Extended abstracts of the

NIPS*97 Workshop

Can Artificial Cerebellar Models Compete to Control Robots?

Editors:

Patrick van der Smagt

Institute of Robotics and System Dynamics
German Aerospace Center (DLR Oberpfaffenhofen)

Oberpfaffenhofen, Germany

Daniel Bullock

Cognitive and Neural Systems Dept.
Boston University

Boston, USA

DLR Technical Report # 515–97–28
December, 1997

Contents

Chapter 1

Dynamic control in new robot structures: Can we learn nonlinear functions of high dimensionality?

Patrick van der Smagt
Institute for Robotics and System Dynamics
German Aerospace Center
P. O. Box 1116, 82230 Wessling
GERMANY
email: smagt@dlr.de

Abstract

If robots are ever to migrate away from factory floors, light-weight concepts are required. However, the novel actuator structures used for such robots results in increasingly difficult dynamics. Control theory cannot cope without learning anymore, but even parameterized models are too inexact to be used. We present the problematics in current day robotics.

1.1 Introduction

Recent successes in robotics have increased the field of application and acceptation of robots. Nevertheless, industrial robotics still has to go a long way. The applicability of classical robots remains limited to factory floors.

Research lab robotics is increasingly moving towards novel actuator schemes with motors which have a high force-to-weight ratio, are rather small (about 1" $\times$ 1" $\times$ 2"), and are therefore useful in the construction of light-weight robot arms. Due to the elasticity of the materials used and the compliance (which is also changing due to wear-and-tear), actuators consisting of agonist-antagonist drive pairs help in maintaining accurate positioning without recalibration, as well as in controlling the stiffness of a joint.

However, when multiple of such joints are used to construct a robot arm, the whole structure is very compliant (in comparison to traditional robot arms), and stable control becomes increasingly difficult up to impossible with known control methods. The only working solution which is used so far consists of low-speed control using (approximated) inverse Jacobians of the robot arm. This control method, however, is either slow or inaccurate.

Clearly, the chosen robot arm construction is growing towards an increased degree of anthropomorphism. Control algorithms, on the other hand, do not. The success that the biological cerebellum has in controlling anthropomorphic arms is in no way matched by artificial neural or other control algorithms, since, to date, the complexity of the function that has to be computed seems to be prohibitive. One outstanding problem is that, when controlling the trajectory of a joint, the position θ, velocity $\dot{\theta}$, and acceleration $\ddot{\theta}$ of all connected joints influence that trajectory. This means that each joint is controlled depending on $3k$ variables (where $k \geq 6$ is the number of degrees-of-freedom of the robot arm). With highly compliant robot arms, the relationship between the $3k$ variables is very complex and highly nonlinear, and therefore very hard to learn. The approach of using recurrent networks which are structurally capable of computing internal representations of $\dot{\theta}$ and $\ddot{\theta}$ have not yet been successful in solving more than toy problems.

1.2 Problems in robot dynamics

The dynamics of a general robot arm can be written as

$$\begin{aligned}\boldsymbol{\tau} =& M(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + C(\boldsymbol{\theta})\left[\dot{\boldsymbol{\theta}}\dot{\boldsymbol{\theta}}\right] + D(\boldsymbol{\theta})\left[\dot{\boldsymbol{\theta}}^2\right] \\ &+ F(\boldsymbol{\theta},\dot{\boldsymbol{\theta}}) + G(\boldsymbol{\theta}) \qquad (1.1)\end{aligned}$$

where $\boldsymbol{\tau}$ is a k-vector of torques exerted by the links, and $\boldsymbol{\theta}$, $\dot{\boldsymbol{\theta}}$, and $\ddot{\boldsymbol{\theta}}$ are k-vectors denoting the positions, velocities, and accelerations of the k joints. $[\dot{\boldsymbol{\theta}}\dot{\boldsymbol{\theta}}]$ and $[\dot{\boldsymbol{\theta}}^2]$ are vectors

$$\left[\dot{\boldsymbol{\theta}}\dot{\boldsymbol{\theta}}\right] = \left[\dot{\theta}_1\dot{\theta}_2, \dot{\theta}_1\dot{\theta}_3, \dots, \dot{\theta}_{k-1}\dot{\theta}_k\right]^T, \qquad (1.2)$$

$$\left[\dot{\boldsymbol{\theta}}^2\right] = \left[\dot{\theta}_1^2, \dot{\theta}_2^2, \dots, \dot{\theta}_k^2\right], \qquad (1.3)$$

$M(\boldsymbol{\theta})$ is the matrix of inertia (the mass matrix), $C(\boldsymbol{\theta})$ is the matrix of Coriolis coefficients, $D(\boldsymbol{\theta})$ is the matrix of centrifugal coefficients, $F(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})$ is a friction term, and $G(\boldsymbol{\theta})$ is the gravity working on the joints.

1.2.1 A taxonomy of problematics in robot dynamics

The dynamics of a robot arm are influenced by the following parts:
(1) **the actuators**. A tendency exists towards using DC motors or step motors for generating the required force; however, pneumatic artificial muscles have also received considerable attention. The dynamic behaviour of an actuator is an important part of the robot arm dynamics.
(2) **the connection between the actuators and the links** (e.g., gear boxes). With a tendency towards light-weight robot arms, for DC or step motor based robot arms it is customary to use high-ratio gearboxes such that the motors used can be kept small and light. On the downside, however, is a considerable elasticity, such that both the rotation at the motor side and at the link side must be measured. *Direct drive* robots are also under consideration; yet, the motors have a very low force-to-weight ratio, and are therefore not suitable for light-weight robots.
(3) **the links**. Finally, the dynamics of the construction has to be taken into account. Very light weight structures may be flexible, leading to a very complex control scheme. It is customary to construct a robot arm to ensure that this part can be neglected.

The rigid body assumption

The simplest kind of robot arm consists of rigid bodies which are connected by rigid links. This assumption is approximately true for industrial robots; the construction of the robot arm is thus that any deformation of links and joints can be neglected. Even current-day research robots are constructed with this principle in mind; even though materials are light-weight, they are supposed to be strong enough not to be flexible, even when payloads are carried. In this case adaptive control is done by linearization of the control equation. Eq. (1.1) is simplified in order to obtain:

$$\boldsymbol{\tau} = Y(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}, \dot{\boldsymbol{\theta}}_r, \ddot{\boldsymbol{\theta}}_r)w \tag{1.4}$$

where w are the parameters which are estimated.

When the actuators used are strong enough, the diagonal elements of the mass matrix M and the centrifugal matrix D are prevalent, while C is approximately 0. Furthermore all matrices are constant, i.e., independent of $\boldsymbol{\theta}$ and its derivatives. These simplifications result in $\tau_i = m_i\ddot{\theta} + d_i\dot{\theta}^2 + f_i$, where i is the joint number; the joints can be independently controlled.

In a light-weight robot arm actuators are used for which the above simplifications no longer hold; the motors are simply not powerful enough such that gravity and other physical influences can be ignored. This means that, apart from having to take the full matrices M, C, and D into account, these and the F and G matrices are parametrised by the joint positions and velocities; Eq. (1.1) cannot be simplified anymore.

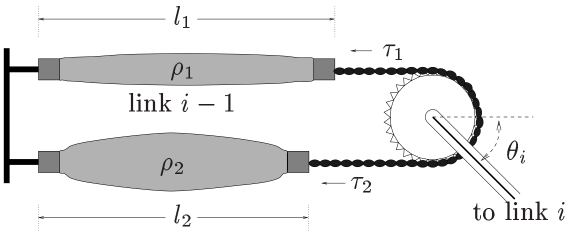

Figure 1.1: An agonist and an antagonist rubbertuator are connected via a chain across a sprocket; their relative lengths determine the joint position θ_i.

Flexible links

There is little research being done on robot arms with flexible links. Some exceptions are the research groups of A. Goldenberg (U. Toronto), J.-J. Slotine (MIT), and M. Spong (UIUC). So far all research in this direction is restricted to two-link robot arms. The general approach here is to attach extra acceleration sensors on the links, and use their signals to correct for their flexibility.

Flexible joints

A somewhat simpler control problem exists with flexible joints. This case is, in fact, very common, namely when high-ratio gearboxes are used.

In the case of elasticity at joint level, an actuator can be modeled by a motor and an arm segment, connected by a spring. The properties of the spring can only be measured when there are joint angle sensors at both the motor (measuring θ_2) and the arm segment (measuring θ_1) side of the spring. Equation (1.1) changes as follows:

$$\begin{aligned}\boldsymbol{\tau}_l =& M(\boldsymbol{\theta}_1)\ddot{\boldsymbol{\theta}}_1 + C(\boldsymbol{\theta}_1)\left[\dot{\boldsymbol{\theta}}_1\dot{\boldsymbol{\theta}}_1\right] + D(\boldsymbol{\theta}_1)\left[\dot{\boldsymbol{\theta}}_1^2\right] \\ &+ F(\boldsymbol{\theta}_1, \dot{\boldsymbol{\theta}}_1) + G(\boldsymbol{\theta}_1), \end{aligned}\tag{1.5}$$

$$\boldsymbol{\tau}_m = J(\dot{\boldsymbol{\theta}}_2)\ddot{\boldsymbol{\theta}}_2 + \boldsymbol{\tau}_l \tag{1.6}$$

where $\boldsymbol{\tau}_m$ is the torque at the motor side and $\boldsymbol{\tau}_l \equiv k(\boldsymbol{\theta}_2 - \boldsymbol{\theta}_1)$ the torque at the link side. J can generally be assumed to be a diagonal matrix.

1.2.2 An exemplar difficult joint structure

The McKibben pneumatic artificial muscle [1], as used in the Bridgestone SoftArm robot, has typical problematic control properties [4]. The use of two artificial muscle in the construction of a joint is depicted in figure 1.1. The sprocket construction, combined with the properties of the artificial muscles, lead to a hysteretic nonlinear pressure-position relationship.

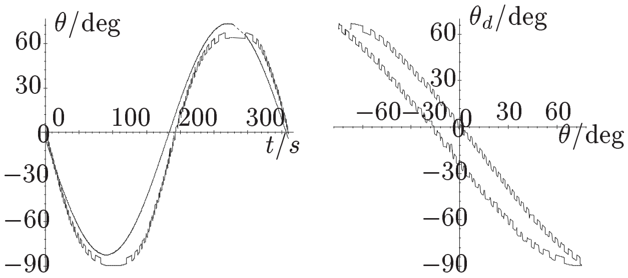

Figure 1.2: Using the internal PID controller to follow the trajectory $\theta_d(t) = c\sin(t)$. The left figure shows the desired and realized trajectories vs. t; the right figure depicts the desired (horizontal axis) vs. the realized (vertical axis) trajectory.

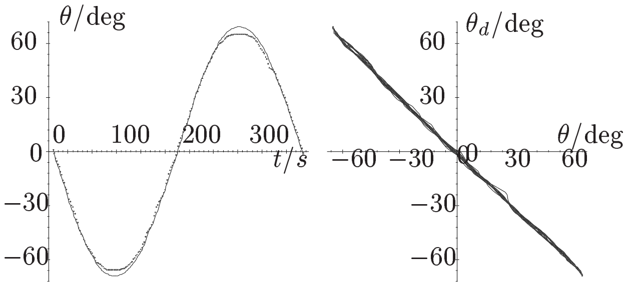

Figure 1.3: Using the neural network controller to follow the trajectory $\theta_d(t) = c\sin(t)$. The left figure shows the desired (solid line) and realized (dotted line) trajectories vs. t; the right figure depicts the desired (horizontal axis) vs. the realized (vertical axis) trajectory. This behaviour is recorded after five minutes of learning.

Naturally, a PID controller cannot control such a robot. A test is shown in figure 1.2. In this figure, a single joint of the robot, controlled via a PD controller, follows a simple trajectory.

1.3 What can we learn?

An obvious way out of the above problematics consists of learning. *If* we presume that samples of the desired input-output behaviour can be recorded, *and* we assume we have a black box which can learn every function $\mathcal{F} : \Re^m \rightarrow \Re^n$ fast enough, *then* any of the above robot architectures can be optimally controlled.

Example: Figure 1.3 shows the pneumatic artificial muscle experiment using a self-learning controller based on fast learning feed-forward networks.

The result looks encouraging; after a relatively short time of learning, the desired trajectory can be followed. The methodology has been shown to generalize well to other trajectories [4]. Unfortunately, this methodology cannot be very well generalized to more dimensions. For the single joint system, a 7-dimensional input space is used: the pressure of one artificial muscle, the trajectory, and the desired trajectory. For a six-Degree-of-Freedom robot this would result in a 42-dimensional input vector. Knowing that the function that must be approximated is highly nonlinear, it is clear that this approach is not feasible. First, it is very difficult to gather sufficient training data for a mapping from $\Re^{42} \rightarrow \Re^6$. Second, a general approximation method will not be able to learn this mapping with sufficient accuracy in waitable time [3].

1.3.1 Are improvements possible?

The method can be improved upon in various ways. First, there is a substantial body of literature frequently updated with methods towards improved approximation of high-dimensional functions. Some examples of these are [3, 5, 6, 7, 2], but many others are currently popping up. These approaches are more than 'personal flavours of back-propagation;' they iterate towards an increasingly good understanding of the representation of high-dimensional surfaces from randomly distributed samples. Put together with the constantly improving computing power, the realm of high-dimensional surfaces which can be accurately and successfully approximated increases steadily.

A second possible improvement is to use recurrent networks which are structurally capable of computing internal representations of time derivatives of the signals. For the above example, this would reduce the input dimensionality from 42 to 18. The dimension reduction, which is otherwise computed by the feed-forward network, has no longer to be performed, resulting at least in a reduction of the input space; possibly also of the network parameter dimensionality.

Yet, in spite of these successes and possible improvements, the larger challenge lies at the third possible improvement: use an adaptive structure which specializes on the control problem at hand. It is in this workshop that the latter point is pursued.

Bibliography

[1] C.-P. Chou and B. Hannaford. Measurement and modeling of McKibben pneumatic artificial muscles. *IEEE Transactions on Robotics and Automation*, 12(1):90–102, 1996.

[2] H. A. B. te Braake, E. J. L. van Can, and H. B. Verbruggen. Training of neural networks by a variable projection method. *Neural Networks*. Submitted.

[3] P. van der Smagt. Minimisation methods for training feed-forward networks. *Neural Networks*, 7(1):1–11, 1994.

[4] P. van der Smagt, F. Groen, and K. Schulten. Analysis and control of a rubbertuator arm. *Biological Cybernetics*, 75(5):433–440, 1996.

[5] P. van der Smagt, F. Groen, and F. van het Groenewoud. The locally linear nested network for robot manipulation. In *Proceedings of the IEEE International Conference on Neural Networks*, pages 2787–2792, 1994.

[6] P. van der Smagt and G. Hirzinger. Solving the ill-conditioning in neural network learning. In J. Orr, K. Müller, and R. Caruana, editors, *Lecture Notes on Computer Science: The Book of Tricks*. Springer Verlag, 1998. In print.

[7] C. Wang and J. C. Principe. On-line stochastic functional smoothing optimization for neural network training. *IEEE Transactions on Systems, Man, and Cybernetics*. Submitted.

Chapter 2

Optimized Weight Smoothing for CMAC Neural Networks

Dr. L.G. Kraft
Electrical and Computer Engineering
University of New Hampshire
Durham, NH 03824
GORDON.KRAFT@UNH.EDU

The CMAC neural network concept has been used very successfully in control and signal processing applications [1, 4, 13]. It has many characteristics suitable for real-time learning problems including rapid training, low memory requirements and faster cycle times than other networks [8]. One limitation to the CMAC neural network in its present form is that, while the approximation to the function being learned may be quite good, the partial derivatives of the functional approximation may not be adequate for predictive control applications. The fundamental reason for this limitation comes from the original design of CMAC. In its original form, CMAC was not designed to develop good partial derivative information. It was developed as fast and efficient associative memory based function approximator similar to an adaptive look-up table. The problem is particularly apparent when CMAC is used to predict system information in areas of the state space where the training data is sparse. In the vibration control problem, for example, the eventual degree of vibration reduction will be limited by the resolution and functional approximation properties of the CMAC network. This limitation can be overcome by adjusting the weight update laws to build in the feature of "weight smoothing". The weight update procedure can be treated as a constrained optimization problem where a penalty is imposed on the difference between adjacent weights. The procedure is similar to penalizing rapid changes in the system input in an optimal control problem. In this paper graphical results for the single input and the two dimensional CMAC will be presented. The early results show dramatic improvement of the CMAC learning performance when compared to the traditional CMAC. Not only is the learning faster, it is much more useful in terms of constructing partial derivative information. This is a key result that may open the field of CMAC control to a large class of optimization problems and predictive controllers such as the adaptive critic [14].

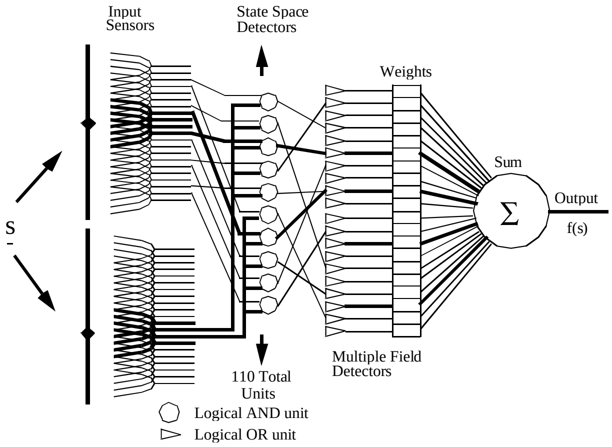

Figure 2.1: CMAC Functional Block Diagram.

2.1 Geometrical Interpretation of the CMAC Network

The basic CMAC algorithm learns a function $f(\underline{s})$ by associating measurements of $\underline{s}$ and $f(\underline{s})$. The implementation algorithm is shown in Figure 2.1.

The first layers of the network determine a mapping of the input vector s to particular weights in the network. Each input vector is associated with a particular set of weights. Once a set of weights is selected, the values of the weights are adjusted so that the output of the network more closely matches the value of the function $f(\underline{s})$. The process of updating weights is very fast due to the parallel architecture and the logic gate implementation. The middle layer of the network implements a random (but fixed) mapping similar to hash coding to avoid the need for large numbers of weights. The details of Miller's realization of CMAC have been reported in the literature [10, 11, 12] and will not be repeated here.

