



## Searching a Scalable Approach to Cerebellar Based Control

JAN PETERS

*Computational Learning and Motor Control Lab, University of Southern California,  
3641 Watt Way, Los Angeles, CA 90*

*jpgeters@usc.edu*

PATRICK VAN DER SMAGT

*Institute of Robotics and Mechatronics, German Aerospace Center, DLR Oberpfaffenhofen,  
P.O. Box 1116, 82230 Wessling, Germany*

*smagt@dlr.de*

**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 might take 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, few applications have been realized.

The currently emerging family of light-weight robots (Hirzinger, in *Proc. Second ecpd Int. Conference on Advanced Robotics, Intelligent Automation and Active Systems*, 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due to their complex dynamics traditional methods, depending on a full analysis of the dynamics of the system, are no longer applicable since the joints influence each other dynamics during movement.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?

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

### 1. Introduction

Throughout the years there has been extensive research in the structure and function of the cerebellum. It is part of the brain of vertebrates that is well understood in various aspects: It is used for the spatial motor coordination of fast movements. The structure of the cerebellum is highly regular, the cells as well as their connections are known up to a large extent. Also the connections to the other parts of the brain and the spinal cord are partially understood. Theories on the learning process of the cerebellum have been presented, resulting into several computational models.

However, although there has been done much research on cerebellar models and their interaction with other parts of the central nervous system, application

of computational cerebellar models has been limited and has remained in its infant state.

Since the beginning of the 1990s a new generation of robots is emerging, the light-weight robots. Due to their intended application outside of factory floors in service and home applications, they are required to have a high payload-to-weight ratio (about 1:1). In comparison, for industrial robots this ratio is an approximate 1:40. Light-weight robots pose difficult control problems: their structures are flexible, and their friction problems cannot be neglected. therefore, full modelling of the robot is difficult or impossible, such that traditional control cannot be used.

Interestingly, primates generate remarkably precise motions in an even more complex control situation; the cerebellum manages this fine motion

control. Can we use cerebellar-based control for robots?

This overview-paper presents gives an in-depth discussion of the problems in robot control (Section 2) and the cerebellum (Section 3). Section 4 discusses four cerebellar models. A discussion is given in Section 5.

## 2. Robot Control

But can robots use a cerebellar model for stable control? Besides the fact that the majority of robot arms, that are a subject of academic research, have an anthropomorphic structure, there is little resemblance between human and robot arms. A first, important, difference is the material that they consist of. The rigid and heavy-weight construction material of industrial robot arms implies the requirement of actuators (typically DC motors) which can generate the necessary forces. The robot motion is incurred through the forces or torques that are exerted by the actuators.

### 2.1. Robot Motion Tasks

Usually, the task that has to be performed by a robot arm with up to six degrees of freedom can be described by a desired Cartesian trajectory  $(\mathbf{x}_d[t], \phi_d[t])$ , where  $\mathbf{x}_d$  is the desired (3D) position and  $\phi_d$  the desired (3D) orientation, that must be followed by its end effector. In sensor-based robotics, this trajectory may not be explicitly available before the whole motion has been completed, but when computed would uniquely describe the movement at hand. Also, in some applications only the final position of the end effector is of importance; in that case, *trajectory planning* or *path planning* is used to determine the trajectory from the current robot position to the desired position.

In traditional robotics, the instantiation of this trajectory is performed in a sequence of steps:

*Inverse Kinematics.* For a robot with revolute joints (e.g., an anthropomorphic robot), this process translates the end effector trajectory into a desired trajectory  $\theta_d[t]$  in joint space. The forward mapping, the *kinematics*, is in fact determined by (the mechanical structure of) the robot.

In many instances, the inverse kinematics is not a function but has multiple solutions. When the dimensionality (degrees of freedom) of the robot exceeds that of the task, there are infinitely many solutions. But also when these two are equal (typically 6), there

may be more than one arm configuration to reach a point, e.g., elbow-up and elbow-down. Additional constraints, such as restrictions in allowed acceleration, can be used to select favourable solutions.

When the inverse kinematics solution is only used to determine a target end effector position, this step must be followed by trajectory planning. At set intervals  $\Delta t$  (typically in the range 1–20 ms) a joint position must be computed which the robot arm should move to.

*Dynamic Control or Tracking.* When a positional increment is available at each  $\Delta t$ , the joint servo control computes the necessary forces or torques at the motor side to realize the requested motion. In traditional industrial robotics, a PD (*Proportional-Derivative*) controller is customarily used. For a rotary joint, this controller sets the torque  $\tau_d$  to a “proportional” constant  $k_p$  times the joint position error, plus a “derivative” constant  $k_v$  times the joint velocity error. For joint  $i$ :

$$\tau_{d,i}(t) = k_{p,i}[\theta_{d,i}(t) - \theta_i(t-1)] + k_{v,i}[\dot{\theta}_{d,i}(t) - \dot{\theta}_i(t-1)]. \quad (1)$$

Clearly, all the joints are independently controlled, and centrifugal and Coriolis forces are assumed to be nonexistent. This control law is known as *servo control*.

