driving one limb influences activity driving the other. The authors note that the movement behaviors of the spinalized cat are extremely robust to perturbations, and they suggest that the spinal circuits provide a basic motor vocabulary for locomotion that could be controlled by supraspinal circuits.

Although the spinal cord provides a basic motor vocabulary for movements requiring automatic coordination of multiple joints, arbitrary coordinated movements probably require a more complicated neural computing architecture than is available in the spinal cord. Behavioral deficits of patients and animals with lesions to the motor cortex support this view. In monkeys, the unilateral lesion of the supplementary motor

area of the cortex (SMA) provokes a characteristic deficit where both arms tend to move identically, rather than share the task between them (Brinkman, 1984). Sectioning of the corpus callosum immediately abolishes this deficit. This finding suggests that coordination is not achieved via a simple addition or merging of neural control from both hemispheres. Lesions of the premotor cortex impair the ability of monkeys to reach around obstacles to a visible food reward (Moll and Kuypers, 1977). The monkeys tend to make straight reaching movements that collide with a transparent barrier, suggesting that the lesion has disrupted a neural system that is important for collision avoidance (Diamond, 1990). Human patients with unilateral damage to SMA demonstrate similar deficits such as mirror-writing or the substitution of bilateral mirror movements for non-mirror movements (Chan and Ross, 1988), and intermanual conflict (McNabb, et. al. 1988).

There is evidence that ipsilateral brain systems are involved in the preparation and execution of both unilateral and bilateral movements (Salmelin, et. al. 1995). Magnetoencephalographic (MEG) recordings during unilateral and coordinated bilateral finger flexion movements reveal ipsilateral movement evoked fields (MEFs) that are stronger for ipsilateral movement than for contralateral movement. Activity of the ipsilateral somatomotor cortex is enhanced with increasing complexity of unilateral movement. This suggests that complicated movements – such as those requiring bilateral coordination – are not planned and executed in one “central” area, but rather involve the cooperative engagement of the sensorimotor cortices of both hemispheres.

There is increasing evidence that the actual neural mechanisms of bilateral coordination are distributed across many brain structures (Liu, et. al. 1997; Wiesendanger, et. al. 1996). Recent neurophysiological studies have begun to probe these neural systems. For example, in primary motor cortex (M1) and SMA a majority of neurons are bilateral. But for a large fraction of these neurons, activities during bilateral

movements are not predictable on the basis of the activities during unilateral movements (Kermadi, et. al. 1997; Donchin, et. al. 1997). For example, (Donchin et. al., 1998) have recorded from M1 and SMA in monkeys performing both unilateral and bilateral reaching movements (Figure 1). Their monkeys were trained to move frictionless manipulanda to lighted targets in a bilateral paradigm similar to Georgopoulos's original reaching experiments (Georgopoulos, et. al. 1982). They found substantial numbers of neurons that participate in unilateral movements of either side. Many neurons also participate in bilateral movements, i.e. neuron responses included a statistically significant bilateral component. They also found neurons selective exclusively for bilateral movements. They note that there seems to be little difference in the distribution of selective neurons between SMA and M1. Since they found bilaterally-specific neurons equally present in M1 and SMA, these researchers suggest that SMA and M1 share the problem of coordinating bilateral movements. This seems reasonable since both M1 and SMA have substantial and comparable projections to the corticospinal tract (Dum and Strick, 1991; He, et. al. 1993).

1.4 Commissurotomy Studies

The functional organization of the callosal fiber bundle resembles that of the cerebral hemispheres. The posterior (splenial) region that connects the occipital lobes conveys visual information (Gazzaniga, 1995), while the mid-posterior portion of the corpus callosum conveys tactile information (Preilowski, 1975; Lutsep, et. al. 1995). There is also evidence that the anterior connections tend to convey "higher order" semantic information that is not purely sensory in nature (Gazzaniga, 1995). Based on these observations, we might expect that the mid-posterior to mid-anterior region of the callosum would be most important for conveying ipsilateral information necessary for

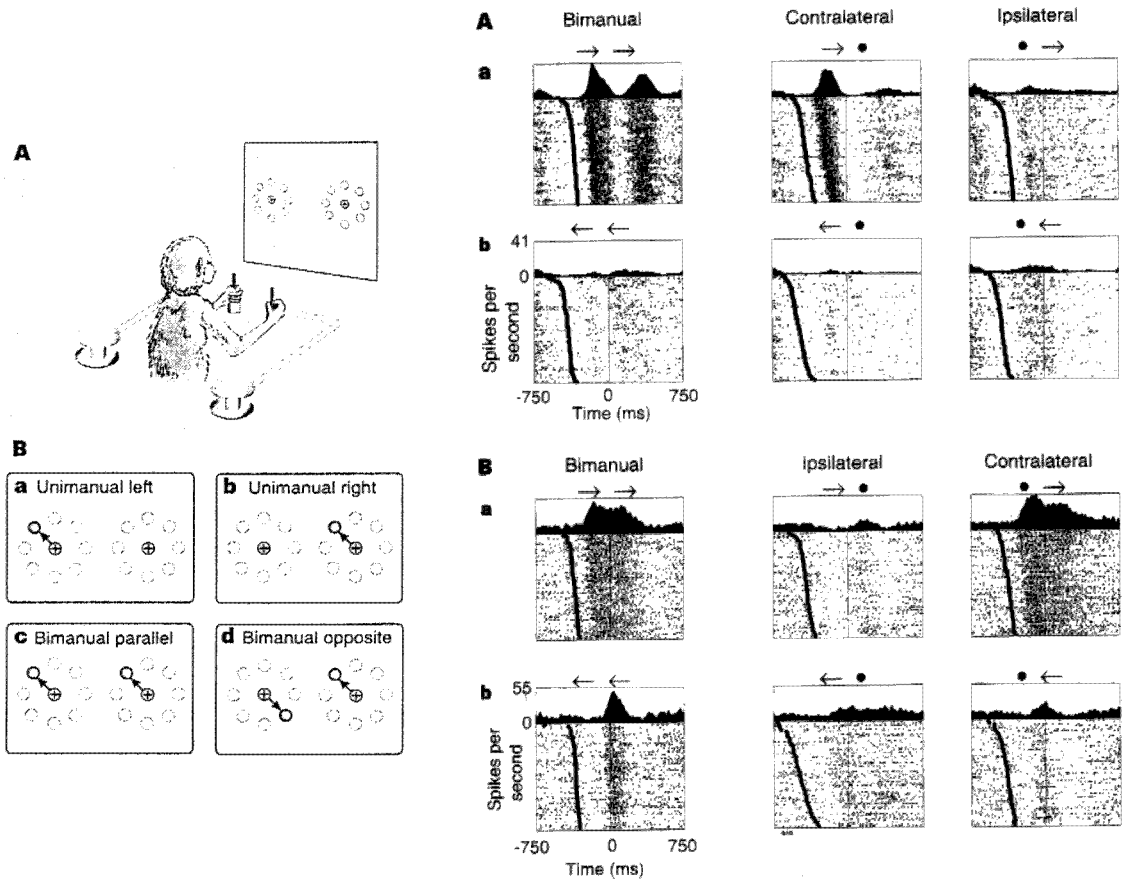


Figure 1. Bilateral movement experiment. (Left) Monkeys were trained to move frictionless manipulanda in a bilateral paradigm analogous to Geogopoulous's one-arm reaching experiments. **A** The monkey and its manipulanda. **B** The four movement conditions the monkey was trained to perform. (Right) Neurons in primary motor cortex and supplementary motor cortex participate in both unilateral and bilateral movements. The top of each box contains a histogram plot of a neuron's activity during one type of movement, with individual raster traces of neuron spikes below, sorted in order of movement onset. The direction of movement of each arm is depicted by an arrow at the top of the box, or by a dot if the arm does not move. **A** The first neuron has a component that is almost exclusively bilateral. **B** The second neuron participates in both unilateral and bilateral movements. (adapted from Donchin, et. al. 1998, without permission).

distal sensorimotor integration and bimanual coordination. Because commissurotomy patients are usually capable of proximal coordination tasks, ipsilateral information necessary for the coordination of these movements is probably carried by subcortical connections (Pashler, et. al. 1994).

Studies of patients who have undergone sectioning of part or all of the corpus callosum and anterior commissure demonstrate a variety of subtle sensorimotor deficits. When information is presented to only one hemisphere, the other hemisphere cannot use this information for perceptual analysis. Sensory integration between the hemispheres becomes difficult: patients are typically unable to integrate visual information between the two visual hemifields, for example. They have difficulty transferring hand posture from one hand to the other (Chen, et. al. 1990). Both hemispheres are able to sense and indicate the location of a simple somatosensory stimulus on either side of the body. But when an object is held in one hand, the commussurotomy patient is often unable to indicate an identical object with the other hand (Preilowski, 1975; Watson and Heilman, 1983; Gazzaniga, 1995).

Disconnection of the cerebral hemispheres also results in a number of subtle motor deficits. Each hemisphere is unable to control movements of the ipsilateral hand by itself. However, either hemisphere can control movements of the arms, torso, and legs (Gazzaniga, 1995). Patients have difficulty performing coordinated bilateral drawing tasks (Jeeves, et. al. 1988a). Well practiced bilateral tasks are preserved with commissurotomy, but simple novel tasks such as coordinated rhythmic finger tapping become difficult to learn (Zaidel and Sperry, 1977). This suggests that commissurotomy may primarily affect fine motor coordination between the fingers and the hands.

Another effect of complete commissurotomy is transient intermanual conflict: one hand grasps an object but the other fails to release it; one hand buttons a shirt, while the other follows along and unbuttons it; one hand repeatedly picks up a newspaper, but

the other puts it down (Preilowski, 1975; Watson and Heilman, 1983). Preilowski suggests that bimanual interference can be explained in terms of ipsilateral efferent influences. Ipsilateral innervation from one hemisphere is normally overridden by contralateral outflow from the opposite hemisphere. If inhibitory control between the hemispheres is lost, conflict between the motor outputs of each may result, especially for asymmetric (e.g. parallel) movements (Preilowski, 1972; Preilowski, 1975).

Commissurotomy may also deny motor areas from access to the hemisphere that normally specializes in a given task (for example movement of the left hand to verbal command, or spatial tracing with the right hand), resulting in a unilateral deficit in performance. Complete commissurotomy patients are sometimes unable to make left-hand movements independently or via verbal command. But after a post-operative recovery period these patients are capable of a variety of spontaneous, well coordinated movements such as scratching parts of their body or pulling up their bedcovers (Preilowski, 1975; Watson and Heilman, 1983).

1.5 Studies of Callosal Agenesis

The effects of hemispheric disconnection can also be studied in patients born without normal callosal connections (callosal agenesis). (Jeeves and Silver, 1988) administered a visual and tactile rod-bending task to a group of commissurotomy and acallosal patients. The task requires the subject to bend the angle of a jointed rod with one hand to match that of an example, with is presented either visually or by touch to the same hand. The task relies on the right hemispheric specialization for spatial analysis: performance in both the left and right hand conditions depends on access to the right hemisphere. They found that the right hand performance of commissurotomy patients was impaired, particularly if the caudal portions of the callosum were sectioned; the

acallosal patients were mostly unimpaired. Their interpretation is that the acallosal patients have developed ipsilateral channels for conveying tactile information from the right hand to the right hemisphere, in addition to compensatory strategies for solving the task. (Reynolds and Jeeves, 1977) found the performance of acallosal patients to be indistinguishable on most behavioral tasks. They did find a young (age 12) acallosal patient who exhibited deficits in cross-localization of light touch stimuli, and who was significantly slower on bimanual pegboard and string winding tasks.

(Jeeves, et. al. 1988b) tested the performance of commissurotomy patients, acallosal patients, normal 6 year old children, and adult controls on a bimanual X-Y plotter tracing task with visual feedback. They found that although the adult controls quickly became skilled in the task, and performed well even in the sudden absence of visual feedback, the commissurotomy and acallosal patients were impaired when visual feedback was removed (Jeeves, et. al. 1988b; Jeeves, et. al 1988a). They note that the performance of the normal children was comparable to that of the acallosal group, and suggest that this may be due to the reduced efficiency of the callosal connection before myelination is complete at age 10 (Jeeves, et. al. 1988b).

1.6 Goals of Research

The ultimate goal of this research is to develop plausible models of the neural computations necessary to support coordination problems. The strategy pursued here is to build simple neural network models, and then increase their complexity in biologically inspired ways. One way to improve the biological plausibility of neural network identification models is to incorporate architectural features known to exist in the brain. Since the brain has a clear bilateral architecture and numerous recurrent connections, these details can be included in neural network models. Recurrent models are more

plausible because they approximate the brain's dynamic representations. The models can be examined in ways analogous to the way neural systems are studied in neurophysiological experiments. If there is a correspondence between what is seen in the model and what is seen in the brain, the models help account for some of the often puzzling phenomena seen in the brain during bilateral actions. They potentially give us the power to predict what neural behaviors may be found during other unperformed but feasible coordination experiments.

Specific questions addressed by the models described in this thesis include: What are the responses of the model cells as a function of time or input parameters such as body posture or location of target point? How do the preferred direction, baseline activity, and dynamic range of the model cells change across space and with different types of movement? How important is internal state in guiding movement? What happens to the model's ability to coordinate the body's limbs when different connections are lesioned? What kind of neural mechanisms can achieve collision avoidance? Are obstacles explicitly represented? These questions are important because they are analogous to those that are or could be asked in neurophysiological experiments.

Chapter 2 : The Motion Planner

This Chapter introduces a motion planner capable of controlling the movements of a simulated robot for a variety of coordination tasks. The robot model was chosen to be as simple as possible while still illustrating the non-trivial complexities of coordinating bilateral arm movements. It consists of a two dimensional body with a pair of two-jointed arms (Figure 2a). The posture of this body in its workspace can be represented by a single point in a four-dimensional joint angle configuration space. The sub-region of this configuration space representing all collision-free postures is called the "safe" region. Its complement is the region representing all colliding postures, i.e. postures with intersecting limbs.

The posture-to-posture problem consists of finding a continuous path through the safe region between any two safe points. This is always possible because, as the problem is constrained, the safe region of configuration space is connected. Since there is typically an infinite number of such paths to pick from, the posture-to-posture problem is ill-posed. The same robot body is used for the self-touching task. A self-touching posture is one in which the endpoint of the contact arm approximately touches any specified point on the arms or body (Figure 2b). Touch targets are specified by their locations relative to a set of fixed touch sensors evenly spaced over the body. There are 8 overlapping touch sensors on each side of each arm segment and 8 more on the connecting body segment for a total of 56 (Figure 2c). The planner can direct a touch anywhere on the reachable body surface by selecting the nearest sensor as its target. A successful touch must bring the tip of the contact arm within a small semicircle around the target on the same side as the target, and avoid collisions (Figures 2d & e). Since

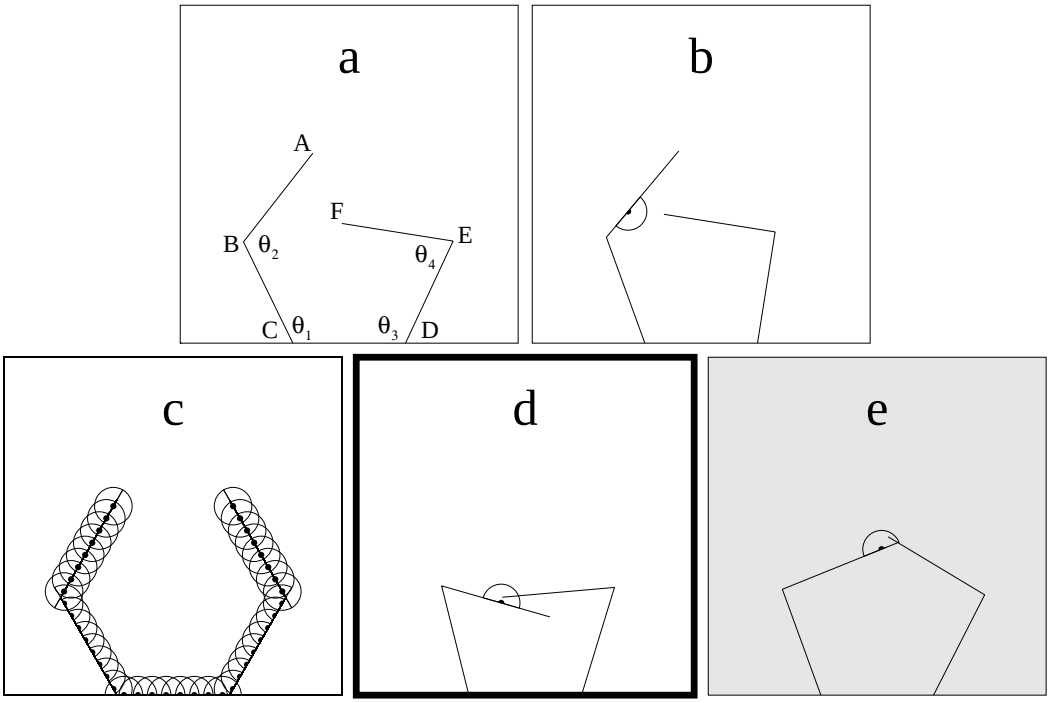


Figure 2. Definition of the workspace and body.

(a) The workspace consists of a two-dimensional unit square. The body consists of two mobile arms each with two segments (AB, BC, FE, ED), plus one immobile body segment between them (CD). Each segment is one-dimensional and has length of 1/3 the unit square. The body has one shoulder joint angle and one elbow joint angle for each arm, for a total of four degrees of freedom (θ_1, θ_2, θ_3, θ_4). Joint angles range from 20° to 180°.

(b) The body may have one somatotopic target ("itch") site, represented by a target semicircle. Targets can be located at any point along either arm or the connecting body segment, on either side, as indicated by the semicircle. Note that given the strict two-dimensional nature of the workspace and body, targets on the outsides of the inner and connecting arm segments are not reachable; they are therefore excluded from consideration.

(c) The 56 touch sensors overlap each other and cover the entire reachable body surface. 8 regularly spaced sensor semicircles are located on each side of each reachable arm segment. Although the sensors on the outer arm segments appear to be whole circles, the planner treats each semicircle as a separate sensor.

(d) A posture is self-touching when the end point of one arm lies within the target semicircle on the opposite arm. Self-touching postures are highlighted with a heavy black outline.

(e) The touching arm must reach the target from the same side of the arm segment as the semicircle; reaching from below is not permitted. Self-touching postures also must not otherwise collide. Since this posture reaches from below, it fails to qualify as self-touching. Instead it is shaded light gray, to indicate that it is near-colliding.

there is typically an infinite number of possible self-touching postures, as well as an infinite number of paths to pick from, the self-touching problem is also ill-posed.

The shapes of the safe region and the boundary separating it from the unsafe region are too complex to compute each time a movement is made, so they are pre-computed and stored. The configuration space has an infinite number of points, but its safe and unsafe regions can be approximated by dividing the space into a rectilinear grid of discrete units ("cells"). A grid with a resolution of 32 cells per dimension was sufficient for accurate planning. Each cell is characterized as safe or unsafe by checking the workspace posture of its center point for collisions. One copy of configuration space suffices for the posture-to-posture task, but for self-touching a different copy was pre-computed for each of the 56 possible touch target sensor locations. Each of these 56 configuration space "maps" has many points representing configurations in which the body touches a particular sensor location. About one megabyte of storage is needed for each pre-computed map.

Cells on the boundary between the safe and unsafe regions of configuration space generally contain both safe and unsafe points, even though they may have been labeled as safe on the basis of their center points. This creates a problem for path planning because, if these cells are considered as truly safe, collisions can occur due to the unsafe points they contain. One solution to this "mixed cell" problem is to remove a small subset of safe points near the boundary from the safe set. This can be accomplished by identifying the set of points representing postures that almost collide, i.e. where the shortest workspace distance between two limbs is less than some threshold. Cells centered on these points are labeled as "near-colliding". They provide a kind of buffer zone around unsafe cells. In the workspace, this has the effect of giving the arms a finite thickness equal to the threshold; in the work reported here this thickness is 1/6 length of a body segment. A configuration space map illustrating these regions is shown in (Figure 3).

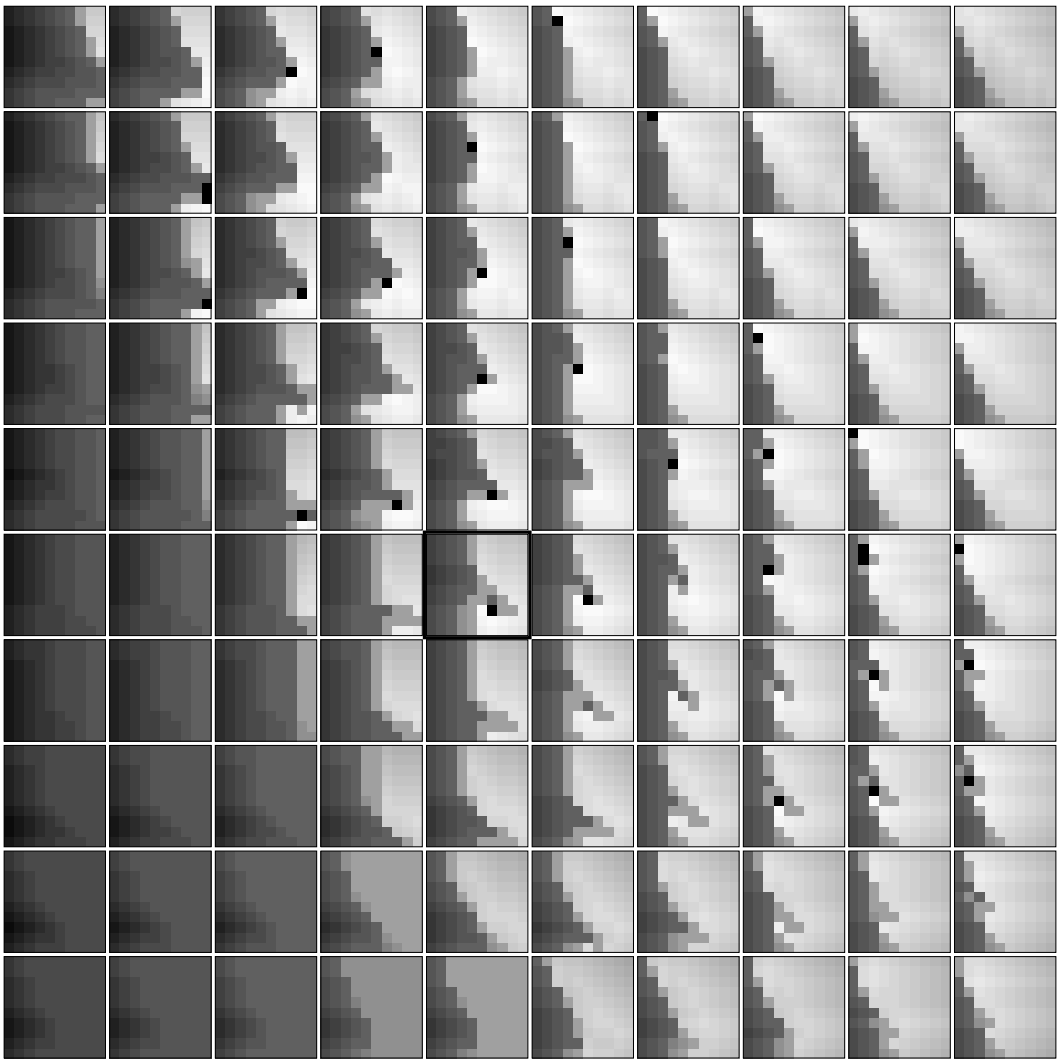


Figure 3. An example four-dimensional configuration space map. Each square is a two-dimensional slice of the map; each pixel represents an individual cell. The map represents different left arm postures across squares, and different right arm postures within squares. The inner joint angles vary horizontally, and the outer joint angles vary vertically. Safe, near-colliding, and unsafe regions are represented by light, medium, and dark gray pixels. For contrast, cells representing self-touching postures for the target shown in Figure 2b are highlighted by black pixels. The two-dimensional configuration space map in Figure 4 is represented in miniature by the outlined square near the center. The numerical values of the NF1 gradient potential function are suggested by fine shades of gray.