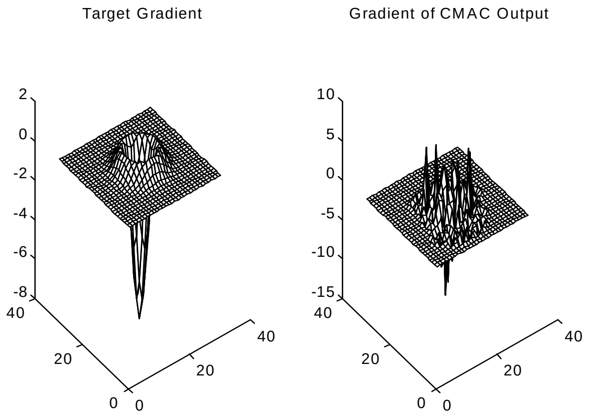

Figure 2.2: True Gaussian Pulse Gradient Compared to CMAC Approximation.

2.2 Optimal Weight Smoothing

In its simplest form the CMAC algorithm can be thought of as set of simultaneous equations. Each data point y_i is associated with C weights in the network. The sum of the associated weights should match the data point.

$$y_i = \underline{s}_i \underline{w}. \tag{2.1}$$

Details of this map may be found in [10] and [11]. Each row vector, $\underline{s}_i$, has exactly C (generalization size) ones, all other elements in the row are zero. Defining $\underline{Y}$ as the vector of training data, $\underline{w}$ as the vector of network weights, and $\underline{S}$ as the selection matrix determined by the CMAC mapping, the equations can grouped into matrix form

$$\underline{Y} = \underline{S}\underline{w}. \tag{2.2}$$

The original CMAC problem was to adjust the weight vector $\underline{w}$ so that the equation above is satisfied. In general there are more weights than equations and thus there are many sets of weights that satisfy the equation. The original CMAC adjusted all the weights within the generalization region the same amount. By doing this the eventual weight pattern may not resemble the function being learned at all and a problem arises when the partial derivative of the function being learned is needed. The derivative must be calculated from the weight vector numerically, usually by taking the difference of nearby weights. These calculations can yield estimates of the derivatives that are not representative of the true derivative of the function. The problem is illustrated in Figure 2.2 below.

To reduce this problem, a different weight update law can be used. The new weights are selected to minimize a scalar measure that includes a penalty for weights that differ widely from their nearest neighbors. A cost function is defined as follows for the constrained optimization problem:

$$J = \underline{w}^T \underline{Q}\underline{w} + (\underline{S}\underline{w} - \underline{Y})^T \underline{R}(\underline{S}\underline{w} - \underline{Y}). \tag{2.3}$$

Mathematically stated the weight vector is selected to minimize J where $\underline{R}$ is a positive definite matrix, $\underline{S}\underline{w} - \underline{Y}$ is the equation error, and $\underline{Q}$ is the penalty matrix associated with the differences between weights. Taking the partial derivative of the cost function with respect to the weights and solving for $\underline{w}$ yields,

$$\underline{w} = (\underline{S}^T \underline{R}\underline{S} + \underline{Q})^{-1} \underline{S}^T \underline{R}\underline{Y}. \tag{2.4}$$

The solution represented by (2.3) is a batch mode weight update law in that it requires all available training data to be present. It can be applied to all the data or just to the data within the generalization window. The same solution to this problem can also be calculated recursively using one sample data point at a time. It will be optimum with regard to whatever penalty weighting matrices are chosen. For this work, the $\underline{Q}$ matrix takes the following form for a case with 4 weights:

$$\underline{Q} = \alpha \begin{pmatrix} 1 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 1 \end{pmatrix} + \beta \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \tag{2.5}$$

$$\text{with } \alpha \geq 0 \text{ and } \beta > 0.$$

The portion of the $\underline{Q}$ matrix pre-multiplied by α represents a penalty on the difference between adjacent weights. This difference penalty is used to enforce the desired amount of smoothness in the weights. The portion of the $\underline{Q}$ matrix pre-multiplied by β represents a penalty on the magnitude of the weights. Magnitude control is always applied so that $\underline{Q}$ is non-singular. This minimum weight magnitude portion of the Q matrix also serves to control the problem of certain CMAC weights drifting to large values as the result of repeated, updates during learning.

Experiments were conducted in software simulation to test the validity of the new approach. A single-input single-output learning case was studied using both the original CMAC and the new CMAC with weight smoothing. The problem was to learn a function, in this case a triangle wave. The same training data was used for each case. There were 100 training points and two passes through the data. Generalization was set at 4 and the learning parameter was set at 1/2 for the original CMAC. The new CMAC used $\alpha = 1$ and $\beta = 0.1$. The results are shown in Figure 2.3.

Figure 2.3 shows graphically that the new CMAC with weight smoothing does a better job of learning the correct derivative of the function. The

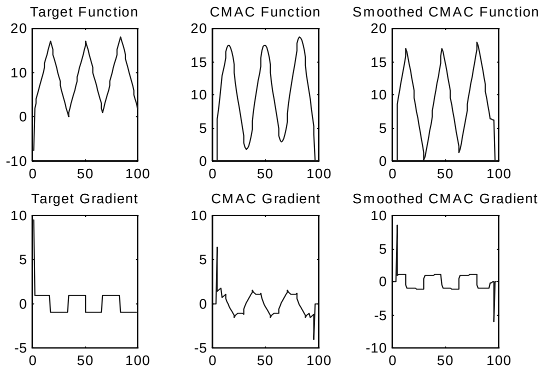

Figure 2.3: Comparison Results for Single Input Case.

weight smoothing CMAC produces a much better estimate of the true gradient than the traditional CMAC. Similar results were found for several different types of functions. Results for higher order examples are the subject of ongoing work.

2.3 Conclusions

While traditional CMAC networks have achieved dramatic results in robot control, image processing, and other applications, it has been limited to function learning problems in which the partial derivative information was not essential. With the new weight smoothing concept, it may be possible to extend the range of real-time applications to include a much wider class of control systems including approximate dynamic programming optimization problems, predictive control, inverse system control and the adaptive critic.

References

[1] Albus, J. S, "A New Approach to Manipulator Control: The Cerebellar Model Articulation Controller (CMAC)," Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME, Vol. 7, pp. 220-227, September, 1975.

[2] Kraft, L. G. and Campagna, D. P., "A Summary Comparison of CMAC Neural Network and Traditional Adaptive Control Systems," IEEE Control Systems Magazine, April, 1990.

[3] Kraft, L. G. and Campagna, D. P. , "Stability of Open-loop Learning in CMAC Neural Networks," American Controls Conference, Seattle, WA, 1995.

[4] Kraft, L. G., "Optimal Control Using CMAC Neural Networks," Neural Networks and Intelligent Control, edited by D. A. White and D. A Sofge, Van Nostrand-Reinhold, New York, NY, 1992.

[5] Kraft, L.G. and Nelson, J. S., "Using CMAC and Optimal Control," IEEE International Conference on Neural Networks, Perth, Australia, November, 1995.

[6] Miller, W, T., "Real-time Application of Neural Networks for SensorBased Control of Robots with Vision," IEEE Transactions on Systems, Man, and Cybernetics, Special Issue on Information Technology for SensoryBased Robot Manipulators, Vol. 19, pp. 825-831, 1989.

[7] Miller, W. T, "Sensor Based Control of Robotic Manipulators Using a General Learning Algorithm," IEEE Journal of Robotics and Automation, Vol. RA-3, pp. 157-165, 1987.

[8] Miller, W. T., Box, B. A., Whitney, E. C., and Glynn, J. M., "Design and Implementation of a High Speed CMAC Neural Network Using Programmable Logic Cell Arrays," Advances in Neural Information Processing Systems 3, edited by R.P. Lippmann, J. E. Moody, and D. S. Touretzky, Morgan Kaufmann, San Mateo, CA, pp. 1022-1027, 1991.

[9] Miller, W. T., Glanz, F. H., and Kraft, L. G., "Application of a General Learning Algorithm to the Control of Robotic Manipulators," International Journal of Robotics Research, Vol. 6.2, pp. 84-98, Summer, 1987.

[10] Miller, W. T., Glanz, F. H., and Kraft, L.G., "CMAC: An Associative Neural Network Alternative to Backpropagation," IEEE Proceedings, Special Issue on Neural Networks, 1990.

[11] Miller, W. T., Glanz, F. H., and Kraft, L. G., "Application of a General Learning Algorithm to the Control of Robotic Manipulators," International Journal of Robotics Research, Vol. 6.2, pp. 84-98, 1987.

[12] Miller, W. T., Hewes, R. P., Glanz, F. H., and Kraft, L. G., "Real-time Dynamic Control of an Industrial Manipulator Using a Neural Network Based Learning Controller," IEEE Journal of Robotics and Automation, February, 1990.

[13] Miller, W. T., Latham, P. J., and Scalera, S. M., "Bipedal Gait Adaptation for Walking with Dynamic Balance: Temporal Difference Learning Using CMAC Neural Networks," Conference on the Simulation of Adaptive Behavior: From Animals to Animats, Paris, France, September 24-28, 1990.

[14] Miller, W. T, Sutton, R. S, and Werbos, P. J., Neural Networks for Control, MIT Press, Cambridge, MA, 1990]

[15] Nelson, J. and Kraft L. G., "Real-time Control of an Inverted Pendulum System using Complementary Neural Network and Optimal Techniques," Proceedings of the American Control Conference, Baltimore, MD, June 29-July 1, 1994.

Chapter 3

Cerebellar learning for context sensitive and critically timed coordination of multiple action channels

Daniel Bullock
CNS Department
Boston University

Whether the cerebellar architecture can compete with alternatives control architectures in robotics depends on two broad factors. First, is it sufficiently competent to serve as a robust basis for control when used in combination with other robust control modules? Second, is it sufficiently efficient in the way that it achieves that competence? These two questions are not unrelated. In particular, as our estimate of the cerebellum's competence increases or decreases, so will our estimate of its efficiency given a fixed number of computational degrees of freedom.

That the cerebellum can serve as a robust basis for control in combination with other control modules is apparent from the many circuits within which it has become embedded during brain evolution. In particular, as emphasized by Ito (1984) and Leiner, Leiner & Dow (1986), among others, the cerebellum serves not just sensory-motor pathways with direct outputs to spinal motor centers, but also serves control pathways with outputs to motor and premotor cortex.

Such 'spread of embedding' tells us that the basic cerebellar module performs very broadly useful functions, but it does not specify what those functions are. Most past theories have emphasized that the cerebellum is capable of learning non-linear sensory-motor transforms, through sparse expansive recoding performed by the mossy-granule fanout and Golgi feedback, combined with adaptation of synaptic efficacies between granule cell axons (parallel fibers) and the giant Purkinje cells. In particular, these are the properties abstracted in the CMAC family of models pioneered by Albus (1975).

Over the years there has also been much discussion of the possibility that the cerebellum is providing a timing function. To cite two early examples, Braitenberg (1961) proposed that the parallel fibers might be functioning as delay lines, and Fujita (1987) proposed that natural variations in Golgi-granule interactions might produce a basis for adaptive lead-lag compensation.

Generally speaking, such proposals are consistent with the expansive recoding idea made explicit in CMAC models. In essence, such proposals argue that expansive recoding occurs in both space and time to provide a plant-independent basis for maximizing the usefulness of input information for purposes of control. In this sense, the cerebellum can be seen to be as much a 'perceptual' as a motor organ, although restricted to serve the kind of non-image-based, direct, perception ('information pickup') emphasized by Gibson (1966). For robotics, the cerebellum can be seen as as existence proof for the usefulness of look-up type perceptual-motor transforms that can be very fast because they avoid the processing overhead associated with complex image processing.

Here it should be noted that cerebellar lesions are associated clinically with striking loss of movement coordination and, in more refined analyses, with loss of precise timing control over motor control actions. Such loss of normal timing is apparent not just in behavior but also in the activities of neurons in frontal cortex (Hore & Flament, 1988). Timing and coordination are intimately related in virtually all animal movement because muscles come in opponent pairs or more complex sets. Even for single joint systems, coordinating an opponent muscle pair during rapid, accurate, self-terminated movement requires precise timing. But these requirements are greatly amplified by multi-joint movement, because of the complex mechanical interactions that arise among linked segments.

In many models, motor planning variables and their derivatives have been assumed to be among the inputs to the cerebellum, and to account for much of the cerebellum's apparent timing competence. This makes sense because of the inherently predictive nature of time derivatives. However, recent experimental data suggest that postulating such inputs cannot fully explain the adaptive timing competence of the cerebellum. In particular, such signals don't exist in certain paradigms used to demonstrate cerebellar adaptive timing, such as the conditioned eye-blink paradigm. Indeed, experimental evidence from this paradigm (reviewed in Bullock, Grossberg & Fiala, 1994; Fiala, Grossberg, and Bullock, 1996) has strongly implicated the cerebellar cortex as site of an adaptive timing function that can learn arbitrary delays in the range of 100 ms to 4 s. In this presentation, I will briefly summarize these data and then contrast two different hypotheses developed in Bullock *et al.* (1994), Fiala *et al.* (1996), and Bullock & Fiala (1996) regarding the mechanisms of this adaptive timing function.

The second hypothesis is of special interest because it postulates that there is an intracellular process—mediated by a metabolic cascade initiated when parallel fiber activation excites metabotropic glutamate receptors on Purkinje cell dendritic spines—whose temporal properties are matched to the macroscopic timing requirements of the control task. Adopting this hypothesis leads to the following summary description of cerebellar function.

Summary of Hypotheses & Conclusions

The cerebellum is composed of a large number of parasaggitally oriented cortico-nuclear microzones that are defined by shared climbing fiber projections and convergence of Purkinje cell axons on subsets of nuclear cells.

Cerebellar granule cells can detect event combinations.

The cerebellum may generate a mGluR mediated spectrum of time-delayed Ca++ pulses for each combination detected in the recent past.

Coincidence of a mGluR-mediated Ca++ pulse with a climbing fiber burst forges a causal link between the delayed pulse and a control action.

Climbing fiber burst rate reflects error-compensating action (broadly interpreted) in the control channel for which the microzone serves as a side-loop, as emphasized in feedback-error-driven models of cerebellar learning (Grossberg & Kuperstein, 1986; Ito, 1984; Kawato, *et al.* 1987).

The dynamics of the Ca++ pulse generation enable the control action to move forward in time. This allows it to counteract, and in many cases cancel, the effect of the perturbation, i.e., the error that initially elicited reactive control action.

The cerebellum searches in parallel through masses of recent signal traffic for leading indicators that will allow the animal to take actions that can preempt errors.

The cerebellum allows a switch from reactive to proactive control.

A common role of the cerebellum is therefore to establish a feedforward process that progressively off-loads feedback control.

The relative size of the cerebellum is larger in animals that use active sensing. This points to the need for further research on the cerebellum as part of perceptual systems, a point recently emphasized by Bower (1997).

The common image of how the time scale of neuronal action relates to the time scale of behavior—that it's a small fraction—is based on iontropic neuronal responses, and has been proven incomplete by data on metabotropic responses, including mGluR cascades.

Because mGluR cascades have now been implicated in neural plasticity in many brain regions—including the striatum, hippocampus, and visual cortex—coincidence detection across temporal gaps may be a generally useful augmentation of associative learning.

By cerebellar learning, a single event detector can control a sequence of control actions, either by a single effector or by a set of effectors.

The cerebellar cortex can be seen as adaptively compiling, and mediating execution of, vectors of context-dependent control actions that can be arbitrarily staggered through time in the 100 ms to 4 second range.

Acknowledgment

This research has been partly supported by grants from the Office of Naval Research, ONR N00014-92-J -1309, ONR N00014-93-1-1364, ONR 14-95-1-0409.

Chapter 4

Adaptive Cerebellar Control of Opponent Muscles

José L. Contreras-Vidal
Motor Control Laboratory
Arizona State University
Tempe, AZ 85287-0404 USA

Juan Lopez-Coronado
Center for Neurotecnology
University of Valladolid
47011 Valladolid, Spain

Abstract

A neural network model of opponent cerebellar learning for movement control is proposed (Contreras-Vidal, Grossberg & Bullock, Learning and Memory (1997) 3:475-502). The model shows how a central pattern generator in cortex and basal ganglia, a neuromuscular force controller in spinal cord, and an adaptive cerebellum cooperate to reduce sources of motor variability during multi-joint movements using mono and bi-articular muscles. An application towards the independent position and force control of a dextrous, compliant artificial hand with a suite of sensors is discussed.

4.1 Introduction

Currently, artificial hands are typically binary and have few rudimentary sensors; very simple mechanical actions which limit the types of functional grasps available, few degrees of freedom, and rudimentary force and position control. Furthermore, they are commonly too bulky, too heavy, too slow, or too complex for use; require reprogramming or changing the terminal device for different tasks/environments, and are not capable of adapting to changes in operating conditions. This is true of prosthetic hands, telethesis devices, or industrial grippers. Therefore, one of the goals of current research on bio-robotics is the development of neurally-controlled artificial hands that are easy to use, safe and not costly to fabricate, maintain or reprogram and that can be used in a large range of settings (e.g., home, school or work). In this paper we present a neural network model of adaptive cortico-spino-cerebellar dynamics for compliant hand movement control.

4.2 Adaptive sensory-motor control

In the system of Fig. 4.1, an opponently-organized central pattern generator (VITE model of Bullock and Grossberg, 1988) is used to compute the desired movement trajectories by smoothly interpolating between initial and final muscle length commands for the antagonist muscles involved in the movement. The rate of interpolation (i.e.,

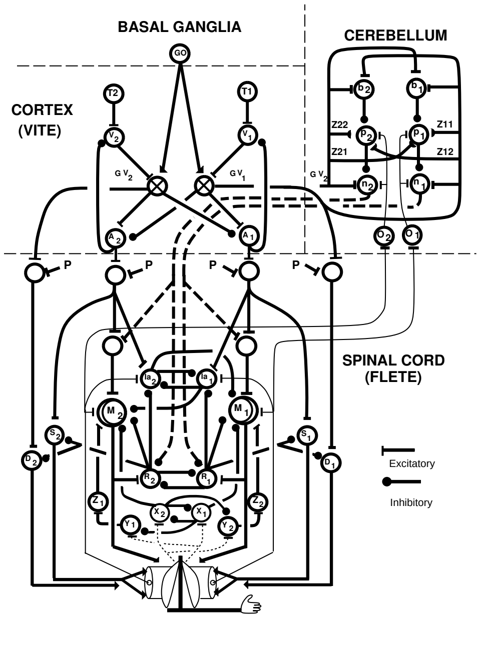

Figure 4.1: Neural network representation of the neuromuscular control system including feedforward cerebellar control. Upper-left part: The VITE model for variable-speed trajectory generation. Lower part: The FLETE model of the opponently organized spino-muscular system. Dotted lines show feedback pathways from sensors embedded in muscles. The two lateral feedback pathways arise in spindle organs sensitive to muscle stretch and its first derivative. The two medial feedback pathways arise in Golgi tendon organs sensitive to muscle force. Signals A_1 and A_2 specify the desired position vector, and the signals V_1G and V_2G specify the desired velocity vector; signals T_1 and T_2 specify the target position vector; signal P scales the level of coactivation, and signal GO scales speed of movement. Upper-right part: Feedforward cerebellar model computes transient inverse-dynamic signals that excite motoneurons and modulate the gain in spinal circuits. Key: B, basket cell; p, Purkinje cells; n, nucleus interpositus cells; O, inferior olive; CF, climbing fibers; z, long-term memory weights. Paths terminated by filled dots are inhibitory; all others are excitatory. Adapted from Contreras-Vidal et al., 1997.

the movement velocity) is controlled by the product of a difference vector which continuously computes the difference between the desired (T_1,T_2) and present position (A_1,A_2) of the limb, and a volitional movement gating signal (GO) (see Contreras-Vidal and Stelmach, 1995).