An important property of the PD control rule is its proven stability. Arimoto and Miyazaki [1] showed that the joint error goes to 0 when using PD control, it was later on shown that it decays at least exponentially.

*Motor Control.* In the study of biological control systems, the term *motor control* refers to the control of movement in general. In robotics, *motor control* refers to the mechanism which controls the DC motors in order to generate a desired torque  $\tau_d$ . Assuming that the joint servo control problem is adequately solved, the motor control problem can be tackled on a joint-independent basis. A fast local feedback loop, using the measured motor current  $i$  as steering signal, suffices here to control the motor.

### 2.2. Robot Dynamics

A robot arm consists of the following parts:

1. *the links* or arm segments. We consider these to be rigid bodies, which is realistic for most small and medium-sized robots, as well as for skeletomuscular systems.

2. *the actuators*. A tendency exists towards using DC motors or step motors for generating the required force; however, other types of actuators such as pneumatic artificial muscles [2] have also received considerable attention.
3. *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.

Under the assumption that the links and joints do not mechanically deform, the forces that the robot structure exerts at the actuators is given by

$$\tau = M(\theta)\ddot{\theta} + C(\theta)[\dot{\theta}\dot{\theta}] + D(\theta)[\dot{\theta}^2] + F(\theta, \dot{\theta}) + G(\theta) \quad (2a)$$

$$\equiv M(\theta)\ddot{\theta} + V(\theta, \dot{\theta}) + G(\theta) \quad (2b)$$

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

$$[\dot{\theta}\dot{\theta}] = [\dot{\theta}_1\dot{\theta}_2, \dot{\theta}_1\dot{\theta}_3, \dots, \dot{\theta}_{N-1}\dot{\theta}_N]^T, \quad (3)$$

$$[\dot{\theta}^2] = [\dot{\theta}_1^2, \dot{\theta}_2^2, \dots, \dot{\theta}_N^2], \quad (4)$$

$M(\theta)$  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. Equation (2b) is a simplified notation of Eq. (2a).

*Agonist/Antagonist Drive*. Biological arms, as well as some experimental robots, use an agonist/antagonist drive principle. An example robot arm using this principle was the Bridgestone rubeuator arm, equipped with two or four McKibben [2] artificial muscles per joint, connected via sprockets [3–5]. However, complex dynamic properties as well as extreme temperature sensitivity and costly maintenance have lead to the production stop of this robot.

The advantage of an agonist/antagonist drive concept is that the *stiffness* of a joint can be very intuitively set: while the joint angle depends on the difference in exerted forces, the joint stiffness depends on the sum

of those forces. There are several reasons, however, for mechanical systems not to use this scheme; the key one being the fact that two drives per joint are required, resulting in a significantly higher weight and more power consumption. These resources are better invested in a robot arm construction with a higher force-to-weight ratio.

**2.2.1. Rigid Robots.** The simplest kind of robot arm consists of rigid links which are connected by rigid joints. This assumption is true enough for industrial robots; the construction of the robot arm is thus that any yield in the links as well as the joints can be neglected. In this case control is done by linearization of the control equation. Equation (2b) can be simplified in order to obtain:

$$\tau = Y(\theta, \dot{\theta}, \ddot{\theta})w \quad (5)$$

where  $w$  are the parameters which have to be estimated.

In industrial robots, the actuators are typically strong enough such that the diagonal elements of the mass matrix  $M$  and the centrifugal matrix  $D$  are prevalent, while  $C$  is approximately 0. Additionally, all matrices are approximately constant, i.e., independent of  $\theta$  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.

Such simplifications usually do not hold for light-weight robot arms, which are a subject of research in many robotics research labs. Due to weight and space limitations, the actuators that are employed are not powerful enough to eliminate the influence of gravity, friction, and Coriolis and centrifugal forces. 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. (2a) cannot be simplified anymore.

**2.2.2. Flexible Links.** Fortunately, material science allows the use of high-carbon wire in the production of robots: a strong light-weight material which, although still costly to produce and especially to process, is used as an alternative to aluminum because of its strength. The little research that is currently being done on flexible link robots is generally restricted to two-link arms (see, e.g. [6]). We consider this topic to be outside the scope of this paper.

**2.2.3. Compliance.** As mentioned before, light-weight robot arms are mostly equipped with actuators which can exert only a limited torque, and therefore

require high-ratio gear boxes. The disadvantage of this approach is the yield in such gear boxes, which has to be taken into account in the control law.