Although there are now "mixed cells" on the boundary between the safe and near-colliding sets, small deviations into the portions of the near-colliding set contained in safe cells will not lead to collisions.

The motion planner finds paths through the safe region and supplies the information needed to train the neural networks. Finding non-colliding trajectories requires several phases. First a discrete, distance related potential function called NF1 (Latombe, 1991) is generated and a rough form of gradient descent is done in this function. Then the paths generated by NF1 are smoothed and shortened. This smoothing and shortening process implicitly generates a new potential function, *V*, and it is gradient descent in *V* that actually traces out the trajectories generated by the motion planner.

The NF1 potential is constructed by starting at cells corresponding to goal configurations and assigning them the value 0; all their safe neighbors are assigned 1, and so on, incrementing the value by 1 at each step. This process stops when the cell containing the starting posture is reached. In the posture-to-posture task the goal posture is given, so the potential is computed on-line for each movement instance. The planner identifies a path to the goal by starting at the cell containing the starting posture and continually moving through cells with lower values until the goal, with value 0, is reached.

In the self-touching problem the final posture is not given, so a different potential field is pre-computed for each touch sensor location. All cells representing self-touching postures get the value 0, and the potential field is extended from these cells to fill the remaining unlabled safe region of configuration space. The same general path finding procedure is used until it ends at the nearest cell representing a self-touching posture. In the workspace, the endpoint of the touching arm is then further adjusted to bring it closer to the actual touch site indicated by the sensory input.

The gradient of NF1 is not unique, i.e. typically many paths are possible. However they all lead to a goal, so any one will do. A single path is selected by looking at each joint angle in a fixed order while searching for lower valued neighboring cells. This fixed selection resolves the ill-posedness of the planning problems. It also leads to jagged paths composed of many right angle bends (Figure 4). Such paths represent jerky arm movements in the workspace because only one joint angle changes at a time.

The final trajectory produced by the motion planer is obtained by doing gradient descent in a new potential, V, obtained by smoothing the NF1 generated paths. Only the gradient of V is needed so the value of V does not actually have to be computed. The negative gradient of V at any point $\boldsymbol{q}$ in configuration space is defined as the vector connecting $\boldsymbol{q}$ to an intermediate point, $\boldsymbol{p}$ on the NF1 path. An intermediate point is the furthest point on the NF1 path that can be connected to $\boldsymbol{q}$ by a straight line in the safe set. Intermediate points may be either intermediate postures $\boldsymbol{p}^{\boldsymbol{I}}$ on the way to the goal posture, or the final goal posture itself, $\boldsymbol{p}^{\boldsymbol{G}}$. Finding $\boldsymbol{p}$ requires a search that consists of connecting $\boldsymbol{q}$ with straight lines to the centers of successive cells along the NF1 path. The line is extended until it reaches the final goal or until further extension would lead to intersection with the unsafe set. More formally, the negative gradient of V at $\boldsymbol{q}$ is given by $-\mathbf{grad}V(\boldsymbol{q}) = \boldsymbol{p} - \boldsymbol{q}$.

The actual trajectory through configuration space is the path generated by following $-\mathbf{grad}V(\boldsymbol{q})$ in small steps. This path is described by the following equation:

$$\boldsymbol{q}(t+1) = \begin{cases} \boldsymbol{q}(t) - \eta\, \mathbf{grad}V(\boldsymbol{q}) / |\mathbf{grad}V(\boldsymbol{q})| & \text{if } |\mathbf{grad}V(\boldsymbol{q})| > \eta \\ \boldsymbol{q}(t) - \mathbf{grad}V(\boldsymbol{q}) & \text{otherwise} \end{cases} \qquad (1)$$

That is, the direction of a step in the trajectory is $-\mathbf{grad}V(\boldsymbol{q})$ and the length of a step is generally equal to a small constant η (1°) except when $\boldsymbol{q}$ is closer to a goal than η,

Figure 4. A two-dimensional slice of the configuration space map. Cells approximating safe, near-colliding, and unsafe regions are shaded light, medium, and dark gray. A goal cell representing a self-touching posture is highlighted with heavy black outline. Cells are drawn with their corresponding postures. Each safe cell is labeled with the value of the NF1 potential, which measures the distance around the unsafe region to the nearest goal cell. The thick gray line is an example NF1 gradient descent path, and the black lines represent the corresponding smoothed path. Terminating extensions off the smoothed path represent intermediate points $\boldsymbol{p}^{\boldsymbol{I}}$; the last terminating extension represents the final goal point $\boldsymbol{p}^{\boldsymbol{G}}$. In most cases, the path changes direction before reaching an intermediate point. Although the two-dimensional path depicted here represents the movement of only the right arm, in general the four-dimensional path represents both arms moving simultaneously.

when the step length becomes exactly the distance to the goal. The path described by (1) is traced by moving the current point $\boldsymbol{q}$ to its next location and then testing to see if the final goal $\boldsymbol{p}^{\boldsymbol{G}}$ has been reached or there is a new intermediate point $\boldsymbol{p}^{\boldsymbol{I}}$. If there is a new $\boldsymbol{p}^{\boldsymbol{G}}$ or $\boldsymbol{p}^{\boldsymbol{I}}$, -**grad**$V(\boldsymbol{q})$ changes direction and the path turns (Figure 4). Typically, a turning point is reached before an intermediate point.

The relatively smooth, four-dimensional configuration space paths generated in this way represent simultaneous, coordinated bilateral arm movements. Example trajectories for the posture-to-posture and self-touching tasks are shown in (Figure 5). For the self-touching task, the configuration space map for the touch target closest to the desired one is chosen from the set of 56 pre-computed maps. Then the path generating procedure is carried out in this space. Once a goal cell is reached, a small final adjustment is performed to bring the touching arm closer to the target actually specified by the input.

The planner's correctness was verified by testing its ability to avoid collisions during both posture-to-posture and self-touching movements. A test set of 10000 posture-to-posture tasks was used. This set was biased to be "difficult", i.e. starting and goal posture joint angles were selected from a Gaussian distribution centered on the colliding posture (100°, 80°, 100°, 80°), with a deviation of 20° per joint angle. This posture is one in which the outer arm segments collide directly in front of the body. However, those postures that were actually colliding or near-colliding were eliminated from the distribution. This sampling method forces the planner into the most difficult part of the configuration space, near the peninsular obstacle. In the workspace, the arms tend to start out close to one another, and must be moved carefully to avoid collisions. For the self-touching task, the starting postures were chosen using the same random-but-difficult algorithm. Touch targets were uniformly distributed along the body surface.

Figure 5. Example motion planner trajectories.

(Top) Example posture-to-posture trajectories. Starting postures are on the left, goal postures on the right. Postures shaded light gray represent near-collisions, which are common when the tips of the arms are passing one another. No collisions occur, however. The planner moves both arms simultaneously – it does not subordinate one arm to the other.

(Bottom) Example self-touching trajectories. Starting postures and somatotopic targets are on the left; successfully self-touching postures are on the right. Occasional near-collisions occur as the contact arm approaches the target semicircle. Once inside the semicircle, the planner attempts one last adjustment to bring the contact point as close as possible to the center of the circle; this leads to the "double" self-touching postures.

There were no collisions when the planner was tested on either set of problem trajectories. When making posture-to-posture movements, the planner always reached the goal posture. When self-touching, the planner reached the target 98.99% of the time; the rest of the time it stoped just shy of the target semicircle. This effect is related to the coarseness of the representation of targets in the planner's database, and could be eliminated with more configuration space maps.

Chapter 3 : Feedforward neural network models

It is unlikely that the brain uses data structures and serial computations similar to those of robotic motion planners. It is more likely that parallel networks of neurons are used, but it is not obvious that simple neural networks can do either task. To show that they can, and to get some idea of how they accomplish it, neural networks were programmed to do the posture-to-posture and self-touching tasks. In the past this approach has provided interesting information about how the brain actually implements computations (Zipser, 1992; Zipser and Andersen, 1988; Moody and Zipser, 1998; Moody, et. al. 1998). In most of these previous examples the tasks have been simple, and generating the data needed to train the networks was not a challenge. That is not the case here. Ideally, natural data obtained from observations of people doing the tasks should be used, but this is currently impractical. Instead the robotic planner described in Chapter 2 was used to provide the required training data.

3.1 Network architecture and training

Many complex and biologically realistic network architectures might be capable of coordinating bilateral movements. Since a major goal of this work was to determine if artificial networks could do the tasks at all, initial study of the neural networks was confined to simple, standard architectures. In particular, fully connected feed forward networks with two hidden layers were used, as shown in Figure 6. This architecture has performed well in other complex geometrical problems (Lang and Witbrock, 1988). Feed forward networks can be used as the key element in a dynamical system that generates

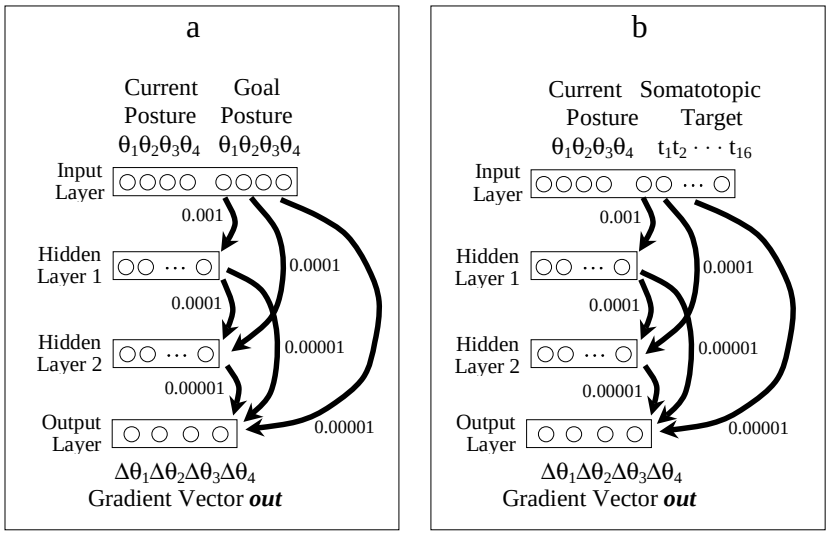


Figure 6. Posture-to-posture (a) and self-touching (b) network architectures. Each network contains two hidden layers of 16 logistic units. Each layer (input or hidden) is connected to all successive layers (hidden or output). Input and output representations of postures use a linear code of posture joint angles, one unit per angle. Input representation of the somatotopic target uses a distributed Gaussian place code, with 16 units evenly spaced across the body surface. Numbers next to each arrow indicate learning rates, which increase with distance from the output layer to compensate for the decrease in error signal propagated back through the network.

trajectories if they provide the information about **-grad***V*(***q***) required to implement the dynamics of (1). Best results were obtained in both tasks when the output of the networks was trained to represent **-grad***V*(***q***). Other networks trained to represent |**-grad***V*(***q***)| or ***q*** **- grad***V*(***q***) were also capable of learning the tasks, albeit with less precision.

The input for the posture-to-posture task consists of 8 joint angles representing the current and goal postures (Figure 6a). The input for the self-touching task consists of 4 joint angles representing the current posture, and a target location on the body (Figure 6b). The touch target site is not represented in configuration space coordinates. Instead, units with Gaussian receptive fields evenly distributed across the body surface are used. This somatotopic representation allows touch targets anywhere on the body to be represented by the appropriate choice of input values. Both of these networks can be interpreted as performing a sensory to motor mapping. The inputs are either joint angle sensors or somatotopic "itch" position sensors, and the output is a goal for motor movement of the joints.

Standard error backpropagation (Rumelhart, et. al. 1986) was used to train the networks. Sets of 45000 posture-to-posture and 30000 self-touching trajectories were generated for training. These trajectories were sampled at small (1°) steps to give approximately 3 million and 2 million different training patterns for each respective task. Training patterns were presented in random order. Learning rates were increased from the output to input layers to compensate for the decrease in the error signal as it propagated back through the network. A momentum of 0.999 was used for all layers. About 100 million training cycles were sufficient to minimize the sum-squared error for all networks. To ensure generality, six networks with different initial random weights were trained and tested on each task.

3.2 Network testing

The neural networks described above were used to generate coordinated movements by replacing -**grad***V*(***q***) in (1) with the network output, ***out***, giving the network dynamical equations:

$$\boldsymbol{q}(t+1) = \begin{cases} \boldsymbol{q}(t) - \eta\ \boldsymbol{out} / |\boldsymbol{out}| & \text{if } |\boldsymbol{out}| > \eta \\ \boldsymbol{q}(t) - \boldsymbol{out} & \text{otherwise} \end{cases} \qquad (2)$$

The accuracy of the networks depends on how closely ***out*** approximates -**grad***V*(***q***). Performance was assessed by testing each trained network on 100000 novel random-but-hard posture-to-posture or self-touching trajectories. A trajectory was scored as successful if the network-driven dynamical system reached the final goal (posture or target) in about the expected time without a collision. Failed trajectories could be assigned to one of four error types: colliding, premature stopping, out of bounds (exceeding the upper or lower limits for joint angles), and perpetual oscillation. The percentage of trajectories that end in each condition is summarized in (Table 1).

Table 1. Feedforward neural network performances during movement.

	reached goal	**colliding**	**stopped**	**out of bounds**	**oscillating**
posture-to-posture (n = 6)	**97.5%**	**1.78%**	**0.559%**	**0.0990%**	**0.0183%**
self-touching (n = 6)	**87.0%**	**9.26%**	**0.900%**	**2.63%**	**0.203%**

For comparison, a simple-minded algorithm that moves directly to the goal posture or target without regard for collisions would collide 44.3% of the time in the posture-to-posture task, and 30.4% of the time in the self-touching task. Glancing collisions between the tips of the arms are the most common kind of error. These errors are rather innocuous from a biological standpoint because they represent contact between fingers,

which are rather compliant. Adjusting the motion planner's geometric parameters can reduce the frequency of these collisions. For example, increasing the near-colliding threshold forces the planner to allow more space between the arms during movement. Collisions can also be reduced by enriching the training set with more trajectories that demonstrate collision avoidance.

Errors arise because ***out*** is not a perfect approximation of -**grad***V*(***q***). Any difference between ***out*** and -**grad***V*(***q***) will alter the direction or length of a trajectory. Small deviations in direction on each step can accumulate and lead to either a collision or an out of bounds condition. In rare cases the path may turn back on itself, leading to oscillation. Premature stops arise from alterations in the length of a step. These are most likely to occur in the vicinity of intermediate goals because |-**grad***V*(***q***)| goes to zero as these points are approached. This is not a problem for the motion planer since it "knows" where these points are and can take a step of just the right length. However, the network has to learn this information with precision to avoid confusing them with the final goal.

Although the neural networks are not as accurate as the robotic motion planner they are much faster and consume less memory. On a 180MHz, 64MB workstation the planner requires an average of 220ms to generate a trajectory, while a neural network requires only 20ms. The planner requires about 3MB of memory space for the posture-to-posture task, and about 58 MB for the self-touching task. Each neural network requires only about 20KB. Neither implementation has been aggressively optimized. It should be noted, however, that neural networks are serial implementations of a fundamentally parallel architecture, and thus stand to benefit significantly from parallel implementation.

3.3 Network mechanisms

The posture-to-posture and the self-touching networks use similar strategies to approximate -**grad***V*(***q***). These strategies were found by examining both the dynamics of hidden unit activity during movement and by looking at key sets of connection weights. The mechanisms are most easily analyzed in cases where there is a small number of well-separated, sharp turning points. The posture-to-posture network, which has the easier task, is analyzed first.

The input to the posture-to-posture network consists of the current posture ***q*** and a final goal posture $\boldsymbol{p}^{\boldsymbol{G}}$. If $\boldsymbol{p}^{\boldsymbol{G}}$ can be reached by a safe straight line, i.e. $\boldsymbol{p} = \boldsymbol{p}^{\boldsymbol{G}}$, then ***out*** can be computed using only the direct connection between the input and the output layers. The computation involves only a subtraction since $\boldsymbol{out} = \boldsymbol{p}^{\boldsymbol{G}} - \boldsymbol{q}$. This will occur if the weights connecting corresponding input and output joint angles are 1 for the $\boldsymbol{p}^{\boldsymbol{G}}$ and –1 for ***q***, while all other weights between input and output are zero. This is a close approximation of what the network actually learns to do. All the weights of the input to output connection matrix go to zero except those on the diagonal which all have a magnitude near 1 (almost exactly 1.2) with the appropriate signs. Thus, the direct input to output connection provides the information required to compute the correct direction, to move toward $\boldsymbol{p}^{\boldsymbol{G}}$ along a straight line in configuration space. While it is generally not safe to move in this direction from all starting postures, it will eventually become safe when ***q*** has passed all possible collision points.

During the initial phase of movement there is generally no safe straight line between ***q*** and the final goal posture. The path must be bent so that it passes around configuration space obstacles. Bending is performed by hidden layer 2, which computes a vector that is added to the output to direct it away from the goal toward a safe intermediate posture. Notice that the activity of the units in hidden layer 2 show large

changes in activity from the start of the trajectory until the point where the danger of collisions has passed (Figure 7). At this point the activity of these units goes to a low, constant value and they no longer make a significant contribution to the output of the network. This is the same point in the trajectory where the input to output connection starts to make the correct computation of the direction of -**grad***V*(***q***). Because these weights are not exactly 1 or -1, the network still needs to make a small correction to the output layer to make |***out***| = |-**grad***V*(***q***)|. This is supplied by the connection between hidden layer 1 and the output. The activities of the units in this layer continue to change until the end of the trajectory (Figure 7). The weights connecting hidden layer 1 to the output are very small, consistent with the need for only a modest correction in the length of ***out***.

Unlike the posture-to-posture network, the self-touching network does not explicitly represent its goal with joint angles or postures. Instead, the final goal posture is implicitly represented by the starting posture and a somatotopically specified touch target. Yet the behavior of the self-touching network still resembles that of the posture-to-posture network (Figure 8). Note that the activity of the hidden units in layer 2 change in the region where the last collision point is passed and then remain nearly constant for the rest of the trajectory. The hidden units in layer 1 have large changes in activity during the final phase of the trajectory. This is analogous to what was observed in the posture-to-posture network as can be seen by comparing Figures 7 and 8. Also, the direct connection between the input and the output is still vital to performance; lesioning this connection has disastrous results.

Further analysis of both networks demonstrates that they learn to encode the kinematic structure of the body in their weight matrices. When a new network with a single hidden layer of 16 units was attached to a fully trained posture-to-posture or self-touching network, it was capable of decoding the network's internal representation of the

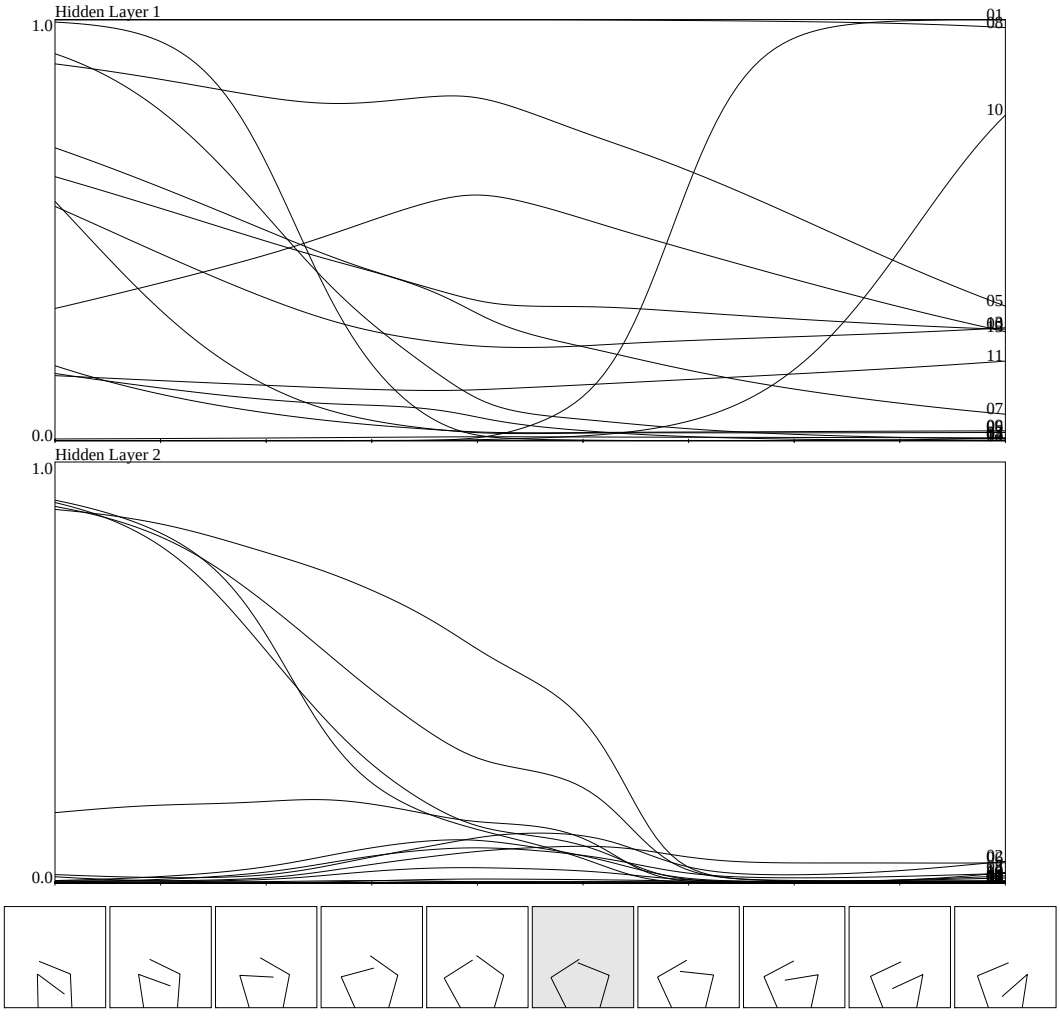


Figure 7. Posture-to-posture network hidden unit activations over trajectory time. Each graph superimposes the activations of all units within a hidden layer. The end of each activation curve is labeled with its unit number. The network's generated trajectory is shown below. Note that when the tips of the arms pass one another (i.e., when there is no longer a danger of collision from making a direct movement), the activations of all units in hidden layer 2 are almost completely shut off.

