Connection Science **10**, pp. 301–320, 1998

Cerebellar control of robot arms

Patrick van der Smagt
Institute of Robotics and System Dynamics
German Aerospace Center / DLR Oberpfaffenhofen
P.O. Box 1116, 82230 Wessling, Germany
email `smagt@dlr.de`

December 11, 1998

Keywords: cerebellum, robot dynamics, robot arm control, computational cerebellar models, neural networks

Abstract

Decades of research into the structure and function of the cerebellum have led to a clear understanding of many of its cells, as well as how learning takes place. Furthermore, there are many theories on what signals the cerebellum operates on, and how it works in concert with other parts of the nervous system.

Nevertheless, the application of computational cerebellar models to the control of robot dynamics remains in its infant state. To date, a few applications have been realized, yet limited to the control of traditional robot structures which, strictly speaking, do not require adaptive control for the tasks that are performed since their dynamic structures are relatively simple. The currently emerging family of light-weight robots (Hirzinger, 1996) poses a new challenge to robot control: due to their complex dynamics traditional methods, depending on a full analysis of the dynamics of the system, are no longer applicable since the joints influence each other dynamics during movement. Can artificial cerebellar models compete here?

In this overview paper we present a succinct introduction of the cerebellum, and discuss where it could be applied to tackle problems in robotics. Without conclusively answering the above question, an overview of several applications of cerebellar models to robot control is given.

1 Introduction

Recent successes in robotics have increased the field of application and acceptance of robots. Nevertheless, robotics still has a long way to go. The applicability of industrial robots remains limited to factory floors. Research lab robotics is increasingly moving towards novel actuators for constructing light-weight, compliant robot arms. For this we require actuators consisting of agonist-antagonist drive pairs or miniature motors with high-ratio gear boxes for maintaining accurate positioning without recalibration, as well as for controlling the stiffness of a joint. However, when multiple of such compliant joints are used to construct a robot arm, known methods can only control the robot arm via low-speed, inaccurate control using (approximated) inverse Jacobians of the robot arm. To date, no generally applicable methods exist which can control the dynamics of any robot arm to follow a given trajectory in joint space. Another unsolved problem is calibration and re-calibration; due to wear and tear, model assumptions of a robot may become increasingly imprecise during operation, leading to imprecise and unstable control when no adaptive methods are employed.

The clear success that the cerebellum has achieved in controlling biological skeletomuscular structures[1] is in no way matched by artificial neural or other adaptive control algorithms, since, to date, the complexity of the mapping that has to be computed seems to be prohibitive. One outstanding problem is that, when controlling the trajectory of a joint, at least 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 at least $3n$ variables (where $n \geq 6$ is the number of degrees-of-freedom of the robot arm). With highly compliant robot arms, the relationship between the $3n$ variables is very complex and highly nonlinear, and therefore very hard to learn. Furthermore, time delays in the control loop introduce additional complexity. The approach of using recurrent networks which are structurally capable of computing internal representations of $\dot{\theta}$ and $\ddot{\theta}$ has not yet been successful in solving more than toy problems.

Starting from Albus' model (1975), an adapted version of which has been successfully used by, e.g., Miller et al. (1989, 1997) for control of a 5 DoF robot arm as well as biped control, or Lang (1997) for 6 DoF robot arm control, neuro-computational models of the cerebellum have been advocated as possible candidates for control of the dynamics of complex robot systems. However, apart from these implementations there have been very few applications of cerebellar models to the control of real robot manipulators in realistic tasks using real robots. On the one hand, cerebellar models have become more refined through specialized investigations based on details of the biological cerebellar system, while insufficient attention has been given to applicability of these refinements in robotics. On the other hand, to keep the increasingly complex robot structures under control, robotics has moved towards the use of established methodologies which rely on as much model knowledge as possible, thus losing flexibility in applications.

It seems to be a good time to use cerebellar modeling as a test case for investigating the prospects for an amalgamation of the fields of robotics and neuro-computational modeling, and raise the question of how neuro-computational models might be incorporated as a standard part of robotics methodologies. How reasonable is the desire for such an amalgamation? What prerequisites are there for having cerebellar models successfully compete with alternative approaches to control of real robots?

This paper follows the 1997 NIPS workshop 'Can Artificial Cerebellar Models Compete to Control Robots?', initiated by Patrick van der Smagt and Daniel Bullock (1997). Section 2 describes the anatomy of the cerebellum. In section 3 the robotics side is investigated: where can artificial cerebellar models help us in robot control? A number of computational cerebellar models are introduced in section 4, starting off with the well-known CMAC model, followed by recent developments in cerebellar theory, and discussing where these developments can be used in robot control. A discussion is given in section 5.

2 The function of the cerebellum

The controversies that were touched upon at the NIPS workshop clearly reflected the current-day understanding we have of various parts of the cerebellum: many conflicting

---

[1] It must be noted, however, that the dynamic properties of this skeletomuscular structure are highly superior to that of current-day robots.

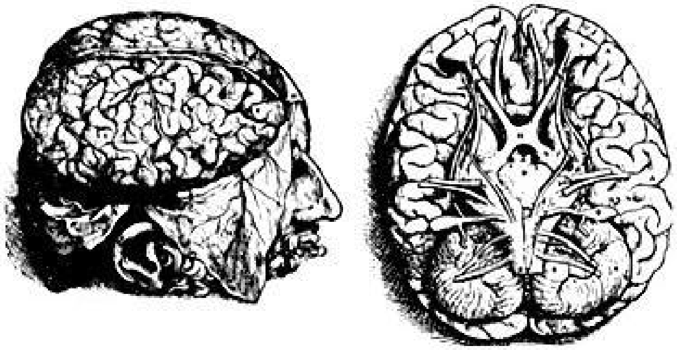

Figure 1: Vesalius' drawings of the brain.

theories exist, none of which have been verified beyond doubt. And although, as we will show in this section, detailed knowledge of the structure of the cerebellum is available, there is no definitive understanding of how the cerebellum as a whole interacts with the cerebrum and the brain stem. Or, as De Schutter (1996) put it, the function of the cerebellum remains a mystery.

2.1 History

Investigation of the cerebellum started as early as in the ancient times, as the Greek surgeon Galen of Pergamum (129–216 / 217 AD), working as a gladiator physician in Pergamum (currently Izmir) and Rome, suspected that the cerebellum is the part of the brain responsible for muscle control. Although the realization that the brain consists of communicating cells was postulated as early as the fourth century AD by Nemesius of Emesa (building on Galen's work in his compendium *Peri physeos anthropou*, Greek: On the Nature of Man) and Saint Augustine (354–430 AD), detailed insight into the brain was not available until experimental science became accepted practice. An important forerunner in this aspect was Andreas Vesalius of Brussels (1514–1564), who was the first person to critically investigate Galenic theory. Through his dissections and anatomical studies he initiated modern physiological science (figure 1).