The effect of a yielding gear box is that the torque at the motor side differs from the torque at the link side. The resulting actuator is modeled by a motor–link pair connected by a spring. To measure the spring properties, angle sensors both at the drive (measuring  $\theta_m$ ) and link (measuring  $\theta_l$ ) sides have to be available. Equation (2b) changes as follows:

$$\tau_l = M(\theta_l)\ddot{\theta}_l + V(\theta_l, \dot{\theta}_l) + G(\theta_l), \quad (6a)$$

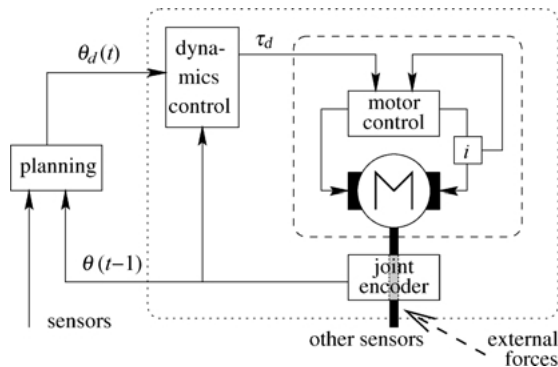
$$\tau_m = J(\dot{\theta}_m)\ddot{\theta}_m + \tau_l \quad (6b)$$

where  $\tau_m$  is the torque at the drive side and, using a linear spring model,  $\tau_l \equiv k(\theta_m - \theta_l)$  the torque at the link side.  $J$  can generally be assumed to be a diagonal matrix.

Naturally, biological skeletomuscular systems are always compliant, and it is this aspect in which cerebellar control may be able to play an important role.

**2.2.4. Delays.** A robot arm is generally controlled in various feedback loops: using imperfect measurements, some model is used to generate new action commands which are subsequently executed by the robot. During action, the actual motion is guarded and corrected where necessary (see Fig. 1).

Feedback loops are necessary since the models on which control commands are based are never in perfect correspondence with the real world. Clearly, the faster the feedback loop works, the less precise the robot mod-



*Figure 1.* Feedback loops in a robot control system. At the lowest level, the motor controller uses the measured electric current  $i$  to effectuate a desired torque  $\tau_d$ . Above that, the joint servo control uses the joint angle measurements  $\theta(t-1)$  to generate  $\tau_d$  such as to reach a desired joint position  $\theta_d(t)$ .

els that are used need to be. Therefore, the tendency in robotics is to decrease the joint servo feedback delays. For light-weight robot arms, delays of 1 ms are not uncommon, whereas industrial robots typically have delays of around 10 ms, necessary to obtain fast and stable control.

The situation is radically different in biological systems. Feedback delays can be as much as 110–150 ms for proprioceptive control [7] and 200–250 ms for visuomotor control [8]. Stable control in the presence of such extensive delays can partly be explained by the fact that skeletomuscular systems have an apparent passive behaviour [9].

### 2.3. Model-Based Control

To control a robot structure, a control law has to be devised which computes torques which, when applied to the joints, make the robot arm stably follow a trajectory  $\theta[t]$ , taking the dynamics Eq. (2b) or Eqs. (6a) and (6b) into consideration.

It is customary to simplify the problem via *control law partitioning*: the control law is dissected in a model-based part and an error-based part. If  $\tau_d$  is the input torque to the plant, we write  $\tau_d = \alpha\tau'_d + \beta$  where  $\tau'_d$  is the torque applied to the unit mass system. By setting  $\alpha = M(\theta)$  and  $\beta = V(\theta, \dot{\theta}) + G(\theta)$  (Eq. (2b)), the control law is

$$\tau'_d = \ddot{\theta}_d + K_v(\dot{\theta}_d - \dot{\theta}) + K_p(\theta_d - \theta), \quad (7)$$

i.e., a simple servo control method (cf. Eq. (1)). This control system is also known as *computed torque control* or *model-based control*.

Clearly, the key question remains: how do we find the optimal  $\alpha$  and  $\beta$ ? There exist various robotics techniques which address this problem; however, it is far from solved, and requires substantial modeling of robot structures. In PD control, the simplified model  $\alpha \equiv I$  and  $\beta \equiv 0$  is chosen: the model-based component is eliminated. Furthermore, the elements of the diagonal matrices  $K_v$  and  $K_p$  are chosen as large as possible (depending on how strong the actuators are), such as to eliminate the errors introduced by the neglected Coriolis and centrifugal forces. If this approach is reasonable, a proven stable control method results [1]. In other cases, however,  $\alpha$  and  $\beta$  can, to date, only be found by extensive modeling of the robot.

#### 2.4. Adaptive Control

An adaptive improvement over model-based control is given by *adaptive control*. It uses changing average values for robot dynamics parameters which are not exactly known or subject to change. Although there exist various adaptive control methods, we will cite one which is similar to the model-based control described above. We define  $\phi$  as a vector containing parameters on dynamics, friction, flexibilities, etc. Then we redefine our robot model from Eq. (2a) using

$$\begin{aligned} \tau = & M(\theta, \phi)\ddot{\theta} + C(\theta, \dot{\theta}, \phi) + D(\theta, \dot{\theta}, \phi) \\ & + F(\theta, \dot{\theta}, \phi) + G(\theta, \phi). \end{aligned} \quad (8)$$