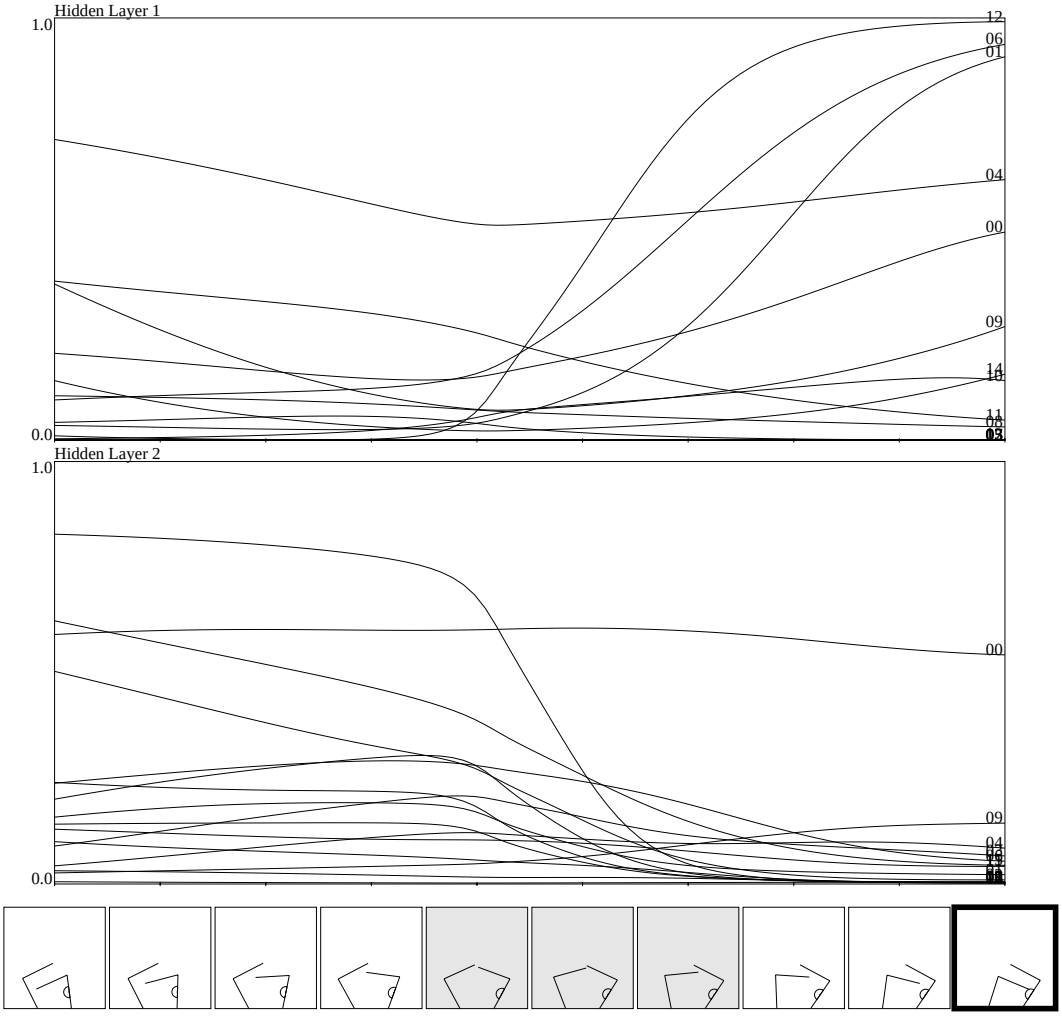


Figure 8. Self-touching network hidden unit activations over trajectory time. Figure format is identical to that of Figure 7. Many units in both layers have large dynamic ranges; once the danger of a collision has passed, however, the activations of units in the second layer tend to remain at constant, non-zero values.

current posture. The new network received input only from the two hidden layers of the trained network. Its output consisted of a single unit that coded for the collision status of the "current posture" input of the preceding network.

Keeping the posture-to-posture or self-touching network weights fixed, the weights of the new network were trained to extract the collision status information from the hidden units of the fixed network. Training patterns were selected in the "random-but-focused" fashion described earlier, while the somatotopic target was randomly chosen with a uniform distribution. The new network was tested by systematically scanning the entire range of possible "current postures" -- i.e., the entire configuration space. For each posture the output of the new network was averaged over 100 goal postures or 100 targets uniformly distributed across the body surface, and this average was plotted in the form of a configuration space map comparable to that used by the motion planner. The configuration space map generated from the posture-to-posture collision detection network is shown in Figure 9, and the map generated from the self-touching collision detection network is shown in Figure 10.

A simple visual comparison of these maps confirms that the new collision detection networks have learned to reconstruct most of the the collision status of the input posture from the internal representations of the posture-to-posture and self-touching networks. This is extremely interesting because the movement networks were not explicitly trained to do collision detection. Neither network has ever seen a colliding posture. The network weights encode enough information about the geometrical structure of the body to allow the safe/unsafe boundary to be implicitly represented in the hidden units.

There are some interesting differences between the networks. These seem to result from the need to convert the somatotopic input representation to the joint angle coordinates used for ***out***. This complex computation is reflected in differences in the

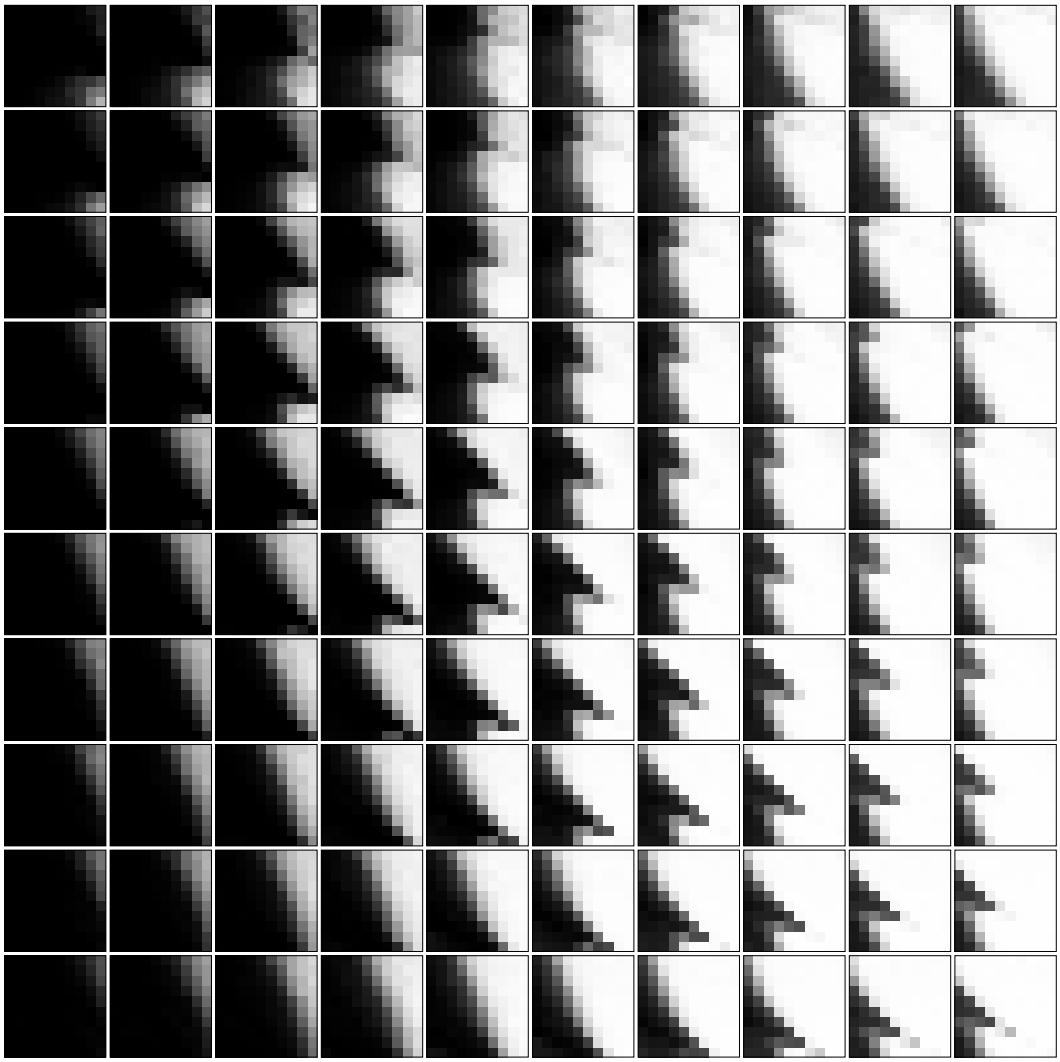


Figure 9. The four dimensional safe/unsafe boundary reconstructed from the hidden unit activations of a posture-to-posture network. Safe regions are light gray, unsafe regions are dark gray. Compare to Figure 3.

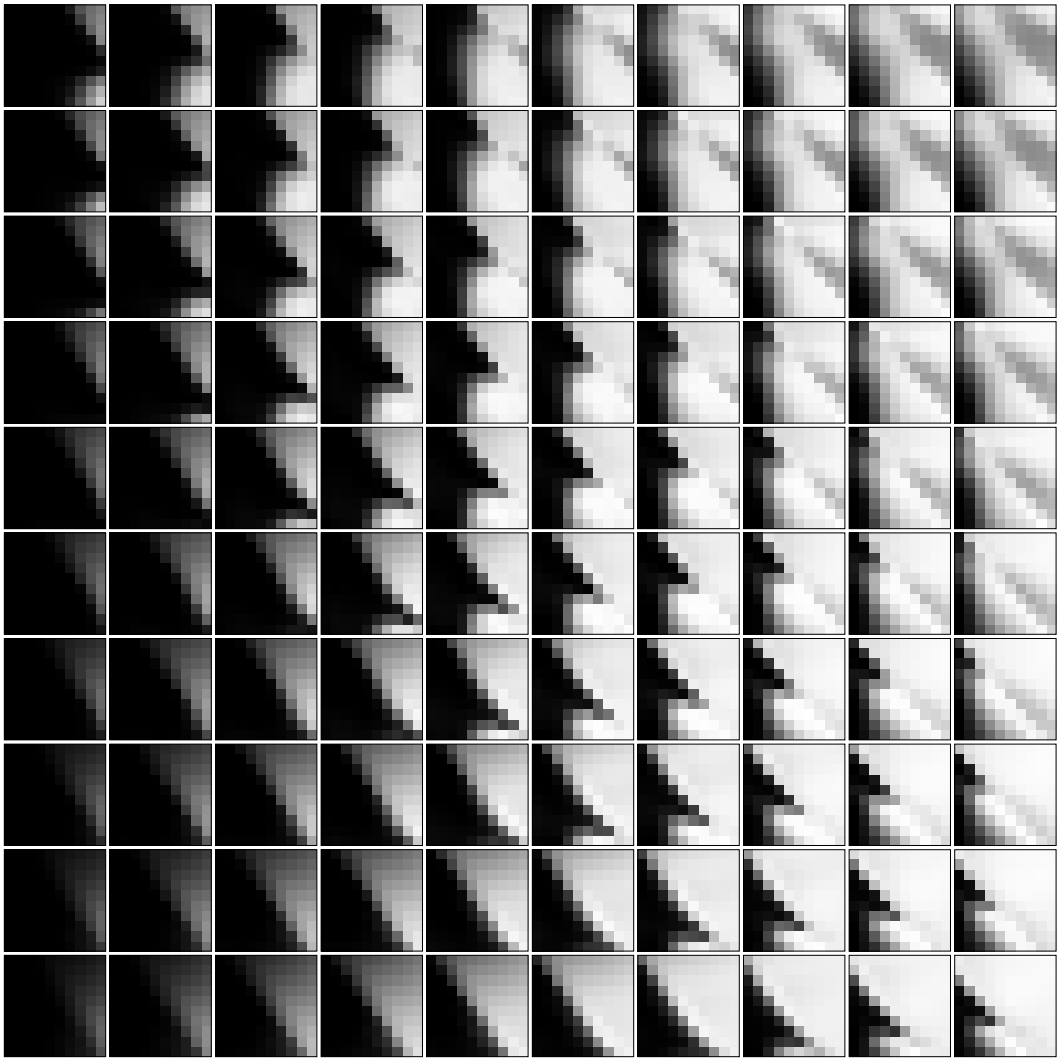


Figure 10. The four dimensional safe/unsafe boundary reconstructed from the hidden unit activations of a self-touching network. Safe regions are light gray, unsafe regions are dark gray. Compare to Figure 3.

patterns of both hidden unit activities and weights. The weight matrix of the direct input to output connection is no longer diagonal, but full of non-zero weights. The near constant activities of hidden layer 2 during the final phase of the trajectory are no longer all near zero, but take on significant values that depend on the touch target input. Also, the connection weights between hidden layer 1 and the output are larger. All of this suggests that while each part of the self-touching network carries out a function analogous to the corresponding structure in the posture-to-posture network, they have the additional task of dealing with the more abstract input representation. In particular, the direct input to output connection can only make a linear approximation to the direction and length of ***out*** during the final phase of the trajectory. Help is needed from hidden layer 2, which makes a nonlinear correction to the direction of ***out***. Since the direction of movement through configuration space during the final phase is constant, the outputs of hidden layer 2 remain constant as well. However, since the length of ***out*** always changes, the activities of hidden layer 1 units change also. The larger magnitude of the weights in the hidden layer 1 to output connection presumably reflect the need for a significant nonlinear correction to the length of ***out*** to deal with the touch target input representation, even when there is no danger of collision.

3.4 Biologically relevant hidden unit properties

The feedforward architectures of the neural networks in this Chapter were chosen to test the ability of simple networks to do the rather difficult coordination tasks. They were not designed to resemble the structures that might be involved in these activities in the brain. For this reason one might expect that hidden unit behaviors would reflect only the most general characteristics of real neurons engaged in similar tasks. Hidden unit activities can be compared to real neurons in several physiologically measurable ways:

by measurement of somatosensory receptive fields on the body, by the modulation of these fields by body posture, and by identification of positional response functions or "posture fields" in the workspace.

The somatotopic receptive fields of several hidden units from the self-touching network are shown in Figure 11. All units have distributed bilateral receptive field structures. Most fields have a patch-like distribution across the entire body surface. This differs from what is found in the primary somatic sensory areas where the neurons have local, unilateral receptive fields. This suggests that, if there is any similarity between these networks and the brain, it will be found at higher levels in the somatosensory processing system or in motor areas. The receptive fields are modulated by changes in posture, which tend to alter the magnitude of the activity more than the shape of the fields, particularly in the first hidden layer.

The positional response functions or posture fields of a neuron show its response as a function of the position of the tip of an arm in the workspace with all other variables held constant (Kettner, et. al. 1993). Posture fields for one arm were determined for every hidden unit in both networks (Figure 12). Most of the hidden units have single, localized posture fields in the workspace. The shape and position of these fields change with goal posture or target position and with the posture of the other arm.

3.5 Conclusions

This chapter has described the movement performances and mechanisms of a set of supervised neural networks trained to emulate the behavior of a simulated robot motion planner. Although it is currently limited to a simple two-dimensional body, the system could be extended to a three-dimensional body using the same methodology. As the planner is scaled up to harder problems with more degrees of freedom, explicitly

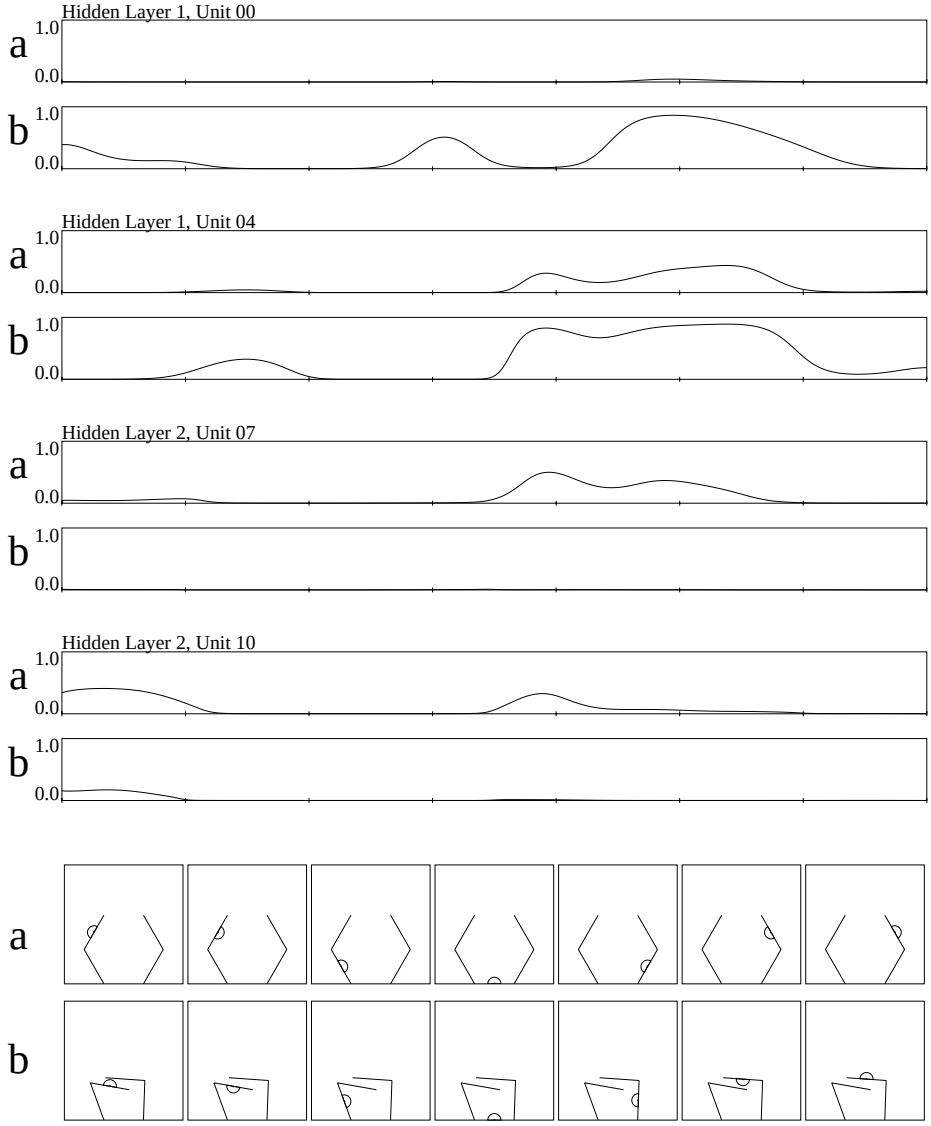


Figure 11. Example self-touching network somatotopic receptive fields. The activations of four hidden units (two from each hidden layer) are shown as a function of target location on the body surface. Target location varies continually along the x-axis, from the outer side of the left outer arm segment, around the inner sides of the left and right arms, to the outer side of the right outer arm segment. Tick marks represent boundaries between arm segment sides. The outer sides of the inner arm segments are not reachable, and are not shown here. Each pair of graphs a and b represents the receptive field of the same unit plotted for two different postures a and b, as depicted by the two rows of postures at the bottom. Note how the magnitudes of the receptive fields are modulated by the body posture.

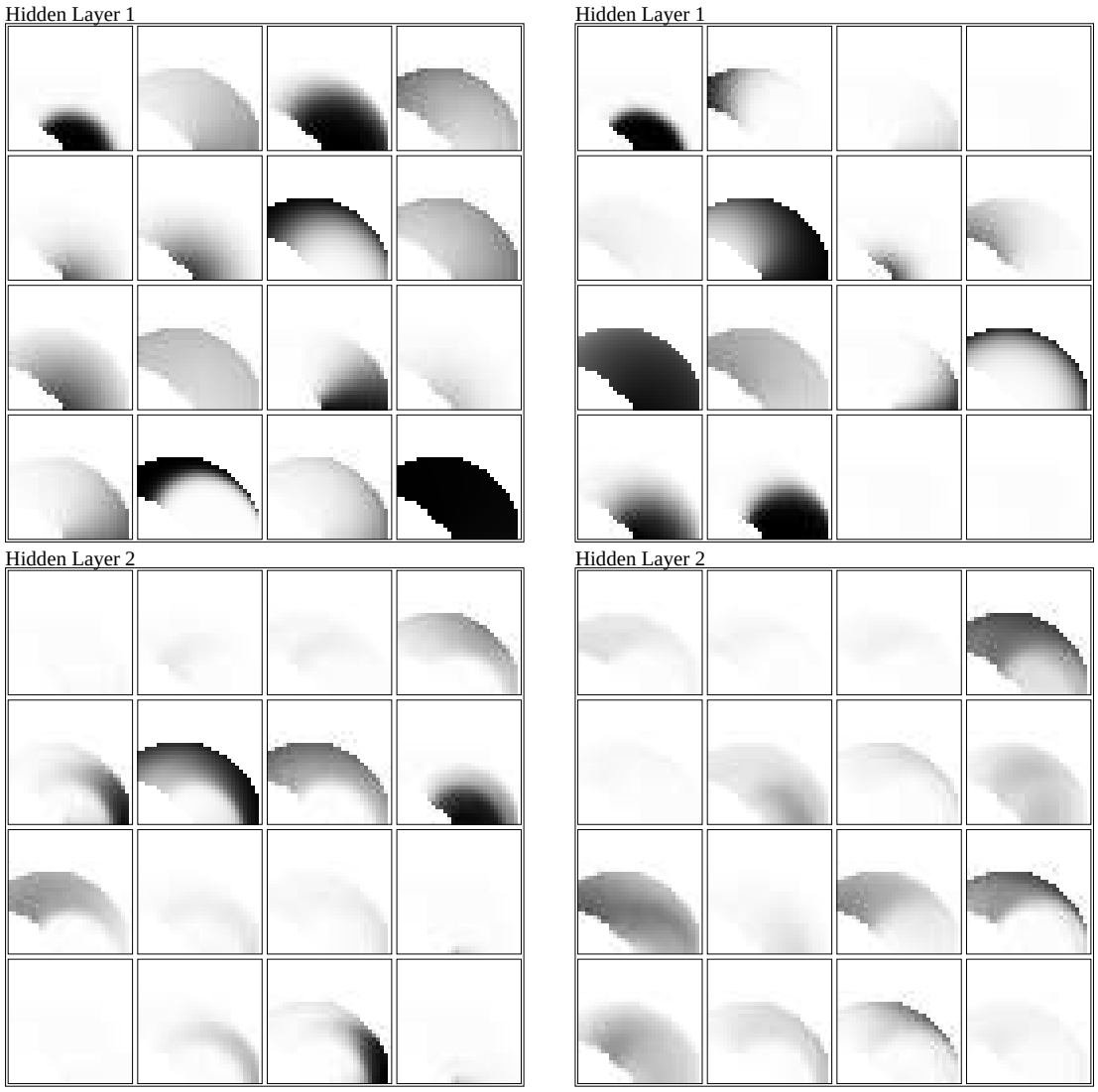


Figure 12. Example positional gradient functions or "posture fields" for the posture-to-posture (left) and self-touching (right) networks. Arcs represent the range of points reachable by the left arm. Unit activations are plotted as a function of the position of the endpoint of the left arm in the workspace. For the posture-to-posture network, the right arm current and goal postures are held fixed in front of the body as in (Figure 2a), and the left arm goal posture tucked underneath the right arm. For the self-touching network, the right arm is as in (Figure 2a), while the somatotopic target is held fixed in the center of the inner side of the right outer arm segment.

representing the entire configuration space as has been done here becomes impractical. Many efficient heuristics have been developed to handle such cases, however, and tractable motion planners for multi-arm, 3D movements have already been built (Kavraki, et. al. 1996, Barraquand and Latombe, 1991, Koga and Latombe, 1994). Although it is quite possible to train neural networks to emulate such planners, a more efficient and biologically plausible approach may be to decompose the movement tasks into subtasks. Then neural networks can be trained to solve these subtasks, and their output reintegrated to achieve movement.