The understanding that the brain is controlled by electrical pulses emerged in the 18th century, when Benjamin Franklin published his *Experiments and Observations on Electricity* (1774). In his 1791 essay *De Viribus Electricitatis in Motu Musculari Commentarius* (Commentary on the Effect of Electricity on Muscular Motion) Luigi Galvani (1737–1798) subsequently showed that frog muscles contracted when excited with electricity, and suspected that these electric pulses were generated by the brain. This assumption was later verified by Emil du Bois-Reymond (1818–1896) around 1850, when instruments were sensitive enough to measure neuron pulses. Bell (1811) proposed that motor fibers originate from the cerebellum; an assumption that was later verified by Flourens (1824). In *The Cerebellum as a Neuronal Machine*, Eccles *et al.* (1967) described a detailed model of the cerebellum, which has been accepted since then and has served as a basis of many subsequent computational models.

2.2 The structure of the cerebellum

All motion commands that originate in the cerebrum pass through the cerebellum (figure 2). The cerebellum subsequently provides timing control of opposing muscles, and

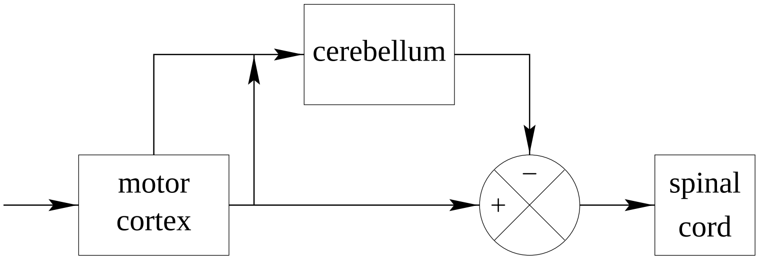

Figure 2: Motor paths in the brain.

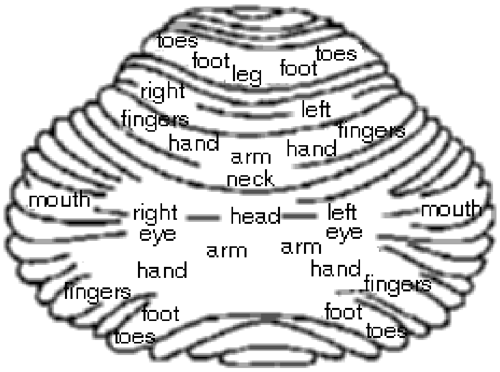

Figure 3: Microzones in the cerebellum.

force as well as stiffness control.

The cerebellar cortex is divided in three layers: the outer synaptic layer (also called molecular layer), the Purkinje layer, and the inner receptive layer (the granular layer). The cortex appears to be organized in microzones (figure 3); each of these microzones contains the parameters for a certain movement.

The human cerebellum (see figure 4) consists of about 10 million Purkinje cells (pc), each receiving about 150,000 excitatory synapses via the parallel fibers (pf). The pf are the axons of the granule cells; these cells are excited by the mossy fibers (mf) originating from the spinal cord. Each pf synapses on about 200 Purkinje cells. A Purkinje cell receives further excitatory synapses from one single climbing fiber (cf); this can fire a cell when active. Basket cells, being activated by pf afferents, can inhibit a Purkinje cell. Finally, Golgi cells receive input from pf, mf, and cf. They inhibit granule cells.

The granule cells operate as pattern separators. The densely ‘coded’ patterns, originating from the spinal cord, have to be ‘preprocessed’ by the granule cells, such that the imprecise giant Purkinje cells can discriminate them. The output of a Purkinje cell is an inhibitory signal to the cerebellar nuclei.

Mechanistic models of the cerebellum did not appear until the paper by Braitenberg and Atwood (1958). Braitenberg (1961) primarily interpreted the cerebellum as a timing organ; in his theory, the pf function as delay lines.

Influenced by Eccles *et al.* (1967), two other early models by Marr (1969) and Albus (1971) view the cerebellum as a learning pattern recognition system. Their more detailed models, as well as a subsequent computational model by Albus (1975), have contributed to a wide acceptance of the pattern recognition theory.

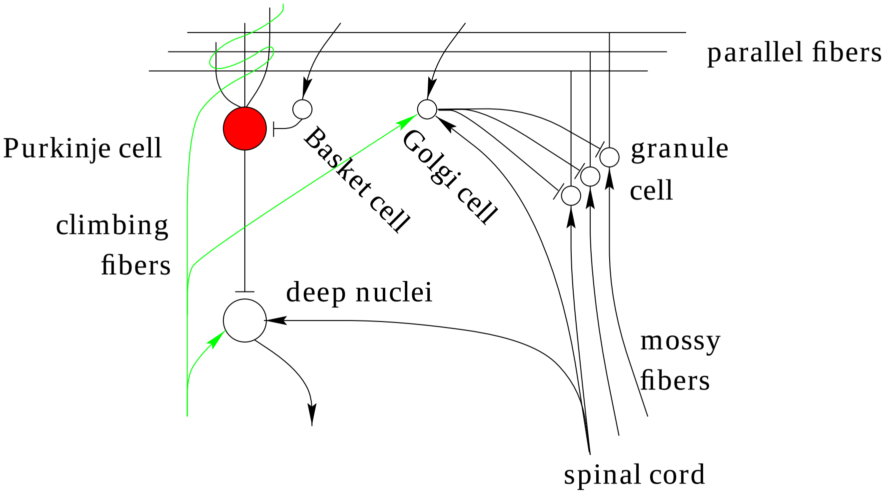

Figure 4: Structure of the cerebellum. An arrow indicates an excitatory connection, a termination (⊣) an inhibitory connection.

2.3 Learning

The major contribution of the papers by Braitenberg, Marr, and Albus was an encompassing theory of how learning in the cerebellum takes place. We know that the cerebellum learns sequences of voluntary movements as well as motor programs[2], but also adapt to external influences. Fortunately, the regular structure of the cerebellum has aided in a good insight in how learning takes place. The key issue is that learning is *context-driven*. In Marr's theory, pf synapses on pc are strengthened when simultaneous activity of the cf occurs. Thus memory traces are stored at the pf synapses. Albus later suggested that the cerebellum functions as an adaptive pattern classifier, where the pf synapses become weaker; the cf signal is thus interpreted as an error signal. This hypothesis is known as the Marr-Albus theory.

This adaptation of the pf synapses is called *long-term depression* (LTD). Next time around that the same pattern of pf appears, the Purkinje cell will *not* fire and thus *not* inhibit its corresponding deep nucleus. Verification of LTD at the pf synapses followed in the early eighties by Ito *et al.* (1982), who demonstrated that, in rabbit cerebellum, simultaneous stimulation of pf and cf result in LTD of the pf/pc synapses. Learning through LTD occurs in many places of the brain, and is usually accompanied by the opposite process called *long-term potentiation* (LTP) (Bliss & Collingridge, 1993).