As shown in ([10], pp. 121–122), the entire model is a linear function of the parameter vector  $\phi$ , expressed by

$$\begin{aligned} K(\theta, \dot{\theta}, \ddot{\theta})\phi = & M(\theta, \phi)\ddot{\theta} + C(\theta, \dot{\theta}, \phi) + D(\theta, \dot{\theta}, \phi) \\ & + F(\theta, \dot{\theta}, \phi) + G(\theta, \phi). \end{aligned} \quad (9)$$

Assuming that an estimate  $\hat{\phi}$  of  $\phi$  is given, we consider the following control algorithm [11]

$$\begin{aligned} \tau = & M(\theta, \hat{\phi})\ddot{\theta} + C(\theta, \dot{\theta}, \hat{\phi}) + D(\theta, \dot{\theta}, \hat{\phi}) \\ & + F(\theta, \dot{\theta}, \hat{\phi}) + G(\theta, \hat{\phi}) + K_D(\dot{\theta}_d - \dot{\theta}) \\ & + K_P(\theta_d - \theta) \end{aligned} \quad (10)$$

$$\ddot{\theta} = \ddot{\theta}_d + 2\lambda(\dot{\theta}_d - \dot{\theta}) + \lambda^2(\theta_d - \theta) \quad (11)$$

where  $\lambda$  is a positive constant which has to be selected by experience.  $K_D$  and  $K_P$  are square parameter matrices,  $\theta_d$  is the desired position and  $\dot{\theta}_d$  is the desired velocity.

For updating the estimated parameter vector, we use

$$\frac{d}{dt}\hat{\phi} = \Gamma K^T(\theta, \dot{\theta}, \ddot{\theta})[(\dot{\theta}_d - \dot{\theta}) + \lambda(\theta_d - \theta)] \quad (12)$$

where  $\Gamma$  is a constant positive-definite symmetric matrix, and where  $K(\theta, \dot{\theta}, \ddot{\theta})$  is given by Eq. (9) with  $\ddot{\theta} = \ddot{\theta}_m$ . Yoshikawa [10] shows that this type of control is stable and that the error  $e = \theta_d - \theta$  converges to zero.

### 3. The Cerebellum

The cerebellum, having a highly regular structure [12], plays a crucial role in movement and posture indirectly by adjusting the output of the major descending motor systems of the brain. This is apparent from cerebellar lesions and deficits, which have various effects on motor performance [13]. Unlike damage to

the cerebral motor cortex, damage to the cerebellum does not prevent voluntary activity, but rather reduces its quality. This is traditionally attributed to a release of motor pathways from supervisory control after damage occurs. Hence, many cerebellar dysfunctions are marked by excessive, undesired motion. Lesions of the vestibulocerebellum and spinocerebellum produce a loss of posture: reduced muscle tone in the trunk and neck, or a staggering, drunken gait. Lesions of the cerebrocerebellum yield a number of complex motor deficits [14]: diminished tendon reflexes (hypotonia), weak, easily tired muscles (asthenia, attributed to inefficient muscular control), decomposition of movements into component parts, inability to gauge distance of movements correctly, past-pointing (dysmetria), inability to perform smooth alternating, repetitive movements (dysdiadochokinesia), oscillation in voluntary motion (intention tremor), ataxic gait (“drunken” walk), and slow, explosive, slurred speech (scanning speech).

It is sometimes presumed that patients with cerebellar lesions can recover from those damages. Supposedly, the cerebral motor cortex takes over functions of the cerebellum. Holmes [15] recorded a large number of motor control impairments due to partial cerebellar damages. All of his experiments demonstrate an inability in the motor programs of effected patients. He points out that a “a striking feature of cerebellar injury [is] that its symptoms gradually decrease in intensity and may in time disappear . . . by compensation by intact parts of the cerebellum . . . in acute stages of disease the disturbances may be only reduced to a certain extent.” Glickstein [16] has carefully analyzed historical medical cases and concluded that there are no reported case histories exist with full recovery of patients with an abnormally developed cerebellum.

#### 3.1. Anatomy

The cerebellum occupies most of the “posterior cranial fossa” (i.e., the lower part at the backside of the skull). It is composed out of a mantle of gray matter (the cerebellar cortex), internal white matter, and three pairs of deep nuclei: the fastigial nucleus, the interposed nucleus and the dentate nucleus.

The cerebellum receives input from three sources: the periphery (e.g., the spinal cord), the brain stem, and the cerebral cortex. The input pathways to the cerebellum synapse on neurons both the deep nuclei and the cerebellar cortex. Most of the outflow from

the cortex projects back to the deep nuclei, in addition some portions of the cerebellar cortex project directly to the vestibular nuclei in the brain stem.

**3.1.1. Regional Organization.** The cerebellum is also divided into three major horizontal divisions: vestibulocerebellum, spinocerebellum, and cerebrocerebellum.

**3.1.1.1. The Vestibulocerebellum.** The vestibulocerebellum, also referred to as the flocculonodular lobe, has direct connections to the vestibular apparatus, and as such it is devoted to vestibular reflexes and eye movements, static and dynamic balance as well as posture [13]. It participates in the equilibrium of the human body during movements. Damage of the vestibulocerebellum has the following effects: truncal ataxia (instability of the body trunk, resembling movements of a drunk person), a tendency to fall, a wide-base stance (attempt to maintain balance in static position), the inability to walk from heel to toe (tandem walk), an intentional axial tremor (titubation), and pathologic nystagmus (persistent movements of eyes).