However, to specify the desired finger-tip direction in 3D space, we must use a direction mapping from spatial to motor coordinates. This direction-to-rotation transform (DIRECT model) maps each spatial direction into an appropriate change in joint angles that causes the finger to move in the commanded spatial direction (Bullock et al., 1993). As directional vectors, rather than target positions, are commanded, tools can be attached to the end-effector without problems. The visuo-motor map is initially learned during a training period, although it can be updated at any time afterwards (Bullock et al., 1993).

4.2.1 Spinal force controller

Bullock and Contreras-Vidal (1993) have shown that the spinal circuitry (see Fig. 4.1) has evolved to provide independent control of muscle length and joint stiffness. In this design, joint stiffness involves simultaneous increments to the contractile states of the joint's opposing muscles, resulting in muscle co-contraction (Humphrey and Reed, 1983). This is accomplished by adding a non-specific co-contraction signal P that adds to both components of the signal pattern (A_1,A_2), producing the net input ($A_1 + P$,$A_2 + P$) to the opponent α-MN pools. The signal P is capable of producing high levels of co-contraction of the opponent muscles. The FLETE model achieves position-code invariance,

$$\theta(A_1, A_2) = \theta(A_1 + P, A_2 + P). \qquad (4.1)$$

Figure 4.2A depicts the steady-state characteristics of the FLETE model controlling a single joint via a pair of antagonist muscles. In this graph, there are 20 separate traces plotted corresponding to 20 different stiffness settings, satisfying Eq. (4.1). Figure 4.2B shows the steady-state response of a planar two-joint system with bi-articular muscles controlled by two VITE-FLETE systems. Joint afferent feedback from the proximal to the distal segment was used to stabilize the distal segment, so that the distal segment could remain constant for a given choice of descending commands ($A_1 - A_2$) over the full range of proximal angles (Contreras-Vidal et al., 1997).

4.2.2 Adaptive cerebellar control of opponent muscles

Recently, Smith (1996) have proposed that pairs of Purkinje cells, as in Fig. 4.1, could use reciprocal inhibition or antagonist cocontraction to control antagonist muscle groups. Suppose the spindle afferents or descending central commands from both opponent channels are active and excite non-specifically the basket cells (b_i in Fig. 4.1). The activation of the basket cells will result in inhibition of both agonist and antagonist Purkinje cells (P_i), therefore disinhibiting the cerebellar nuclear cells (n_i) and causing co-contraction of the opponent pair of muscles. On the other hand, suppose that during learning, long-term depression (LTD) of parallel fiber-Purkinje cell (PF-P) synapses through crosscorrelation of parallel fiber and climbing fiber discharges in the same opponent channel (Z_{11} and Z_{22}, but not in opposing or unrelated channels (Z_{12} and Z_{21}), has caused the pattern of projections of Fig. 4.1 to emerge. Then differential agonist channel stretch will activate the Purkinje cell from the antagonist channel therefore inhibiting the antagonist nuclear cell (Contreras-Vidal et al., 1997).

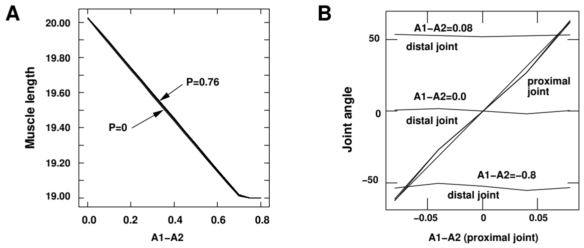

Figure 4.2: (A) Steady state response of the VITE-FLETE system for a (A) single joint and (B) 2 joint, planar system with bi-articular muscles and feedback joint compensation. Both simulations show that linearity is preserved across the work space. In (A), lines correspond to several levels of joint coactivation given by signal P. In (B) the input to the FLETE distal system was constant during three different input settings corresponding to three regions covering the workspace of the planar, two-joint system ($A_1 - A_2 = .08$, 0, and $-.08$). The input to FLETE controlling the proximal joint was varied from -0.08 to 0.08. Adapted from Contreras-Vidal et al., 1997 and Contreras-Vidal, 1994).

Figure 4.3 shows a schematic representation of the cerebellar learning process just described. Panel A lists two possible shoulder joint movements. During flexion, pectoralis (PEC) is the agonist muscle. In extension, the posterior deltoid (PDEL) is the agonist. Panel B summarizes the initial connectivity (unitary) matrix of the PF-P synapses. The rows correspond to granule cells and their associated parallel fibers. The columns correspond to distinct muscle channels with their associated Purkinje cells and climbing fibers. Each granule cell sends a parallel fiber to a Purkinje cell in both channels. Each climbing fiber (C_1 or C_2) innervates only one Purkinje cell. Panel C uses an enclosing box to indicate CF or PF activity. Conjunctive occurrence of parallel fiber and climbing fiber discharge is shown by shading. This conjunction will produce a decrease of the synaptic strength of the PF-to-Purkinje connection (Ito, 1991). After enough LTD trials, the synaptic matrix would look as in Panel D. Only the parallel fiber-Purkinje connections in unshaded cells sur-

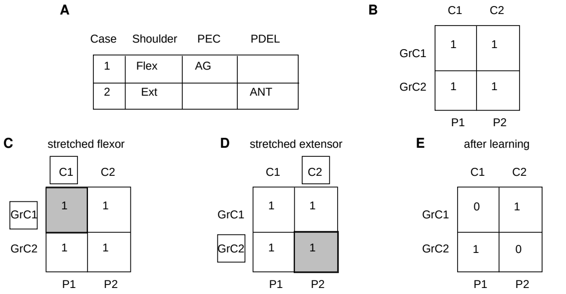

Figure 4.3: Cerebellar learning in the single joint case. (A) Two possible scenarios for shoulder joint flexion and extension; (B) initial parallel fiber-Purkinje cell weights are set to one; (C) during motor learning, the conjunctive excitation of parallel fiber (through granule cells) and climbing fiber would produce a decrease in the synaptic strength of that connection through LTD; (D) after learning, only those weights with uncorrelated parallel fiber-climbing fiber activity will survive LTD.

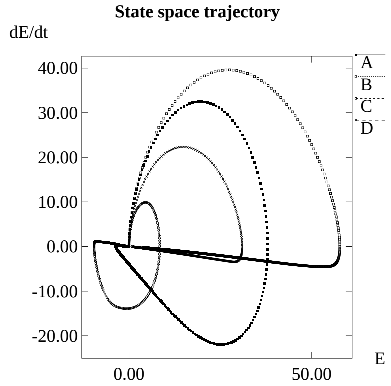

Figure 4.4: State–space representation of the shoulder and elbow position and velocity tracking for both intact and decerebellate two-joint model simulations. Three variables are shown: The plots show curves of joint position error (x-axis) vs. error derivative (y-axis) for both shoulder and elbow joints over time. Key: $E = \Theta - \Theta_d$ (shoulder) or $E = \Phi - \Phi_d$ (elbow). Plots A (shoulder) and C (elbow) are for the intact system; plots B (shoulder) and D (elbow) correspond to the decerebellate system.

vive the long-term depression.

In the model of Fig. 4.1, the cerebellar nuclear cells excites α-motorneurons and inhibits their associated Renshaw cells via the rubrospinal tract (Henatsch et al., 1986). However, Purkinje cell inhibition of the nuclear cells will prevent excitation of α-MNs by this pathway unless parallel fiber signals, after learning in particular contexts, will produce a transient inhibition of Purkinje cells, which will disinhibit the nuclear cells whenever those contexts recur.

Figure 4.4 shows a state-space representation of the tracking capabilities for an intact and a decerebellate system (cf. Fig. 4.1). It illustrates how the composite system moves the limb to the configuration specified by the target position vector input to the VITE modules, in terms of the error between the actual and the intermediate or desired trajectory values for joint angles and velocities generated by VITE. Note that the intact system follows the desired joint positions and velocities better than the decerebellate system. Absolute deviations from zero error are smaller and the error curves are more evenly distributed above and below zero on both the position and velocity dimensions. In summary, the cerebellar projections to spinal centers improve the tracking dynamics of the two-joint system.

4.3 Application to bio-robotics

In order to transfer the biological principles of movement control to robotics, we are investigating how the model might be incorporated as a standard part of the robotic control. Therefore, an artificial hand with two fingers and one thumb is being developed to demonstrate the neural algorithms taking into account all the physical laws which otherwise may be neglected or simplified in a mathematical model (Chou and Hannaford, 1997). Each multi-jointed finger prototype has 3 DoF's (for the metacarpophalangeal (MCP), proximal interphalangeal (PIP), and distal interphalangeal (DIP) joints, respectively). The hand is attached to the wrist. The angles of rotation about the wrist are limited to flexion (or volar flexion) and extension (or dorsiflexion). Also, hand supination and pronation orients the finger opposition space during hand preshaping. The hand geometry is similar to that of the Stanford/JPL hand (Mason and Salisbury, 1985). Both the PIP and the DIP joints have flexion and extension, and the MCP joint has abduction/adduction, thereby allowing changes in the finger bending direction, thus permitting the hand to perform a variety of prehensile tasks, such as cylindrical grasp for heavy, convex workpieces, power grasp with fingers and thumb curled about the object such as a hammer, modified hook grasp with thumb along tool, three fingertip grasp for light circular or cylindrical objects, three finger precision or tripod grip, two finger precision or palmar grip and lateral pinch grip (Cutkosky and Wright, 1986).

Muscle-like actuators attach to steel tendon wires which wrap around a pulley embedded with an encoder at each joint using a novel mechanical design for routing the tendons along the finger, ensuring a linear displacement-joint angle relationship. Opponent, muscle-like actuators are critical components of the robotic gripper as they need to be used if one is to achieve the position-code invariance relationship using the model described herein.

4.4 Discussion

In principle, the use of sensors can allow an artificial hand to examine, *feel* and detect changes and variations in the workpiece and adapt its operation to them. This requires, however, the sensor output signals to be integrated directly in the neural (spinal) force controller (Contreras-Vidal, 1994; Contreras-Vidal et al., 1997). Although predictive (cerebellar) feedforward sensory control mechanisms support grasp stability even before object contact, sensory feedback generated by discrete mechanical events (e.g., initial object contact and release) is essential for monitoring and successful completion of the series of events related to the different grasping and manipulation phases (Johansson and Cole, 1994). Indeed, Edin et al. (1992) have shown that the grip/load force ratios employed at each finger engaged in a lifting task are controlled independently and are based on current sensory tactile information and anticipatory parameter control.

Hand compliance is also important to compensate for positioning errors and to improve stability when contacting rigid or fragile environments. Compliance control is already being used in robotic gripper applications, but these systems provide limited control capacity at present (e.g., Remote Center Compliance). The present model of cortico-spino-cerebellar dynamics provide a novel mechanism for overcoming these limitations, and suggests that the cerebellum is involved in the learning and coordination of opponent muscle synergies.

Acknowledgements

Part of this work was done while the first author was at the Department of Cognitive and Neural Systems at Boston University, in collaboration with Profs. Daniel Bullock and Stephen Grossberg.

References

Bullock, D. and Contreras-Vidal, J.L. (1993). How spinal neural networks reduce discrepancies between motor intention and motor realization. In K. Newell and D. Corcos (Eds.), *Variability and motor control*, Champaign, Illinois: Human Kinetics Press, pp. 183-221.

Bullock D., Grossberg S., & Guenther, F.H. (1993) A self-organizing neural model of motor equivalent reaching and tool use by a multijoint arm, *J. Cognitive Neuroscience*, **5**:408-435.

Chou C-P. & Hannaford B. (1997) Study of human forearm posture maintenance with a physiologically-based robotic arm and spinal level neural controller, *Biol Cybernetics*, **76**:285-298.

Contreras-Vidal, J.L. (1994). **Neural networks for motor learning and control of posture and movement.** Unpublished Doctoral dissertation: Boston University (no. 9334218).

Contreras-Vidal, J.L., Grossberg S. & Bullock, D. (1997) A neural model of cerebellar learning for arm movement control: Cortico-spino-cerebellar dynamics, *Learning and Memory*, **3**:475-502.

Contreras-Vidal, J.L., and Stelmach, G.E. (1995) A neural model of basal ganglia-thalamocortical relations in normal and Parkinsonian movement. *Biol. Cybern.*, 73:467-476.

Cutkosky MR, Wright PK (1986) Modeling manufacturing grips and correlations with the design of robotic hands, *IEEE Robotics and Automation*, **3**:1533-1539.

Edin BB, Westling G, Johansson RS (1992) Independent control of fingertip forces at individual digits during precision lifting in humans, *J Physiol (London)*, **450**:547-564.

Henatsch, H.D., Meyer-Lohman J., Windhorst U. & Schmidt J. (1986) Differential effects of stimulation of the cat's red nucleus on lumbar alpha motorneurons and their Renshaw cells, *Exp Brain Res*, **62**:161-174.

Humphrey, D.R. and Reed, D.J. (1983). Separate cortical systems for control of joint movement and joint stiffness: Reciprocal activation and coactivation of antagonist muscles. In J.E. Desmedt (ed.), **Motor control mechanisms in health and disease**. New York: Raven Press, pp. 347-372.

Ito, M. (1991) The cellular basis of cerebellar plasticity. *Curr. Opin. Neurobiol.*, **1**:616-620.

Johansson RS, Cole KJ (1994) Grasp stability during manipulative actions, *Can J. Physiol. Pharmacol.*, **72**:511-524.

Mason M.T., Salisbury J.K. (1985) **Robot Hands and the Mechanics of Manipulation**, MIT Press, Cambridge, MA.

Chapter 5

Multiple Internal Models in the Cerebellum

Mitsuo Kawato 2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-02.
ATR Human Information Processing Research Laboratories
Dynamic Brain Project, ERATO, JST
Kyoto, 619-02 Japan

5.1 Feedback-error-learning model of the cerebellum and its physiological examination

In this part, I will discuss the following three questions regarding the internal models of a motor apparatus and the external world in visual-motor transformations such as visually guided reaching movements. (1) Why are internal models necessary for visual-motor transformation? (2) Where could internal models be located in the brain? (3) How can internal models be acquired in the brain?

(1) At least three computational problems must be solved for visually-guided arm reaching movements: coordinate transformation, trajectory formation and motor command generation. All three of these problems require forward/inverse kinematics and dynamics models of the arm. Especially, for the control problem, the low stiffness of the arm which is observed during multi-joint movements suggests the existence of an inverse dynamics model of the arm (Gomi and Kawato, 1996).

(2) Internal models could possibly be located in many places in the brain. However, the cerebellar cortex is one of the few places experimentally supported. An inverse dynamics analysis of cerebellar Purkinje cell firing patterns in the ventral parafloculus (VPFL) suggests that the VPFL constitutes an inverse dynamics model of the eye plant for the ocular following responses (OFR) (Shidara et al., 1993)

(3) Acquiring inverse models through motor learning is computationally difficult. We proposed a cerebellar feedback-error-learning model to resolve this difficulty (Kawato and Gomi, 1992). This scheme was recently experimentally supported by a generalized linear model analysis of the complex spike firing probability in the VPFL during OFR (Kobayashi et al., 1995, 1997; Kawato et al., 1997). Figure 5.1 summarizes the current schema of the neural circuit controlling the OFR in monkey, and it provides a direct support to the feedback-error-learning model of the cerebellum. The cerebellar VPFL is the essential site for this visuo-motor transformation.

5.2 Robotics applications

The feedback-error-learning model has been applied to several different controlled objects ranging from a conventional industrial robotic manipulator, PUMA (Miyamoto et al., 1988), an automatic braking system of an automobile (Ohno et al., 1994), Bridgestone's rubbertuator (pneumatic rubber artificial muscles) SoftArm robots (Katayama and Kawato, 1991). More high-level, task learning applications are teaching by demonstration in Sarcos dextrous arm (Miyamoto et al., 1996) using the model of the phylogenetically newer part of the cerebellum, which is illustrated in the next section.

5.3 Embedding of multiple internal models in the lateral cerebellum

We hypothesize that the computational mechanism is common for different parts of the cerebellum. In the lateral hemisphere, which is extensively developed in man, internal models of the external world, such as tools, other people's brains, and other parts of the brain, are assumed to be acquired through learning. Imamizu et al. (1997) have obtained fMRI data supporting this view.

The ventrolateral part of the dentate nucleus and its corresponding part of the cerebellar lateral hemisphere are almost unique to man. I propose that this phylogenetically newest part of the cerebellum provides many different internal models of the external world as well as other parts of one's own brain which are essential for cognitive functions such as mental imagery, non-verbal communication, language, thoughts and self-consciousness. Especially, I emphasize mimesis of Merlin Donald (1991): motor skill to use the whole body as a representational device as the central part of human intelligence which is enabled by bi-directional interactions between these internal models.

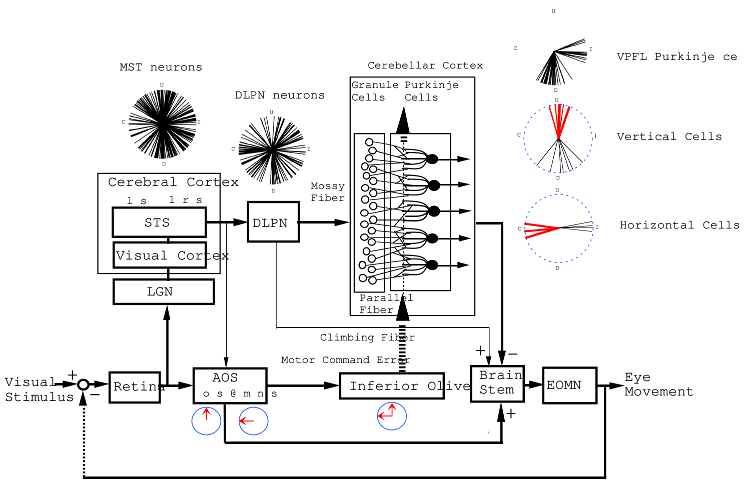

Figure 5.1: Neural circuit of OFR.