The architecture of the networks used here does not incorporate important brain features such as bilateral symmetry or the separation of sensory and motor processing. Nevertheless, some of the properties of the hidden units in these models may qualitatively reflect what will be found in the brain. For example, experiments on bilateral coordination might expect to find neurons with somatotopic receptive fields modulated by both the body posture and the movement goal. Perhaps of more importance is the observation that many of the hidden units in the models have near constant activity when there is no need to avoid a self-collision. This suggests that there may be a population of neurons devoted to obstacle avoidance that shows little activity change during unobstructed movements. To find such neurons, it will be necessary to design experiments where interlimb collisions are possible. If verified experimentally, it would suggest that the system used for controlling coordinated movements may be partially distinct from and operate in parallel with simpler reaching mechanisms. Neurons in the coordination system would then not be very active in reaching tasks not involving bilateral movement or collision avoidance.

Chapter 4 : Biologically motivated models

This chapter introduces several identification models that partially account for a narrow but growing set of neurophysiological evidence on the brain mechanisms underlying bilateral coordination. The models build on and extend the networks described in the previous chapter by making them recurrent, altering their tasks to be closer to those used in neurophysiological experiments, and by giving them an anatomically inspired bilateral architecture. The identification paradigm is used to explore a set of reasonable model architectures. Plausible architectures can be accepted or rejected as models by comparing their hidden unit activations to known data. Each architecture embodies a conjecture about the neural algorithm underlying coordination. The identification paradigm allows conjectures that are consistent with the data to be verified, and thus establish a candidate mechanism that may be used in the brain. It also allows conjectures that are inconsistent with the data to be falsified. In either case, the methodology supports the goal of understanding and explaining cognition in the brain by providing testable hypotheses to guide experimental work.

Two architectural families are studied. In one family, network structure is partially or fully segregated in a bilateral fashion. In the other family, networks were trained on either the original posture-to-posture problem, a new "visual target" task similar to the posture-to-posture problem, or on both the visual target and self-touching tasks. The networks achieve performance comparable to that of the feedforward networks. They make heavy use of recurrent connections. In bilaterally segregated networks, lesioning the connection between left and right hidden layers causes catastrophic failure. The hidden units have preferred directions, baseline activations, and

dynamic ranges that vary with body posture, task type, and movement context in a biologically plausible way. They also have peri-movement activation profiles that account for some but not all single unit recordings in primates performing bilateral tasks.

4.1 Methodological Issues

Neural networks are known to be universal function approximators. Any formal functional mapping can be learned to an arbitrary approximation if the network is given sufficient resources (e.g. hidden units, connections, and training time). Neural networks can thus learn any desired behavior, as long as the behavior can be formally encoded in a function mapping. When a neural network has correctly learned to emulate a behavior, it can be considered a model of the brain mechanism serving that behavior. This thesis makes the distinction between models with correct input/output behavior and models that correctly emulate the neural mechanism the brain uses to accomplish the behavior.

Neural system identification models may yield several kinds of negative results. They may fail to correctly emulate the input/output behavior of the neural system in question. This occurs when a network does not have sufficient resources to solve its task, or when the task is not appropriately encapsulated in the formal encoding presented to the network. Often relatively minor changes in a network's training parameters have a critical effect on its performance on the task. This lack of robustness to parameter values means that its usefulness for understanding the properties of neural systems in the brain is limited.

The model neurons may exhibit behaviors that have not been published in the neurophysiological literature. This does not mean that the model is incorrect. The vast majority of data collected in neurophysiological experiments are not reported in published form. Often, only data that have a clear or obvious explanation are published.

The behavior of model neurons may correspond to behaviors recorded in poorly understood, unpublished data. The model may thus provide an explanation for data that was previously unfathomable (Zipser, 1992).

The model neurons may fail to exhibit behaviors that have been found in analogous biological systems. This difference in internal behavior implies that the neural algorithm embodied by the network's architecture is at least partially inconsistent with that used in the brain. This kind of result shows that specific neural architectures are not sufficient to account for the way the brain accomplishes a task.

4.2 Bilateral movement tasks

The feasibility of using neural networks to model the mechanisms of bilateral coordination has already been established (Farrar and Zipser, 1999; Chapter 3 of this dissertation). The feedforward and monolithic architectures used, however, were not biologically inspired. To address this shortcoming, and to improve the usefulness of the models, recurrent networks were trained on a variety of coordination tasks: the original posture-to-posture task, a new "visual target" task analogous to that used in neurophysiological experiments, and a combination of the visual target and self-touching tasks.

Several networks were trained to perform the posture-to-posture task, with input and output formats similar to those of the feedforward networks described in Chapter 3 (Figure 13). Training data consisted of consecutive steps from example posture-to-posture trajectories. Each step in each trajectory was presented to the networks in order (not randomized, as in the feedforward networks), and the starting posture was randomly reset at the beginning of every trajectory. Recurrent networks with resources comparable

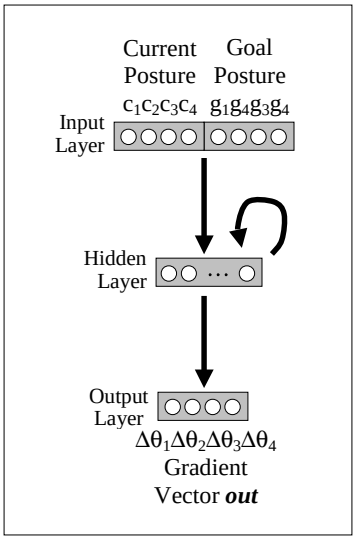


Figure 13. Recurrent posture-to-posture network architecture. Input and output formats are the same as the feedforward posture-to-posture network (Figure 6a). The single hidden layer contains 32 units, which are recurrently connected to one another.

to the feedforward networks described in Chapter 3 and with the architecture shown in Figure 13 were capable of learning the task in approximately the same epoch time.

In the network architecture depicted in Figure 13, all input units connect to all hidden units; all hidden units connect to all hidden units; and all hidden units connect to all output units. There are no direct connections between input and output units, however. This constraint prevents the networks from developing a diagonal weight matrix as in the feedforward network described in Chapter 3 (Figure 6a). It forces the them to develop an internal representation for all postures, not just those that require collision avoidance.

These networks demonstrate that a single layer of recurrent hidden units is capable of performing the same computation as several layers of feedforward hidden units. This is important for modeling. Feedforward networks are known to be universal function approximators, and in principle can solve any problem with a single layer of hidden units. As a practical matter, however, networks with multiple hidden layers perform better because they can create abstract representations that build on one another. To exploit this useful feature, the theorist must guess how many computational steps (i.e. hidden layers) are needed to solve the problem efficiently. Recurrent networks, because they maintain state over time, are capable of building such abstract representations automatically, within a single layer of units (Williams and Zipser, 1995). The use of recurrent units helps avoid guessing how many computational steps are required to solve a problem. The network decides for itself how to break up the problem, and how many computational steps to use. Although the input changes on every cycle, the changes are very gradual because the step size of the movement is rather small (equal to 1° in configuration space). The input at previous time steps is likely to approximate the current input, which gives the network enough time steps to solve the problem. Thus the

choice of input dwell is unlikely to be a strong constraint on number of computational steps the network could use in constructing its internal representation(s).

In a variation of the posture-to-posture training paradigm, networks with the architecture shown in Figure 13 were trained on data representing continuous movements. The final goal posture of one trajectory was the starting posture of the next. No significant difference was found between networks trained on continuous movements and those trained on movements with randomly reset starting postures. The insertion of "hold" commands (i.e. a goal posture equal to the starting posture) of various lengths (e.g. 0 ~ 10 steps) between each posture-to-posture trajectory also had no discernable effect.

The posture-to-posture problem does not require the networks to remember an internal state for long periods of time. The network is able to rapidly recompute its internal state as the current posture is reset at the beginning of a trajectory. Since each trajectory is composed of an average of 40 ~ 60 steps, resetting is relatively rare and does not appear to have an important impact on the network's performance.

Several networks were trained on a posture-to-visual-target task similar to the posture-to-posture task (Figure 14). In this task, the network moves the left and right arms to reach left and right target points in the workspace. Points were chosen at random, but selected so that the body can simultaneously touch each target point with the tip of one arm, with no collisions between the arms. These workspace targets are analogous to the endpoints of the arms of the goal posture in the posture-to-posture problem. The current posture is represented to the network via a 4 unit linear code (e.g. Figure 14, top left). Each visual target is presented to the to the network in the form of a distributed, two-dimensional Gaussian place code. In the work reported here, this code was represented by a 4x4 matrix of input units (e. g. Figure 14, top). The overlapping Gaussian receptive fields allow visual targets to be represented anywhere in the

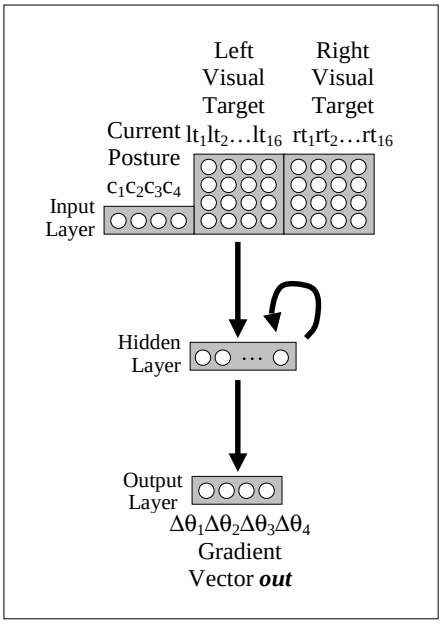


Figure 14. Recurrent posture-to-visual-target network architecture. Input format encodes the current posture and the locations of two workspace targets, one for each arm. Output format and hidden layer structure same as Figure 13.

workspace by an appropriate choice of input values. The output is the goal for motor movement, and is represented by a linear code of the difference between the current posture and the next desired posture. The motion planner's posture-to-posture trajectories were adapted to provide this training data.

The posture-to-visual-target task was further adapted so that either one or two targets might be presented. The absence of a target was represented by setting all the units for that target to 0. Training data consisted of trajectories representing movements of the arms to either one or two workspace targets, selected at random. For trajectories with only one target, the motion planner was instructed to move toward a goal posture where one arm touches the workspace target and the other arm is at its original starting posture. This behavior emulates one-arm reaching for most workspace targets.

The purpose of this variation was to investigate a class of behaviors that are more directly comparable to those studied in neurophysiological experiments. The eight-target visual reaching or manipulation task (Georgopoulos, et. al. 1984) is widely used in the neurophysiological literature, and has been extended to bilateral movements (Donchin, et. al. 1998). In the bilateral paradigm, the monkey or subject is sometimes instructed to move only one arm to a target. The extended posture-to-visual target task allows for the possibility that an arm will have no target. It is possible to run analogous unilateral and bilateral reaching experiments with the model. Neural networks trained to emulate this task are therefore well-formed models of the neural systems supporting bilateral, goal-directed reaching.

4.3 Bilateral network architectures

The architectural distinctiveness of the cortical and subcortical motor systems can be incorporated by subgrouping the hidden units of the models into left and right hidden

layers, analogous to the left and right hemispheres of the brain. Connections between the segregated layers are analogous to the fibers of the corpus callosum. The bilateral organization of the model allows more direct comparisons to be made between hidden unit activities and emerging neurophysiological data. The movement disorders of networks with callosal or unilateral lesions can also be compared to the coordination deficits manifested in commissurotomy patients, and in animal models of high level motor system lesions.

Networks with restricted architectures were constructed and tested (Figure 15). The hidden units were divided into left and right hidden layers. Units within a hidden layer connect to all units within that layer, and to all units in the hidden layer on the other side. Both layers connect to all input units, but only to the output units on the same side, left or right respectively (Figure 15, center). The cleaving of the hidden-to-output connection makes this architecture partially bilateral: all units receive information about both sides, but are responsible for constructing the movement vector of only one side. The bilateral distinctiveness of this architecture was enhanced in a training variation. Connections from one hidden layer to the other – connections spanning the "commissure" – were constrained to propagate activation but not error during training. This subgrouping strategy has been shown to improve training times (Zipser, 1989), and might also play the role of a catalyst for the functional specialization of the hidden units.

In another model architecture, the bilateral distinctiveness was increased by segregating both the input and output connections (Figure 16). Input and output units were divided into blocks that carry information about the left or right sides of the body. Each hidden layer connects to the respective left or right input and output units. Units within a hidden layer connect to all units within that layer, and to all units in the hidden layer on the other side (Figure 16, center). The purpose of this architecture was to test a conjecture about how the hemispheres of the brain might divide coordination tasks, and

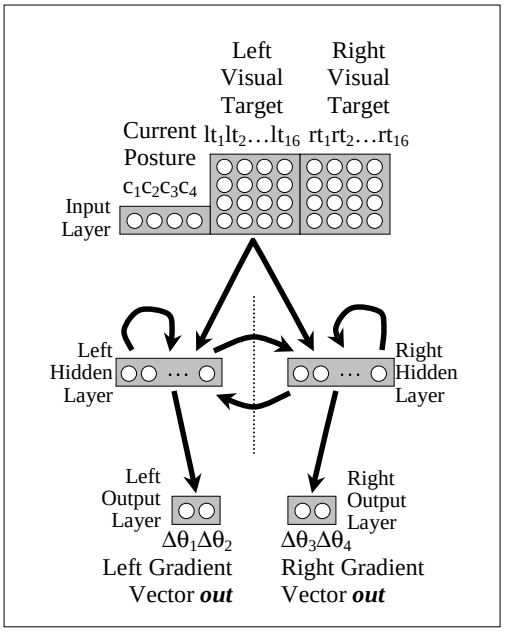


Figure 15. Partially segregated recurrent posture-to-visual-target network architecture. Input format encodes the current posture and the locations of two workspace targets, one for each arm; all inputs go to both hidden layers. Output format encodes the movement commands for the left and right arms separately. The hidden units are segregated into left and right layers, which project to the respective left or right output layers and to one another. Units in each hidden layer are also recurrently connected to all the other units in that layer. The dotted vertical line represents the model's commissural connections. In a training variation, error was not allowed to propagate across this line.

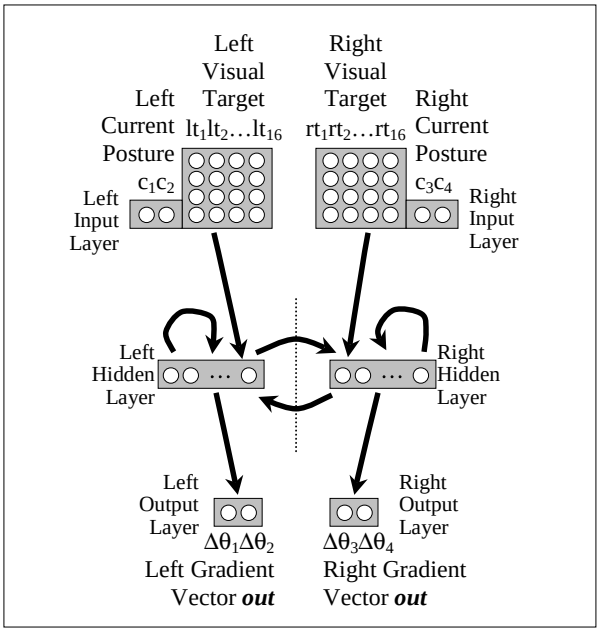


Figure 16. Fully segregated recurrent posture-to-visual-target network architecture. Input format encodes the current posture and location of workspace target for each arm separately. Output format and hidden layer structure the same as in Figure 15.

to investigate a class of models that were constrained to develop a pair of cooperating internal representations. The architecture forces the hidden layers to exchange encoded information about the contralateral input across the commissural connections to coordinate the arms. In another architectural variation, the subgrouping strategy described above was applied to this architecture to test an "extreme" form of bilateral segregation.

There is a certain danger in adding constraints that shape the network's mechanisms, if the purpose of the study is to discover and explore new mechanisms. Inappropriate constraints may force the network into a solution that the theorist thinks is reasonable, but that is not used in biological nervous systems. The addition of constraints is appropriate if there is a reasonable biological basis for them. This is an important issue for identification modeling, but it does not invalidate the method. Rather, it forces the cognitive neuroscientist to make conjectures and constraints explicit. Even if those conjectures turn out to be incorrect, the modeling exercise is still useful because it gives both theorists and experimentalists a common framework that they can work within.

4.4 Dual task architectures

Several networks were trained to perform both posture-to-visual-target and self-touching tasks. Both tasks were represented in the input units (Figure 17). A particular task was selected by activating the input units appropriate for the task and setting the units for the other task to 0. Training data consisted of alternating posture-to-visual-target and self-touching trajectories. Starting postures were randomly reset on every trajectory.

The purpose of this variation was to explore how networks might develop a more general internal representation than that developed by networks trained on either task

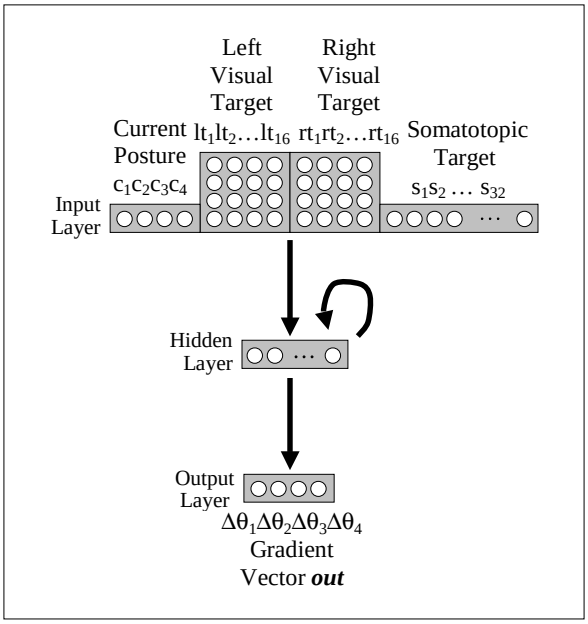


Figure 17. Recurrent dual task architecture. Input format simultaneously encodes the current posture, two workspace targets, and the location of a somatotopic target. The appropriate movement task is selected by activating the units appropriate for the target, and setting the units for all other tasks to 0. Output format and hidden layer structure same as in Figure 14.

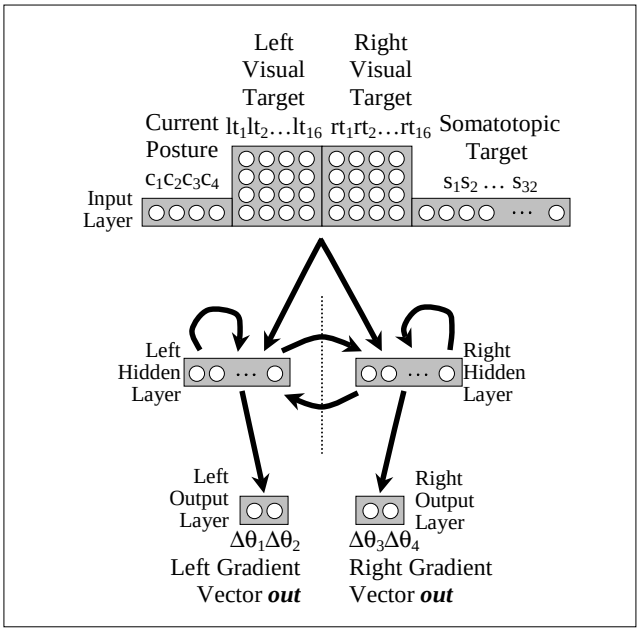


Figure 18. Partially segregated recurrent dual task architecture. Input format simultaneously encodes the current posture, two workspace targets, and the location of a somatotopic target. The appropriate movement task is selected by activating the units appropriate for the target, and setting the units for all other tasks to 0. Output format and hidden layer structure same as in Figure 15.

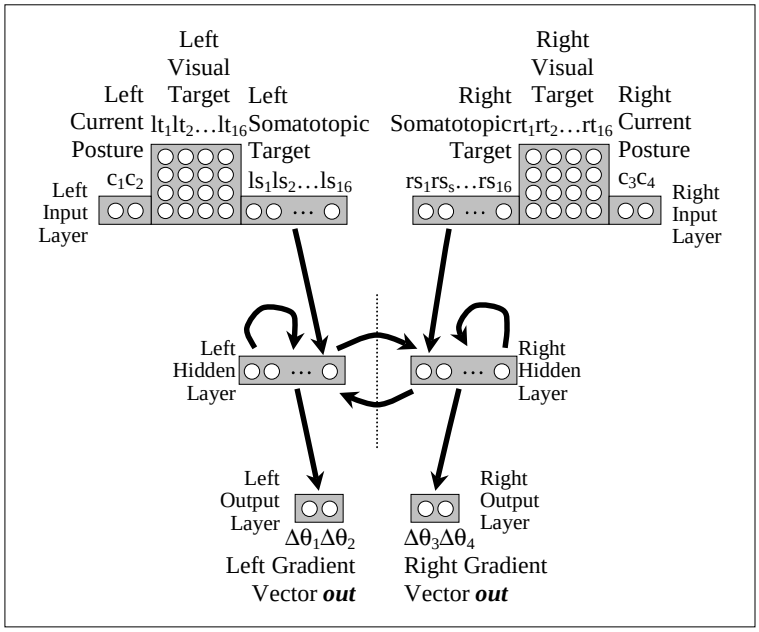


Figure 19. Fully segregated recurrent dual task architecture. Input format simultaneously encodes the current posture, two workspace targets, and the location of a somatotopic target. The appropriate movement task is selected by activating the units appropriate for the target, and setting the units for all other tasks to 0. Output format and hidden layer structure same as in Figure 16.

alone. Requiring a single network to perform both tasks is a much harder problem, but networks with the architecture shown in Figure 17 were still able to adequately learn both tasks. Partially (Figure 18) or fully (Figure 19) segregating the inputs does not have a strong effect on the network's ability to perform either task. In these models the tasks are represented simply through a sensory modality. The networks described here do not behave correctly or even rationally if, for example, the task inputs are summed. The network architecture also oversimplifies the problem of task attention and specification as it occurs in the brain. The models described here do not address this problem in a general way.

4.5 Network training and testing

Standard backpropagation through time (Williams and Zipser, 1995) was used to train all networks. During training, input patterns were presented on every cycle (input dwell = 1 cycle), and weight changes for a given input pattern were computed after a short delay period (training delay = 4 cycles). Weight changes were computed and accumulated in batch mode; weights were updated only at the end of the batch (batch size = 10 cycles). A learning rate of 0.1 and a momentum rate of 0.9 were used for all layers except the output layer, which had a learning rate of 0.001. About 100 million training cycles were sufficient to minimize the sum-squared error.