**3.1.1.2. The Spinocerebellum.** The spinocerebellum focuses on control of axial musculature through vermo-fastigial system of connections and on control of the distal and proximal limb musculature through paravermal-globose/emboliform system of connections. Thus, this region of the cerebellum oversees the execution and coordination of the motor events. It is presumed to contain topographical maps of the body that receives sensory information of the spinal cord [12]. In neurology, damage to the spinocerebellum is described as the anterior lobe syndrome, it has the following effects: gait ataxia if paravermis is involved (staggering gait due to discoordination of the muscles of the lower extremities), and hypotonia of muscles of trunk and/or extremities.

**3.1.1.3. The Cerebrocerebellum.** The cerebrocerebellum is the newest portion of the cerebellum, on an evolutionary part with the cerebral neocortex. It receives input from all of the senses and is known to be necessary for the coordination of fine motor skills. This is the portion of the cerebellum that may also contribute to cognitive skills. According to [12] it coordinates the planning of limb movements. Lesions in the cerebrocerebellum have shown to produce delays in movement initiation, decreased muscle tone (hypotonia), decomposition of movement (dyssyner-

gia), distortion of muscular coordination especially in upper extremities (ataxia), inability to gauge distance of movements correctly, past-pointing (dysmetria), intention tremor or kinetic tremor, disorders in speech (dysarthria), and coordinated movement disorders, especially difficulties with rapidly alternating movements (dysdiadochokinesis).

**3.1.2. Cellular Organization.** The cerebellar cortex has an amazingly regular cellular structure, a structure which is similar for all vertebrates [13]. The cortex is divided into three layers, their name is determined by the cell bodies which each layer contains. The granule layer (200–300  $\mu\text{m}$  thick) contains mossy fiber axons and granule cells. The Purkinje cell layer (approximately 100  $\mu\text{m}$  thick) contains the Purkinje cells and the molecular layer (approximately 400  $\mu\text{m}$  thick) the basket and stellate cells as well as the axons of granule cells. The cerebellar cortex comprises the outermost 1 mm of the cerebellar surface. This surface is deeply folded, first into lobes, then smaller lobules, then sublobules, and finally into folia. All of the folds and resulting folia run horizontally, parallel to each other [17]; unfolded, the cortex is actually a strip of tissue measuring approximately 15 cm  $\times$  100 cm. This strip is deeply folded, only 15 percent of it are part of the surface. Although the cerebellar cortex contains only 10 percent of the brain by weight, its area ( $>100,000 \text{ mm}^2$ ) is a full three-quarters of the area of the cerebral cortex [18]. In addition, the cerebellar cortex contains nearly half of the total number of neurons in the brain [19]. The cerebellar cortex is composed of six types of neurons: Purkinje, granule, basket, stellate, Golgi, and Lugaro and three types of fibers: mossy, parallel and climbing fibers. These neurons are arranged in a uniform geometric structure not found exclusively in the cerebellum. Inputs to the cortex arise from the mossy fibers and climbing fibers which receive input from nuclei in the brain stem and spinal cord. The output of the cortex is provided solely by the Purkinje cells, which project on deep cerebellar nuclei, fastigial and vestibular nuclei (see Fig. 2).

**3.1.2.1. Mossy Fibers.** The primary input path to the cerebellar cortex are mossy fibers. They form axons from neurons in spinal and brain stem nuclei. Each mossy branches in a plane perpendicular to the long axis of the folium, yielding between two and 35 terminals or rosettes, with an average of about four. A glomerulus is composed of a rosette surrounded by

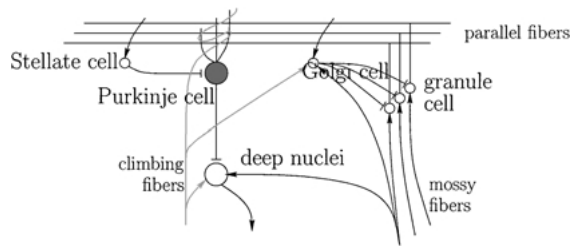


Figure 2. The cellular organization of the cerebellum with all major cells and fibers.

approximately 25 granule cell dendrites and a few Golgi cell axons. The mossy fibers form excitatory synapses with the granule and Golgi cell dendrites as well as with the deep nuclear neurons.

**3.1.2.2. Granule Cells.** The granule cells are the smallest (each cell has a diameter of 5–8  $\mu\text{m}$ ) and most frequent cells in the cerebellar cortex (there are about  $10^{10}$ – $10^{11}$  in the cerebellar cortex) [13]. Granule cells receive input from mossy fibers as well as from Golgi cells. Granule cells have 2–7 dendrites which terminate at separate glomeruli in the granular layer. The axon of a granule cell projects radially through the middle layer of the cortex into the molecular layer. In there it forms “T” parallel to the long axis of the folium: a parallel fiber. Each parallel fiber is 2 mm long and lies parallel to the others in a plane of the molecular layer. These fibers form excitatory synapses with Purkinje cell dendrites. The parallel fibers also form synapses with the dendrites of stellate, basket, and Golgi cells in the molecular layer. Granule cells are the only excitatory neurons inside the cerebellar cortex.