Although Artola and Singer (1993) concluded that LTD learning in the cerebellum distinguishes itself through the absence of LTP, this conclusion is not generally accepted (Kano, 1996). It be noted that LTD cannot be the only learning mechanism for the pc/pf synapses, since otherwise they would eventually be driven to zero responsiveness.

Later investigations resulted in a more complex picture. When lacking a metabotropic glutamate receptor subtype, LTD is absent in mice. It was recently shown (Aiba et al., 1994) that such mice, when trained on the eye blink reflex[3], learn as normal animals in

---

[2] A motor program is a sequence of muscle commands that can be executed without feedback, using the correct timing, e.g., speech, cycling, etc.

[3] This experiment works as follows: a test 'person' (in this case, a mouse) is given a puff of air in the eye, before which a single tone is played. When the delay between the playing of the tone and the airpuff is constant, the test person learns to close the eyelid on hearing the tone, before the airpuff is administered.

their first three days of training. During the following days, however, the mice learn significantly less than normal mice. This effect cannot be explained using the Marr-Albus theory.

Another controversy concerns the vestibulo-ocular reflex (VOR). The function of this reflex is to stabilize an image on the retina while the head is subject to rotational or translational movements. Lesions in the flocculus, the oldest part of the cerebellum, prevent learning of the VOR, and Ito (1984) proposed the flocculus as an example of the Marr-Albus theory. Later experiments, however (Lisberger, Pavelko, & Broussard, 1994; Lisberger, Pavelko, Bronte-Stewart, & Stone, 1994; Lisberger, 1994) show that the same Purkinje cell can change its activity in the same as well as in the opposed direction of the VOR gain. Again, this phenomenon cannot be explained with the Marr-Albus theory.

Thirdly, there are conflicting reports on whether the cf signal is an error signal; other properties of the cf signals are also not explained by the Marr-Albus theory (De Schutter, 1997). Even the function of the cf signal is a mystery (Simpson, Wylie, & De Zeeuw, 1996): while some researchers assume that the cf signal leads to LTD at the pc / pf synapses, other conclude that it leads to short-term enhanced responsiveness at the pc / mf synapses. Also, the cf has been proposed to serve as internal timing signal.

Related to this problem is the *credit assignment problem* (Houk, Buckingham, & Barto, 1996). When an external error signal is present, how is this distributed to the responsible nodes (Purkinje cells)?

2.4 Timing function of the cerebellum

One of the important omissions in the pattern matching models is the timing behaviour of the cerebellum (Bullock, 1997). For instance, cerebellar lesions result in a loss of precise timing control over motor control actions, as well as in the activities of neurons in the frontal cortex (Hore & Flament, 1988). Timing in muscle control is important, since muscles always come in pairs or more complex groups. Furthermore, the motor control system suffers from considerable time delays, which have to be taken into account in any control method that is used (see section 3.1.4).

Many cerebellar models have been augmented by feeding it with motor inputs as well as their derivatives; however, this alone does not suffice to explain the timing behaviour. A counterexample is the eye-blink reflex, which has been shown to be solved by the cerebellum. This experiment shows that the cerebellum can learn delays in the range 100ms–4s.

Bullock (1997) summarizes the following important features of the cerebellum, which should be matched by a computational cerebellar model:

- the microzones in the cerebellum are defined by shared climbing fiber projections and convergence of Purkinje cell axons on nuclear cells;
- the granule cells detect combinations of events;
- the burst rate of the climbing fibers reflects the error compensating action in the control channel for which the microzone serves as a loop;
- the cerebellum performs a parallel search through signals for indicators that can be used to reduce errors;

This reflex is known to be solved by the cerebellum.

- the cerebellum allows switching from reactive to proactive control, and attempts to minimize feedback control;
- the cerebellum can generate sequences actions that lie in the range 100ms–4s.

2.5 Is a cerebellum necessary?

There is a popular belief that patients with cerebellar agenesis eventually gain normal motor control, even when the cerebellum is totally absent. However, a careful analysis of literature (Glickstein, 1994) reveals that recorded case histories of patients with cerebellar agenesis always show retarded and deficient walking, speech, and other motor functionality. Furthermore, diminished intellectual capabilities have been observed.

On the other hand, nature also shows that adaptive motor control is possible without a cerebellum. As Nelson (1997) argues, insects solve complex motor control problems without a cerebellar structure. The insect motor control system consists of four groups: sensor afferents, spiking interneurons, nonspiking interneurons, and motor neurons. The spiking interneurons, which are topographically arranged, receive sensor information. The precise function of nonspiking interneurons is not known; they have multimodal receptive fields and react to sensor signals from different modalities in the same behavioural context. They control groups of leg motorneurons during walking, and can therefore be compared to the vertebrate cerebellum at a functional level. Nelson (1997) demonstrates this theory with an insect-like walking hexapod.

3 Problems in robot control

We consider a robot arm as a machine consisting of three separate parts. First, the arm itself is a mechanical structure consisting of a number of **links** (arm segments) which are connected by **joints**. The joints, which are usually rotational instead of sliding for various reasons, are driven by motors. The result of a robot control algorithm (be it neural-network based or using a traditional technique) often is the position and orientation $(\mathbf{x}[t], \boldsymbol{\phi}[t])$ of the end-effector with respect to the robot's base.

The second part of a robot arm consists of the electronic hardware to control and drive the robot. This part consists of one or more processors, D / A and A / D converters, power amplifiers, etc. The third part is given by the software to control the robot arm. In order to simplify the control problem, this control is usually multi-stage. First, most tasks are defined in terms of the desired position and orientation of the end-effector. In Cartesian coordinates, the desired position is specified as a trajectory $(\mathbf{x}_d[t], \boldsymbol{\phi}[t])$. Naturally, a specification in sensor space (Smagt, 1995) is also a likely possibility; this depends on the task that is performed.

The task for the robot control software is now twofold. First, a desired end-effector trajectory must be translated to a trajectory $\theta[t]$ in joint space. This transformation is called the **inverse kinematics**. When this transformation is underdetermined, additional constraints, such as the avoidance of undesired configurations of the arm to prevent collision with obstacles, but also dynamic constraints, have to be taken into account to find a unique solution (see, e.g., DeMers & Kreutz-Delgado, 1997, for a good overview of traditional and neural approaches to solve this problem).

In the second stage, the desired trajectory in joint space have to be translated to forces (torques τ or even motor commands) which are applied to the joints: **dynamic control**, also known as **joint servo control** or simply **tracking**. In this stage, physical limitations have to be taken into account.