All networks reported here are of moderate size: unsegregated networks have 32 hidden units, and the segregated networks have two sets of 16 hidden units. Although larger networks with more hidden units (e.g. segregated networks with two sets of 64 hidden units) could eventually achieve a slightly better level of performance in some architectures, they required excessive amounts of training time. It was usually faster to achieve a desired performance by training a smaller network for a longer time. The

addition of large numbers of hidden units also had little discernable effect on the type of network solutions found. Extra units tend to die off during training, maintaining a low constant output that contributes little to the network's behavior.

Each trained network was tested on a set of 10000 novel reaching problems to verify that it had correctly learned to reach postural goals, somatotopic targets, or visual targets. A trajectory was scored as successful if the arm tips reached the target locations in about the expected time with no collisions between the arms. Failed trajectories could be assigned to one of four error types: colliding, premature stopping, out of bounds (exceeding the upper or lower limits for joint angles), and perpetual oscillation. The percentage of trajectories that end in each condition is summarized in (Table 2).

Table 2. Recurrent neural network performances during movement.

(n = 1 for all networks)	Fig.	**reached goal**	**colliding**	**stopped**	**out of bounds**	**oscillating**
posture-to-posture	**13**	96.960%	2.680%	0.000%	0.120%	0.240%
visual target	**14**	98.460%	1.070%	0.000%	0.000%	0.470%
visual target	**15**	96.930%	2.570%	0.000%	0.040%	0.460%
visual target	**16**	95.250%	4.660%	0.000%	0.020%	0.070%
dual task (visual target)	**17**	94.680%	3.420%	1.390%	0.000%	0.510%
dual task (visual target)	**18**	97.000%	2.210%	0.380%	0.030%	0.380%
dual task (visual target)	**19**	94.920%	1.730%	0.380%	0.130%	2.840%
dual task (self-touching)	**17**	72.550%	18.430%	0.070%	8.950%	0.000%
dual task (self-touching)	**18**	81.970%	12.970%	0.000%	5.050%	0.010%
dual task (self-touching)	**19**	74.610%	22.930%	0.420%	1.690%	0.350%

Dual task networks were tested on both the visual target and the self-touching problems, and are thus represented twice in the table. Collisions between the arms were the most frequent kind of error. Collisions could be reduced by adjusting geometric parameters of the body such as the thickness of the arms, and by training for extremely long periods of time. Collision errors are the result of imperfections in the network's learning of the reaching tasks. In most oscillating trajectories, the arm is near its target location, but the oscillation does not bring the arm close enough to the target to reach the termination criterion. Oscillating, premature stopping, and out of bounds conditions are

artifacts of the way the networks are used to generate movement, and tend to be quite rare.

Surprisingly, full connectivity between the somatotopic input units and all hidden units was not essential for networks trained on the dual-task self-touching problem. When these input connections were segregated (Figure 19), network performance was comparable to that of the partially or non-segregated networks. In particular, the hidden units were able to obtain adequate information about the locations of targets on the opposite side (Figure 20).

Another way to evaluate a network's correctness is to trace the trajectories of the arms as they move. Figure 21 traces the arm movements of a fully segregated visual target network (Figure 16) for a variety of movement conditions analogous to those used in neurophysiological experiments. In the "unilateral left" condition, the left arm is instructed to move to visual targets located at 45° intervals around the initial resting posture, and a constant distance of 1/3 the length of an arm segment. The right arm is instructed to hold its position while the left arm moves. Traces of the resulting 8 movements are shown by the dotted lines, which form a star pattern centered on the initial location of the arm. The "unilateral right" condition shows the analogous movement traces for the right arm. The traces demonstrate that this network is reasonably accurate at making unilateral movements of either side. In the "bilateral parallel" condition, both arms are instructed to move in the same direction (e.g. both arms move to the upper right). In the "bilateral opposite" condition, the arms are instructed to move in opposite directions (e.g. the left arm moves to the upper right, while the right arm moves to the lower left). In the bilateral symmetric condition, the arms are instructed to move symmetrically about a vertical line down the middle of the workspace (e.g. the left arm moves to the upper right, while the right arm moves to the upper left). The bilateral interactive condition is like the bilateral opposite condition, except that the

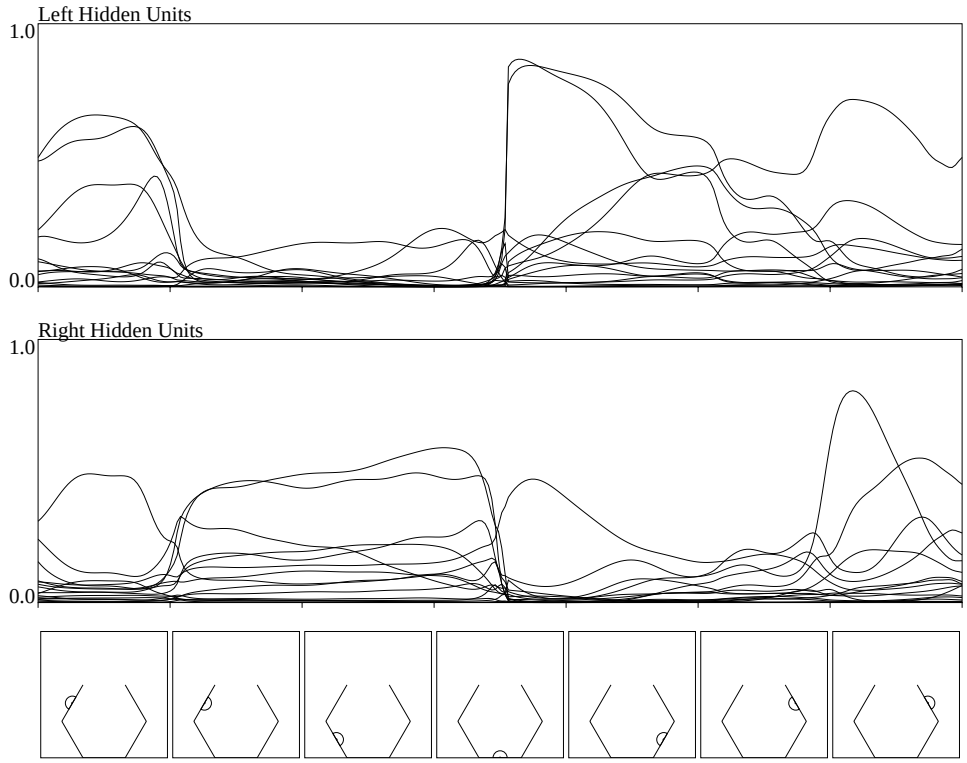


Figure 20. Unit activation as a function of location on the body surface for a segregated network trained on both the posture-to-posture and self-touching tasks. (Top) left units; (bottom) right units.

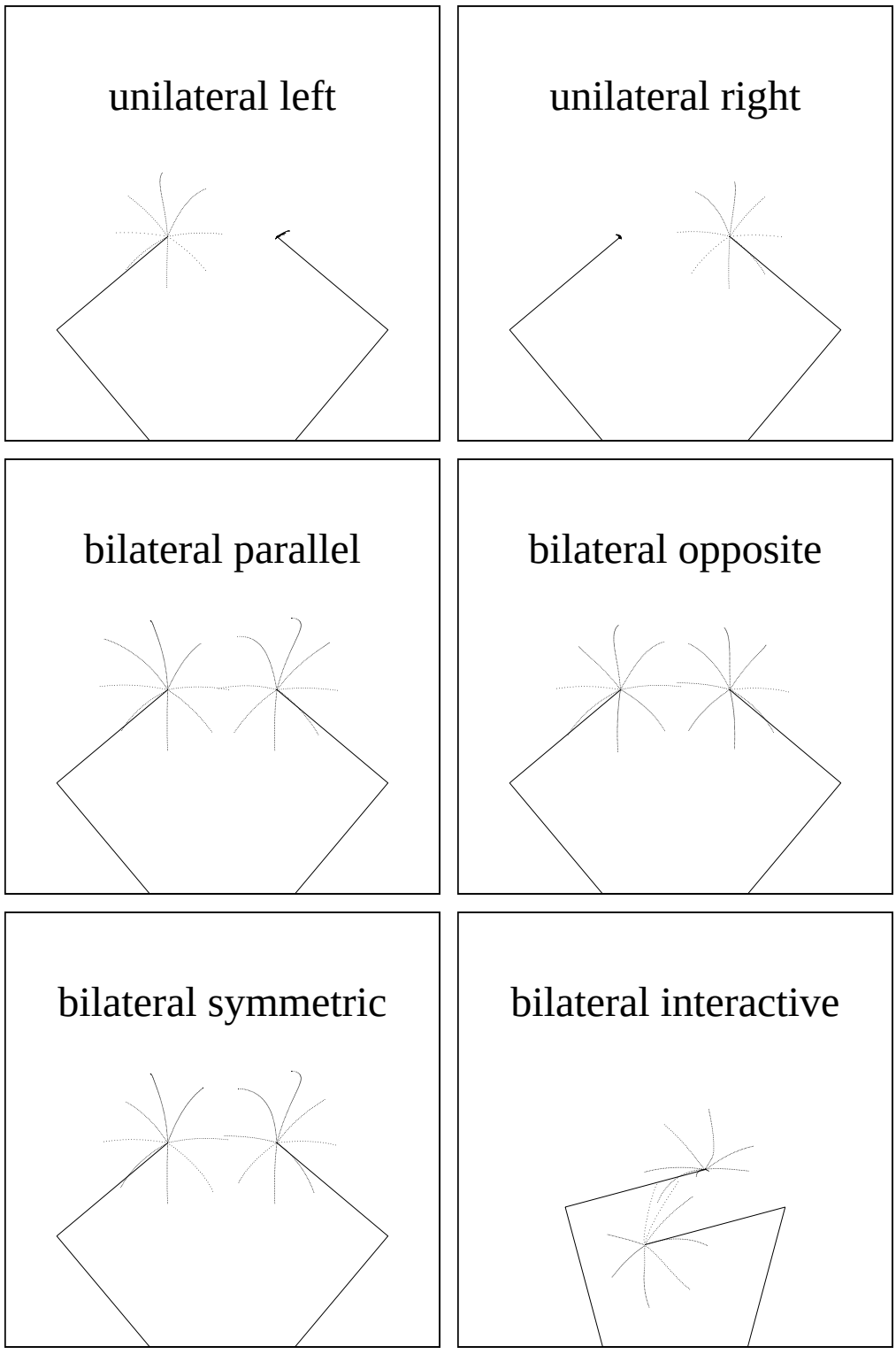


Figure 21. Traces of six different movement conditions generated by the posture-to-visual-target network. The initial resting posture is represented for each condition. Successive dots represent the location of an arm at successive time steps.

starting posture is one in which the arms are deliberately placed near one another. For some bilateral opposite movements (e.g. the left arm moves down, while the right arm moves up) the arms must move around one another to avoid colliding. The deflection of movement causes the warped star pattern. The traces demonstrate that this network is capable of making a variety of coordinated bilateral movements similar to those studied in neurophysiological experiments.

4.6 Network mechanisms

Simple inspection of the recurrent weight matrices shows that all recurrent networks make heavy use of recurrent weights. This is somewhat surprising, since the problems can be solved using strictly feedforward computations, albeit with two hidden layers. Lesions of the recurrent, trans-commissural weights in any segregated network (i.e. Figures 16 and 19) cause catastrophic failure: the network is completely unable to construct a meaningful movement vector and the body posture collapses. This means that each component of the movement vector becomes highly negative, and the arms to fold into one another regardless of the current posture or movement task.

The ipsilateral recurrent connections provide a mechanism by which the network can develop abstract internal representations (Figures 16 and 19). On each time step, each hidden layer constructs a new internal representation based on the current sensory input and its previous representation. The internal representation evolves over time, and is not limited (as in the case of the feedforward networks) to one or two computational steps. The recurrent connections allow the network to determine for itself how many computational steps to use.

The contralateral “commissural” connections provide a mechanism by which the network’s two internal representations cooperate (Figures 16 and 19). Since lesioning

these connections causes the body posture to collapse (the arms fold into one another regardless of the goal), these connections are necessary for either layer to maintain a meaningful internal representation. Each hidden layer by itself is insufficient to perform useful movement; both are required. The network's two hidden layers thus form another larger representation, distributed across layers. This is an interesting solution. One might imagine that the network could have developed a different solution whereby each hidden layer would be capable of making direct movements of the ipsilateral arm. The commissural connections would then mediate deflections of these direct movements to accomplish collision avoidance. Lesion of this connections would then result only in the network's inability to avoid collisions between the arms, not in a complete loss of meaningful movement.

An interesting extension of this work would be to perform a selective lesion, rather than a lesion of the entire weight matrix. There are several possible modes of failure: the network's performance may degrade gracefully with lesion size; the network's performance may collapse when the lesion size reaches some critical threshold; or the network's deficits may be partial, specific, and severe (they may be restricted to movements in one region of the workspace, for example).

If the brain mechanisms supporting coordination receive input information that is segregated into left and right streams, and must combine this information to achieve coordination, lesioning of this connection should impair the ability to coordinate the left and right limbs. Sectioning of the corpus callosum causes numerous motor deficits, but typically does not impair coordinated bilateral movements analogous to the posture-to-posture or visual target tasks. Thus, the bilaterally segregated network described here is an incomplete model of the brain mechanisms of bilateral coordination. There are numerous subcortical routes for the exchange of information between hemispheres, such as through the thalamus, basal ganglia, and the cerebellum. It is possible that such routes

provide enough contralateral information to the hemispheres even in cases of commissurotomy. This presents an opportunity for future modeling, as such connections can be included via architectural modifications to the network. It is also possible that the brain systems supporting coordination receive input information from both sides of the body, or that the input information is already bilaterally integrated. In this case, the partially segregated network is probably more appropriate as a model than the fully segregated network.

These models do not map a functional decomposition of bilateral coordination problems onto their architectural structure. They have not learned to make functional modules out of the architectural modules they were given. This is interesting from a theoretical viewpoint, because it confirms that function and architecture – or function and anatomy – are not isomorphic. In many neurophysiological studies, there is a tendency to assume that anatomically distinct brain areas (e.g. SMA and M1) play different functional roles. This assumption leads many experimenters to study one area in isolation, with the hope of understanding its role as a functional module. An alternative suggested by the segregated model is that functional "modules" may be distributed across several anatomically distinct structures. To understand a functional system, it may be necessary to examine its representative parts in several brain areas. The conjecture is not that functional modules do not exist, but rather that they may be anatomically distributed. A corollary of this conjecture is that multiple functional systems may co-exist in the same brain area. Experimental work focused on one brain area may uncover a bewildering variety of phenomena, which may not be functionally related to one another.

4.7 Biologically relevant hidden unit properties

One common method of analyzing neuron function in motor systems to compute the neuron's preferred direction (Georgopoulos, et. al. 1982). It is known that preferred directions are dependent on movement context, and vary with parameters such as the current body posture and task type (Caminiti, et. al. 1990; Scott and Kalaska, 1994). The model units also demonstrate preferred directions (Figure 22). In the figure, the normalized preferred directions for an example unit are shown as a function of the location of the endpoint of the left arm in the workspace. The preferred directions form a vector field that illustrates how preferred directions change as a function of arm posture.

The preferred directions of all hidden units of the visual target network shown in Figure 16 strongly depend on starting posture. For some units, the preferred direction may change by 180° in different parts of the workspace (e.g. Figure 23, middle upper left unit). Preferred directions tend to point either toward or away from a particular point in the workspace (e.g. Figure 23, middle upper left unit). In larger networks with more hidden units, a larger proportion of units demonstrates this behavior. These preferred or anti-preferred workspace points tend to be clustered in the area of the workspace immediately in front of the body, where the arms are most likely to collide.

The baseline activities of the hidden units also vary as a function of the position of the arm in the workspace. The dynamic range of the unit's baseline usually parallels the dynamic range of its preferred direction, in agreement with previous modeling studies (Moody and Zipser, 1998). As the endpoint of the arm approaches a unit's preferred or anti-preferred workspace point, the unit's activation usually approaches the minimum or maximum of its dynamic range.

Most units demonstrate preferred directions for movements of either arm (e.g. Figures 23 and 24, upper right unit). The model allows us to make the interesting

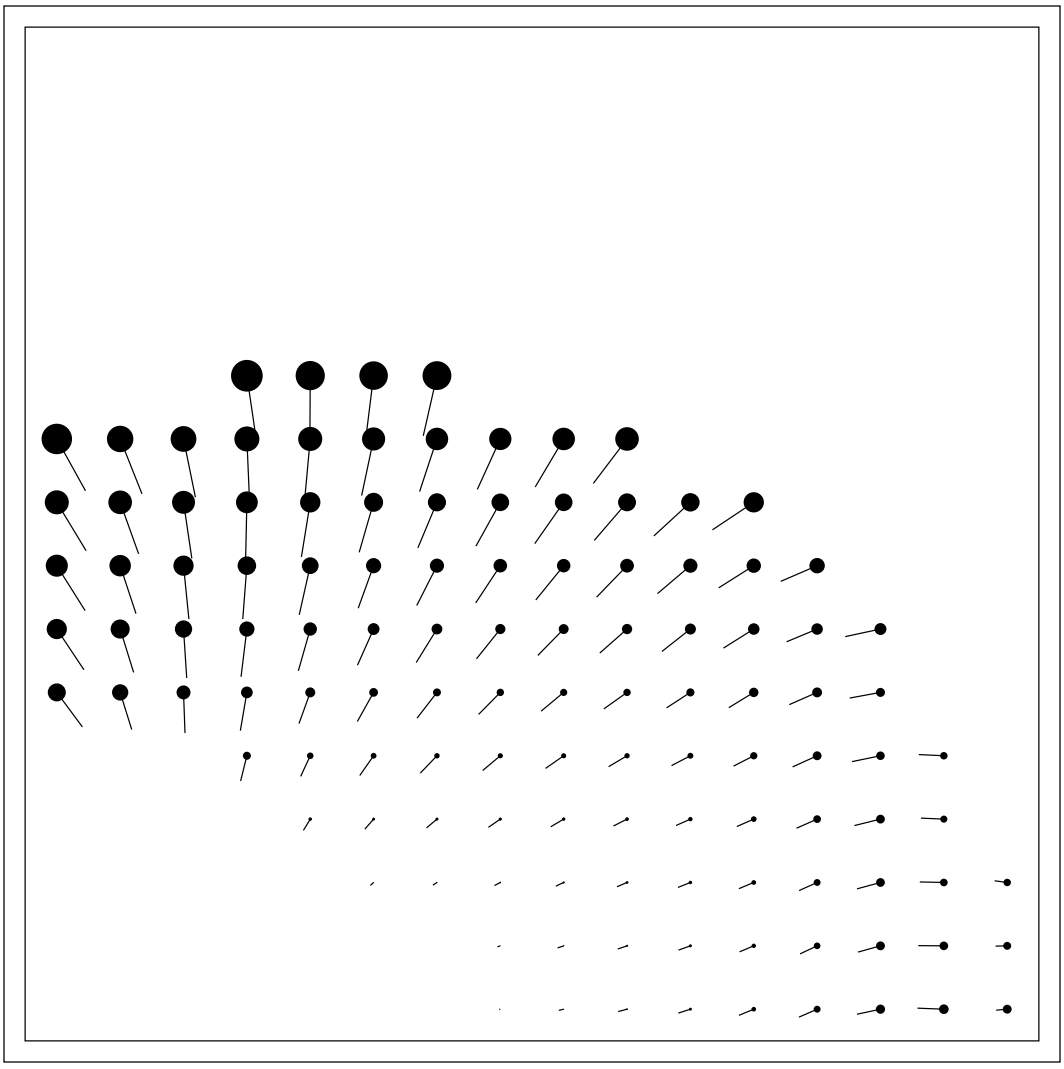


Figure 22. Example preferred directions for one unit. The diagram shows the preferred directions of a single unit as a function of the location of the endpoint of the left arm in the reachable workspace. Line length represents the normalized dynamic range, and spot site represents the normalized baseline activation. Lines form a vector field that spans the reachable range of the workspace. Unit from the left hidden layer of the fully segregated visual target network depicted in Figure 16.

pv2-vf0.eps (unilateral_left)

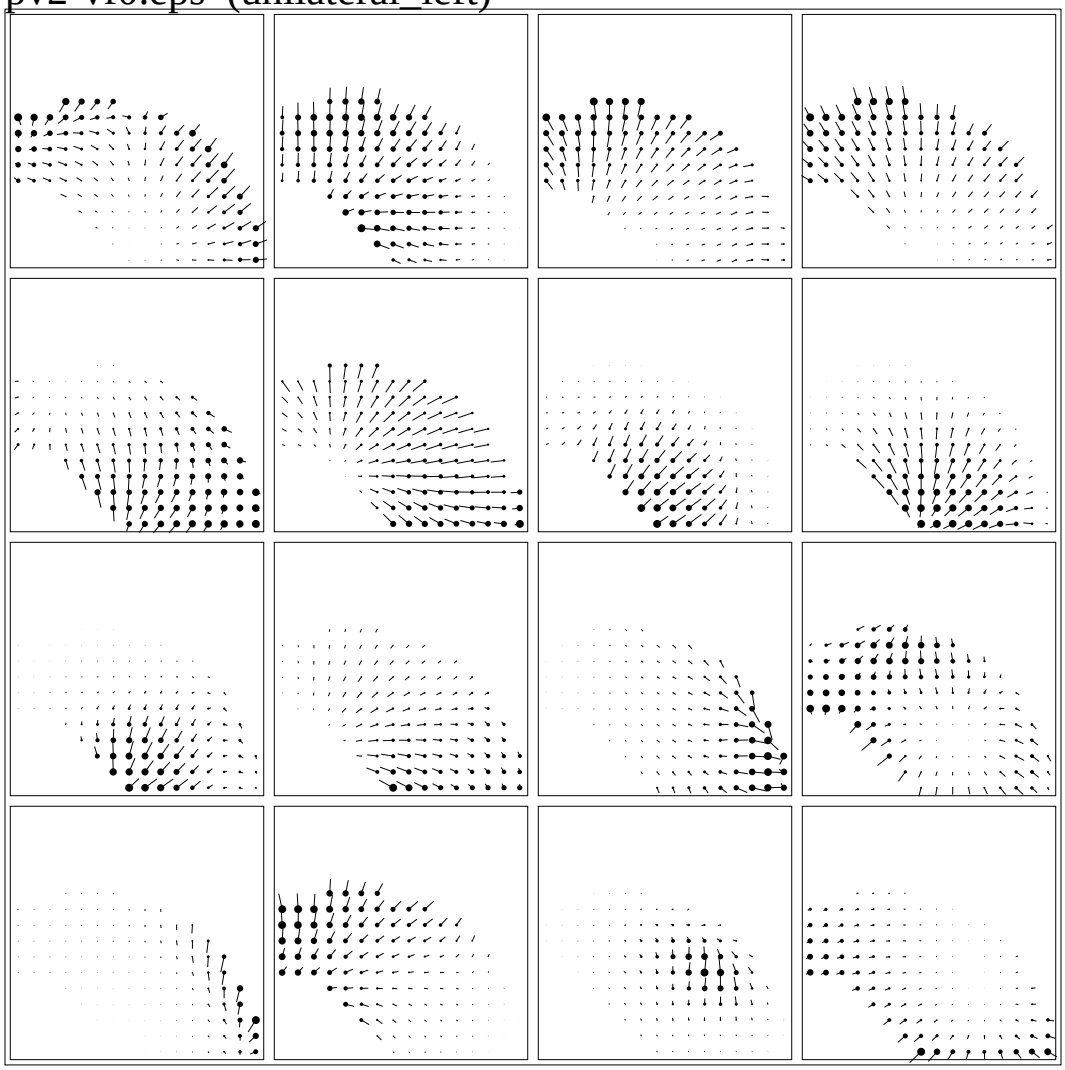


Figure 23. Left unit preferred direction vector fields for unilateral left movements. Each graph represents the preferred directions of one hidden unit. The graphs show the preferred direction vector fields for movements of the left arm, with no right arm movement (unilateral left movement condition). Vectors are plotted with respect to the position and movement of the endpoint of the left arm. Units from the left hidden layer of the fully segregated visual target network depicted in Figure 16. Figure format otherwise the same as Figure 22.