**3.1.2.3. Purkinje Cells.** Purkinje cells are the large (21–40  $\mu\text{m}$  long) planar output neurons of the cerebellar cortex [13]. All Purkinje cell bodies are inside of the Purkinje cell layer. In the human cerebellum there are approximately 15 million Purkinje cells, one for every 3000–9000 granule cells [13]. Purkinje cell axons have an inhibitory effect on the deep cerebellar nuclei. Each axon contacts about 35 nuclear cells. The axon collaterals also synapse on stellate, basket, Golgi, and other Purkinje cells. The dendrites of Purkinje cells extend outward into the molecular layer where they fan out perpendicular perpendicular to the parallel fibers. Although about 300,000 parallel fibers pass through the dendritic tree of a Purkinje cell, only 60,000 to 120,000 form synapses with this particular cell. Differently put: Each parallel fiber passes approximately 500 dendritic

planes though it forms only approximately 45 synapses. Purkinje cells generate complex spikes when activated by a climbing fiber and by parallel fibers as well as surrounding cells. It generates mostly simple spikes when activated only by parallel fibers, basket and stellate cells.

**3.1.2.4. Climbing Fibers.** Climbing fibers originate in the spinal and brain stem nuclei. Particularly in mammals, climbing fibers originate exclusively from the inferior olivary nuclei in the brain stem and are unique to the cerebellum. Every Purkinje cell has only a single climbing fiber which branches out into its dendritic structure. These fibers have a powerful direct excitatory effect on the Purkinje cells. Due to the one-to-one mapping of Purkinje cells to climbing fibers, these fibers are likely candidates for mediation and adaption of synaptic plasticity. Climbing fibers are also often associated with long-term changes in the cerebellum because their spontaneous activity is at the low rate of 1–3 Hz [13], which is too slow for any contribution to motor control. Each climbing fiber contacts about 15 Purkinje cells and excites Golgi and Lugaro cell as well as to granule cells.

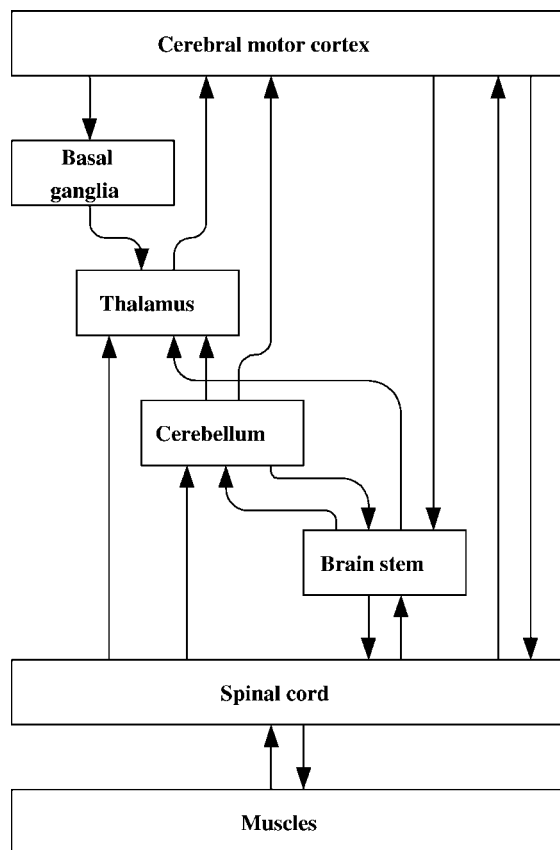
**3.1.2.5. Golgi Cells.** Golgi cells are interneurons which are in the granular layer. They receive input from parallel fibers, climbing fibers, and Purkinje cell collaterals. They send inhibitory signals to thousands of glomeruli. The general view is that Golgi cells serve as a feedback loop to modulate granule cell excitation. It is presumed that they also have a role in restructuring the synapses between mossy fibers and granule dendrites [20].

**3.1.2.6. Basket and Stellate Cells.** They receive input from Purkinje collaterals along horizontal dendrites, and they have axons which extend mostly horizontally in the molecular layer. Basket and stellate cells have similar forms and it is assumed that also their function is similar. They are inhibitory interneurons in the molecular layer with input from climbing fibers, parallel fibers, and Purkinje collaterals. Axons synapse on Purkinje dendrites. While stellate cells can be found in the cerebelli of all species, basket cells are exclusively in cerebelli of birds and mammals. Their name originates from the fact that they form basket-like structures of 50 axons around Purkinje cell bodies. All baskets are lined-up perpendicular to the parallel fibers.