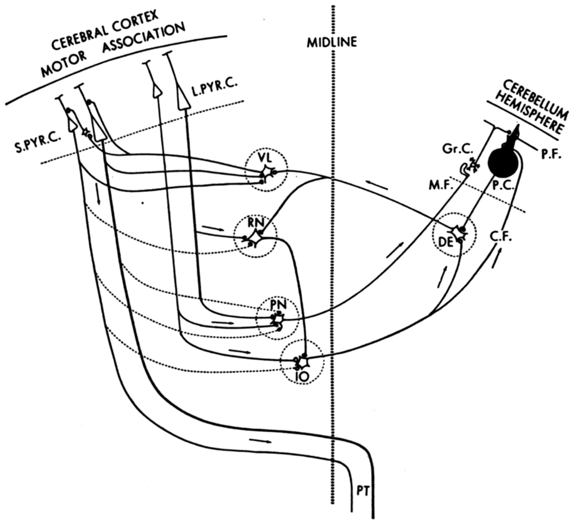

Figure 5.2: Ogawa's triangle.

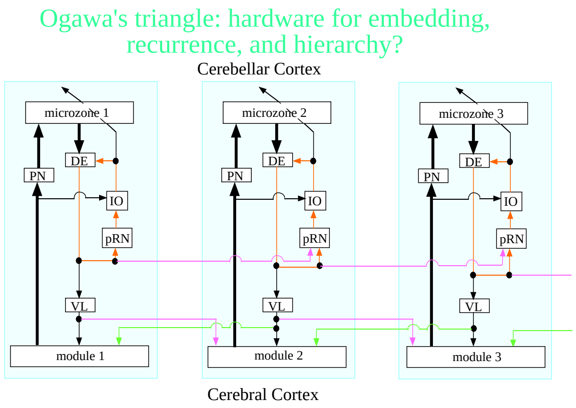

Figure 5.3: Embedding model of cerebellar modules

The difference between human and chimpanzee's intelligence is often characterized in their ability of hierarchical embedding in tool usage, motor plan and language. The newest part of the cerebellum is characterized by a unique anatomical structure called Ogawa's triangle (see Figure 5.2): closed loop formed by the dentate nucleus, parvocellular part of the red nucleus, and the inferior olive. I propose that this closed loop makes one internal model could be trained based on an error signal which is generated by another internal model, and thus makes learning of the hierarchical embedding structure possible (Figure 5.3). This hypothesis is based on the following data and computational reasoning.

Recently, flood of data suggest cognitive functions of the human cerebellum. PET, fMRI, patients, anatomy: the broad range demonstrated is quite impressive: motor imagery, noun-verb association, insane task, visual shape discrimination, mental rotation, sensory discrimination, attention, tower of Hanoi, motion perception, autism, working memory, and general IQ!

The cerebellum expands (2.8) just as the cerebrum (3.2) compared with an average of non-human primates with the same body weight. Especially, dentate nucleus, and furthermore, its ventro-lateral part was enormously expanded. Taken this with the fact that the cerebellum is 10%, 50% and more than 100% of the cerebrum in weight, surface area and the cell number, respectively, it is not surprising if human intelligence is mainly due to the newest part of the cerebellum, or at least its close connection with the cerebral frontal cortex by enormous number of cerebro-pontine fiber (20 million).

Human brain started to take the present form some 3 million years ago. On the other hand, language is supposed to be much more recent event (probably between 50 to 300 thousands years ago). Meanwhile, human intelligence which enabled making stone tools, domestication of fire, immigration across continents, was characterized as mimesis by Merlin Donald. This motor skill includes or is very much related to phenomena like imitation, perception of biological motion, motor theory of movement-pattern perception, mirror neurons, mental motor imagery. From computational viewpoint, most of the elements could be implemented by bi-directional information processing using both forward and inverse models of the external world including other's brain.

From computational reasoning, our group postulated to use the same optimal principle and neural network hardware which are used for motor planning and execution for perception of communication signals (Kawato, 1996). Recent physiological and brain imaging studies support this general idea: mirror neurons and motor regions are lit up during movement perception.

In summary, the hypothesis is based on the following factors: (1) Neural circuit is uniform over the cerebellum, thus some essential computational principle should be preserved. (2) Internal models and error-driven learning are the computational principles of the phylogenetically older part of the cerebellum. (3) Brain imaging study also supports the internal model hypothesis and the error-driven learning for the phylogenetically newer part of the human cerebellum (cf. Raichle et al., 1994; Imamizu et al., 1997). (4) Because the inputs and outputs of the newest part of the cerebellum are not connected directly with motor apparatus or sensory organs, internal models acquired there should not be simple inverse or forward models of the motor apparatus or sensory organs. (5) Because the newest part of the cerebellum are almost unique to human brain, it should have functions unique to human intelligence, especially mimesis. Thus, internal models are modeling other parts of one's own brain, other person's brain, other person's body, the external world other than motor apparatus or sensory organs, for example tools.

Acknowledgements

Supported by HFSP Grant.

References

Donald M: *Origins of The Modern Mind.* Harvard University Press, Cambridge, Massachusetts. (1991).

Gomi H, Kawato M: Equilibrium-point control hypothesis examined by measured arm-stiffness during multi-joint movement. *Science*, **272**, 117-120 (1996).

Imamizu H, Miyashita S, Sasaki Y, Takino R, Pütz, Kawato M: Cerebellar activity related to acquisition of internal models of a new tool: A functional MRI study. *Abstracts of the 27th Annual Meeting Society for Neuroscience*, **23**, 1053 (1997).

Katayama M, Kawato M: Learning trajectory and force control of an artificial muscle arm by parallel-hierarchical neural network model. In Lippmann RP, Moody JE, Touretzky DS (Eds.) *Advances in Neural Information Processing Systems* **3**. Morgan Kaufmann, San Mateo. 436-442 (1991).

Kawato M, Gomi H: The cerebellum and VOR/OKR learning models. *Trends in Neurosciences* **15**, 445-453 (1992).

Kawato M: Bi-directional theory approach to integration. In Inui T, McClelland J (Eds.) *Attention and Performance, XVI* MIT Press, Cambridge, Massachusetts. 335-367 (1996).

Kawato M, Kobayashi Y, Kawano K, Takemura A, Inoue Y, Kitama T, Gomi H: Cell-to-cell negative correlations between the simple spike and the complex spike firing characteristics of individual Purkinje cells. *Abstracts of the 27th Annual Meeting Society for Neuroscience*, **23**, 1299 (1997).

Kobayashi Y, Kawato M, Kawano K, Inoue Y, Shidara M, Gomi H: Inverse-dynamics representation of complex spike discharges of Purkinje cells in monkey cerebellar ventral paraflocculus during ocular following responses. *Abstracts of the 25th Annual Meeting Society for Neuroscience*, **21**, 140 (1995).

Kobayashi Y, Kawano K, Takemura A, Inoue Y, Kitama T, Gomi H, Kawato M: Temporal firing patterns of Purkinje cells in the cerebellar ventral paraflocculus during ocular following responses in monkeys. *Abstracts of the 27th Annual Meeting Society for Neuroscience*, **23**, 1299 (1997).

Miyamoto H, Kawato M, Setoyama T, Suzuki R: Feedback-error-learning neural network for trajectory control of a robotic manipulator. *Neural Networks* **1**, 251-265 (1988).

Miyamoto H, Schaal S, Gandolfo F, Gomi H, Koike Y, Osu R, Nakano E, Wada Y, Kawato M: A Kendama learning robot based on bi-directional theory. *Neural Networks*, **9**, 1281-1302 (1996).

Ohno H, Suzuki T, Aoki K, Takahashi A, Sugimoto G: Neural network control for automatic braking control system. *Neural Networks*, **7**, 1303-1312 (1994).

Raichle ME, Fiez JA, Videen TO, MacLeod AK, Pardo JV, Fox PT, Petersen SE: Practice related changes in human brain functional anatomy during nonmotor learning. *Cerebral Cortex*, **4**, 8-26 (1994).

Shidara M, Kawano K, Gomi H, Kawato M: Inverse-dynamics model eye movement control by Purkinje cells in the cerebellum. *Nature*, **365**, 50-52 (1993).

Chapter 6

Using Crude Corrective Movements to Learn Accurate Motor Programs for Reaching

Andrew H. Fagg
Department of Computer Science
University of Massachusetts
Amherst, MA 01002

Leo Zelevinsky
Department of Computer Science
University of Massachusetts
Amherst, MA 01002

Andrew G. Barto
Department of Computer Science
University of Massachusetts
Amherst, MA 01002

James C. Houk
Department of Physiology
Northwestern University
School of Medicine
Chicago, IL 60611

Abstract

A computational model that uses crude corrective movements to learn accurate reaching programs is presented. The model learns a feed-forward motor program in the form of pulse-step commands which drive a dynamic, 2 DOF, planar arm. The arm is actuated by two pairs of opposing muscles, which drive the shoulder and elbow joints. The model is able to learn motor programs that accurately bring the arm to the target, while producing in many cases bell-shaped tangential velocity profiles.

6.1 Introduction

Young infants reaching to touch or grasp an object make movements that differ greatly from adult grasping movements. Instead of tangential velocity profiles that are dominated by a single peak and are approximately bell-shaped, infant profiles consist of a sequence of peaks, which define segments of movement that are referred to as "movement units" [von Hofsten, 1979]. As the infant develops, the number of movement units required to reach a target is reduced, until adult-like velocity profiles are ultimately achieved. Berthier (1994) suggested that some of the individual peaks result from the discrete triggering of a movement-generating mechanism, and that the individual movement units are attempts at correcting errors in the previous movement. Berthier, Singh, Barto and Houk (1993) hypothesized in a model of cerebellar learning that similar corrective movements, though crude, may serve as a source of training information for an adaptive process that tunes a motor program.

In this paper, we focus on the issue of using crude corrective movements to learn accurate motor programs for controlling a simulated dynamic arm. The arm is actuated by two pairs of opposing muscles, which drive the shoulder and elbow joints. For the purposes of the experiment described in this paper, the learning controller represents a simplistic motor program for moving the arm from a single initial position to a single target, and takes the form of pulse-step motor commands that are sent to the muscles. The pulse-step commands are tuned as a function of the corrective movements that follow the learned movement. Despite the crude nature of the corrective information, and the simple representation of the motor program, the system learns to generate accurate reaching movements that exhibit, in some cases, straight paths and bell-shaped velocity profiles.

6.2 A Simple Control System

The control system consists of two separate modules which generate motor signals in the form of activation levels for each of the four muscles (Figure 6.1). The *learning controller* stores a feed-forward motor program that attempts (with learning) to bring the arm to the target. The hard-wired *corrector module* is responsible for generating crude corrective movements when the learning controller does not produce accurate reaches. The output control signals of these two modules are summed, and then passed through a cascade of three first-order, low-pass filters before arriving at the muscles.

The muscles are modeled as visco-elastic (spring) elements. A single muscle control signal sets the threshold for the muscle's length-tension curve. We assume a linear dependence of tension on

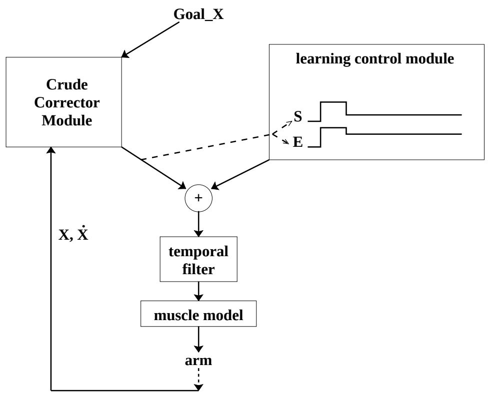

Figure 6.1: The Control Architecture. Both the *crude corrector* and *learning* modules produce muscle activation signals. These two sets of signals are summed, the result of which is passed through a temporal filter, before affecting the state of the muscles. The learning module produces pulse-step activation patterns that are intended to bring the arm from a given state to a target. The corrector module generates pulses of activation when the arm comes to rest at a point far from the target.

length [Houk and Rymer, 1981]. The dependence of tension on velocity is described by the Hill equation for shortening [Winters and Stark, 1987], and is dominated by the stretch reflex during lengthening. The non-linear damping provided by the stretch reflex is approximated by $-\dot{l}^{1/5}$, where $\dot{l}$ is the velocity of muscle stretch [Gielen and Houk, 1984]. The passive forces are modeled as exponentials, the magnitudes of which are only significant near the joint limit, as in Winters and Stark (1985).

For an agonist/antagonist pair of muscles, the control signals specify a unique equilibrium point for the joint. However, due to the high, non-linear damping forces generated by the stretch reflex, there exists a region around the equilibrium in which the joint velocity is rapidly forced towards zero, effectively causing the joint to "stick". A more detailed treatment of related muscle models may be found in [Wu et al., 1990, Fagg et al., 1997, Barto et al., 1998]. For simplicity, the muscle moment arms are assumed to be constant over the range of movement.

The learning controller represents a motor program in the form of a muscle activation pattern (the *pulse*), which is followed at a specified time by a second pattern of activation (the *step*)[1]. The details of these pulse-step programs are collapsed in the learning controller into a set of two parameters for each joint: the heights of the pulse and step (the activation pattern for each flexor/extensor muscle pair is specified by a single parameter). For simplicity, we focus here on the case of learning to reach from a single initial point to a single target. Therefore, only four parameters are represented by the learning controller. The time of transition from the pulse to step is linearly scaled with the maximum of the shoulder and elbow pulse heights. When appropriately tuned, the pulse phase serves to initiate the movement of the arm towards the target. The step phase sets the threshold of the stretch reflex that is responsible for slowing the arm, and thereafter maintaining the arm at the target [Ghez and Martin, 1982].

In addition to the learning controller, our model assumes the existence of a pre-wired corrective control module that is capable of making crude movements in the approximate direction of the target. This module produces a corrective movement only in cases in which the arm stops moving before the target is reached. The movement is generated using fixed-duration, constant-magnitude bursts of muscle activity. For a given correction, the burst magnitudes for the shoulder and elbow agonists are chosen according to three heuristics: 1) Inspired by the observations made by Karst and Hasan (1991), the relative activation level of the shoulder and elbow muscles is derived from the target's position in a visual coordinate system that is rooted at the wrist, and whose orientation is that of the forearm (Figure 6.2). Target locations along the axis of the forearm are translated into pure shoulder movements, whereas those located along the perpendicular axis result in pure elbow movements. Off-axis targets recruit a mixture of shoulder and elbow muscles; 2) The magnitude of the muscle bursting pattern is scaled with distance to the target; and 3) If the arm has not moved significantly since the last correction, then the magnitude of the muscle bursting pattern is increased by a constant factor.

In many (but not all) instances, the movements generated by this set of heuristics bring the arm closer to the target. However, the target is rarely reached on the first correction. Instead, a sequence of corrections is typically required to complete a given reach.

6.3 The Learning Algorithm

The protocol for training is as follows: At the beginning of a trial, the arm is placed in a starting configuration, and a target location is specified. The reaching movement is then triggered – the learning controller executes its pulse-step motor program. If the arm stops moving at a position not near the target, then the corrector module produces a brief pulse of activation. If the arm again stops far from the target, then the correction process is repeated until the wrist is within 1 cm of the target.

[1] Pulse-step control of rapid eye movements has been well established [Robinson, 1975]; this form of control appears to be a reasonable abstraction in the context of arm movements [Ghez and Martin, 1982].

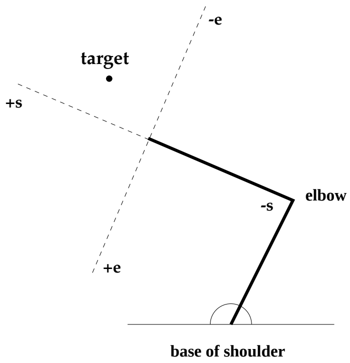

Figure 6.2: Heuristic method for choosing the relative magnitudes of the elbow and shoulder flexor/extensor muscle bursts. Targets located in the half-plane denoted by $+e$ recruit elbow flexors; the $-e$ half-plane recruits elbow extensors; $+s$: shoulder flexors, and $-s$: shoulder extensors.

Not only do the pulses generated by the corrector module take the arm closer to the target, but they also provide information about how the learning module's pulse-step motor program should be updated. If the step levels are miscalibrated, then even if the pulse is able to bring the arm near the target, the arm will subsequently drift away. When the arm stops moving, this will result in the production of a corrective movement, whose direction will indicate how the learning controller's *step command* should be updated.

In contrast, if the pulse command is drastically miscalibrated (but the step is correct), the position of the arm shortly after the execution of the learning controller's pulse-step program will be far from the target, resulting in a sequence of corrections. In this case, the training information derived from the corrections is used to update the *pulse command* of the learning controller.

The full learning procedure, in which the pulse and step are learned in parallel, is summarized below. L_p and L_s are vectors representing the pulse and step commands of the learning controller, respectively; $C(X,T)$ represents the corrector motor signal, which is a function of the arm (X) and target (T) positions.

FOR each trial
- Set the target and initial arm positions: T and $X(0)$, respectively
- Execute the pulse command: L_p;
- Wait for pulse duration
- Execute the step command: L_s;
- Wait for arm to stop
- UNTIL arm is at the target
  - Execute the corrective pulse: $L_s + C(X,T)$
  - IF the arm has been near the target during this trial
    - $L_s \leftarrow L_s + \alpha C(X,T)$
  - ELSE
    - $L_s \leftarrow L_s + \hat{\alpha} C(X,T)$
  - $L_p \leftarrow L_p + \beta(i) C(X,T)$
  - Wait for arm to stop

α, $\hat{\alpha}$, and $\beta(i)$ are step-size parameters in which $\hat{\alpha} \ll \alpha$, and $\beta(i)$ decays exponentially with the number of corrections, i, made since the beginning of the trial.

In the above algorithm, the corrective pulse is taken to be the sum of the learning controller's step (L_s), and the pulse generated by the corrector module ($C(X,T)$). Without the learning controller's step, it would be necessary for the corrector module to also take into account the absolute position of the target in order to produce movements in the correct direction, thus adding to the "intelligence" that would have to be hard-wired into this module.