It is this second problem that we want to discuss in this paper. From section 2 it is clear that the applicability of cerebellar models to robot control problems can be found in the improvement of adaptive dynamic control methods. But where are such adaptive methods required? In this section we will discuss the dynamics of a general robot arm, and give an example where classical methods are insufficient to stably control a robot arm.

3.1 A taxonomy of robot dynamics problems

The dynamics of a robot arm can be written as

$$\boldsymbol{\tau} = M(\theta)\ddot{\theta} + C(\theta)\left[\dot{\theta}\dot{\theta}\right] + D(\theta)\left[\dot{\theta}^2\right] + F(\theta,\dot{\theta}) + G(\theta) \tag{1}$$

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

$$\left[\dot{\theta}\dot{\theta}\right] = \left[\dot{\theta}_1\dot{\theta}_2, \dot{\theta}_1\dot{\theta}_3, \ldots, \dot{\theta}_{k-1}\dot{\theta}_k\right]^T, \tag{2}$$

$$\left[\dot{\theta}^2\right] = \left[\dot{\theta}_1^2, \dot{\theta}_2^2, \ldots, \dot{\theta}_k^2\right], \tag{3}$$

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

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 gear boxes 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 thus, such that this part can be neglected.

3.1.1 The rigid body assumption

The simplest kind of robot arm consists of rigid links which are connected by rigid joints. 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; although 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) is simplified in order to obtain:

$$\boldsymbol{\tau} = Y(\theta, \dot{\theta}, \dot{\theta}_r, \ddot{\theta}_r)w \tag{4}$$

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 θ 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 parameterized by the joint positions and velocities; Eq. (1) cannot be simplified anymore.

3.1.2 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.

3.1.3 Flexible joints

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

In the case of elasticity at joint level, an actuator can be modeled as 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_m) and the arm segment (measuring θ_l) side of the spring. Equation (1) changes as follows:

$$\boldsymbol{\tau}_l = M(\theta_l)\ddot{\theta}_l + C(\theta_l)\left[\dot{\theta}_l\dot{\theta}_l\right] + D(\theta_l)\left[\dot{\theta}_l^2\right] + F(\theta_l, \dot{\theta}_l) + G(\theta_l), \tag{5}$$

$$\boldsymbol{\tau}_m = J(\dot{\theta}_m)\ddot{\theta}_m + \boldsymbol{\tau}_l \tag{6}$$

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

3.1.4 Time delays

A final important problem in joint servo control is taking count of delays. Due to the fact that, in a feedback control loop, delays are present in the digital controllers as well in their communication, predictive control is required to obtain stable tracking. There are two problems that have considered:

1. **feedforward delays in the robot.** When the robot is given a new joint position, delays result due to the joint-interpolated trajectory planning, joint position readout, and communication. These factors usually add up to a delay in the range 5ms–200ms.

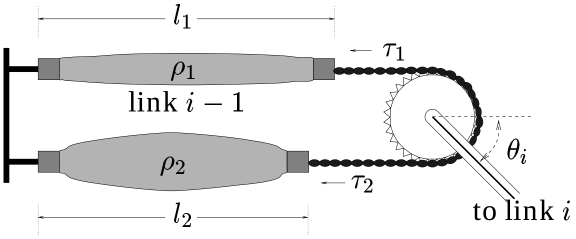

Figure 5: An agonist and an antagonist rubbertuator are connected via a chain across a sprocket; their relative lengths determine the joint position θ_i.

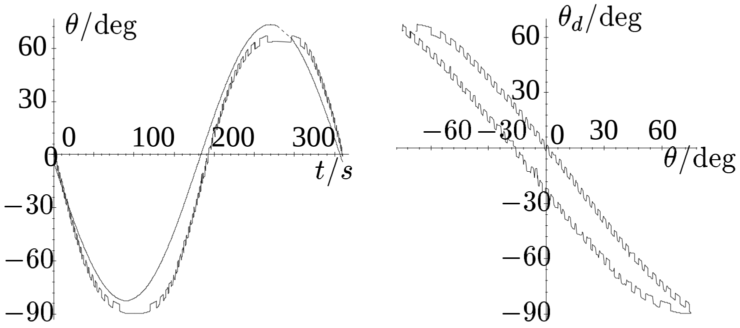

Figure 6: Using the internal PD controller to follow the trajectory $\theta_d(t) = c\sin(t)$. The left figure shows the desired and realized trajectories vs. t; the right figure depicts the desired (horizontal axis) vs. the realized (vertical axis) trajectory.

2. **feedback delays in the communication.** When the current joint position of the robot is probed, a communication delay between the various components of the controller will ensue. This kind of delay is usually negligible.

Any joint servo controller has to take these delays into account; otherwise, unstable control will result.

3.2 An exemplar difficult joint structure

The McKibben pneumatic artificial muscle (Chou & Hannaford, 1996), as used in the Bridgestone SoftArm robot, has typical problematic control properties (Smagt, Groen, & Schulten, 1996). The use of two artificial muscles in the construction of a joint is depicted in figure 5. The sprocket construction, combined with the properties of the artificial muscles, leads to a hysteretic nonlinear pressure-position relationship.

Naturally, a Proportional-Derivative (PD) controller cannot control such a robot. A test is shown in figure 6. In this figure, a single joint of the robot, controlled via a PD controller, follows a simple trajectory in joint space.

3.2.1 Using adaptive control

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 $\mathcal{F} : \Re^m \to \Re^n$ fast enough, *then* any of the above robot architectures can be optimally controlled.

Example: Figure 7 shows the pneumatic artificial muscle experiment using a self-learning controller based on fast learning feed-forward networks.

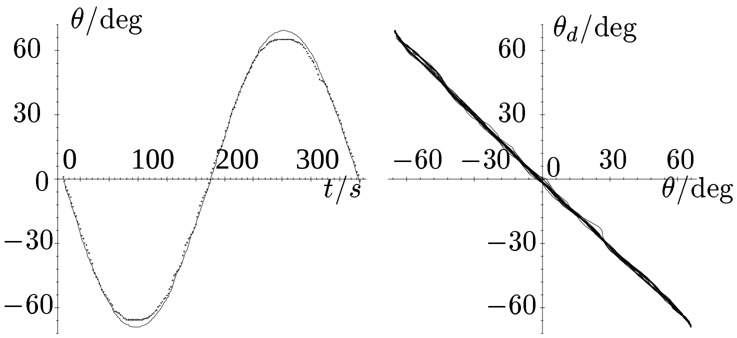

Figure 7: Using the neural network controller to follow the trajectory $\theta_d(t) = c\sin(t)$. The left figure shows the desired (solid line) and realized (dotted line) trajectories vs. t; the right figure depicts the desired (horizontal axis) vs. the realized (vertical axis) trajectory. This behaviour is recorded after five minutes of learning.