**3.1.2.7. Lugaro Cells.** Lugaro cells are found sparsely in the granular layer, near the middle layer. They receive input from Purkinje collaterals along horizontal dendrites and have axons which extend mostly horizontally in the molecular layer. It is currently assumed that they do not have an important function in the cerebellum.

### 3.2. *The Cerebellum in the Central Nervous System*

The central nervous system contains various subsystems specialized to different tasks. Unfortunately, the knowledge on interaction and function of those modules is limited, and even the connections between those modules is only partially understood. In this section, we will try to give an overview using largely accepted hypotheses. Involved in motor control are the following areas: the (1) cerebral motor cortex, the (2) basal ganglia, the (3) thalamus, the (4) cerebellum, the (5) brain stem, and the (6) spinal cord. The connections are shown in Fig. 3.



*Figure 3.* The embedding of the cerebellum in the central nervous system.

The cerebellum is connected only to (1), (3), (5) and (6). These areas will be reviewed in the following subsections:

**3.2.1. Cerebral Motor Cortex.** In the control hierarchy of the central nervous system, the motor areas of the cerebral cortex are at the highest level. The structure and function of this cortex are still a mystery, but lesions in this area give an impression of its role. As shown in [21], removing the cat cerebral cortex does not interfere with basic abilities like walking and eating. These functions are controlled at a lower level. Still, it lacks in the purposefulness in its movement and will sit still for hours. Of course this effect varies in different live forms. As encephalization takes place, in more intelligent life forms, more functions are subsumed in the cerebral motor cortex. A human being for example would not be able to cope with the situation if his motor cortex would be decorticated.

**3.2.2. Thalamus.** The thalamus is often referred to as the gateway to the brain [22]. It is a structure deep within the brain that acts as a relay station for incoming stimuli from the senses. Sensory stimuli pass through the thalamus and are relayed to the cortex for further processing. The cortex sends signals back to the thalamus for further processing.

The thalamus serves as hub, sending signals to every part of the cortex at the same time. It is responsible for supplying information to the cerebral motor cortex [23]. It plays a role in helping us decide what to pay attention to and it is also largely accepted that it plays a role in cognition [22].

The cerebellum does not receive input from the thalamus but projects out on it.

**3.2.3. Brain Stem.** The brain stem consists of a large column of nerve fibers that connects the brain and the spinal cord. The brain stem receives incoming sensory information and regulates the degree of attention payed to this information by sending it to the right points. The brain stem also governs the body's involuntary activities, such as breathing and heartbeat control [22]. The cerebellum receives input from the brain stem and sends output to it in return.

**3.2.4. Spinal Cord.** The spinal cord is the main communication module between the brain and the mammal body. Almost all peripheral sensory and motor nerves connect to the spinal cord at some point. Sensory nerves enter the spinal cord through the sensory roots.



Motor nerves to the muscles originate from spinal cord neurons and leave through the motor roots.

The spinal cord contains two important reflexes:

- *Stretch reflex*: This reflex opposes the stretching of a muscle beyond its present length, any movement of the joint will be opposed. It assures the stability of the motor control system.
- *Tendon reflex*: The tendon reflex is a feedback circuit that can maintain muscle tension at a certain value. The set-point for this reflex is controlled by higher brain centers, e.g. the cerebellum. This is useful for tasks that need a constant muscle force, not a constant muscle length.

### 3.3. Cerebellar Functions in Motor Control

After gaining an insight into the interesting cerebellar physiology as well as its embedding in the central nervous system, we will now discuss its functions in motor control. The motor control functions related to the cerebellum can be divided into three categories [13]: the (a) elementary reflexes, the (b) posture and locomotion, and the (c) voluntary movements which will be discussed in the following paragraphs.

**3.3.1. Elementary Reflexes.** The fastest responses in the central nervous system are the elementary reflexes. The most basic reflexes take place in the spinal cord and interact with other modules in central nervous system. Some seem to be strongly influenced by the cerebellum. Two reflexes for which the hypotheses that they are implemented by or interact with the cerebellum is largely accepted are the vestibulo-ocular reflex (VOR) and the optokinetic response (OKR) [13]. Still, there are more ocular reflexes which interact with these two and therefore also have to be influenced by the cerebellum. Also some basic reflexes for different functions are presumed to be modulated by the cerebellum.

**3.3.1.1. Vestibulo-Ocular Reflex (VOR).** The VOR produces eye movements compensatory for head movements, preventing images of the visual world from moving across the retina during head movement. The VOR thus allows a human to fixate the environment while his head is moving. According to [13] the adaptability of VOR due to change in body and environment can be shown. From lesions in the vestibulo-cerebellum it can be shown that this adaptability takes place in this area. Electro-physiological exper-

iments have confirmed that several Purkinje cells modulate phase relationship and head velocity [13]. These Purkinje cells can be divided into two groups: the in-phase-modulating cells depress the VOR while the out-phase-modulating ones enhance the VOR. The VOR can be explained using the sidepath model which will be discussed later.