pv2-vf1.eps (unilateral_right)

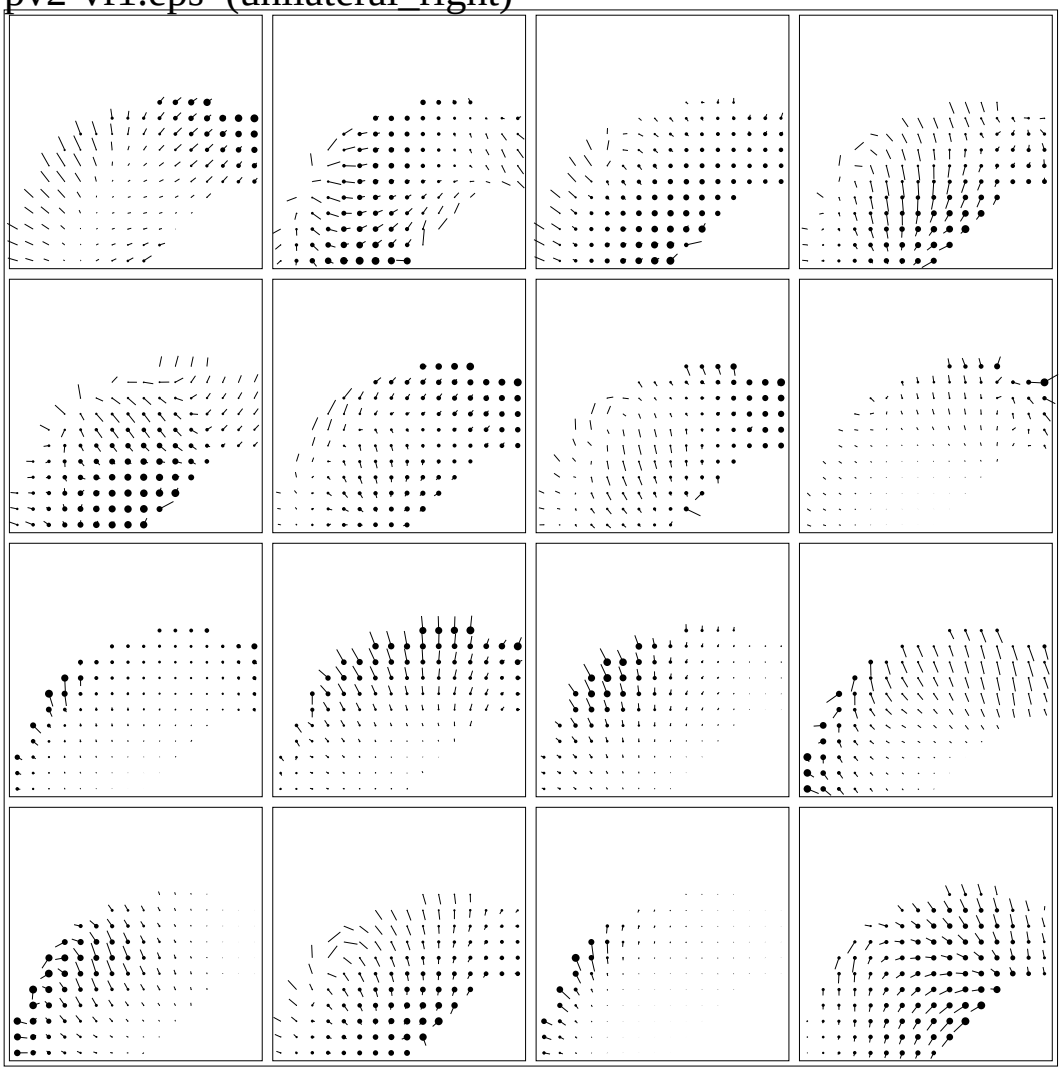


Figure 24. Left unit preferred direction vector fields for unilateral right movements. The graphs show the preferred direction vector fields for movements of the right arm, with no left arm movement (unilateral right movement condition). Vectors are plotted with respect to the position and movement of the endpoint of the right arm. Units from the left hidden layer of the fully segregated visual target network depicted in Figure 16. Figure format otherwise the same as Figures 22 and 23.

pv2-vf2.eps (bilateral_parallel)

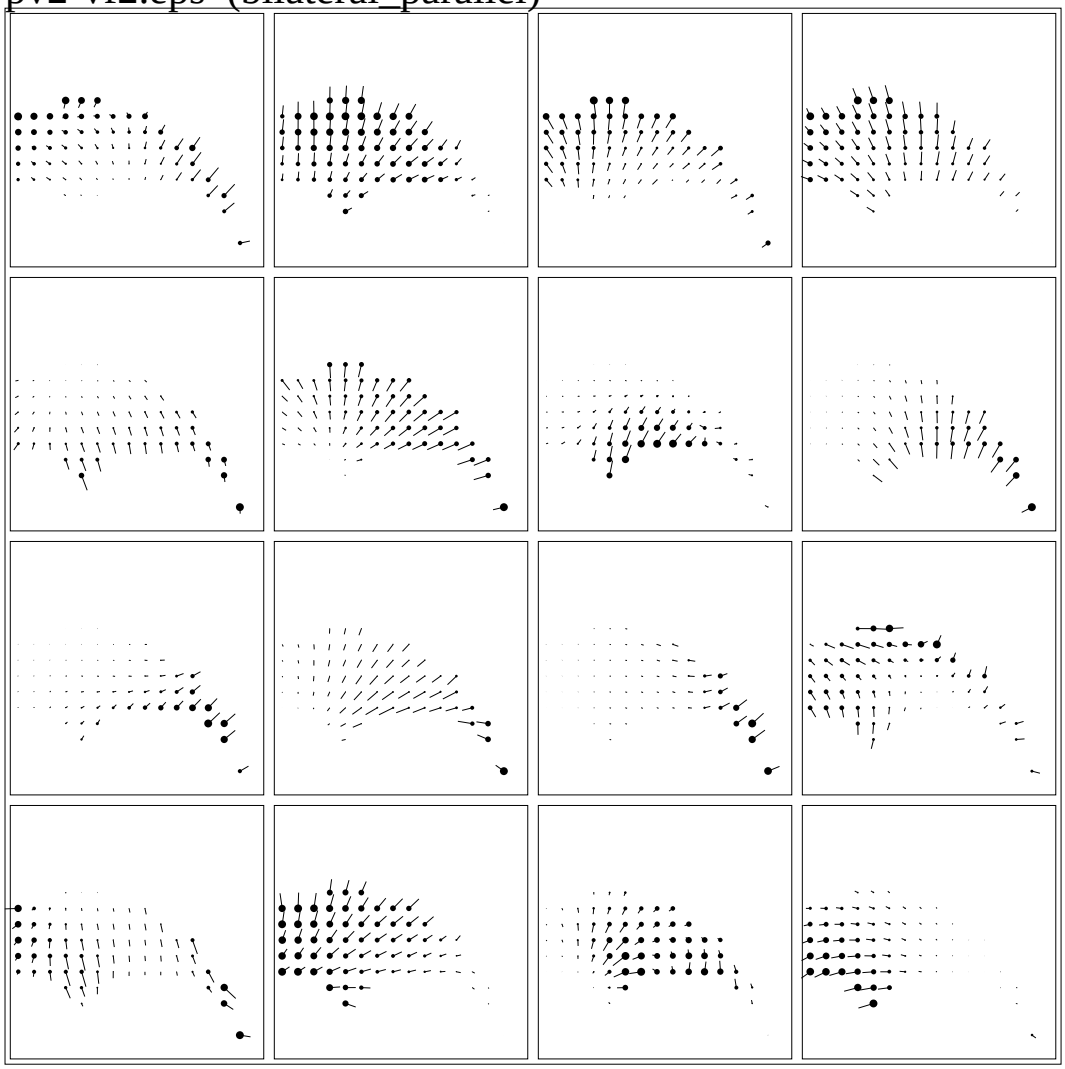


Figure 25. Left unit preferred direction vector fields for bilateral parallel movements. The graphs show the preferred direction vector fields for arms moving in the same direction (bilateral parallel movement condition). Vectors are plotted with respect to the position and movement of the endpoint of the left arm. Units from the left hidden layer of the fully segregated visual target network depicted in Figure 16. Figure format otherwise the same as Figures 22 and 23.

pv2-vf3.eps (bilateral_opposite)

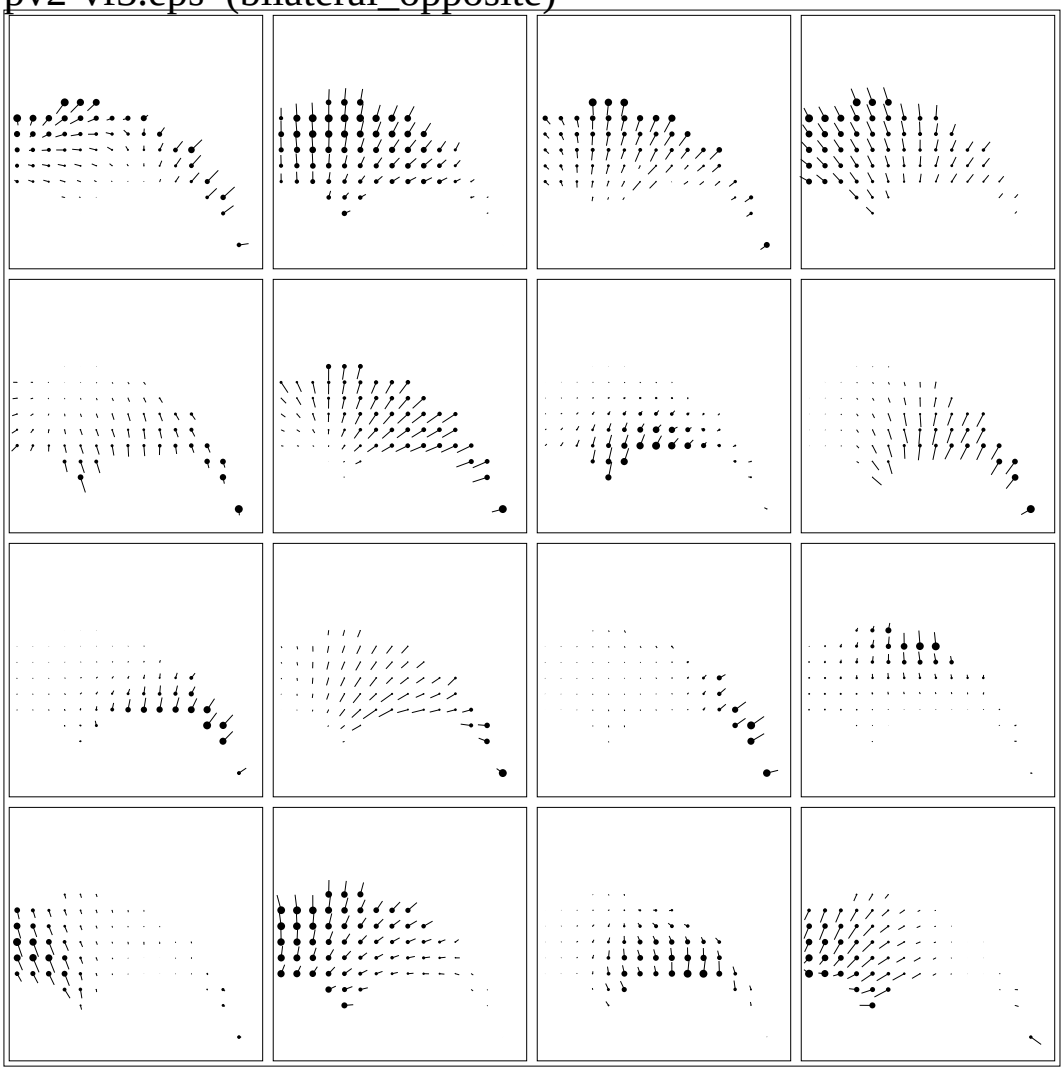


Figure 26. Left unit preferred direction vector fields for bilateral opposite movements. The graphs show the preferred direction vector fields for arms moving in opposite directions (bilateral opposite movement condition). Vectors are plotted with respect to the position and movement of the left arm. Units from the left hidden layer of the fully segregated visual target network depicted in Figure 16. Figure format otherwise the same as Figures 22 and 23.

pv2-vf4.eps (bilateral_symmetric)

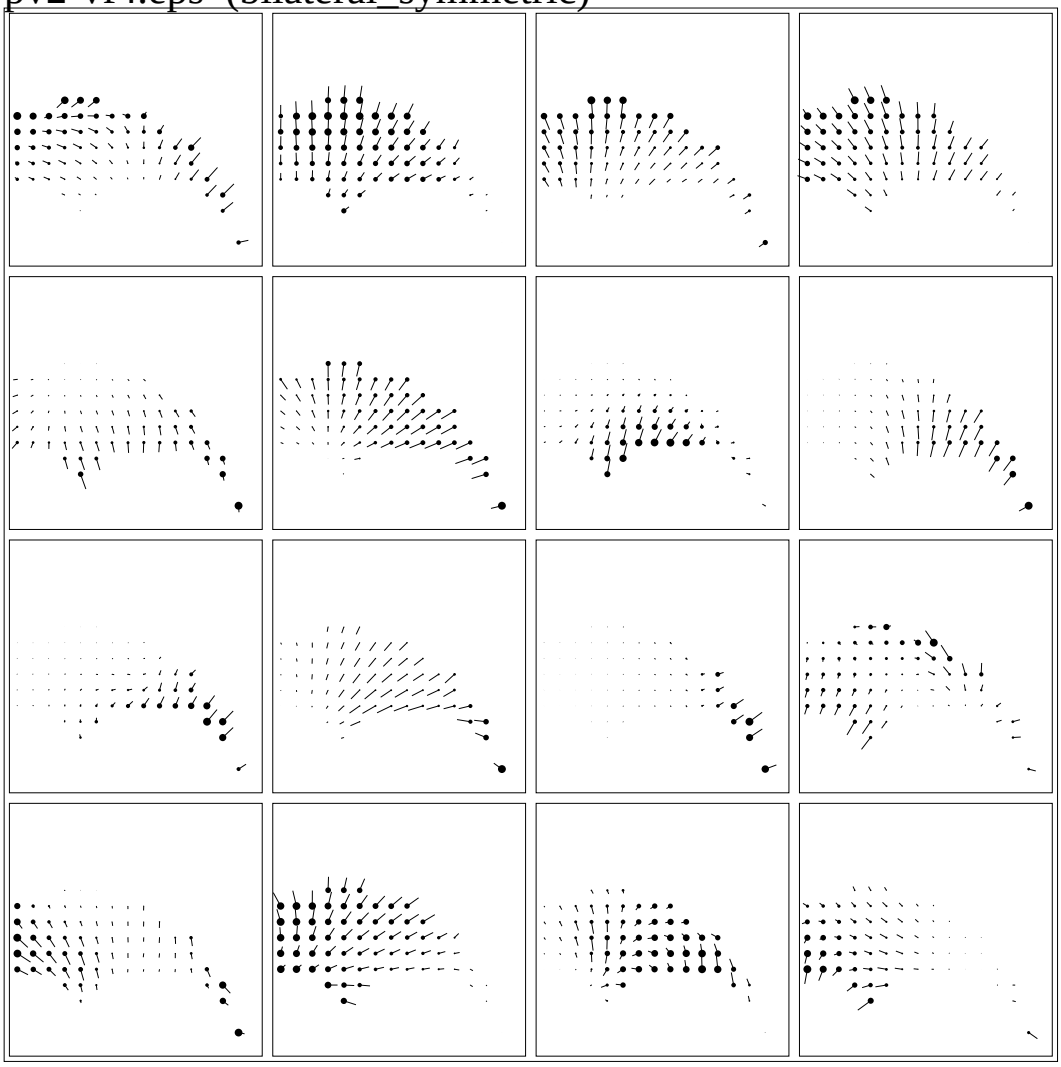


Figure 27. Left unit preferred direction vector fields for bilateral symmetric movements. The graphs show the preferred direction vector fields for symmetric movements of the arms (bilateral symmetric movement condition). Vectors are plotted with respect to the position and movement of the endpoint of the left arm. Units from the left hidden layer of the fully segregated visual target network depicted in Figure 16. Figure format otherwise the same as Figures 22 and 23.

pv2-vf5.eps (bilteral_interactive (opposite))

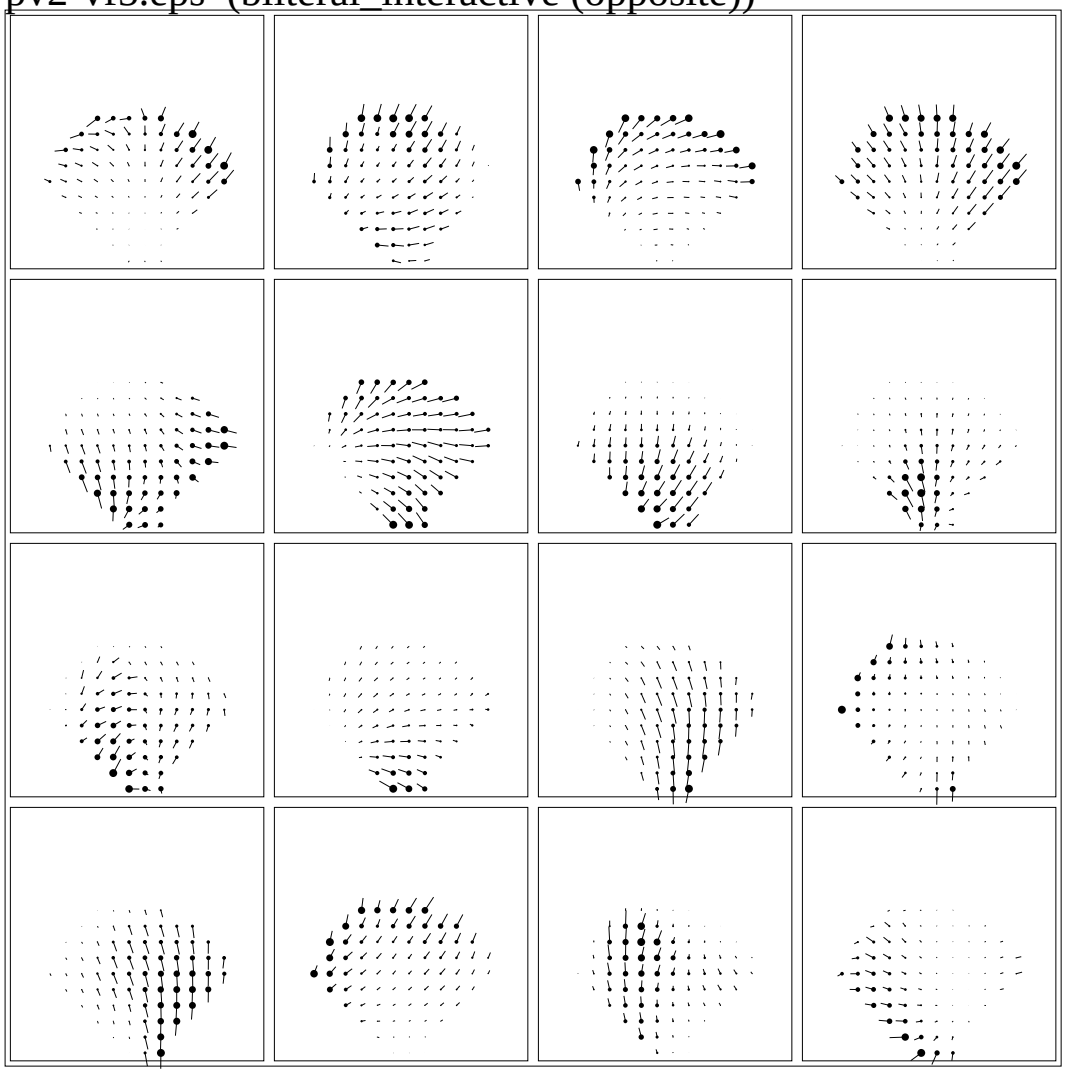


Figure 28. Left unit preferred direction vector fields for bilateral interactive movements. The graphs show the preferred direction vector fields for arms moving in opposite directions, with the initial arm posture near collision at all possible workspace points (bilateral interactive movement condition). Vectors are plotted with respect to the position and movement of the endpoint of the left arm. Units from the left hidden layer of the fully segregated visual target network depicted in Figure 16. Figure format otherwise the same as Figures 22 and 23.

prediction that neurons in both motor cortices will also demonstrate preferred directions for movements of either arm. This prediction has been supported by data gathered in early studies of bilateral movements (Donchin, et. al. 1997). Most neurophysiological studies have examined preferred directions with respect to movements of one side only. In the model, there does not appear to be any strong correlation between the preferred directions for movements of the left and right arms for similar regions of the workspace (Figures 23 and 24).

The preferred directions are also modulated by task context. The baseline activities of units during ipsilateral movements (Figure 23) are less varied and complex than they are during contralateral movements (Figure 24). This may be because all units are at least one step farther away from contralateral input information. When units participate in contralateral movements, their contribution may be more complex and nonlinear. There are minor variations in the preferred direction vector fields across most of the unilateral (e.g. Figure 23) and bilateral (e.g. Figure 25) movement conditions in most units. However, a few units demonstrate large variations in the preferred direction vector fields between different movement conditions (e.g. Figures 25 and 26, lower right unit). The preferred directions of some units change by more than 45° when compared to the preferred directions in the same workspace locations for the other movement contexts (Figure 29). The model allows us to predict that preferred directions may change when multiple parts of the body must be moved in coordination with one another.