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 (Smagt et al., 1996). 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 (Smagt, 1994).

3.2.2 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 (e.g., Smagt, Groen, & Groenewoud, 1994; Braake, Can, & Verbruggen, 1998; Wang & Principe, 1997; Smagt & Hirzinger, 1998). 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. This path has been followed by several researchers; as we will see in the following section, however, more often than not applications of such controllers are restricted to the control of simulated 2-link robot arms.

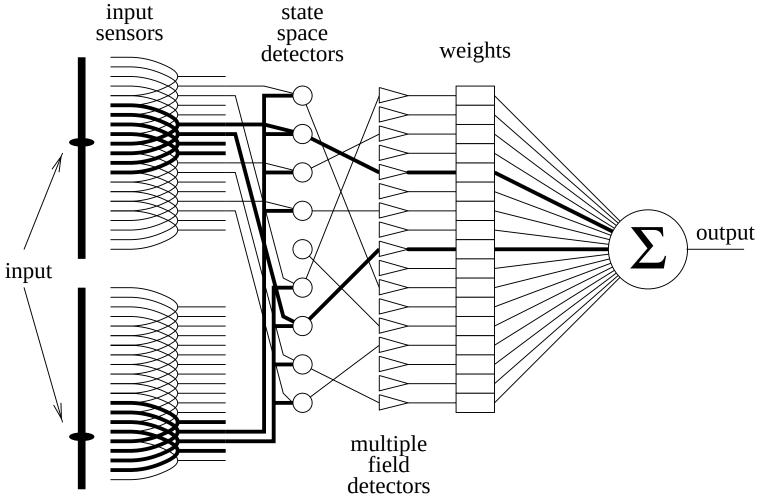

Figure 8: The CMAC model of the cerebellum.

4 Computational models of the cerebellum

Since the development of the CMAC model, many other cerebellar models have been developed, some of which have been used in robot control. A primary cause for this further development has been a better understanding of the structure of the cerebellum, but also how it interacts with other parts of the brain.

Unfortunately, many of these recent developments are not well enough understood to be applicable in a simulated cerebellum. In some cases, however, the applicability of the model is taken into account, and new computational models have been successfully used. We will discuss three mainstream models in this section in some extend, and mention various implementations and applications of such models.

4.1 Model I: The CMAC

One of the earliest computational models of the cerebellum is the *Cerebellar Model Articulation Controller* (CMAC), introduced by Albus (1975). The CMAC has been subsequently used in various adaptive robot control tasks by Miller et al. (1989, 1997), Lang (1997), as well as by other authors, and has thus received extensive attention; also, its extensive mathematical analysis (Miller, Glanz, & Kraft, 1990) has furthered its acceptance.

The original idea of CMAC is based on the BOXES approach by Michie and Chambers (1968). This approach basically implements a table lookup process. Of course the major problem with straightforward table lookup is, that during ‘learning’ no generalization takes place. In each cell a value must be separately stored, and the whole input–output relationship is only known when all input combinations are learned. This may take prohibitively long, esp. for applications such as robot control where the input has a relatively high dimensionality. The CMAC solves this problem.

The basic architecture of the CMAC (in the implementation of Miller et al.) is depicted in figure 8. The first important feature of the CMAC is the discretization of the input signals through the *input sensors*. Each signal activates a number of input sensors (four in the figure). The input sensors are connected to the *state space detectors* in a regular fashion; these detectors are AND-units and therefore only switch on when all their inputs are

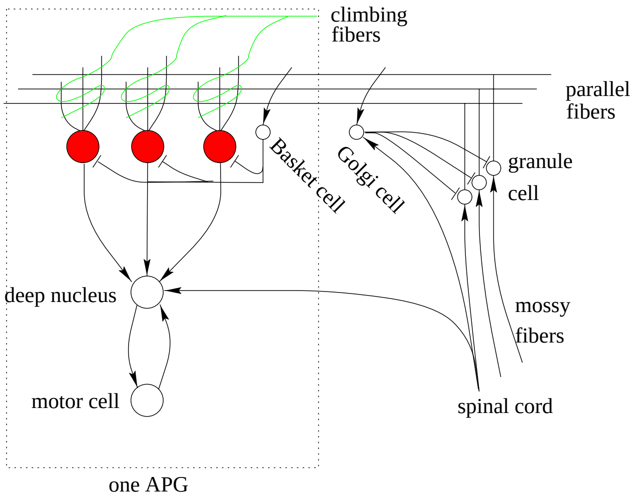

Figure 9: The structure of the Adjustable Pattern Generator model of the cerebellum. An arrow indicates an excitatory connection, a termination (⊣) an inhibitory connection.

active. The state space detectors are connected to the *multiple field detectors* via random hashing. These detectors compute the OR function. Finally, the weighted outputs of the multiple field detectors are summed together, constituting the output of the CMAC.

The input sensors together with the state space detectors implement the functionality of granule cells in the cerebellum. The multiple field detectors with their weighted output mimic a Purkinje cell.

Learning takes place by modifying the weights according to a delta rule. Due to the fact that the network output is linearly related to the weights, learning a certain pattern is instantaneous and no iterative procedure is required.

A recent improvement is the smoothed CMAC function (Kraft, 1997). This CMAC has the advantage that, due to the smoothing, the gradient of the network output can be accurately obtained, which is useful for robot dynamics control.

Albus' CMAC is clearly only a crude approximation of the cerebellum. For instance, an important omission in the model is the fact that it ignores the inhibitory output of the Purkinje cells. The CMAC has often been considered a function approximation model only, and can be compared with other general function approximation models (such as feed-forward networks and self-organizing feature maps) as such.

4.2 Model II: The adjustable pattern generator

The Adjustable Pattern Generator model (APG) was introduced by Houk and colleagues (Houk, 1989; Houk, Singh, Fisher, & Barto, 1990; Barto, Buckingham, & Houk, 1995). The term 'APG' was coined since the model can generate a burst command with adjustable intensity and duration.

In the APG a nucleus cell, connected to a motor cell in a feedback circuit, is inhibited by a group of Purkinje cells. The structure of the cerebellum mostly matches the structure described in section 2.2, and includes basket cells, cf, Golgi cells, granule cells, mf, pf, and the stellate cells. Each motor cell is connected to the nucleus cell in its own APG, but also to nucleus cells in neighbouring APGs. The structure of the network is shown in figure 9.

In this model, a basket cell spike causes the pc to change its state. The motor neuron / nucleus cell loop is no longer inhibited and a feedforward motion can begin. Subsequent firing patterns of the pc influence the firing of the motor neurons, and therefore the planned movement.

Applications of this model are currently restricted to simulated two-link robot arms (Fagg, Sitkoff, Houk, & Barto, 1997).