6.4 A Learning Experiment

The behavior of the system while learning to move from a single initial position to a single target is illustrated in Figure 6.3. After 10 learning trials (Figure 6.3A), the pulse signals for the shoulder and elbow flexors that are generated by the learning controller are not large enough to bring the arm to the target. This results in a sequence of six corrective movements that make smaller and smaller "hops" towards the target. The corrector module recruits the shoulder flexor for all six corrections (indicated by six additional peaks in the flexor motor command). For the elbow, however, the first two corrections recruit the flexor, but the remaining corrections involve the extensor. We also see flexor activity during these latter corrections; this is due to the fact that the learning controller's step phase recruits the elbow flexor (recall that the learning controller's step is combined with the corrector's pulse in order to generate a corrective movement).

After 34 trials of learning (Figure 6.3B), the wrist makes a relatively straight-line movement to the target, and the velocity profile is approximately bell-shaped. Furthermore, the learned movement does not require the invocation of the corrector module.

Figure 6.4 demonstrates the paths learned by the system for 3 initial positions and 47 target positions[2]. For each initial position/target pair, the pulse-step commands of the learning controller were first cleared; learning trials were then presented to the system until the movement was com-

[2]The one path not shown (center starting position of Figure 6.4B, movement to the right and down) was not learnable by the system due to limitations in the algorithm in dealing with the high passive forces near the joint limit.

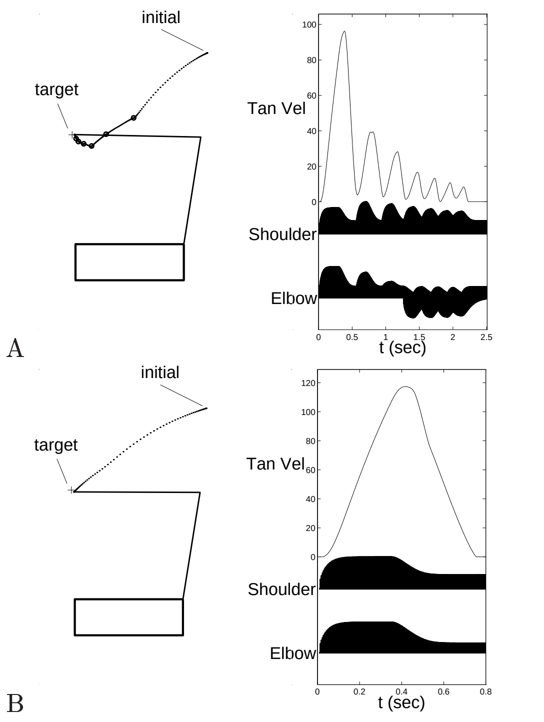

Figure 6.3: System behavior at trial 11 (A) and trial 35 (B) during learning for a single initial position/target pair. The small dots indicate the path taken by the wrist; larger dots show points at which the corrective movements were generated. The temporal plot shows the tangential velocity of the wrist (cm/sec), and the filtered motor commands for the shoulder flexor (extending upward), extensor (downward), and the elbow flexor (up) and extensor (down).

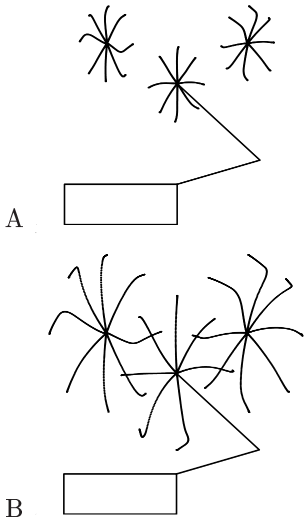

Figure 6.4: Learning movements for a set of center-out movement tasks of length 10 (A) and 20 (B) cm. Each path is the result of separate learning runs, each consisting of an average of 33 learning trials.

pleted with no corrections. For these movements, an average of 33 trials were required for each training pair (the range was 16–98 trials). The hooks that occurred at the end of many of the movements resulted from one joint completing its motion before the other. This happened typically when the magnitudes of movement in joint space differed significantly between the two joints. Also, in these cases, the tangential velocity profile showed a high velocity peak, followed by either a second (lower) velocity peak or a flat region.

6.5 Discussion

This paper has attempted to demonstrate that a crude error correction device may be used to drive the adaptation of a high-quality motor program. Our model is able to learn feed-forward motor programs for reaching that accurately bring the arm to the target, while producing in some cases tangential velocity profiles that are single-peaked and bell-shaped. The trajectories generated by our model that do not reflect these adult-like properties end with hooked paths, which are the result of one joint completing the movement before the other.

Hollerbach and Atkeson (1987) have suggested that by staggering the onset of movement of one joint relative to the other, curved movements may be made to appear straighter. Several authors have reported experiments which suggest that such a strategy may be utilized by humans [Kaminski and Gentile, 1986, Karst and Hasan, 1991b]. Karniel and Inbar (1997) have proposed a model similar to ours that makes use of pulse-step control to drive a 2 degree-of-freedom arm. In their model, the motor programs are described not only by the height of the pulse and step for each joint, but also by the relative timing of the joint pulses. By using these additional degrees of freedom, they are able to learn movements that do not reflect the hooks that are seen in our model. Their learning procedure, however, relies on a systematic variation of the control parameters to estimate an error gradient, which is then used to further reduce movement errors.

Our approach is also related to Kawato's feedback error learning, in that training is driven by movement produced by a controller external to the learning system [Kawato and Gomi, 1992]. However, our model differs from this work in two key ways. First, feedback in our model is not provided continuously, rather it is available sporadically. Second, our model does not require the use of a high-quality reference trajectory from which the corrective movements are computed.

Although what is learned in the present model is a pure feedforward motor program for one initial position/target pair, the learning approach is more generally applicable at two levels. First, it would be possible to learn to compute the pulse-step parameters as a function of the current state of the arm and target position, allowing the model to learn reaching movements over the entire workspace. Second, in the context of a more realistic cerebellar learning model, it is possible to learn to combine contextual information, motor efference copy, and delayed sensory inputs in order to compute, in an on-line fashion, the outgoing motor commands [Fagg et al., 1997, Barto et al., 1998].

Acknowledgments

This work is supported by the National Institutes of Health (grant #NIH 1-50 MH 48185-02).

Bibliography

[Barto et al., 1998] Barto, A. G., Fagg, A. H., Houk, J. C., and Sitkoff, N. (1998). A cerebellar model of timing and prediction in the control of reaching. *Submitted to Neural Computation*.

[Berthier, 1994] Berthier, N. E. (1994). Infant reaching strategies: theoretical considerations. *Infant Behavior & Development*, 17:521.

[Berthier et al., 1993] Berthier, N. E., Singh, S. P., Barto, A. G., and Houk, J. C. (1993). Distributed representations of limb motor programs in arrays of adjustable pattern generators. *Journal of Cognitive Neuroscience*, 5:56–78.

[Fagg et al., 1997] Fagg, A. H., Sitkoff, N., Barto, A. G., and Houk, J. C. (1997). A model of cerebellar learning for control of arm movements using muscle synergies. In *Proceedings of the IEEE Symposium on Computational Intelligence in Robotics and Automation*, pages 6–12.

[Ghez and Martin, 1982] Ghez, C. and Martin, J. M. (1982). The control of rapid limb movment in the cat. III. Agonist-antagonist coupling. *Experimental Brain Research*, 45:115–125.

[Gielen and Houk, 1984] Gielen, C. C. A. M. and Houk, J. C. (1984). Nonlinear viscosity of human wrist. *Journal of Neurophysiology*, 52:553–569.

[Hollerbach and Atkeson, 1987] Hollerbach, J. M. and Atkeson, C. G. (1987). Deducing planning variables from experimental arm trajectories: Pitfalls and possibilities. *Biological Cybernetics*, 56:67–77.

[Houk and Rymer, 1981] Houk, J. C. and Rymer, W. Z. (1981). Neural control of muscle length and tension. In Brooks, V. B., editor, *Handbook of Physiology: Sec. 1: Vol. 2. Motor Control*, pages 247–323. American Physiological Society, Bethesda, MD.

[Kaminski and Gentile, 1986] Kaminski, T. and Gentile, A. M. (1986). Joint control strategies and hand trajectories in multijoint pointing movements. *Journal of Motor Behavior*, 18(3):261–278.

[Karniel and Inbar, 1997] Karniel, A. and Inbar, G. F. (1997). A model of learning human reaching-movements. *Biological Cybernetics*, to appear.

[Karst and Hasan, 1991a] Karst, G. M. and Hasan, Z. (1991a). Initiation rules for planar, two-joint arm movements: Agonist selection for movements throughout the work space. *Journal of Neurophysiology*, 66(5):1579–1593.

[Karst and Hasan, 1991b] Karst, G. M. and Hasan, Z. (1991b). Timing and magnitude of electormyographic activity for two-joint arm movements in different directions. *Journal of Neurophysiology*, 66(5):1594–1604.

[Kawato and Gomi, 1992] Kawato, M. and Gomi, H. (1992). A computational model of four regions of the cerebellum based on feedback–error learning. *Biological Cybernetics*, 68:95–103.

[Robinson, 1975] Robinson, D. A. (1975). Oculomotor control signals. In Lennerstrand, G. and Bach-y-rita, P., editors, *Basic Mechanisms of Ocular Motility and their Clinical Implications*, pages 337–374. Pergamon Press, Oxford.

[von Hofsten, 1979] von Hofsten, C. (1979). Development of visually directed reaching: The approach phase. *Journal of Human Movement Studies*, 5:160–168.

[Winters and Stark, 1985] Winters, J. and Stark, L. (1985). Analysis of fundamental human movement patterns through the use of in-depth antagonistic muscle models. *IEEE Transactions on Biomedical Engineering*, BME-32(10):826–839.

[Winters and Stark, 1987] Winters, J. and Stark, L. (1987). Muscle models: What is gained and what is lost by varying model complexity. *Biological Cybernetics*, 55:403–420.

[Wu et al., 1990] Wu, C. H., Houk, J. C., Young, K. Y., and Miller, L. E. (1990). Nonlinear damping of limb motion. In Winters, J. and Woo, S., editors, *Multiple Muscle Systems: Biomechanics and Movement Organization*. Springer–Verlag, New York.

Chapter 7

Adaptive Motor Control Without a Cerebellum

Mark E. Nelson
Department of Molecular & Integrative Physiology and
The Beckman Institute for Advanced Science and Technology
University of Illinois, Urbana-Champaign
Urbana, IL 61801

Abstract

Insects solve complex adaptive motor control problems without the benefit of a cerebellum. Analysis of the sensory-motor networks underlying the control of leg movements in insects reveals neural elements that perform functions similar to those often associated with the cerebellum: adjusting reflex gains and time constants, coordinating the activity of multiple joints, modulating the amplitude and timing of underlying pattern generator outputs, and adapting motor output to account for changes in behavioral context. In constructing neurally-inspired controllers for robots, it is useful to examine both vertebrate and invertebrate control system architectures to look for common functional features which may have quite different neural implementations.

7.1 Cerebellum in vertebrate motor control

Despite decades of intense research, the precise functional role of the cerebellum in vertebrate motor control is still largely unresolved. There seems to be a general consensus that the cerebellum is somehow involved in processes of adapting, calibrating, coordinating, and tuning motor output, although there are many different models for how this comes about. The classic Marr-Albus models view the cerebellum as a perceptron-based associative memory that controls elemental movements (Marr 1969, Albus 1971). More recently, other investigators have described the functional role of the cerebellum in terms of controlling an array of adjustable pattern generators (Houk *et al.*, 1996), as a means for establishing context-response linkages (Thach 1996), as an internal model of inverse-dynamics (Kawato and Gomi, 1995) and as an optimal state-estimator (Paulin, 1993).

7.2 Invertebrate motor control: Insect adaptive leg movements

Insects are quite adept at walking over irregular terrain. Under such circumstances, they do not use rigid gaits as observed when walking on smooth horizontal surfaces. Instead, as the insect negotiates obstacles and gaps in the terrain, individual leg and joint movements exhibit a wide variation in amplitude, timing, and duration as the insect adjusts its motor output in order to move efficiently through the environment (Pearson and Franklin, 1984).

Adaptive leg movements in the insect are controlled by neurons in the thoracic ganglia which can be classified into four groups: sensory afferents, spiking interneurons, nonspiking interneurons, and motor neurons. These groups are interconnected in a predominantly feedforward manner, however there are extensive recurrent connections within the interneuron layers. Fig. 7.1 summarizes the connections between these different groups of neurons (Burrows, 1992; Laurent 1993).

Among the many sense organs found on an insect leg, three classes of mechanoreceptors are relevant for locomotion control. Chordotonal organs provide information about the relative position and movements of individual limb segments, campaniform sensilla measure local forces on segments, and tactile hairs and spines signal contact with external objects or other limb segments (review: Delcomyn *et al.* 1996).

Spiking interneurons typically receive information from just one of these three sensory modalities, and there is evidence that these neurons are arranged topographically within the ganglion. Hence spiking interneurons can be thought of as forming unimodal, topographic sensory maps.

Non-spiking interneurons (NSIs) are so named because they do not generate action potentials, rather they communicate by graded synaptic transmission. NSIs have more complicated receptive fields than those of the spiking sensory interneurons and often have multimodal receptive fields. Interestingly, NSI activity is frequently responsive to sensory signals that arise from different modalities in the same behavioral context.

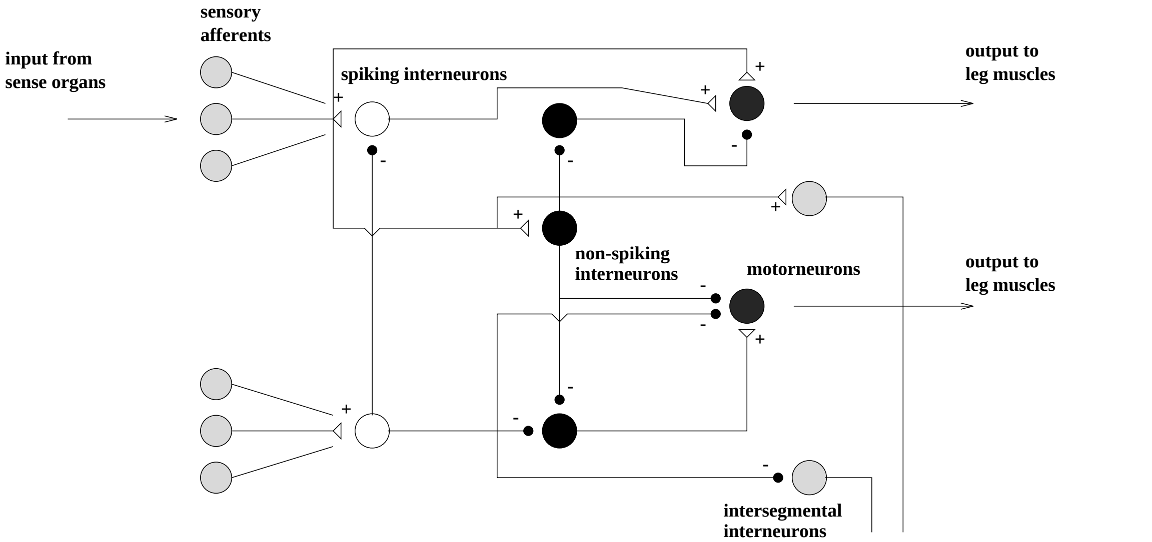

Figure 7.1: Organization of the neural circuitry controlling leg movements in the insect thoracic ganglion (from Burrows, 1992). *Sensory afferents* arise from sense organs on the leg. Afferents make contact with two classes of interneurons: *spiking interneurons* are sensory neurons; *non-spiking interneurons* are premotor neurons. Spiking and non-spiking interneurons make synapses onto *motor neurons*, which drive the leg muscles. In insects, some non-spiking interneurons have functional roles similar to those associated with the cerebellum in vertebrates.

NSIs are important premotor elements involved in posture control and locomotion. Some NSIs form central pattern generating (CPG) networks, while other NSIs appear to modulate the the properties of the CPG output (Büschges, 1995). Some NSIs receive input from inter-segmental interneurons and are thus involved in coordinating activity between legs. Other NSIs are involved in controlling the gain and time constants of local leg reflexes (Büschges and Schmitz, 1991). NSIs receive, integrate, and process information from multiple sensory modalities and drive functionally-related groups of leg motoneurons during walking (Schmitz *et al.*, 1992).

From the above description, it can be seen that NSIs in the insect thoracic ganglia are associated with functions similar to those associated with the cerebellum in vertebrate motor systems. They are involved in adjusting reflex gains and time constants, coordinating the activity of multiple joints, modulating the amplitude and timing of underlying pattern generator outputs, and adapting motor output to account for changes in behavioral context.

7.3 Substrate-finding reflex

In insects, a stereotyped searching movement of an individual leg can be elicited when support is removed from that leg. When the leg encounters a suitable substrate, it attempts to establish a foothold. This behavior is known as the substrate-finding reflex (Bässler 1993). To explore some of the adaptive motor control capabilities of the neural circuitry of the thoracic ganglia, we have imple-

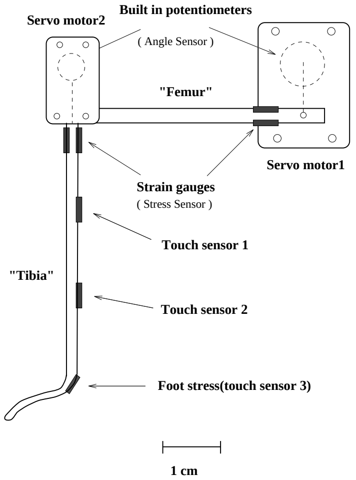

Figure 7.2: A two-joint insect-like robot leg. Each joint is powered by a servo actuator with muscle-like dynamic properties. Potentiometers provide joint angle measurements; strain gauges provide force and touch measurements.

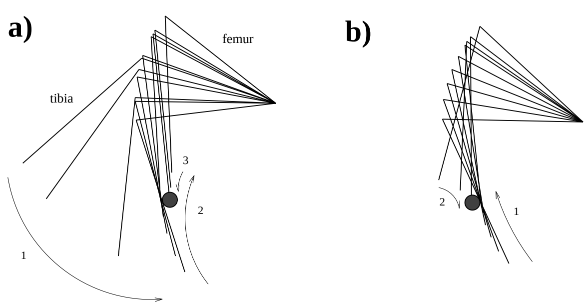

Figure 7.3: Substrate-finding behavior performed by a two joint robot leg. a) The leg encounters an object during the downward sweep of a search cycle; once contact is made, the leg slides up until it just clears the object and then comes back down to establish a foothold. b) The leg encounters the object during the upward sweep of a search cycle (from Ding and Nelson, 1995).

mented a biologically-inspired model of the insect substrate-finding reflex in a 2-joint robot leg (Ding and Nelson, 1995). The structure of the robot leg is shown in Fig. 7.2.