The model allows us to predict the existence of neurons in the motor cortices that demonstrate preferred directions that are sensitive to the context of a movement. This prediction could be tested in an experimental paradigm that allows the arms to interact in the same region of the workspace. The human or animal subject could be instructed to perform the same bilateral opposite reaching or manipulation movement that is currently

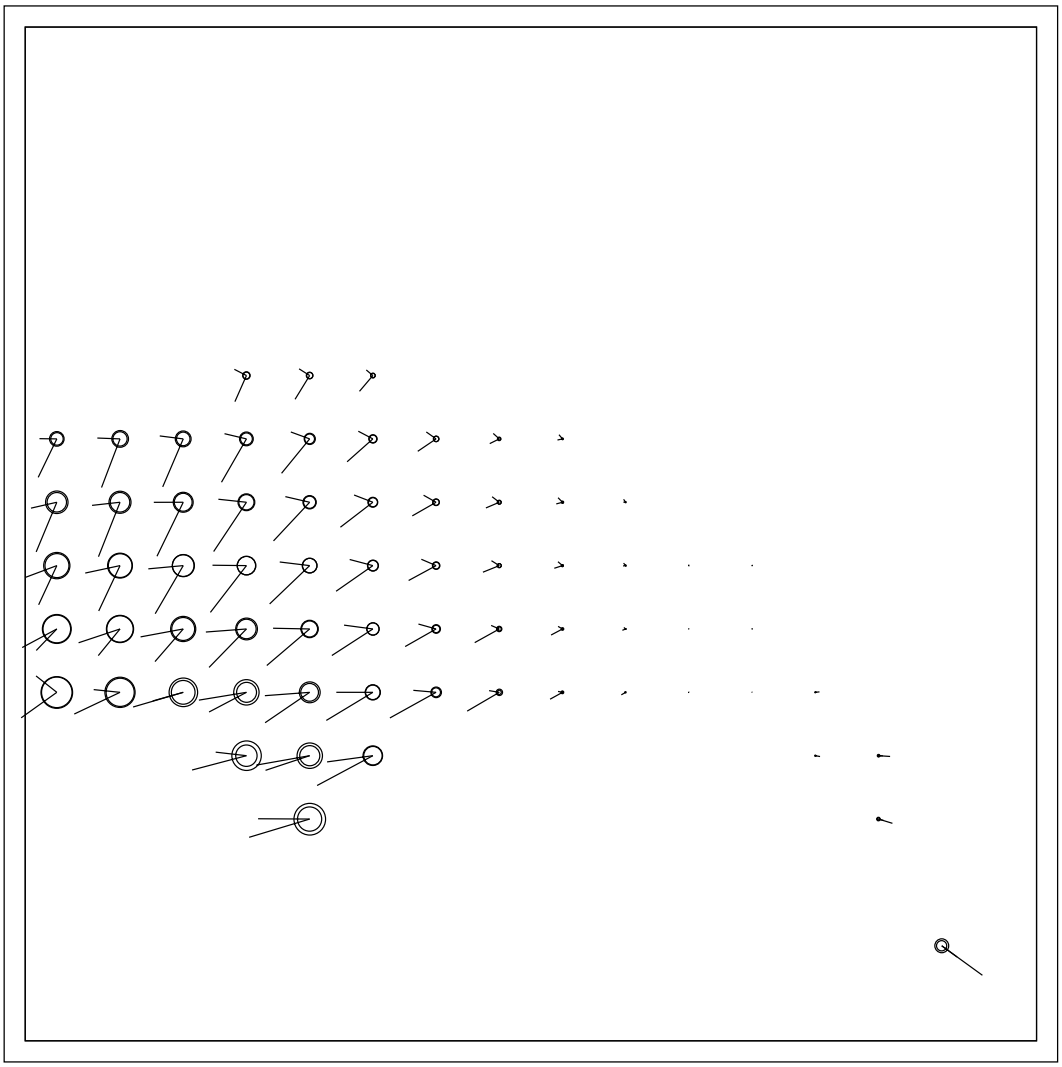


Figure 29. Change in preferred direction between two movement tasks. The preferred direction vector fields of the lower right unit from Figure 25 (bilateral parallel movements) and Figure 26 (bilateral opposite movements) are superimposed to emphasize their differences, particularly in the left half of the workspace.

performed, but from a different starting or resting posture, where the arms are near one another.

The model's state includes both the activities of the network's hidden units and the current posture of the body. Each movement task defines one or more attractors in the model's activation space, where each attractor is either an intermediate goal or a final goal. Attractors corresponding to intermediate goals are generally not stable, and give way to other attractors corresponding to subsequent intermediate goals or a final goal. Final goal attractors are stable, and have basins of attraction that span a substantial region of the high dimensional space of model states. When the state enters one of these regions, it may either stop or oscillate in the region indefinitely. The stopping criteria described above are an attempt to regulate the effects of these attractors.

Examination of the model state over time demonstrates that the hidden units have developed an interesting functional decomposition (Figures 30 and 31). Consider the fully segregated visual target network model. Initially, the body is fixed at the starting posture and network's state is allowed to settle. Then the visual targets are presented. This causes an abrupt change in the network's internal state, as the hidden units begin to compute the proper movement vector. Figure 30 illustrates the model state over a trajectory that does not involve collision avoidance. The units generally take on near-constant values. Figure 31 illustrates the model state where the network must avoid a collision. Several units take on larger values, and most units undergo a shift in activation midway through the trajectory, at the point when the right arm is passing under the end of the left arm. The behavior is similar to that seen in the feedforward self-touching networks. The difference is that in the recurrent network, the functions of the layers are mixed: units that compute a movement deflection coexist in the same hidden layer with units that compute movement direction. This is different from the feedforward network,

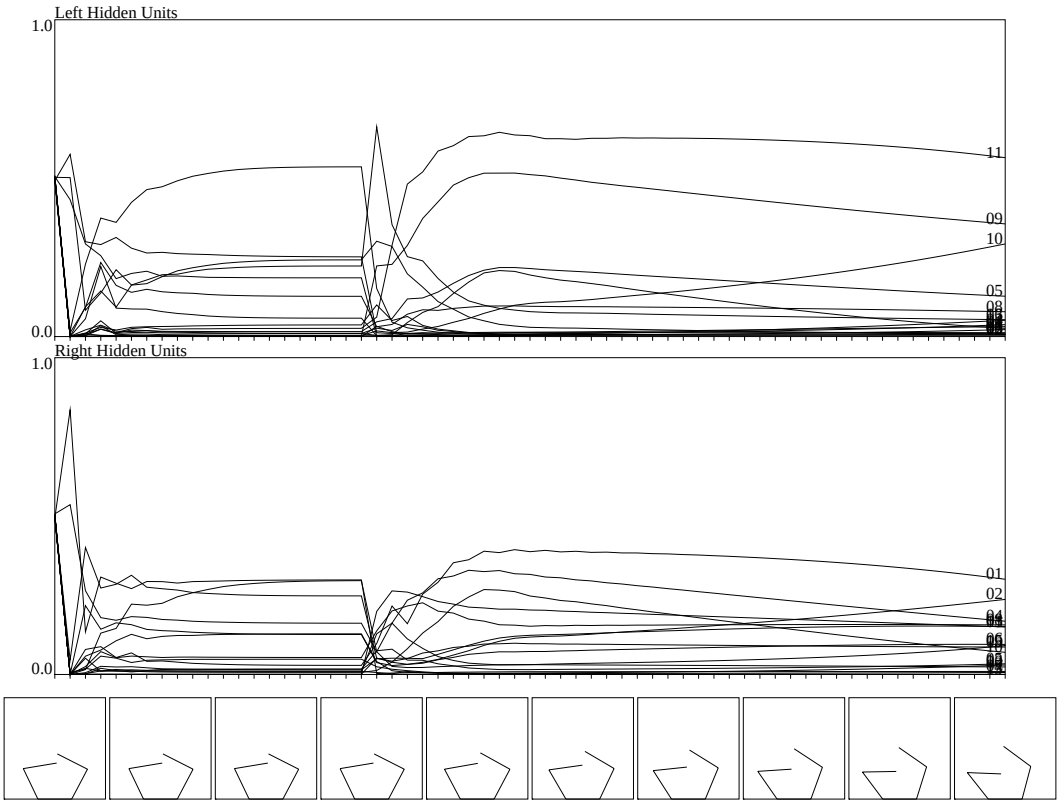


Figure 30. Posture-to-visual-target hidden unit activations over trajectory time. Time steps are indicated by tick marks at the bottom of each graph. The network units start at random values tightly distributed around 0.5. The current posture is presented immediately, and the network is allowed to settle for some number of steps before the goal (visual target) is presented (about time step 20). Units from the left hidden layer of the fully segregated visual target network depicted in Figure 16.

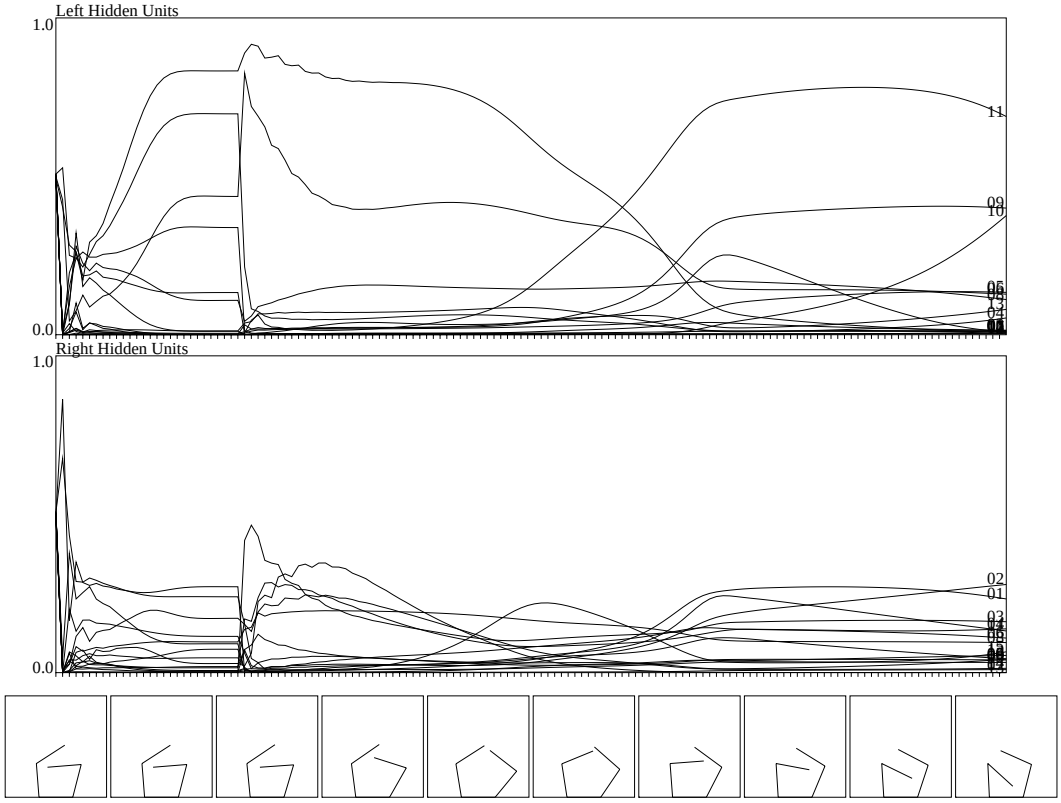


Figure 31. Posture-to-visual-target hidden unit activations over trajectory time. Figure format same as Figure 30. Units from the left hidden layer of the fully segregated visual target network depicted in Figure 16.

where units that computed the deflection were in a different layer from the units that computed the direction.

Many neurons in primate primary and supplementary motor cortices demonstrate participation in both unilateral and bilateral movements (Donchin, et. al. 1998). The behaviors of the model units (Figures 32 and 33) resemble those of neurons in these cortices recorded during unilateral and bilateral manipulation tasks (Figure 1). Figure 32 shows the normalized activations of left hidden units to unilateral and bilateral movements to the left and right. The hidden unit activity during a bilateral movement is the weighted sum of its responses to each unilateral component. For example, the activity of the last unit to a bilateral movement where both arms move to the left (Figure 32, bottom row, third column) is the weighted sum of its activities during unilateral movements of the left arm to the left and the right arm to the left (Figure 32, bottom row, first and second columns). Figure 33 shows similar behavior for units on the right side. This behavior plausibly resembles that of the second neuron in Figure 1. Some real neurons are primarily or exclusively bilateral, and participate only when both arms move (Figure 1, first neuron). No exclusively bilateral units were found in these models, however. The models can thus only partially, but not completely, account for the experimental data.

The positional response functions (Kettner, et. al. 1993) of the units in the segregated visual target network demonstrate sharply bounded regions of activation (Figure 34). In the figure, activation of an example left unit is plotted with respect to the entire range of possible postures. The network is allowed to settle after each posture is presented. No visual target is presented, however. The range of the workspace reachable by the left arm is approximately outlined by the grayscale region of unit activity for each right arm posture. The unit may represent an abstraction such as the range of the workspace that is unsafe for the left arm.

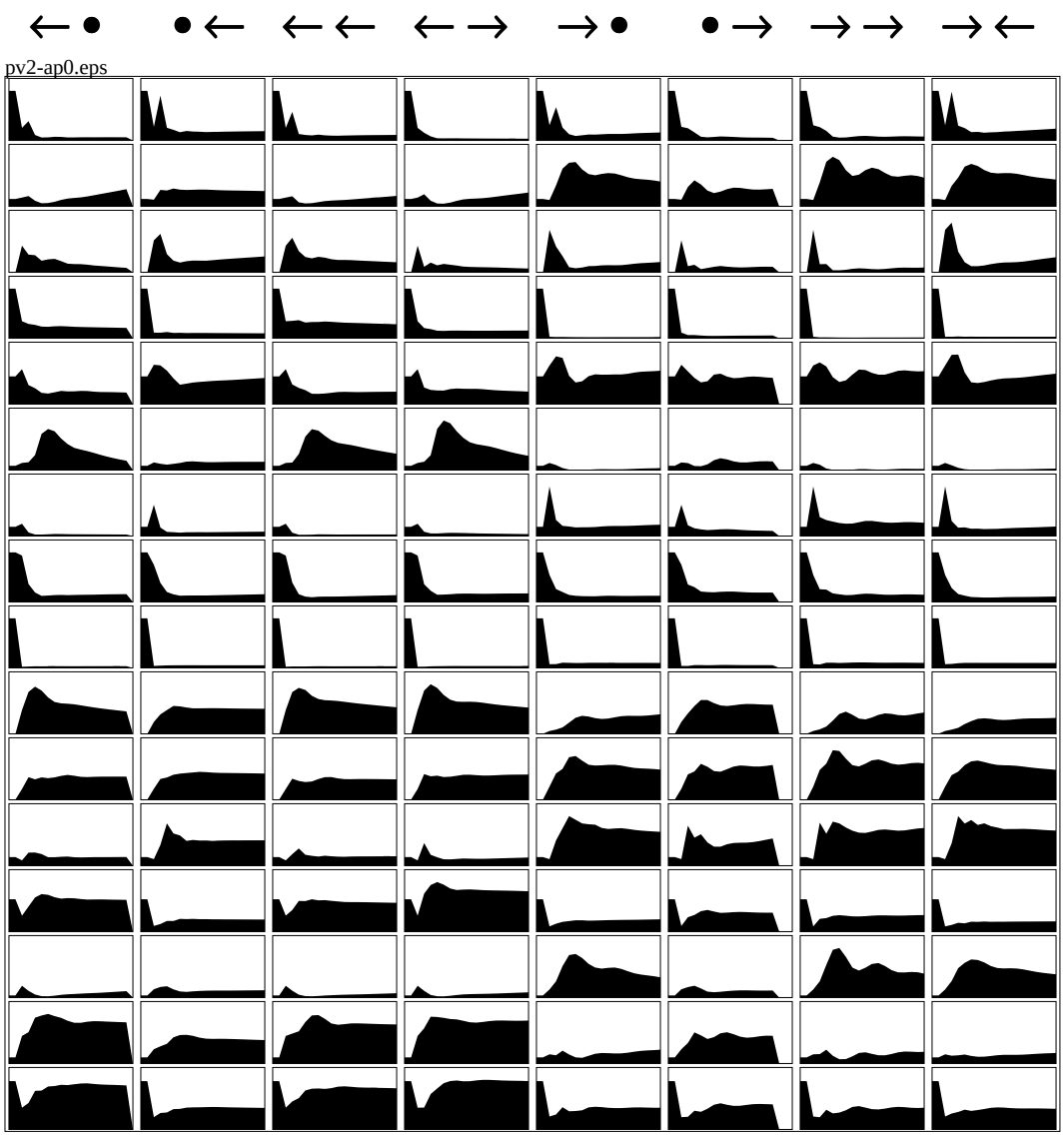


Figure 32. Left hidden unit responses for 8 different movements. Each row shows the normalized activity of one hidden unit for 20 time steps. Each column represents a different movement condition. Left and right symbols at the top of each column represent actions of the left and right arms respectively (i.e. arm moves left, arm moves right, or arm holds). At the beginning of each movement, the network is first allowed to settle at a resting posture, with no movement goal. The sudden change in activity at the beginning of each diagram indicates when the target(s) are presented. Compare to Figure 1 and Donchin, et. al. 1998.

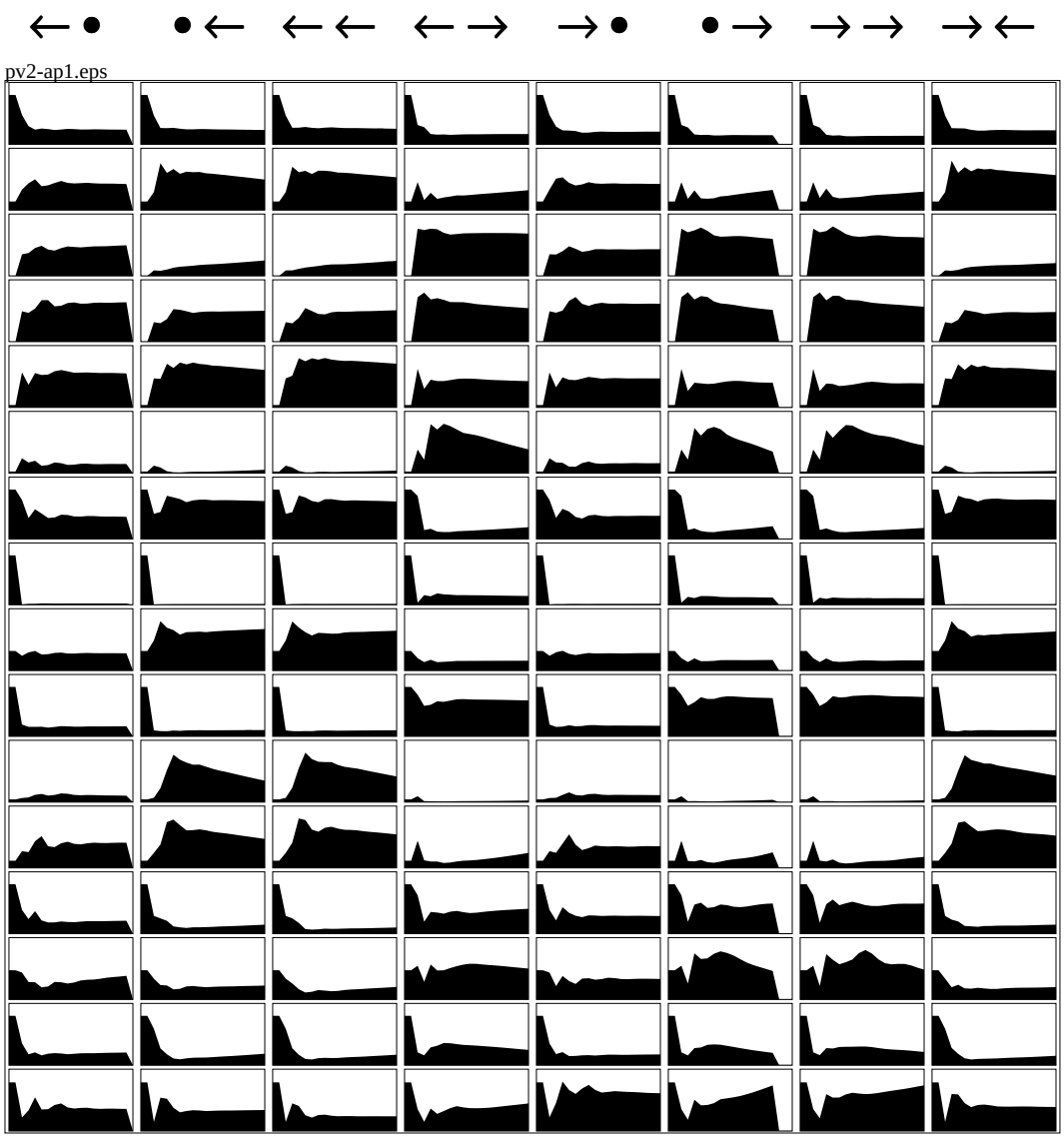


Figure 33. Right hidden unit responses for 8 different movements. Figure format same as in Figure 32. Compare to Figure 1 and Donchin, et. al. 1998.

pv2-uapm0.eps

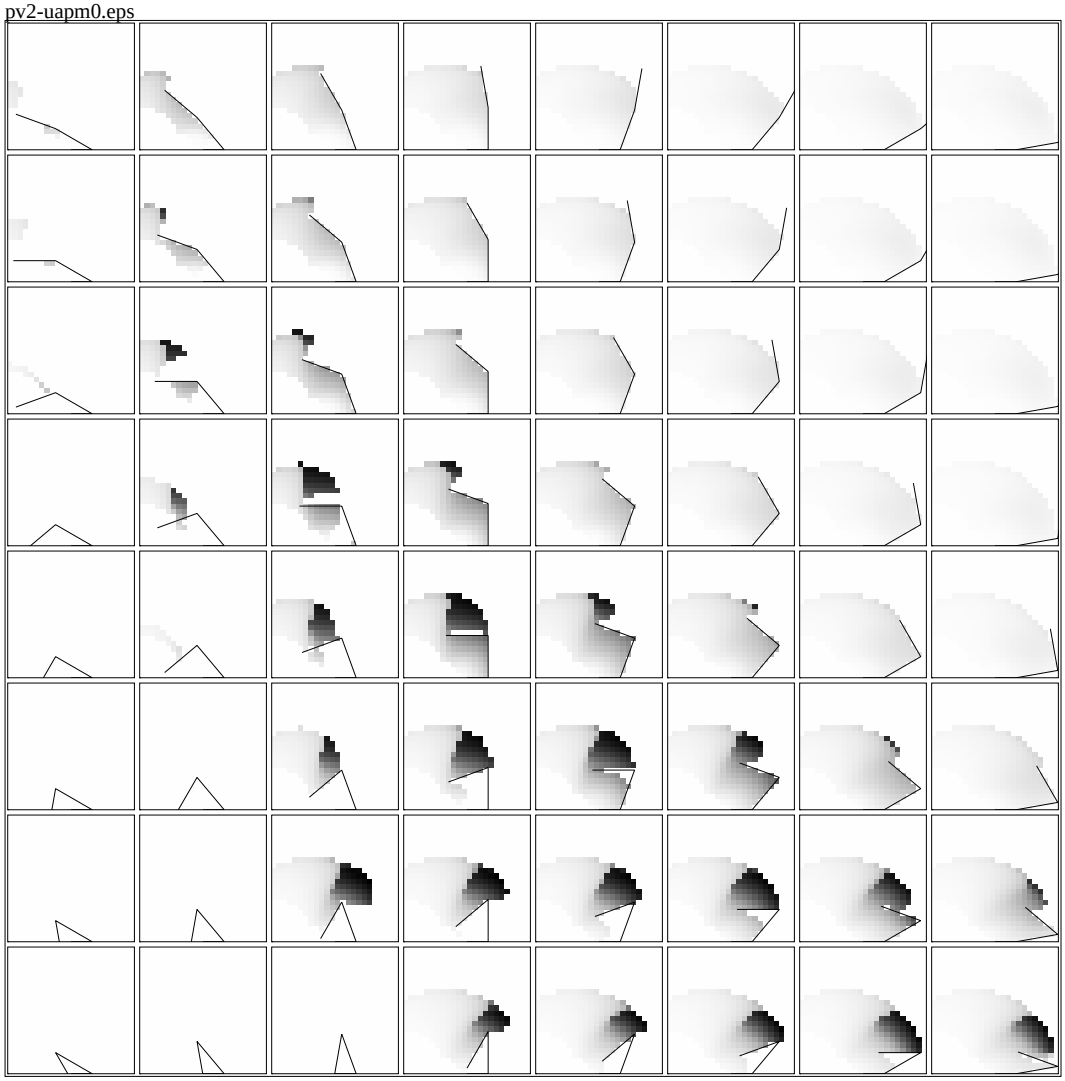


Figure 34. Activation of one unit in the visual target network as a function of posture. The graph shows the normalized activation profile of an example left hidden unit with respect to the workspace locations of both arms. Within each square, unit activation is represented as a function of the position of the left arm in the workspace, with the right arm fixed in the position shown. Across squares, unit activation is represented as a function of the position of the right arm in the workspace. Dark gray represents regions of high activity, light gray low activity. Regions of the workspace that are unreachable or that would result in collisions between the arms are left blank.