4.3 Model III: Internal models of the inverse dynamics

Kawato (Kawato & Gomi, 1992) has long argued that internal models of the proprio-skeleto-muscular structure, as well as one's environment, are a necessity for optimal control. An implementation of a system which learns its own internal models (viz. the feedback-error-learning model) was applied to control of the SoftArm robot described in section 3.2 (Katayama & Kawato, 1991) as well as other robotic tasks.

Kawato (1997) defends his assumption as follows:

1. **Why are internal models necessary for visual-motor coordination?**

   The major reason for this is that the low stiffness of the human arm during multi-joint movement suggests the existence of an inverse dynamics model of the arm. Furthermore, such models are extremely helpful in visuo-motor control, when pose estimation, trajectory formation, and motor command generation have to be performed.

2. **Where could internal models be located in the brain?**

   Such models have been experimentally found in the cerebellum. For instance, analysis of cerebellar Purkinje cell firing suggests that an inverse dynamics model of the eye musculature is used.

3. **How can internal models be acquired in the brain?**

   Kawato and Gomi (1992) have proposed their cerebellar feedback-error-learning model to solve this problem.

Furthermore, the fact that the newest part of the cerebellum is almost unique to humans, the assumption that it plays a key role in intelligence and self-observation is supported. This supports the assumption that the cerebellum not only contains models of the motor apparatus, but also of tools, one's own brain, etc.

Although Kawato's models has been criticized since it lacks a mechanism to take time delays in the control loop (see section 3.1.4) into account, this problem has been tackled through the work by Schweighofer (1995).

4.3.1 Learning motor programs

The key interest of course is solving the problem: how do I learn a whole motor program with a cerebellar structure? There are several attempts to tackle this problem.

An important structure is the parallel-hierarchical feedback-error-learning model (Katayama & Kawato, 1991; Kawato & Gomi, 1992) shown in figure 10 (cf. also figure 2). It consists of three parts. The Inverse Static Model (ISM) solves the posture equations when the system is in rest ($\dot{\theta} = 0$). The main role of the ISM is to control the equilibrium posture and mechanical stiffness (Hogan, 1984). Otherwise the Inverse Dynamic Model

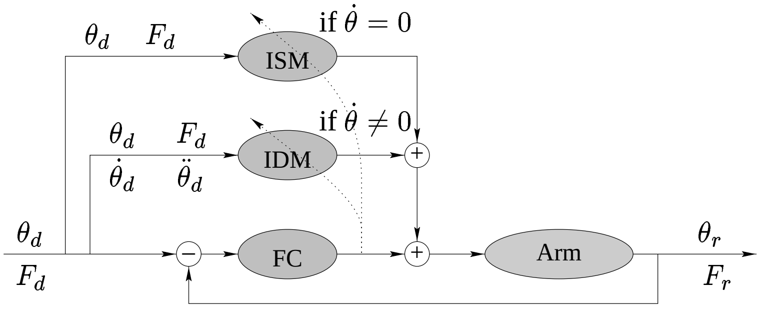

Figure 10: The parallel-hierarchical feedback-error-learning model (Katayama & Kawato, 1991; Kawato & Gomi, 1992).

(IDM) takes over control. The output of the ISM and IDM is augmented with the feedback signal from the feedback controller FC. This approach to control was previously suggested by Atkeson and Hollerbach (1985). Although the ISM and IDM were initially feed-forward neural networks, the IDM is typically implemented in the form of a cerebellar model.

Fagg et al. (Fagg, Zelevinsky, Barto, & Houk, 1997) investigate the behaviour of young infants when grasping an object. In contrast to adults, whose velocity profiles consist of a single peak and are approximately bell-shaped, their profiles consists of a sequence of peaks which define segments known as ‘movement units’ (Hofsten, 1979).

Their control system consists of a learning controller, which stores a feed-forward motor program. A hard-wired corrector module generates crude corrective movements when the learning controller does not produce accurate reaches. The output of the two modules is summed and passed to the muscles. Motor programs are represented in the form of a muscle activation pattern (the *pulse*) plus a time (the *step*).

This approach is similar to the feedback-error-learning model, since the system learns from an external controller. However, in contrast to the feedback-error-learning this system needs only sporadic feedback, and it does not need a high-quality trajectory to learn from.

Spoelstra, Arbib and Schweighofer (1997) use the feedback-error-learning model as a basis for their fast movement controller. They complement the ISM / IDM system by adding a reflex feedback from the spinal chord, from which the IDM is trained. Using a simulated planar (2 DoF) arm with McKibben muscles they demonstrate the superiority of their ISM / IDM system over PD control.

4.4 Other models

Jabri et al. (Jabri, Coenen, Huang, & Sejnowski, 1997) use a cerebellar model for sensory-motor integration and demonstrate their model on a Khepera miniature mobile robot. They follow up on the timing function of the cerebellum, and use a simple cerebellar model to predict infrared sensor readings. The single common feature that most cerebellar models have, they argue, is its predictive capabilities. Predictive motor commands are encoded in the nucleus cells, which are steered by the pc. The inhibitory inputs from the nucleus cells to the inferior olive may carry a delayed feedback of the predictions being established in the nucleus cells (Miall, Weir, Wolpert, & Stein, 1993).

A different adaptive control structure is given by the VITE–FLETE model of Bullock and Grossberg (1988a). The *Vector Integration To Endpoint* (VITE), an improved version of which was proposed by Gaudiano and Grossberg (1991), provides a simplified model of the cerebral cortex, and is basically a trajectory generator. The *Factorization of muscle Length and muscle Tension* (FLETE), modelling the spinal cord (Bullock & Grossberg, 1988b, 1989), is basically a model of the neuromuscular system. These models have been applied to mobile robot control (Aguilar & Contreras-Vidal, 1994; Gaudiano, Guenther, & Zalama, 1997), speech production (Gaudiano et al., 1997), handwriting generation (Bullock, Grossberg, & Mannes, 1993), and visually guided movements (Li & Öğmen, 1994).

Using a feed-forward cerebellar model which computes inverse dynamics signals, Contreras-Vidal et al. (Contreras-Vidal, Grossberg, & Bullock, 1997; Contreras-Vidal & Lopez-Coronado, 1997) extend the model to the the control of opponent muscles. This model, combined with the cerebral cortex (VITE) and the spinal cord (FLETE), is proposed as a viable method to control the fingers of an artificial hand.

Various other models, some of which lack the detail to create a computational model, have been proposed (Paulin, 1989; Kawato & Gomi, 1992; Thach, Goodkin, & Keating, 1992; Berthier, Singh, Barto, & Houk, 1993; Miall et al., 1993; Paulin, 1993; Buonomano & Mauk, 1994; Lisberger, 1994; De Schutter, 1997). For an overview see (Houk et al., 1996).