The robot leg is controlled using a neural architecture similar to that shown in Fig. 7.1. The controller dynamics are such that when the leg is in contact with the substrate and bearing weight (signaled by femur stress), the system moves to a stable fixed point corresponding to the *stance* phase of the substrate-finding behavior. In the absence of a femur stress signal, the system converges to a periodic attractor corresponding to the *search* phase of the behavior. When the leg encounters an object during the search phase (signaled by an increase in tibia stress), the controller output gives rise to a coordinated pattern of movements in which the leg slides up along the object maintaining a relatively constant tibia stress, until it just clears the object signaled by a drop in tibia stress, and comes back down to find support on top of the object. Typical robot leg trajectories are shown in Fig. 7.3.

7.4 Summary

The substrate-finding reflex is a simple model system for exploring adaptive motor control strategies in insects. In this system, non-spiking interneurons (NSIs) have functional roles similar to those associated with the cerebellum in vertebrate motor systems. They are involved in adjusting reflex gains and time constants, coordinating the activity of multiple joints, modulating the amplitude and timing of underlying pattern generator outputs, and adapting motor output to account for changes in behavioral context. In constructing neurally-inspired controllers for robots, it is useful to examine both vertebrate and invertebrate control system architectures to look for common functional features which may have quite different neural implementations.

Acknowledgments

This work supported by grant N000149610656 from the Office of Naval Research.

References

Albus J.S. (1971) A theory of cerebellar functions. *Math. Biosci.* **10** 25-61.

Bässler, U. (1993) The walking- (and searching-) pattern generator of stick insects, a modular system composed of reflex chains and endogenous oscillators. *Biol. Cybern.* **69**, 305-317.

Burrows M. (1992) Local circuits for the control of leg movements in an insect. *Trends Neurosci.* **15**, 226-232.

Büschges A. (1995) Role of local nonspiking interneurons in the generation of rhythmic motor activity in the stick insect. *J. Neurobiol.* **27**, 488-512

Büschges A. and Schmitz J. (1991) Nonspiking pathways antagonize the resistance reflex in the thoracocoxal joint of stick insects. *J. Neurobiol.* **22**, 224-237.

Delcomyn F., Nelson M.E., Cocatre-Zilgien J.H. (1996) Sense organs of insect legs and the selection of sensors for agile walking robots. *Intl. J. Robot. Res.* **15** 113-127.

Ding Z. and Nelson M.E. (1995) A neural controller for single-leg substrate-finding: a first step toward agile locomotion in insects and robots. In: *The Neurobiology of Computation*, J.M. Bower, ed., Kluwer Academic Press, pp. 379-384.

Houk J.C., Buckingham J.T., Barto A.G. (1996) Models of the cerebellum and motor learning. *Behav. Brain Sci.* **19**, 368-383.

Kawato M. and Gomi H. (1992) The cerebellum and VOR/OKR learning models. *Trends Neurosci.* **15**, 445-453.

Laurent G. (1993) Integration by spiking and non-spiking local neurons in the locust central nervous system: The importance of cellular and synaptic properties for network function. In: *Biological Neural Networks in Invertebrate Neuroethology and Robotics*, R.D. Beer, R.E. Ritzmann, T. McKenna (eds.), pp. 69-85, Academic Press.

Marr D. (1969) A theory of cerebellar cortex. *J. Neurophysiol.* **22**, 437-470.

Pearson K.G. and Franklin R. (1984) Characteristics of leg movements and patterns of coordination in locusts walking on rough terrain. *Intl. J. Robot. Res.* **3**, 101-112.

Paulin M.G. (1993) The role of the cerebellum in motor control and perception. *Brain Behav. Evol.* **41**, 39-50.

Schmitz J., Büschges A. and Kittmann R. (1992) Intracellular recordings from nonspiking interneurons in a semiintact, tethered walking insect. *J. Neurobiol.* **22**, 907-921.

Thach W.T. (1996) On the specific role of the cerebellum in motor learning and cognition: Clues from PET activation and lesion studies in man. *Behav. Brain Sci.* **19**, 411-431.

Chapter 8

Cerebellar control of a simulated biomimetic manipulator for fast movements

Jacob Spoelstra
Department of Computer Science
University of Southern California
Los Angeles, CA 90089

Michael A. Arbib
Department of Computer Science
University of Southern California
Los Angeles, CA 90089

Nicolas Schweighofer
ERATO - JST
Advance Telecommunication Research
2-2 Hikaridai, Seika-cho, Soraku-gun
Kyoto 619-02, Japan

Abstract

We present a learning neural model, based on the circuitry and functional connections of the cerebellum, to control reaching movements of a simulated biomimetic manipulator.

Basing our work on the [Schweighofer et al., 1997b] model, we propose a new system that utilizes the servo mechanism of the spinal reflex circuitry as a key element to allow kinematic control of slow movements while learning predictive dynamic compensation, and demonstrate the ability to rapidly learn to control stereotyped fast movements. The key elements of the model are 1) parallel postural and dynamics controllers [Katayama and Kawato, 1991,]; 2) learning neural network models explicitly modeled on the physiology of the cerebellum; 3) detailed neural model of spinal reflex circuitry; and 4) a muscle-like actuator model based on real artificial muscles.

Simulation results show the effectiveness of the system to learn accurate trajectory control for fast movements.

8.1 Introduction

[Schweighofer et al., 1997a] proposed a model of the role of the intermediate cerebellum in the control of voluntary movement. The model embeds a neural network, based on known cerebellar circuitry, in a simulation of the mammalian motor control system to control a 6-muscle 2-link planar arm. In this model the cerebellar module acts in parallel with a controller composed of a proportional-derivative feedback controller, and a feedforward controller (PDFF), the latter implementing a crude inverse dynamic model of the arm. The simulation results of [Schweighofer et al., 1997b] suggest that this cerebellar model was able to learn parts of the inverse dynamics model not provided by the PDFF controller, as indicated by an improved tracking performance of desired trajectories after learning. However, the system required a large number of training repetitions and could only learn to control relatively slow movements.

Basing our work on the Schweighofer model, we propose a new system that utilizes the servo mechanism of the spinal reflex circuitry as a key element to allow kinematic control of slow movements while learning predictive dynamic compensation, and demonstrate the ability to rapidly learn to control stereotyped fast movements. The key elements of the model are 1) parallel postural and dynamics controllers; 2) learning neural network models explicitly modeled on the physiology of the cerebellum; 3) detailed neural model of spinal reflex circuitry; and 4) a muscle-like actuator model based on real artificial muscles.

8.2 The model

The model uses parallel inverse static and dynamic controllers as shown in Figure 8.1 and is conceptually similar to the parallel hierarchical control scheme proposed by [Katayama and Kawato, 1991], where an inverse static model (ISM) and inverse dynamic model (IDM) are connected in parallel. The ISM learns the part of the inverse dynamics that depend only on posture, such as gravity terms, while the IDM learns terms that depend on derivatives of joint angles. Such an idea is also related to [Atkeson and Hollerbach, 1985] who showed that separating the static and dynamic terms of the in-

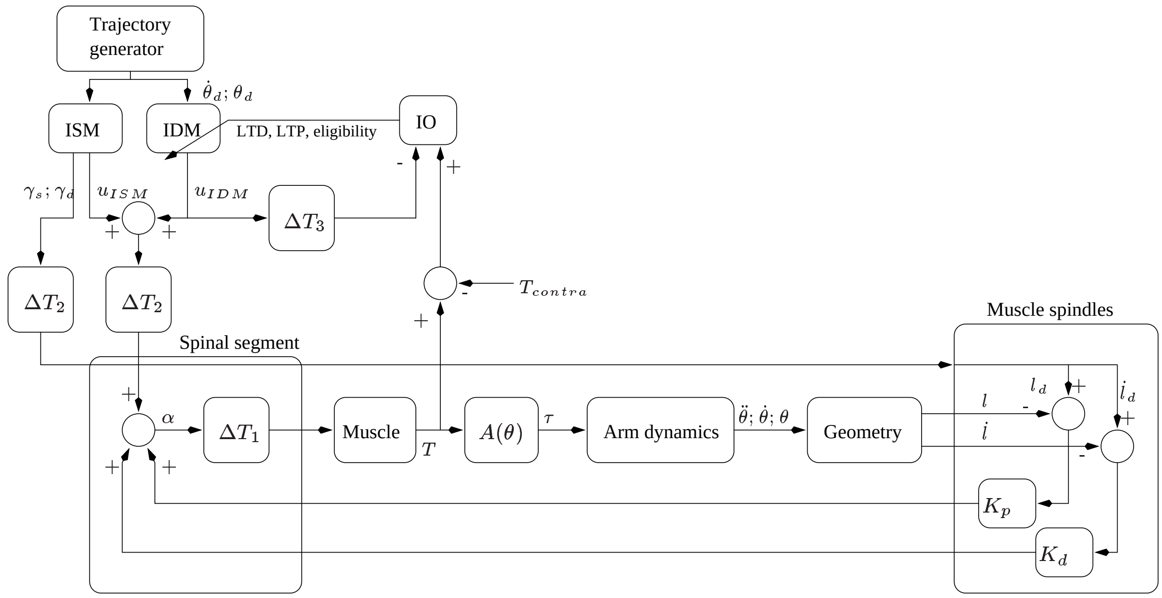

Figure 8.1: Simplified schematic of the control system. The trajectory generator defines a kinematic trajectory for each joint. The ISM provides alpha-muscle control input to define an equilibrium at the current desired position, but also generates static and dynamic gamma drive for the spindles to detect length and velocity errors. The IDM (implemented as a cerebellar model) uses the sensed muscle tension produced by the spinal reflex circuitry as teacher signal and learns to associate this with the system state (provided by the trajectory generator and muscle spindles) to provide corrective control signals in a feed-forward manner.

verse dynamics, gives the dynamic terms simple linear scaling properties for changes in speed and load.

8.2.1 Plant model

To provide a more realistic test-bed of the cerebellar model, we have provided detailed simulations not only of cerebellum but also of a two-segment planar arm and the spinal segment circuitry.

A neural model of the spinal segment circuitry implemented the motor servo. We used a model that was implemented in DSP hardware by [Chou and Hannaford, 1996b] for single joint posture control. As shown in Figure 8.2(A), the model incorporates Alpha- and Gamma-motoneurons, Renshaw cells, Ia - and Ib interneurons.

With an eye on robotic implementation, the plant simulated was a two-segment planar arm, actuated by 6 antagonistic McKibben pneumatic artificial muscles [Chou and Hannaford, 1996a] as shown in Figure 8.2(B). In addition, Ia-pathway kinematic feedback was provided from each muscle by static- and dynamic spindle pairs: separate gamma drive inputs set a “desired” length and velocity, the spindle output is a clipped (positive only) function of the position or velocity error. Simulated Golgi tendon organs provide force feedback to the Ib interneurons.

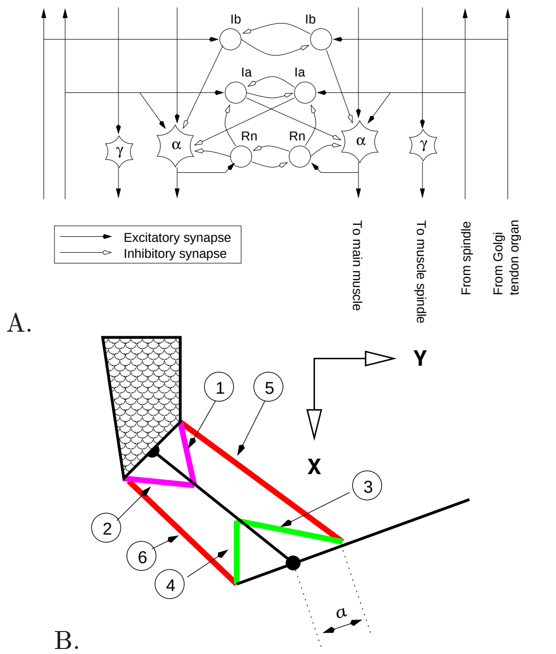

Figure 8.2: A: The neural circuit used to implement a spinal segment (redrawn from [Chou and Hannaford, 1996b]). B: Schematic representation of the muscle attachments on the 2-link arm.

8.2.2 Postural control

Our postural module (ISM) provides alpha drive to set muscle tensions so that the limb would have an equilibrium point at the desired position. Muscles are often modeled as damped springs with resting length determined as a function of the alpha motor command [Ozkaya and Nordin, 1991]. The result of this property is that a given vector of motor commands to a set of antagonistic muscles defines a point attractor for the limb in joint space. The equilibrium-point (EP) theory [Bizzi et al., 1984, Bizzi et al., 1992, Feldman, 1986] proposes that movements are effected by moving this equilibrium point from the starting to target configuration. The model has been criticized because measured muscle stiffness values would lead to unrealistic trajectories. [McIntyre and Bizzi, 1993] have suggested that the spinal reflex path could serve as a low latency position and velocity feedback servo, but [Schweighofer, 1995] has shown that fast multi-joint movements still require some feedforward compensation.

We propose that the ISM also drives intrafusal muscle spindles through the gamma system so that deviations from the desired position would activate the spinal servo to provide additional restoring force. This feedback system ensures that novel trajectories are approximately followed, but as suggested by [Gomi and Kawato, 1993], also serves to train the IDM as described below.

8.2.3 Cerebellar inverse dynamics controller

In parallel with the postural controller, alpha motoneurons in the model receive a second input from the cerebellar module (related to Gomi and Kawato's IDM) which implements a nonlinear predictive regulator by learning the inverse dynamics of the plant and spinal circuit. The intermediate cerebellum receives extensive input from motor- and premotor cortex (which we model as containing the representations of the descending postural command, and only desired joint position and velocity), as well as position and velocity error feedback from muscle spindles via the dorsal spinal tract. We propose that the intermediate cerebellum provides predictive assistive alpha-drive to reduce trajectory errors during fast movements.

We used the same cerebellar module described in a previous paper for learning visuo-motor transformations when throwing while wearing wedge prism glasses [Spoelstra and Arbib, 1997]. All cells are modeled as leaky integrators with membrane potential defined by

$$\tau \frac{dm}{dt} = -m + x \tag{8.1}$$

where x is the current synaptic input. The output of each cell is a positive real number representing the instantaneous firing rate and is derived from the membrane potential as

$$y(m) = \frac{y_m ax}{1 + \exp(\frac{m-o}{s})} \tag{8.2}$$

with o and s parameters that determine respectively the baseline firing rate and linear range of the cell.

Although the model is rather high-level, we have tried to incorporate the basic circuitry of the cerebellum [Ito, 1984]. Inputs arrive via mossy fibers to the granule cells whose axons bifurcate to form parallel fibers in the cerebellar cortex. Each Purkinje cell receives input from a large number of parallel fibers (PF) and one climbing fiber (CF) originating in the inferior olive (IO). The Purkinje cells are the sole output from the cerebellar cortex and inhibit the nuclear cells. Using mechanisms based on current theories of cerebellar learning as long term depression (LTD) of parallel fiber-Purkinje cell synapses after coactivation of parallel- and climbing fibers, and by incorporating known recurrent projections between cerebellar nuclear cells and inferior olive (IO) cells, we demonstrated that this produced a stable input-following learning system.

8.2.4 Learning

The operation of the IDM is defined by the information in the climbing fiber system. In the model the IO compares the output of the IDM with the muscle tension produced by the spinal feedback circuitry and descending commands combined, to drive the IDM to generate corrective control signals in predictive manner to eventually replace the feedback controller. By subtracting the tension produced by the contralateral muscle, co-contraction is minimized to produce effective reciprocal muscle activations.

The servo feedback controller gain is limited by delays, which lead to errors during fast movements. If the feedback controller is placed on the same loop with the learning inverse dynamics controller (as in the standard feedback-error learning scheme), delays cause the controller to produce a control signal (used for training the IDM) given by:

$$\tau_{fb} = K_p[\theta_d(t) - \theta_d(t - \Delta T)]. \tag{8.3}$$

The problem is that even if the trajectories were perfect, the feedback controller would continue to generate torques that would lead to an incorrect inverse dynamics model. By detecting the errors at the spindles, though, the feedback torque becomes

$$\tau_{fb} = K_p[\theta_d(t - \Delta T) - \theta_d(t - \Delta T)]. \tag{8.4}$$

The delay is on the forward path and can be eliminated by providing the same signal at time $(t - \Delta T)$.

A crucial element is the concept of synaptic eligibility [Sutton and Barto, 1981, Klopf, 1982]. The process of LTD has been shown to involve second messengers whose concentration tags synapses eligible for modification. [Schweighofer et al., 1996] has suggested that the second messenger concentration might follow second order dynamics to peak at a specified instant after parallel fiber activity, and that this could be used to solve the temporal credit assignment problem in saccade adaptation where the error information is available only after the control action. We used the same principle and show that the cerebellum learns to associate current control actions with prior states and learns to act as a predictive feedforward controller.

A further effect of the synaptic eligibility trace is to smooth the control signal over the temporal evolution of the movement, favoring solutions that minimize motor command changes.

The learning rule for each PF-PC synapse can be formalized as:

$$\Delta w(t) = -\alpha e(t)[y_{IO} - y_{IO}(0)] \tag{8.5}$$

with y_{IO} the firing rate of the climbing fiber input and $e(t)$ defined as

$$\tau \frac{de_1}{dt} = -e_1 + y_{GC} \tag{8.6}$$

$$\tau \frac{de}{dt} = -e + e_1 \tag{8.7}$$

with y_{GC} the firing rate of the parallel fiber. From Equation (8.5) it is clear that IO firing above baseline will produce LTD, while firing below baseline will produce LTP.

8.3 Results

Simulation results to show the effectiveness of the system to learn accurate trajectory control for fast movements is presented in Figure 8.3. Movements are made from a central position to eight radial targets at a rate of 0.6s per movement. As can be seen in Figure 8.3(D), accurate trajectories can be generated in a small number visits to each target.

8.4 Conclusion

We have implemented a modified version of the parallel hierarchical control model proposed by [Katayama and Kawato, 1991] and have shown that our cerebellar network was able to learn the control function. The new features of our model are

1. the use of the spinal reflex feedback circuit as a source for **training** signals to the IDM;
2. an ISM that also generates gamma-drive so that muscle-state errors can be detected at the spindles with no delay;
3. the use of synaptic eligibility to learn inverse dynamics feedforward control, and also to force smooth control actions.

Acknowledgements

This work was supported in part by the Human Brain Project (with funding from NIMH, NASA, and NIDA) under the P20 Program Project Grant HBP: 5-P20-52194 for work on "Neural Plasticity: Data and Computational Structures" (M.A. Arbib, Director). Stefan Schaal provided many helpful comments and suggestions.

Bibliography

[Atkeson and Hollerbach, 1985] Atkeson, C. and Hollerbach, J. (1985). Kinematic features of unrestrained vertical arm movements. *Journal of Neuroscience*, 5:2318–2330.