The hidden units of the visual and somatotopic target network have bimodal receptive fields that are modulated by posture (Figures 35 and 36). In the figures, normalized activity of the left units is plotted as a function of the workspace location of a visual target presented to the left side. In the Figure 35 the posture is relaxed, with the arms far apart; in Figure 36 the posture is contracted, with the arms close to one another. The change in posture modulates the intensity of some of the fields, and alters the fundamental shapes of others. There seems to be no relation between the locations of visual receptive fields in the workspace, and the locations of somatotopic receptive fields mapped onto the arm locations in the workspace. In other words, the model does not demonstrate coherent bimodal receptive fields analogous to those found in premotor cortex (Graziano, et. al. 1994; Graziano, et. al. 1997). This is probably due to the fact that the network was not trained to explicitly correlate the locations of targets in the workspace with targets on the arms. The internal representations used to solve the visual target and somatotopic target tasks are overlapping but independent.

Based on its failure to account for an important aspect of the experimental data, this dual-task architecture can be ruled out as a model of visual and somatomotor integration. In order for a model to account for neurons with localized, mobile, and bimodal receptive fields, a conceptual change in the network's architecture and task will be required. One conjecture is that the network needs to be trained to report the somatotopic location nearest to a visual target in the workspace. In order to report this location independently of posture, the network would be forced used the current posture to translate points in workspace coordinates to points in the somatotopic coordinates. The network might do this in a chunked fashion, using hidden units to identify regions of workspace that map onto regions of the body surface. These regions would correspond to bimodal receptive fields of the type observed experimentally (Graziano, et. al. 1994; Graziano, et. al. 1997).

pvst2-bm0.eps (Posture 0, Left Targets)

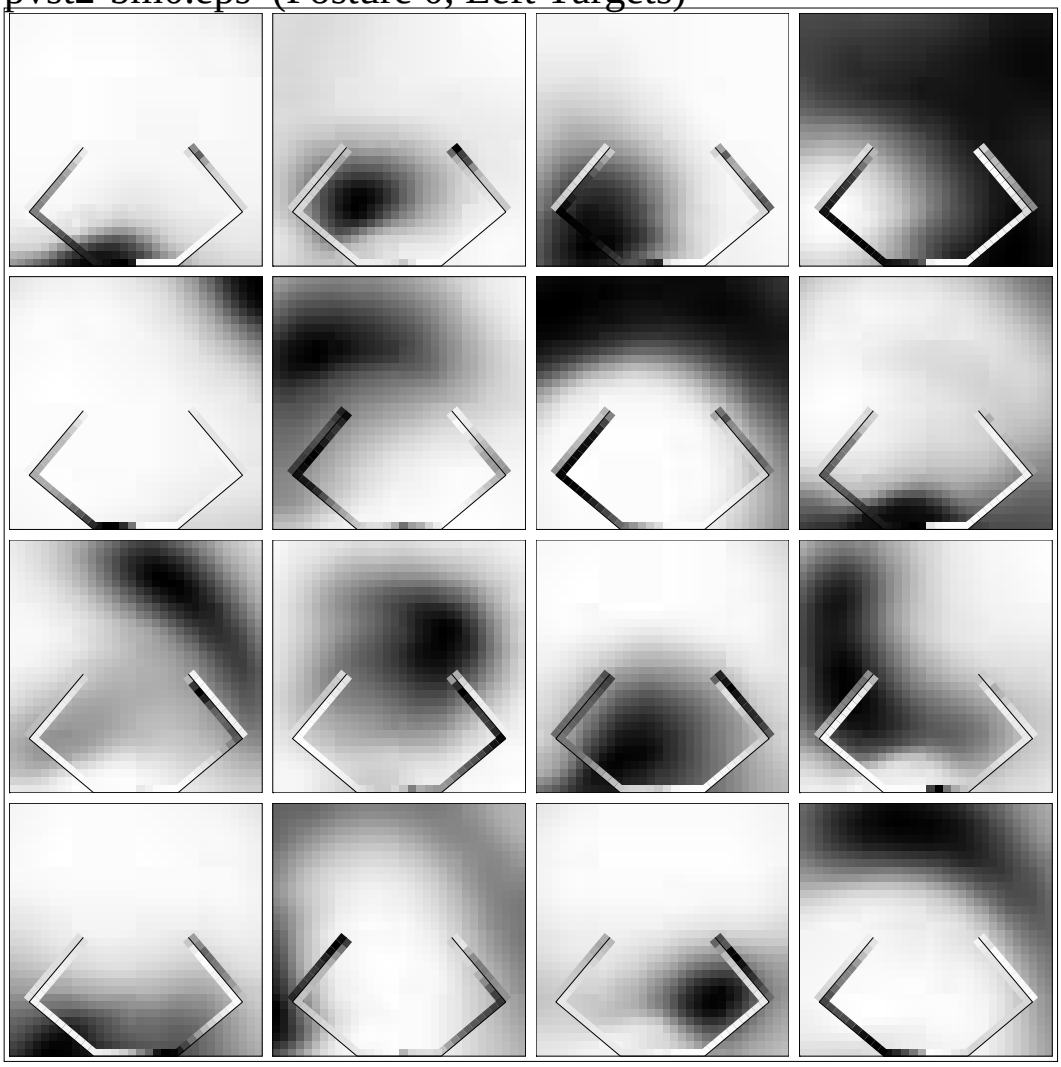


Figure 35. Bimodal receptive fields. Each square represents the normalized activity of one unit with respect to a left-arm visual target in the workspace. Superimposed is a grayscale plot of the unit's activation with respect to targets on the body surface, projected onto the workspace according to the (fixed) input posture. The combination shows the complicated nature of the bimodal receptive fields. Compare to (Graziano, et. al. 1994; Graziano, et. al. 1997).

pvst2-bm2.eps (Posture 1, Left Targets)

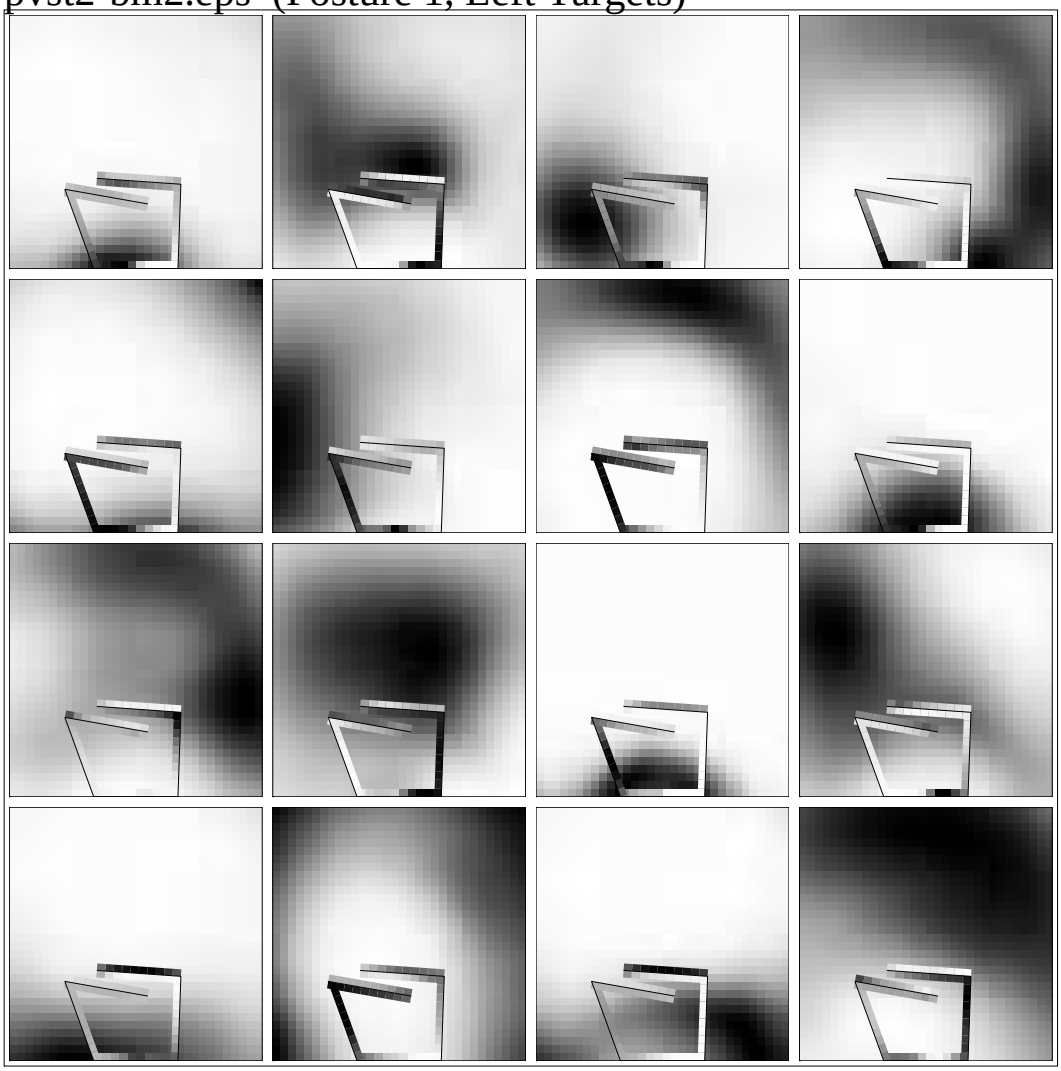


Figure 36. Bimodal receptive fields. Figure format same as Figure 35.

4.8 Discussion

The hidden units of the recurrent network do not represent collisions or obstacles explicitly the way hidden units in the feedforward network do. The units implicitly represent parts of the collision space via the dynamic ranges of their preferred directions. The preferred direction vector field leads to preferred or anti-preferred workspace points that are clustered in the regions of the workspace where collisions are most likely. This is a very interesting prediction for the neural mechanism underlying reaching, as most proposed reaching systems achieve collision avoidance by explicitly representing obstacles.

The change of a unit's preferred direction as a function of arm posture suggests an additional interpretation. If a unit were clamped to the maximum of its dynamic range, it would tend to move the endpoint of the arm toward its preferred direction. During normal operation, the contributions of the units are summed. The arm's position would "flow" with the field in a gradient descent fashion. The network superimposes and weights these flow fields to achieve movement in any direction. This may be a kinematic complement to muscle force equilibrium attractors in the spinal cord (Bizzi, 1993; Mussa-Ivaldi and Giszter, 1992).

One of the benefits of the identification paradigm is that it allows the theorist to develop plausible models with few assumptions about the neural architecture under investigation. The network optimization procedure discovers its own neural algorithm, one that is often different from what might imagined. To improve the model, however, it is necessary to constrain its architecture in plausible ways. One way to do this is to decompose the coordination task into subtasks, and enforce functional and architectural modularity. A module may consist of a network trained to perform one simple task, such as collision avoidance or goal-seeking. The outputs of these modules are combined by

yet another module that judges which is more relevant in the current movement context. Another way to constrain the model architecture is to use a layering technique. One set of networks may be trained to solve a simple problem such as direct one-arm reaching. The weights of these networks are tenured. Another set of networks is layered on "top" of these, and trained modify the output of the system to perform a more complicated behavior such as collision avoidance. Further layers of networks may enrich the repertoire of behaviors. This approach to modeling is inspired by the brain's layered anatomy: "high" level brain systems (e.g. cortex) are often layered on "low" level (e.g. spinal cord) systems. Both are required for complicated movements.

4.9 Conclusion

This thesis has described feedforward and recurrent networks capable of solving a variety of coordination problems. The networks have been tested in ways analogous to neurophysiological experiments to better understand their internal mechanisms. Some of the unit properties, such as the presence and variation of preferred directions, are consistent with properties of neurons in the motor cortices, especially the primary motor cortex. But the networks do not account for some of the more recent data on bilateral coordination, such as the existence of neurons that are exclusively bilateral.

These models are a clear example of how the identification paradigm can be used to generate neurobiologically plausible models. Model behaviors are robust under a wide variety of parameter and training variations. Although the model conjectures are reasonable, they do not account for all of the known data. This implies that non-trivial modifications to the model are necessary.

Bibliography

Barraquand J, Latombe J-C. (1991) Robot motion planning: A distributed representation approach. *International Journal of Robotics Research*. 10(6):628-649.

Beer RD, Chiel HJ, Quinn RD, Espenschied KS. (1992) A distributed neural network architecture for hexapod robot locomotion. *Neural Computation*. 4:356-365.

Bizzi, E. (1993). Intermediate representations in the formation of arm trajectories. *Current Opinion in Neurobiology*. 3:925-931.

Bizzi E, Giszter SF, Loeb E, Mussa-Ivaldi FA, Saltiel P. (1995) Modular organization of motor behavior in the frog's spinal cord. *Trends in Neuroscience*. 18(10):442-446.

Brinkman C. (1984) Supplementary motor area of the monkey's cerebral cortex: short- and long-term deficits after unilateral ablation and the effects of subsequent callosal section. *Journal of Neuroscience*. 4(4):918-929.

Brooks RA. (1986) A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*. RA-2(1):14-23.

Brooks RA. (1990) Elephants don't play chess. *Robotics and Autonomous Systems*. 6(1-2):3-15.

Brooks RA. (1991) Intelligence without reason. MIT Artificial Intelligence Laboratory Memo No. 1293.

Caminiti T, Johnson PB, Urbano A. (1990) Making arm movements within different parts of space: dynamic aspects in the primate motor cortex. *Journal of Neuroscience*. 10(7):2039-2058.

Chan J-L, Ross ED. (1988) Left-handed mirror writing following right anterior cerebral artery infarction: evidence for nonmirror transformation of motor programs by right supplementary motor area. *Neurobiology*. 38:59-63.

Chen YP, Campbell R, Marshall JC, Zaidel DW. (1990) Learning a unimanual motor skill by partial commissurotomy patients. *Journal of Neurology, Neurosurgery, and Psychiatry*. 53:785-788.

Diamond A. (1990) Developmental time course in human infants and infant monkeys, and the neural bases of, inhibitory control in reaching. *Annals of the New York Academy of Sciences*. 608:637-676.

Donchin O, Gribova A, Steinberg O, Bergman H, Vaadi E. (1997) How do SMA and M1 contribute to bimanual coordination: evidence from simultaneous recordings of single unit activity and local field potentials in the two hemispheres of a behaving monkey. *Society for Neuroscience Abstracts*. 606.20.

Donchin O, Gribova A, Steinberg O, Bergman H, Vaadi E. (1998) Primary motor cortex is involved in bimanual coordination. *Nature*. 395:274-278.

Dum RP, Strick PL. (1991) The origin of corticospinal projections from the premotor areas in the frontal lobe. *Journal of Neuroscience*. 11(3):667-689.

Forssberg S, Grillner S, Halbertsma J, Rossignol S. (1980) The locomotion of the low spinal cat: II Interlimb coordination. *Acta Physiologica Scandinavica*. 108(3):283-295.

Gazzaniga MS. (1995) Principles of human brain organization derived from split-brain studies. *Neuron*. 14:217-228.

Georgopoulos AP, Kalaska JF, Caminiti R, Massey JT. (1982) On the relations between the direction of two-dimensional arm movements and cell discharge in primate motor cortex. *Journal of Neuroscience*. 2(11):1527-1537.

Graziano MSA, Yap GS, Gross CG. (1994) Coding of visual space by premotor neurons. *Science*. 266:1054-1057.

Graziano MSA, Hu XT, Gross CG. (1997) Coding the locations of objects in the dark. *Science*. 277:239-241.

Grossberg S, Pribe C, Cohen MA. (1997) Neural control of interlimb oscilations. I. Human bimanual coordination. *Biological Cybernetics*. 77:131-140.

He S-Q, Dum RP, Strick PL. (1993) Topographic organization of corticospinal projections from the frontal lobe: motor areas on the lateral surface of the hemisphere. *Journal of Neuroscience*. 13(3):952-980.

Jeeves MA, Silver PH. (1988) Interhemispheric transfer of spatial tactile information in callosal agenesis and partial commissurotomy. *Cortex*. 24:601-604.

Jeeves MA, Silver PH, Jacobson I. (1988a) Bimanual co-ordination in callosal agenesis and partial commissurotomy. *Neuropsychologia*. 26(6):833-850.

Jeeves MA, Silver PH, Milne AB. (1988b) Role of the corpus callosum in the development of a bimanual motor skill. *Developmental Neuropsychology*. 4(4):305-323.

Kavraki LE, Svestka P, Latombe J-C, Overmars MH. (1996) Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Transactions on Robotics and Automation*. 12(4):566-580.

Kermadi I, Liu Y, Rouiller EM. (1997) Comparison of single unit activities in the primary and supplementary motor areas related to bimanual motor control. *Society for Neuroscience Abstracts*. 454.4.

Kettner R, Marcario J, Port N. (1993) A neural network model of cortical activity during reaching. *Journal of Cognitive Neuroscience*. 5(1):14-33.

Kjaerulff O, Kiehn O. (1996) Distribution of networks generating and coordinating locomotor activity in the neonatal rat spinal cord *in vitro*: A lesion study. *Journal of Neuroscience*. 16(18):5777-5794.

Koga, Y.; Latombe, J.-C. (1994) On multi-arm manipulation planning. *Proceedings 1994 IEEE International Conference on Robotics and Automation*. IEEE Computer Society Press. 2:945-952.

Lang KJ, Witbrok MJ. (1988) Learning to tell two spirals apart. In Touretzky D, Hinton G, Sejnowski T (Eds.) *Proceedings of the 1988 Connectionist Models Summer School*. 52-59.

Latome J-C. (1991) *Robot Motion Planning*. Kluwer Academic Publishers. Boston.

Liu Y, Kermadi I, Moret V, Rouiller EM. (1997) Neuronal activity related to bimanual coordination in motor cortical areas and basal ganglia in macaque monkeys, mapped with Fos-like Immunoreactivity. *Society for Neuroscience Abstracts*. 606.18.

Lutsep HL, Wessinger CM, Gazzaniga MS. (1995) Cerebral and callosal organization in a right hemisphere dominant "split-brain" patient. *Journal of Neurology, Neurosurgery, and Psychiatry*. 59:50-54.

McNabb AW, Carroll WM, Mastaglia FL. (1988) "Alien hand" and loss of bimanual coordination after dominant anterior cerebral artery territory infarction. *Journal of Neurology, Neurosurgery, and Psychiatry*. 51:218-222.

Moll L, Kuypers HGJM. (1977) Premotor cortical ablations in monkeys: contralateral changes in visually guided reaching behavior. *Science*. 198:317-319.

Moody SL, Wise SP, Pellegrino G, Zipser D. (1998) A model that accounts for activity in primate frontal cortex during a delayed matching-to-sample task. *Journal of Neuroscience*. 18(1):399-410.

Moody SL, Zipser D. (1998) A model of reaching dynamics in primary motor cortex. *Journal of Cognitive Neuroscience*. 10(1):35-45.

Mussa-Ivaldi FA, Giszter SF. (1992) Vector field approximation: a computational paradigm for motor control and learning. *Biological Cybernetics*. 67:491-500

Mussa-Ivaldi FA, Giszter SF, Bizzi E. (1994) Linear combinations of primitives in vertebrate motor control. *Proceedings of the National Academy of Sciences*. 91:7534-7538.

Nichols TR. (1994) A biomechanical perspective on spinal mechanisms of coordinated muscular action: an architecture principle. *Acta Anatomica*. 151(1):1-13.

Pashler H, Luck SJ, Hillyard SA, Magnun GR, O'Brien S, Gazzaniga MS. (1994) Sequential operation of disconnected cerebral hemispheres in split-brain patients. *Neuroreport*. 5:2381-2384.

Preilowski BFB. (1972) Possible contribution of the anterior forebrain commissures to bilateral motor coordination. *Neuropsychologica*. 10:267-277.

Preilowski BFB. (1975) Bilateral motor interaction: perceptual-motor performance of partial and complete "split-brain" patients. In *Cerebral Localization*, Zülch KJ, Creutzfeldt O, Galbraith GC (Editors). Springer, Berlin. 115-132.

Reynolds DM, Jeeves MA. (1977). Further studies of tactile perception and motor coordination in agenesis of the corpus callosum. *Cortex*. 13:257-272.

Rumelhart DE, Hinton GE, Williams RJ. (1986) Learning internal representations by error propagation. In Rumelhart DE, McClelland JL (Eds.) *Parallel distributed processing: Explorations in the microstructure of cognition*. Volume 1. MIT Press. Cambridge. 318-362.

Scott SH, Kalaska JF. (1994) Cell activity changes in monkey motor cortex is altered by changes in arm posture for movements with identical hand trajectories. *Society for Neuroscience Abstracts*. 403.3

Stein PSG, McCullough ML. (1998) Example of 2:1 interlimb coordination during fictive rostral scratching in a spinal turtle. *Journal of Neurophysiology*. 79(2):1132-1134.

Stein PSG, Victor JC, Field EC, Currie SN. (1995) Bilateral control of hindlimb scratching in the spinal turtle: contralateral spinal circuitry contributes to the normal ipsilateral motor pattern of fictive rostral scratching. *Journal of Neuroscience*. 15(6):4343-4355.

Watson RT, Fleet WS, Gonzalez-Rothi L, Heilman KM. (1986) Apraxia and the supplementary motor area. *Archives of Neurology*. 43:787-792.

Watson RT, Heilman KM. (1983) Callosal apraxia. *Brain*. 106:391-403.

Wiesendanger M, Rouiller EM, Kazennikov O, Perrig S. (1996) Is the supplementary motor area a bilaterally organized system? In Luders HO (Ed.) *Advances in Neurology, Vol. 70: Supplementary Sensorimotor Area*. Lippincott-Raven. Philadelphia.

Williams RJ, Zipser D. (1995) Gradient-based learning algorithms for recurrent networks and their computational complexity. In Chauvin Y, Rumelhart DE (Eds.) *Backpropagation: Theory, architectures, and applications*. Lawrence Erlbaum Associates. Hillsdale. 433-486.

Zaidel D, Sperry RW. (1977) Some long-term motor effects of cerebral commissurotomy in man. *Neuropsychologia*. 15:193-204.

Zipser D. (1992) Identification models of the nervous system. *Neuroscience*. 47(4):853-862.

Zipser D, Andersen RA. (1988) A back-propagation programmed network that simulates response properties of a subset of posterior parietal neurons. *Nature*. 331:679-684.