5 Discussion

Taking into account the fact that the new generation light-weight robots poses a challenge to traditional robot control methodologies, model-free adaptive control seems to be the way to go. Some early artificial cerebellar models have demonstrated their capabilities in robot control, and seem to be a viable approach to solve the problem.

However, there is a large discrepancy between those early models and the current-day understanding of the cerebellum. One important omission is the fact that the Purkinje cells generate an *inhibitory* output; a fact for which many theories exist but the significance of which is not yet fully explained. Yet many other differences between the computational models and the cerebellum exist.

Newer, usually adapted, models have been proposed. Currently, however, most of these models have not been capable of solving robot control tasks which are more complex than 2 DoF, and usually on simulated robot structures only. Another important problem is the existence of time delays: delays are present in the biological motor system as well as in robot structures, and therefore have to be taken into account in any joint servo control method. Most cerebellar models, however, ignore this problem.

The development of computational cerebellar models could be greatly advanced when they would be compared on a set of realistic robot arm dynamics benchmarks at a functional level. This approach may shed more light on the applicability of such models to real-life problems. However, can these models compete with engineering solutions?

Acknowledgments

Many fruitful discussions with Daniel Bullock have strongly influenced this research. The author kindly acknowledges the continuing support of Gerd Hirzinger, who leads the Department of Robotic Systems at the German Aerospace Center in Oberpfaffenhofen.

References

Aguilar, J. M., & Contreras-Vidal, J. L. (1994). NAVITE: A neural network system for sensory-based robot navigation. In *Proceedings of the World Congress on Neural Networks* (Vol. II, pp. 177–182). Lawrence Erlbaum, Hillsdale, New Jersey.

Aiba, A., Kano, M., Chen, C., Stanton, M., Fox, G., Herrup, K., Zwingman, T., & Tonegawa, S. (1994). Deficient cerebellar long-term depression and impaired motor learning in mGluR1 mutant mice. *Cell*, *79*, 377–388.

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

Albus, J. (1975). A new approach to manipulator control: The cerebellar model articulation controller (CMAC). *Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME, series G*, *97*, 270–277.

Artola, A., & Singer, W. (1993). Long-term depression of excitatory synaptic transmission and its relationship to long-term potentiation. *Trends in Neuroscience*, *16*, 480–487.

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

Barto, A. G., Buckingham, J. T., & Houk, J. C. (1995). A predictive switching model of cerebellar movement control. In D. S. Touretzky, M. C. Mozer, & M. E. Hasselmo (Eds.), *Advances in Neural Information Processing Systems 8* (pp. 138–144). Boston, MA: MIT Press.

Bell, C. (1811). *Idea of a New Anatomy of the Brain; Submitted for the Observations of His Friends.* London: Strahan and Preston.

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

Bliss, T. V. P., & Collingridge, G. (1993). A synaptic model of memory: Long-term potentiation in the hippocampus. *Nature*, *361*, 31–39.

Braake, H. A. B. te, Can, E. J. L. van, & Verbruggen, H. B. (1998). Training of neural networks by a variable projection method. *Neural Networks.* (In print.)

Braitenberg, V. (1961). Functional interpretation of cerebellar histology. *Nature*, *190*, 539–540.

Braitenberg, V., & Atwood, R. P. (1958). Morphological observations on the cerebellar cortex. *Journal of Comparative Neurology*, *109*, 1–34.

Bullock, D. (1997). *Cerebellar learning for context sensitive and critically timed coordination of multiple action channels.* (In van der Smagt & Bullock, pp. 11–12, NIPS*97)

Bullock, D., & Grossberg, S. (1988a). Neural dynamics of planned arm movements: Emergent invariants and speed-accuracy properties during trajectory formation. *Psychological Review*, *95*, 49–90.

Bullock, D., & Grossberg, S. (1988b). Neuromuscular realization of planned trajectories. *Neural Networks*, *1*(Suppl. 1), 329.

Bullock, D., & Grossberg, S. (1989). VITE and FLETE: Neural modules for trajectory formation and tension control. In W. Hershberger (Ed.), *Volitional Action* (pp. 253–297). Amsterdam: North-Holland.

Bullock, D., Grossberg, S., & Mannes, C. (1993). A neural network model for cursive script production. *Biological Cybernetics, 70*, 15–28.

Buonomano, D. V., & Mauk, M. D. (1994). Neural network model of the cerebellum: Temporal discrimination and the timing of motor responses. *Neural Computation, 6*, 38–55.

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

Contreras-Vidal, J., 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., & Lopez-Coronado, J. (1997). *Adaptive Cerebellar Control of Opponent Muscles.* (In van der Smagt & Bullock, pp. 13–16, NIPS*97)

De Schutter, E. (1996). One cannot build theories of cerebellar function on shaky foundations: Induction properties of long-term depression have to be taken into account. *Behavioral and Brain Sciences, 19*, 440–441.

De Schutter, E. (1997). A new functional role for cerebellar long term depression. *Progress in Brain Research, 114*, 529–542.

DeMers, D., & Kreutz-Delgado, K. (1997). Inverse kinematics of dextrous manipulators. In O. Omidvar & P. van der Smagt (Eds.), *Neural Systems for Robotics* (pp. 75–116). Boston, Massachusetts: Academic Press.

Eccles, J. C., Ito, M., & Szentagothai, J. (1967). *The Cerebellum as a Neuronal Machine.* Berlin: Springer-Verlag.

Fagg, A., Zelevinsky, L., Barto, A., & Houk, J. (1997). *Using Crude Corrective Movements to Learn Accurate Motor Programs for Reaching.* (In van der Smagt & Bullock, pp. 20–24, NIPS*97)

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

Flourens, P. (1824). *Recherches expérimentales sur les propriétés et les fonctions du système nerveux, dans les animaux vertébres.* Paris: Cervot.

Franklin, B. (1774). *Experiments and Observations on Electricity.* London.

Gaudiano, P., & Grossberg, S. (1991). Vector associative maps: Unsupervised real-time error-based learning and control of movement trajectories. *Neural Networks, 4*(2), 147–183.

Gaudiano, P., Guenther, F. H., & Zalama, E. (1997). The neural dynamics approach to sensory-motor control. In O. Omidvar & P. van der Smagt (Eds.), *Neural Systems for Robotics* (pp. 153–194). Boston, Massachusetts: Academic Press.

Glickstein, M. (1994). Cerebellar agenesis. *Brain*, *117*, 1209–1212.

Hirzinger, G. (1996). Towards a new robot generation for space, terrestrial, and medical applications. In *Proc. Second ECPD Int. Conference on Advanced Robotics, Intelligent Automation, and Active Systems.*

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

Hogan, N. (1984). An organizing principle for a class of voluntary movements. *Journal of Neuroscience*, *4*(11), 2745–2754.