[Bizzi et al., 1984] Bizzi, E., Accornero, N., Chapple, W., and Hogan, N. (1984). Posture control and trajectory formation during arm movement. *Journal of Neuroscience*, 4:2738–2744.

[Bizzi et al., 1992] Bizzi, E., Hogan, N., Mussa-Ivaldi, F., and Giszter, S. (1992). Does the nervous system use equilibrium-point control to guide single and multiple joint movements? *Behavioral and Brain Sciences*, 15:603–613.

[Chou and Hannaford, 1996a] Chou, C.-P. and Hannaford, B. (1996a). Measurement and modeling of mckibben pneumatic artificial muscles. *IEEE Transactions on robotics and automation*, 12(1):90–102.

[Chou and Hannaford, 1996b] Chou, C.-P. and Hannaford, B. (1996b). Study of human forearm posture maintenance with a physiologically based robotic arm and spinal level neural controller. Submited to Biological Cybernetics.

[Feldman, 1986] Feldman, A. (1986). Once more on the equilibrium-point hypothesis (λ model) for motor control. *Journal of Motor Behavior*, 18:17–54.

[Gomi and Kawato, 1993] Gomi, H. and Kawato, M. (1993). Neural network control for a closed-loop system using feedback-error-learning. *Neural Networks*, 6:933–946.

[Ito, 1984] Ito, M. (1984). *The Cerebellum and Neural Control*. Raven Press, New York.

[Katayama and Kawato, 1991] Katayama, M. and Kawato, M. (1991). A parallel-hierarchical neural network model for motor control of a musculo-skeletal system. *Systems and Computers in Japan*, 22(6).

[Klopf, 1982] Klopf, H. (1982). *The Hedonistic Neuron: A theory of memory, learning and intelligence*. Hemisphere, Washington, DC.

[McIntyre and Bizzi, 1993] McIntyre, J. and Bizzi, E. (1993). Servo hypotheses for the biological-control of movement. *Journal Of Motor Behavior*, 25(3):193–202.

[Ozkaya and Nordin, 1991] Ozkaya, N. and Nordin, M. (1991). *Fundamentals of biomechanics: equilibrium, motion, and deformation*. Van Nostrand Reinhold, New York.

[Schweighofer, 1995] Schweighofer, N. (1995). *Computational models of the cerebellum in the adaptive control of movements*. PhD thesis, University of Southern California, Los Angeles, CA.

[Schweighofer et al., 1996] Schweighofer, N., Arbib, M., and Dominey, P. (1996). A model of the cerebellum in adaptive-control of saccadic gain .1. the model and its biological substrate. *Biological Cybernetics*, 75(1):19–28.

[Schweighofer et al., 1997a] Schweighofer, N., Arbib, M., and Kawato, M. (1997a). Role of the cerebellum in reaching movements: 1. Distributed inverse dynamics control. To appear in the European Journal of Neuroscience.

[Schweighofer et al., 1997b] Schweighofer, N., Spoelstra, J., Arbib, M., and Kawato, M. (1997b). Role of the cerebellum in reaching movements: 2. A neural model of the intermediate cerebellum. To appear in the European Journal of Neuroscience.

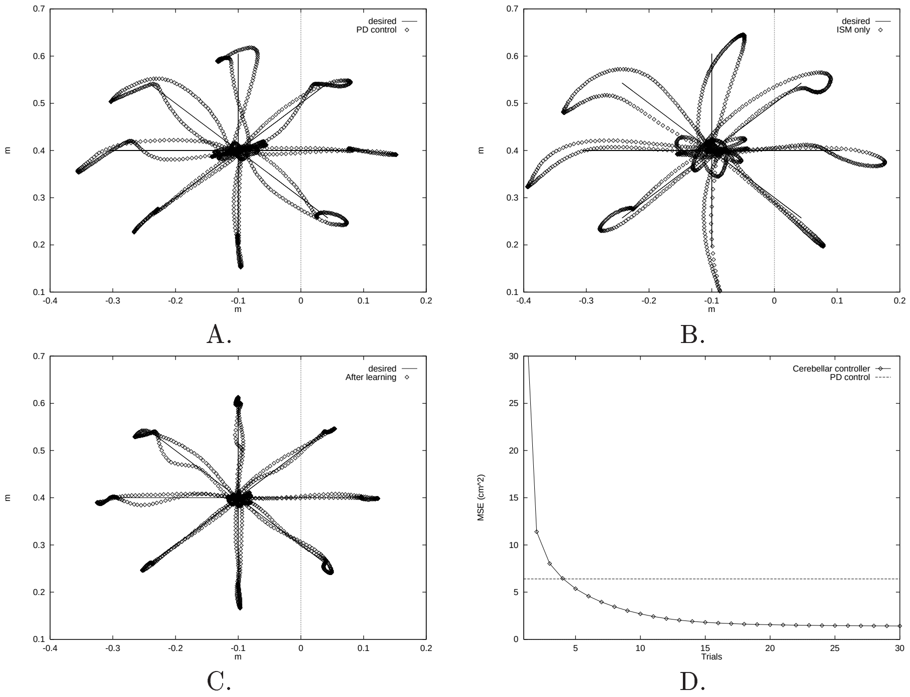

Figure 8.3: Simulation results. Movements are made from a central position to eight radial targets. Each segment of the pattern is completed in .6s. A: The best tracking performance using a proportional-derivative (PD) controller; B: Tracking performance before learning, i.e., using only postural control; C: Performance after learning; D: Tracking error (MSE in cm^2) as a function of learning trials.

[Spoelstra and Arbib, 1997] Spoelstra, J. and Arbib, M. (1997). A computational model of the role of the cerebellum in adapting to throwing while wearing wedge prism glasses. In *Proceedings of the 4th Joint Symposium on Neural Computation*, volume 7, pages 201–208, Los Angeles, CA.

[Sutton and Barto, 1981] Sutton, R. and Barto, A. (1981). Toward a modern theory of adaptive networks: Expectation and prediction. *Psychology Review*, 88:135–170.

Chapter 9

Sensorimotor integration and control

Marwan A. Jabri
Neuromorphic Engineering Group
Systems Eng. & Design Automation Lab.
Dept. of Electrical Engineering
University of Sydney
NSW 2006 Australia
marwan@sedal.usyd.edu.au

Olivier J.-M. D. Coenen
Computational Neurobiology Laboratory
and UCSD Biophys/Phys. Dept.
Howard Hughes Medical Institute
10010 North Torey Pines Rd
La Jolla, CA 92037
olivier@salk.edu

Jerry Huang
Neuromorphic Engineering Group
Systems Eng. & Design Automation Lab.
Dept. of Electrical Engineering
University of Sydney
NSW 2006 Australia
jerry@sedal.usyd.edu.au

Terrence J. Sejnowski
Computational Neurobiology Laboratory
and UCSD Biology Dept.
Howard Hughes Medical Institute
10010 North Torey Pines Rd
La Jolla, CA 92037
terry@salk.edu

Abstract

We describe in this paper models and an apparatus of sensory-motor integration in the context of a micro-robot. The models attempt to represent the interactions between the pre-frontal cortex, the basal ganglia, the ventral tegmental area and the cerebellum. We describe various experiments to perform simple tracking and obstacle avoidance tasks. The micro-robot, a Khepera, is equipped with infra-red proximity sensors and two motors, and is connected to a personal computer through a serial line. All neural computation is performed on the personal computer and the motor commands are communicated to the robot via the serial line. Learning and motor decisions are performed in real-time within a multi-threaded environment. Two sensory-motor integration models are described. The first implements a simple decision making system inspired from temporal difference learning and uses the sensory inputs and the last motor commands to produce the next motor commands. The second model augments the first by introducing a simple abstract model of a cerebellum acting as a predictor of the robot's environment. The output of the cerebellum model is used as an additional input to a network similar to the first model. Our experiments show that the inclusion of the predictions of the environment greatly improves the performance of the robot in tracking external sources. The experiments will be demonstrated at the NIPS'97 workshop on "Can Artificial Cerebellar Models Compete to Control Robots?".

9.1 Introduction

Sensory-motor integration is about combining sensory cues and motor information in order to generate motor commands, typically to maximize future sensory information and related awards. Sensory-motor integration is a corner-stone of active perception in the context of roving robots.

The development of biologically plausible models of sensory-motor integration is receiving an increasing attention from the robotics and neuromorphic engineering communities. Researchers have investigated functional sensory-motor integration models involving roles for the pre-frontal cortex (PFC), the basal ganglia (BG), the ventral tegmental area (VTA), and more recently the cerebellum. It has been hypothesized [16, 7, 43] that the pre-frontal cortex provides some working motor memory, the basal ganglia provides decision making and that VTA is involved in reward generation.

We describe in this paper neural computation architectures for sensory-motor integration for simple obstacle avoidance and source tracking. The architectures are based upon simplified abstract models reported in [16, 7, 43, 45, 6]. We have tested our models using a micro-robot (Khepera) and our experiments show that learning is achieved efficiently and in real-time on an average desktop personal computer. Two sensory-motor integration models are described. The first implements a simple decision making system inspired from Temporal Difference (TD) learning and uses the sensory inputs and the last motor commands to produce the next motor commands. The second model augments the first by introducing a simple abstract model of a cerebellum acting as a predictor of the robot's environment (sensory input). The output of the cerebellum model is used as an additional

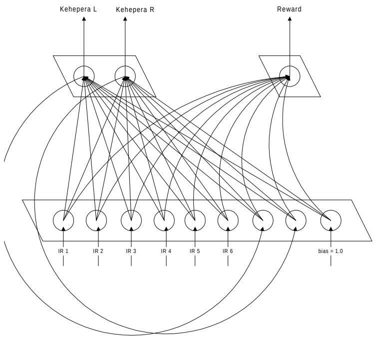

Figure 9.1: Simple sensory-motor integration architecture.

input to network similar to the first model. Our experiments show that the inclusion of the predictions of the environment greatly improves the performance of the robot in tracking external sources.

The paper is structured as follows. In Section 9.2 we describe the simple sensory-integration model that attempts to model the interactions between the pre-frontal cortex (PFC), basal ganglia (BG) and the ventral tegmental area. This model is tested in obstacle avoidance and source tracking tasks. We then describe the motivations in augmenting this model with a cerebellum-based prediction model. Section 9.3 reviews biological background on the role of the cerebellum in motor learning and presents the sensory-motor architecture augmented with an abstract model of a cerebellum.

9.2 A simple sensory-motor interaction model

9.2.1 Architecture

The simple model we describe here is based the Sutton-Barto TD learning approach [45] and the Dayan-Sejnowski predictive Hebbian learning architecture [33]. The architecture consists (see Figure 9.1) of two single layer networks, the first behaving as an "action" network and the second acts as predictor of the expected future reward. The inputs to both networks are the micro-robot sensory inputs (6 infra-red proximity sensors at the front side of the robot) and the last values of the left and right motor commands. The action network has two outputs producing the left and right motor commands by applying a *sign* function and then multiplying the results by a value corresponding to a constant speed.

9.2.2 Obstacle avoidance

The basic operation of the networks and its learning rule are depicted in Figure 9.2 for the object avoidance case. Note in this case the reward and learning are implemented as follows:

1. a *clearance* measure is computed according to the infra-red proximity sensors. Only sensor values above a threshold are used (set to 60 out of 1023 for our robot). The total sensory inputs are accumulated and then mapped to a [0,1] range using a tanh() function. The clearance is one minus this value.
2. if the clearance value is zero, no obstacle in proximity is present and the motor speed are set to maximum forward (2 in out case). We restart from 1.
3. the robot is approaching an obstacle. The learning is triggered and the reward is to achieve total clearance again. The expected reward prediction error δ is computed [43] according to

   $$\delta(t) = \text{clearance}(t) + \gamma V(t+1) - V(t) \tag{9.1}$$

   When learning, δ is then used to update the network according to

   $$e_{ij} = (1-\lambda)e_{ij} + \lambda \text{in}_j \text{out}_i \tag{9.2}$$

   $$w_{ij} = w_{ij} + \eta\delta e_{ij} \tag{9.3}$$

   Where e_{ij} represents the *eligibility* of weight $w[i][j]$ (from input j to output i).

Therefore, in the case of a simple obstacle avoidance task, the reward is the clearance. While the robot is in the clear no learning is performed and the robot will continue its straight-a-head movement.

The networks above have been implemented in C++ and run on a PC communicating with the micro-robot via a serial line. In the case of obstacle avoidance the obstacles were the walls of the box surrounding the robots, or objects deposited inside the box. The system works very well with the network learning to turn from the first encounter with a wall or any other object it encounters. The experiments will be demonstrated at the workshop.

9.2.3 Homing & Tracking Moving Objects

In the case of homing and tracking of moving objects a programmable infra-red emitting hardware was developed. It consists of infra-red light emitting diodes (LEDs) organised in a horizontal array. When the LEDs are activated, the robot's proximity sensors will detect the radiation and will confuse it with its own radiation that it uses to detect proximity to objects. LEDs with 8 degrees radiation angle were used in our setup. Each LED in the array can be dynamically switched on or off by software. By switching the appropriate sequence of LEDs, we can simulate objects (to a degree of discretisation that depends on the robot sensor reading speed) moving at various speeds.

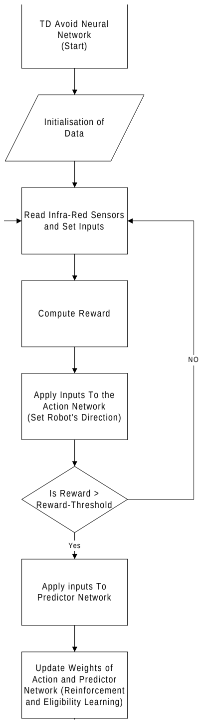

Figure 9.2: Basic operation of the sensory-motor network of Figure 9.1 in the case of a simple obstacle avoidance task.

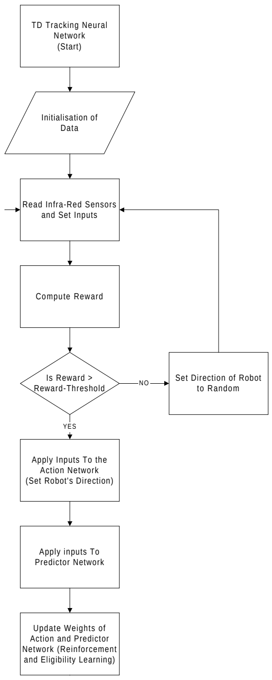

Figure 9.3: Basic operation of the sensory-motor network of Figure 9.1 in the case of simple source tracking.

This hardware setup was used in our homing and tracking experiments. For homing, at least one LED is activated and the robot positioned at various angles. In the tracking experiments, the LEDs are switched on/off in groups of 2 or more. The adjacent LEDs are activated as to keep 1 LED activated in 2 consecutive groups. The speed of LED group switching is controlled by software and can be at least of 20ms.

In the homing and tracking tasks, the first experiments we describe here use a network architecture identical to that of Figure 9.1. The operation procedure is shown in the flowchart of Figure 9.3 and the definition and computation of the reward is done as follows:

1. a reward measure based on the proximity sensors is computed (threshold of 60 out of 1023 is used). The tanh() of the total sensor values is then taken as the reward.
2. if the reward is above a threshold (0.0 in our experiments) the output of the "action" network is used to drive the robot left and right motors, and learning is activated. The predicted reward error δ is computed using
$$\delta(t) = \text{reward}(t) + \gamma V(t+1) - V(t) \tag{9.4}$$
3. otherwise a random move is performed using a uniform random number generator.

The weights of the network are updated using Equation 9.3.

In the homing experiments, the system starts by "searching" and seeking a reward, doing random moves when the present reward is below a threshold. After few turns searching for the infra-red source, the robot homes on it. In the tracking experiments, the system will behave similarly but will continue tracking the "moving" LEDs until it homes on a LED which effectively ends the trial.

Again, the experiments will be shown at the workshop.

9.3 Additing an abstract model of a cerebellum

The cerebellum is involved in motor timing [18], motor coordination [15], motor learning [32, 12] and sensorimotor integration [44]. Cerebellar contributions have been inferred in situations as diverse as timing of the conditioned eyelid response [41], shifting of attention [1, 10], adaptation of the vestibulo-ocular reflex [27] and coordination of eye and hand motor systems [48]. Some of these studies also suggest that the cerebellum may be involved in cognitive aspects of information processing [4, 23, 25, 31, 46, 42]. Several theories of cerebellar function have been proposed, including the original motor learning theories of Marr (1969), Albus (1971) and others [2, 8, 9, 11, 14, 17, 21, 20, 24, 28, 30, 36, 37, 38, 39, 47]. Nevertheless, few of these theories have provided a consistent view of the cerebellum's role in these diverse tasks. The possibility that the common thread across these different tasks is a predictive ability that the cerebellum brings to the central nervous system is the focus of our investigation. The predictive functionality of the cerebellum may include prediction of neural signals carrying sensory, motor, and even cognitive information. The predictive functionality may go beyond direct prediction and encompass predictive control in motor tasks. For example, when sensory and motor signals are combined in the cerebellum under the control of motor errors, the outcome may be the implementation of a predictive controller. The difference with previous predictive models of the cerebellum is in the details of the implementation.

9.3.1 Anatomy and predictive functionality

The basic anatomic and idea of the approach is shown in fig. 9.4. The deep cerebellar nuclei and the cerebellar cortex form together in our framework a predictive machine that is under the regulatory control of the inferior olive. The predictions being constructed are predictions of neural activities related to the excitatory inputs reaching the in-

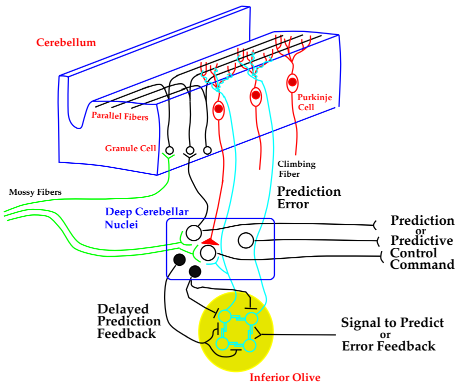

Figure 9.4: Simplified cerebellar anatomy and diagram of the hypothesis on cerebellar functions.
Anatomy: Inputs to the cerebellum enter either as mossy fibers contacting granule cells or as climbing fibers coming from the inferior olive. The mossy fibers contact granule cells and neurons in the cerebellar nuclei. One mossy fiber contacts approximately 400 to 1000 granule cells in one folia of the cerebellum. One mossy fiber can also run in other folia. Each granule cells receive inputs from 1 to 7 mossy fibers. Granule cell axons form the parallel fibers which contact the dendrites of Purkinje cells. One Purkinje cell may have from 100 000 to 400 000 parallel fibers passing through its dendritic tree, but only 20% (80 000) may have synaptic contacts – these are estimates calculated from different assumptions on synaptic contacts. The length of a parallel fiber ranges from 0.6 mm in the mouse to 2.6 mm in man; it passes through the dendritic tree of about 225 Purkinje cells, and synapses presumably onto 45 only. The climbing fiber has multiple contacts with Purkinje cell dendrites. Each Purkinje cell receive only one climbing fiber and one climbing fiber contact approximately ten Purkinje cells. The climbing fiber also sends collaterals to the deep cerebellar nuclei which receive projections from Purkinje cells contacted by the climbing fiber. Parallel fibers drive Purkinje cells to fire simple spikes at a rate of 20 to 100 Hz, generating modulated inhibition at the deep cerebellar nuclei. Climbing-fiber activity produces complex spikes in Purkinje cells; these short bursts (10 ms) of about five spikes strongly inhibits the deep cerebellar neurons. All connections are excitatory except for the Purkinje cells projections which are inhibitory. The inferior olivary neurons are electrotonically coupled, this is indicated by junctional contacts between short dendrites. They received two types of inhibitory gabaergic inputs from the deep cerebellar nuclei, one type terminates at the dendro-dendritic gap junctions, and another type terminates at the perikaryon (cell body) [34, 35, 3, 29]. **Hypothesis:** Predictions or predictive motor commands are encoded in the deep cerebellar nuclei with contributions from the Purkinje cell inputs they receive from cerebellar cortex. The actual representation may depend on whether the inferior olive receives direct inhibitory input feedback from the deep cerebellar nuclei or indirect inhibitory inputs from other nuclei. The gabaergic fibers terminating at the perikaryon are used to calculate the prediction error. The fibers terminating at the gap junctions may be used to modulate the number of olivary neurons firing in synchrony, and therefore the number of climbing fibers reporting a particular prediction error. This consequently may change the number of Purkinje cells involved in a particular prediction.

ferior olive. Depending on the source of inhibitory inputs to the inferior olive, the output of the cerebellum are predictions or predictive control commands. The inhibitory inputs from the deep cerebellar nuclei to the inferior olive may carry a delayed feedback of the predictions being established in the cerebellar nuclei. The internal circuitry of the cerebellar nuclei and feedback connections with other nuclei may provide the time delays required for the hypothesis [30].

9.3.2 Prediction in the cerebellum: Biological evidence

Experimental results have shown that a predictive representation is constructed in the cerebellum [13]. Monkeys were trained to grasp, lift and hold an object between thumb and index finger for one second. On selected blocks of trials, a perturbation was applied to the held object to simulate object slip. Purkinje cells located in the hand representation area lateral to the vermis in lobules IV-VI acquired an anticipatory response to the perturbation in order to stabilize the position of the hand. They also showed a gradual extinction of the anticipatory response after the perturbation was withdrawn. The contribution of the cerebellum in this task is similar to its role in eyelid conditioning. In eyelid conditioning, the conditioned response timing is apparently controlled by the anterior lobe of the cerebellum and occurs before the unconditioned stimulus [41, 40]. Thus, one may infer that the cerebellum participates in constructing a predictive response from the conditioned stimuli. Purkinje cell responses in the flocculus and ventral paraflocculus are modulated during smooth pursuit eye movements [44]. Predictive control of smooth pursuit eye movements occurs for complex two-dimensional tracking trajectories in monkey [26] and human [5]. The cerebellum is therefore a possible site for the predictive learning of smooth eye movements, and it has been modeled with this perspective [22].

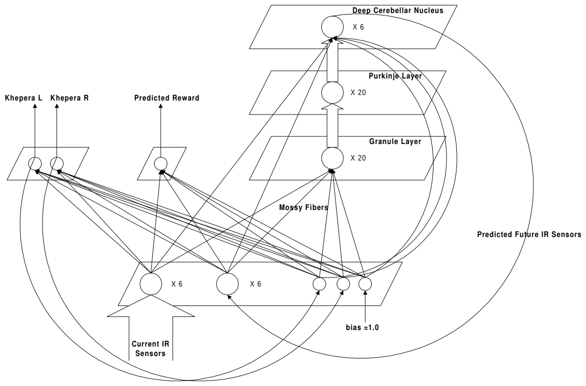

Figure 9.5: Augmented sensory-motor integration architecture including an abstract model of a cerebellum.

The structure of the cerebellum, and the great convergence of mossy fiber inputs projecting from all areas of the brain and senses, make the cerebellum a very suitable brain area to construct short-term predictions and predictive control commands which can be further processed by the central nervous system. The cerebellum is the only place in the brain where cells receive such a massive number of inputs, and with such a diverse input representation: there are more granule cells than all other cells in the brain combined. The input representation, presumable nonlinear, is extremely rich since one mossy fiber contacts between 400 to 1000 granule cells, and a granule cell receives from 4 to 7 mossy fibers. An extremely large number of different input combinations, presumably nonlinear, are then available to the Purkinje cells. Such diversity would be expected in order

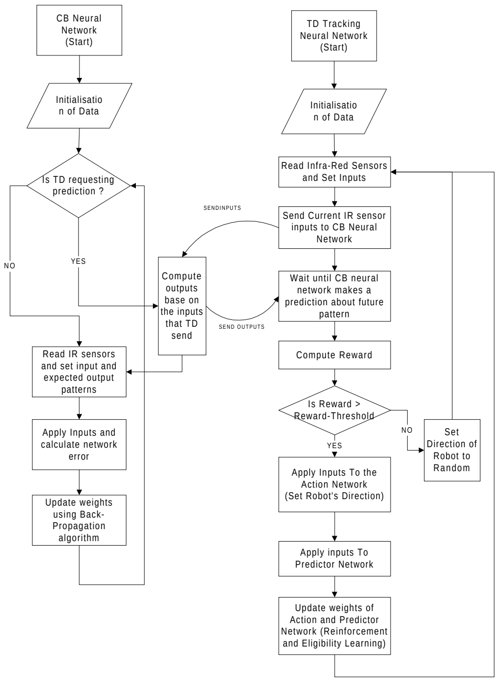

Figure 9.6: Basic operation of the sensory-motor network of Figure 9.5 in the case of simple obstacle avoidance.

to construct predictions, and predictive controllers from a changing environment where causal relationship between events may be changing and may depend on context. As an example, large representation at the inputs is required for building inverse-dynamics model for motor control [19]. The combination of at least 26 precise nonlinear terms made of nine state variables is necessary to construct the inverse-dynamics model of a three degrees of freedom robot manipulator. By analogy, the rich representation at the input of the cerebellum may provide the first stage to construct such nonlinear terms.

9.3.3 Predictive Model Architecture

The architecture of the model augmented with a functional model of the cerebellum is shown in Figure 9.5. In this architecture, the cerebellum is predicting the sensory environment at some Δt from the present. The operation of the augmented model is shown by the flowchart of Figure 9.6. The length Δt is determined by several factor including the speed of the PC, the speed of the robot, and the neuron unit delays in our model of the cerebellum. In the experiments described below Δt is not fixed and varies between 10-20ms.

The experiments we have conducted consisted of tracking a fixed and moving external infra-red source. Our aim was to investigate:

- the prediction of the robot sensory environment using the cerebellar model.
- the impacts of providing the sensory prediction as an additional input to the sensory-motor integration model described in Section 9.2.

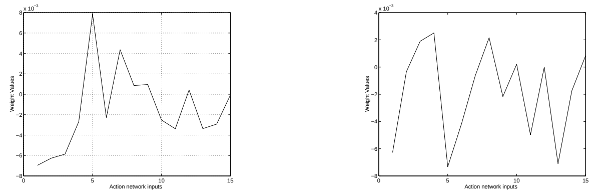

Figure 9.7: Weights of left and right motor outputs of the augmented sensory-motor integration network. The inputs are current infra-red sensor (1-6), predicted infra-red sensors by the cerebellum (7-12), last left and right outputs, and the bias. For groups 1-6 and 7-12, the smaller index represents the rightmost sensory input.

The architecture was operated as follows (see flowchart in Figure 9.6):

The original sensory-motor integration and the cerebellum model were operated in independent threads with the Win32 multithreaded operating system of the PC. The SMN and the CBN both start learning from random states. When the SMN requires the CBN outputs it sends a message to its thread message queue and wait for a reply with the prediction. It then uses this prediction to proceed with its operations. The delays in the messaging system are about 20 ms on average.

The experiments we have performed (which will be shown at the workshop) shows an improved performance in the tracking. Not only that the robot movements becomes smoother but it also homes at the target faster.

We show in Figure 9.7 the weights of the sensory-motor integration network (action network) in the tracking case. Note the mapping performed by the left and right weights for the current sensory inputs (inputs 1-6 in the plots) is straight forward: The weights are the largest for the left sensory inputs.

9.4 Conclusions

We have described in this paper architectures for sensory-motor integration and their applications to simple obstacle avoidance and tracking by a micro-robot. We have shown that predictive Hebbian learning can be used to train the robot in real-time. We have also described an augmented sensory-motor integration architecture that includes a model of the cerebellum. Our experiments show that the robot movements become smoother and it homes on targets faster.

Bibliography

[1] N. A. Akshoomoff and E. Courchesne. A new role for the cerebellum in cognitive operations. *Behavioral Neuroscience*, 106(5):731–738, 1992.

[2] NA Akshoomoff, E Courchesne, and J Townsend. Attention coordination and anticipatory control. *Int Rev Neurobiol*, 41:575–598, 1997.

[3] J. S. Albus. A theory of cerebellar function. *Math. Biosci.*, 10:25–61, 1971.

[4] Nancy C. Andreasen, Daniel S. O'Leary, Stephan Arndt, Ted Cizadlo, Richard Hurtig, Karim Rezai, G. Leonard Watkins, Laura L. Boles Ponto, and Richard D. Hichwa. Short-term and long-term verbal memory: A positron emission tomography study. *Proc. Natl. Acad. Sci.*, 92:5111–5115, 1995.

[5] G.R. Barnes and P.T. Asselman. The mechanism of prediction in human smooth pursuit eye movements. *Journal of Physiology*, 439:439–461, 1991.

[6] A.G. Barto. Adaptive critics and the basal ganglia. In J.L. Davis J.C. Houk and D.G. Beiser, editors, *Models of information processing in the basal ganglia*, pages 215–232. Cambridge, MIT Press, 1995.

[7] G.S. Berns and T.J. Sejnowski. How the basal ganglia make decisions. In A.R. Damasio et al, editor, *Neurobiology of Decision Making*. Springer-Verlag, Berlin Heidelberg, 1996.

[8] J.R. Bloedel. Functional heterogeneity with structural homogeneity: How does the cerebellum operate? *Behavioural and Brain Science*, 15:666–78, 1992.

[9] F. Chapeau-Blondeau and G. Chauvet. A neural network model of the cerebellar cortex performing dynamic associations. *Biological Cybernetics*, 65:267–279, 1991.

[10] Maurizio Corbetta, Francis M. Miezin, Gordon L. Shulman, and Steven E. Petersen. A PET study of visuospatial attention. *Journal of Neuroscience*, 13(3):1202–1226, 1993.

[11] C. Darlot. The cerebellum as a predictor of neural messages -I. The stable estimator hypothesis. *Neuroscience*, 56:617–646, 1993.

[12] Sascha du Lac, Jennifer L. Raymond, Terrence J. Sejnowski, and Stephen G. Lisberger. Learning and memory in the vestibulo-ocular reflex. *Annual Review of Neuroscience*, 18:409–41, 1995.

[13] Claude Dugas and Allan M. Smith. Responses of cerebellar Purkinje cells to slip of a hand-held object. *Journal of Neuroscience*, 67:483–495, 1992.

[14] M. Fujita. Adaptive filter model of the cerebellum. *Biological Cybernetics*, 45:195–206, 1982.

[15] H.P. Goodkin, J.G. Keating, T.A. Martin, and W.T. Thach. Preserved simple and impaired compound movement after infraction in the territory of the superior cerebellar artery. *The Canadian Journal of Neurological Sciences*, 20(Suppl. 3):S93–S104, 1993.

[16] J.C. Houk, J.L. Adams, and A.G. Barto. A model of how the basal ganglia generate and use neural signals that predict reinforcement. In J.L. Davis J.C. Houk and D.G. Beiser, editors, *Models of information processing in the basal ganglia*, pages 249–270. Cambridge, MIT Press, 1995.

[17] M. Ito. *The Cerebellum and Neural Control*. Raven Press, New York, 1984.

[18] Richard B. Ivry and Steven W. Keele. Timing functions of the cerebellum. *Journal of Cognitive Neuroscience*, 1(2):136–152, 1989.

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

[20] M. Kawato and H. Gomi. The cerebellum and VOR/OKR learning models. *Trends in Neuroscience*, 15:445–453, 1992.

[21] James D. Keeler. A dynamical system view of cerebellar function. *Physica D*, 42:396–410, 1990.

[22] R.E. Kettner, S. Mahamud, H.-C. Leung, A.G. Barto, J.C. Houk, and B.W. Peterson. Predictive smooth pursuit eye movements are generated by a cerebellar model. In *25th Annual Meeting Society for Neuroscience Abstracts*, volume 21. Society for Neuroscience, 1995.

[23] S.-G. Kim, Uğurbil, and P.L. Strick. Activation of a cerebellar output nucleus during cognitive processing. *Science*, 265:949–951, 1994.

[24] H.C. Leiner, A.L. Leiner, and R.S. Dow. Reappraising the cerebellum: what does the hindbrain contribute to the forebrain? *Behavioral Neuroscience*, 103(5):998–1008, 1989.

[25] H.C. Leiner, A.L. Leiner, and R.S. Dow. Cognitive and language functions of the human cerebellum. *Trends in Neurosciences*, 16(11):444–447, Nov 1993. Debate over the evidence for non-motor function of the cerebellum.

[26] H.-C. Leung and R.E. Kettner. Predictive control of smooth pursuit eye movements along complex two-dimensional trajectories in monkey. In *25th Annual Meeting Society for Neuroscience Abstracts*, volume 21. Society for Neuroscience, 1995.

[27] S. G. Lisberger, T. A. Pavelko, H. M. Brontë-Stewart, and L. S. Stone. Neural basis for motor learning in the vestibuloocular reflex of primates. II. Changes in the responses of horizontal gaze velocity Purkinje cells in the cerebellar flocculus and ventral paraflocculus. *Journal of Neurophysiology*, 72(2):954–973, August 1994.

[28] Rodolfo Llinas and John P. Welsh. On the cerebellum and motor learning. *Current Opinion in Neurobiology*, 3:958–965, 1993.

[29] D. Marr. A theory of cerebellar cortex. *J. Physiol.*, 202:437–470, 1969.

[30] R. C. Miall, D. J. Weir, D. M. Wolpert, and J. F. Stein. Is the cerebellum a smith predictor ? *Journal of Motor Behavior*, 25(3):203–216, 1993.

[31] Frank A. Middleton and Peter L. Strick. Anatomical evidence for cerebellar and basal ganglia involvement in higher cognitive function. *Science*, 266:458–461, 1994.

[32] Susan E. Molchan, Trey Sunderland, A. R. McIntosh, Peter Herscovitch, and Bernard G. Schreurs. A functional anatomical study of associative learning in humans. *Proc. Natl. Acad. Sci.*, 91:8122–8126, 1994.

[33] P.Read Montague, Peter Dayan, and T.J. Sejnowski. Foraging in an uncertain environment using predictive Hebbian learning. In J. Cowan and J. Alespector, editors, *NIPS'6*, pages 598–605. Morgan Kaufmann, 1994.

[34] R. Nieuwenhuys, J. Voogd, and Chr. van Huijzen. *The human central nervous system*. Springer-Verlag, 1988.

[35] S. L. Palay and V. Chan-Palay. *Cerebellar Cortex, Cytology and Organization*. Springer-Verlag, 1974.

[36] Michael G. Paulin. A Kalman filter theory of the cerebellum. In Michael A. Arbib and Schun-ichi Amari, editors, *Dynamic Interactions in Neural Networks: Models and Data*, chapter III, pages 239–259. Springer-Verlag, New York, 1989. Research Notes in Neural Computing, Vol. 1.

[37] A. Pellionisz. Brain modeling by tensor network theory and computer simulation. the cerebellum: distributed processor for predictive coordination. *Neuroscience*, 4(3):323–348, 1979.

[38] A. Pellionisz. Tensorial approach to the geometry of brain function: cerebellar coordination via a metric tensor. *Neuroscience*, 5(7):1125–1138, 1980.

[39] A. Pellionisz. Space-time representation in the brain. the cerebellum as a predictive space-time metric tensor. *Neuroscience*, 7(12):2949–2970, 1982.

[40] Stephen P. Perrett and Michael D. Mauk. Extinction of conditioned eyelid responses requires the anterior lobe of cerebellar cortex. *Journal of Neuroscience*, 15:2074–2080, 1995.

[41] Stephen P. Perrett and Blenda P. Ruiz. Cerebellar cortex lesions disrupt learning-dependent timing of conditioned eyelid responses. *Journal of Neuroscience*, 13:1708–1718, 1993.

[42] Marcus E. Raichle, Julie A. Fiez, Tom O. Videen, Ann-Mary K. Macleod, Jose V. Pardo, Peter T. Fox, and Steven E. Petersen. Practice-related changes in human brain functional anatomy during nonmotor learning. *Cerebral Cortex*, 4:8–26, 1994.

[43] W. Shultz, P. Dayan, and P.R. Montague. A neural substrate of prediction and reward. *Science*, 275:1593–1599, 1997.

[44] L. S. Stone and S.G. Lisberger. Visual response of Purkinje cells in the cerebellar flocculus during smooth-pursuit eye movements in monkeys. I. simple spikes. *Journal of Neurophysiology*, 63:1241–1261, 1990.

[45] R.S. Suttin and A.G. Barto. Towards a modern theory of adaptive networks: Expectation and prediction. *Psychological Review*, 88 2:135–170, 1981.

[46] Hisao Tachibana. Event-related potentials in patients with cerebellar degeneration: electrophysiological evidence for cognitive impairment. *Cognitive Brain Research*, 2:173–180, 1995.

[47] W. T. Thach, H. P. Goodkin, and J. G. Keating. The cerebellum and the adaptive coordination of movement. *Annual Review of Neuroscience*, 15:403–442, 1992.

[48] Paul van Donkelaar and Robert G. Lee. Interactions between the eye and hand motor systems: Disruptions due to cerebellar dysfunction. *Journal of Neurophysiology*, 72:1674–1685, 1994.