Hore, J., & Flament, D. (1988). Changes in motor cortex neural discharge associated with the development of cerebellar limb ataxia. *Journal of Neurophysiology*, *60*, 1285–1302.

Houk, J. C. (1989). Cooperative control of limb movements by the motor cortex, brainstem, and cerebellum. In R. M. J. Cotterill (Ed.), *Models of Brain Function* (pp. 309–325). Cambridge University Press.

Houk, J. C., Buckingham, J. T., & Barto, A. G. (1996). Models of the cerebellum and motor learning. *Behavioral and Brain Sciences*, *19*(3), 368–383.

Houk, J. C., Singh, S. P., Fisher, C., & Barto, A. G. (1990). An adaptive sensorimotor network simulation inspired by the anatomy and physiology of the cerebellum. In W. Miller, R. Sutton, & P. Werbos (Eds.), *Neural Networks for Control* (pp. 301–348). MIT Press.

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

Ito, M., Sakurai, M., & Tongroach, P. (1982). Climbing fiber induced depression of both mossy fiber responsiveness and glutamate sensitivity of cerebellar Purkinje cells. *Journal of Physiology*, *324*, 113–134.

Jabri, M., Coenen, O., Huang, J., & Sejnowski, T. (1997). *Sensorimotor integration and control.* (In van der Smagt & Bullock, pp. 33–38, NIPS*97)

Kano, M. (1996). Long-lasting potentiation of GABAergic inhibitory synaptic transmission in cerebellar Purkinje cells: Its properties and possible mechanisms. *Behavioral and Brain Sciences*, *19*, 354–361.

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

Kawato, M. (1997). *Multiple Internal Models in the Cerebellum.* (In van der Smagt & Bullock, pp. 17–19, NIPS*97)

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

Kraft, G. (1997). *Optimized Weight Smoothing for CMAC Neural Networks.* (In van der Smagt & Bullock, pp. 8–10, NIPS*97)

Lang, M. (1997). *A Real-Time Implementation of a Neural-Network Controller for Industrial Robotics.* Unpublished doctoral dissertation, Institute of Aerospace Studies, University of Toronto.

Li, L., & Öğmen, H. (1994). Visually guided motor control: Adaptive sensorimotor mapping with on-line visual error correction. In *Proceedings of the World Congress on Neural Networks* (Vol. 2, pp. 127–134). Hillsdale, New Jersey: Lawrence Erlbaum.

Lisberger, S. G. (1994). Neural basis for motor learning in the vestibuloocular reflex of primates. III. Computational and behavioral analysis of the sites of learning. *Journal of Neurophysiology*, *72*, 974–998.

Lisberger, S. G., Pavelko, T. A., Bronte-Stewart, H. M., & Stone, L. S. (1994). 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*, 954–973.

Lisberger, S. G., Pavelko, T. A., & Broussard, D. M. (1994). Neural basis for motor learning in the vestibuloocular reflex of primates. I. Changes in the responses of brain stem neurons. *Journal of Neurophysiology*, *72*, 928–953.

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

Miall, R. C., Weir, D. J., Wolpert, D. M., & Stein, J. F. (1993). Is the cerebellum a Smith predictor? *Journal of Motor Behavior*, *25*, 203–216.

Michie, D., & Chambers, R. A. (1968). BOXES: An experiment in adaptive control. In E. Dale & D. Michie (Eds.), *Machine Intelligence* (Vol. 2, pp. 137–152). Edinburgh: Oliver and Boyd.

Miller, W. T., Glanz, F. H., & Kraft, L. G. (1990). CMAC: An associative neural network alternative to backpropagation. *Proceedings of the IEEE, Special Issue on Neural Networks*, *78*, 1561–1567.

Miller III, W. T. (1989). Real-time application of neural networks for sensor-based control of robots with vision. *IEEE Transactions on Systems, Man, and Cybernetics*, *19*(4), 825–831.

Miller III, W. T., & Kun, A. L. (1997). Neural systems for robotics. In O. Omidvar & P. van der Smagt (Eds.), *Dynamic Balance of a Biped Walking Robot.* Boston, MA: Academic Press.

Nelson, M. (1997). *Adaptive Motor Control Without a Cerebellum.* (In van der Smagt & Bullock, pp. 25–27, NIPS*97)

Omidvar, O., & Smagt, P. van der (Eds.). (1997). *Neural Systems for Robotics.* Boston, Massachusetts: Academic Press.

Paulin, M. G. (1989). A Kalman filter theory of the cerebellum. In M. A. Arbib & S. Amari (Eds.), *Dynamic Interactions in Neural Networks* (pp. 241–259). Heidelberg-Berlin: Springer-Verlag.

Paulin, M. G. (1993). The role of the cerebellum in motor control and perception. *Brain, Behavior, and Evolution*, *31*, 39–50.

Schweighofer, N. (1995). *Computational models of the cerebellum in the adaptive control of movements.* Unpublished doctoral dissertation, University of Southern California.

Simpson, J. I., Wylie, D. R., & De Zeeuw, C. I. (1996). On climbing fiber signals and their consequence(s). *Behavioral and Brain Sciences, 19,* 384–398.

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

Smagt, P. van der. (1995). *Visual Robot Arm Guidance using Neural Networks.* Unpublished doctoral dissertation, Dept of Computer Systems, University of Amsterdam.

Smagt, P. van der, & Bullock, D. (Eds.). (1997). *Can Artificial Cerebellar Models Compete to Control Robots?* (Tech. Rep. No. 515-97-28). Oberpfaffenhofen. (NIPS*97 Workshop. Downloadable from `http://www.robotic.dlr.de/Smagt/papers/`)

Smagt, P. van der, Groen, F., & Groenewoud, F. van het. (1994). The locally linear nested network for robot manipulation. In *Proceedings of the IEEE International Conference on Neural Networks* (pp. 2787–2792).

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

Smagt, P. van der, & Hirzinger, G. (1998). Solving the ill-conditioning in neural network learning. In J. Orr, K. Müller, & R. Caruana (Eds.), *Tricks of the Trade: How to Make Neural Networks Really Work.* Lecture Notes in Computer Science, Springer Verlag.

Spoelstra, J., Arbib, M., & Schweighofer, N. (1997). *Cerebellar control of a simulated biomimetic manipulator for fast movements.* (In van der Smagt & Bullock, pp. 28–32, NIPS*97)

Thach, W., Goodkin, H., & Keating, J. (1992). The cerebellum and the adaptive coordination of movement. *Annual Review of Neuroscience, 15,* 403–442.

Wang, C., & Principe, J. C. (1997). On-line stochastic functional smoothing optimization for neural network training. *IEEE Transactions on Systems, Man, and Cybernetics.* (Submitted)