**3.3.1.2. Optokinetic Response (OKR).** The OKR is an eye movement which follows a relatively small movement of the visual surroundings. Since this might be caused by head movements, there is a strong interrelationship between OKR and VOR. Under stationary visual surroundings, the OKR stabilizes the eye position which would otherwise be continuously drifting as can be observed in darkness. The OKR produces eye movements in response to the visual stimulus movement. Lesions show that the cerebellum is responsible for this interaction. Unlike the VOR, the control of the OKR cannot be achieved by the open-loop sidepath model, but a closed loop model [13].

**3.3.1.3. Interacting Ocular Reflexes.** There are several more ocular reflexes which act in concert with the VOR and OKR toward the stabilization of retinal images. These are not as well studied, but there is evidence that the cerebellum is also involved in their control. The two most prominent of these reflexes are:

- The *Cervico-ocular reflex (COR)* induces eye movement modifications due to neck torsion. Its adaptability and interaction with the VOR has been demonstrated [13].
- *Trunk-ocular reflex (TOR)* causes eye movements to stabilize retinal images during movements of lower and upper trunk.

**3.3.1.4. Other Reflexes.** In addition to the ocular reflexes, postural reflexes, and scratch reflex shown in previous paragraphs and later sections, there are a number of somatic and autonomic reflexes in which the cerebellum has been suggested to play a role [13]. These include the *pupillary light reflex*, the *lens accommodation reflex*, *conditioned reflexes*, *cardiovascular reflexes*, *respiratory reflexes*, and *acoustic and startle reflexes*.

**3.3.2. Posture and Locomotion.** Important complex reflexes are controlling *posture*, *scratching*, *stepping*, and *locomotion*.

**3.3.2.1. Posture.** The postural state is held due to the concerted activity of numerous reflexes, the so-called postural reflexes. These can be divided into three categories: the (a) local static reactions acting on individual limbs, the (b) segmental static reactions linking one extremity to the opposite extremity, the (c) general static reactions arising from the head in space which affects the overall stance of the mammal.

The local static reactions include the stretch reflex and the positive posture reaction. As we have explained before, the stretch reflex is located in the spinal cord. Still, there are indications which point out, that the cerebellum might change the dynamic characteristics of the stretch reflex and might stabilize it to prevent oscillations.

**3.3.2.2. Scratching.** When the skin is irritated at a point of the body surface, the animal *scratches* that point rhythmically with its foot. It is largely accepted that this movement is a spinal reflex due to transections at the upper cervical segments. The cerebellum is not involved in generation of the rhythms, but it appears to play roles in adapting motor systems to the rhythmic inputs and in coordinating postural reflexes to the generated rhythmic movements. Ito [13] suggests that the rhythm generator in the spinal cord sends messages to the cerebellum to activate it. Subsequently, the cerebellum adapts motor commands on the rhythm generated by the spinal cord. Therefore the cerebellum serves as a modifiable element in the scratch reflex according to the sidepath model. Still, it is suggested that the cerebellum has even more roles in this reflex: modifying relative duration of the flexion and extension phases, the velocity of limb movements and most important, the spatial organization for transmission of rhythm generator to the segmental output neuron pool which is important for accurate performance of the scratch reflex. The cerebellum modifies the activity of limb muscles using spatial information about the limb position and movement and also about the irritated skin spot.

**3.3.2.3. Locomotion.** Another reflex which is strongly related to the scratching reflex is the *stepping reflex* or *locomotion*. Hindlimbs of animals perform locomotion on a treadmill even if the spinal cord is transected in the middle. Therefore it is generally assumed that the stepping rhythm generation takes place in the spinal cord. Various spinal models for this effect have been proposed and verified by simulation.

Still, by activity studies of cerebellar neurons, it was shown that the cerebellum modulates the stepping rhythm generation and that it is activated by peripheral signals. However, it has been shown that decerebellate cats still produce locomotion, therefore the cerebellum is not essential.

Roles of the cerebellum in locomotion may be multifold. First, the cerebellum might contribute to the adjustment of the timing of locomotive movements of limbs and therefore for its accuracy. A second role of the cerebellum would be interlimb coordination during locomotion. Finally, the cerebellum may be responsible for adjusting various reflex activities to be compatible with locomotion.

**3.3.3. Voluntary Movements.** In the previous subsections, we have studied reflexes which apparently are influenced by the cerebellum. We will now discuss cerebellar mechanisms for voluntary motor control. Voluntary movements are movements which are initiated or mediated from the cerebral cortex. In this area there is still a wide gap between knowledge on neural circuitry and events in the cerebellum. To date, two types of voluntary movements have been studied, *voluntary eye movement* and *voluntary limb control*.

**3.3.3.1. Voluntary Eye Movements.** There are two types of voluntary eye movements, *saccade* and *smooth pursuit eye movements*. A *saccade* is a quick jerky eye movement which positions a visual target on a region of the retina specialized for high acuity. Some saccade movements are triggered by the visual target and therefore reflexive while others are conditioned by rewards. The saccade generator is located in the brain stem and it is operated by signals from the brain stem visual system, superior colliculus, and the cerebral cortex. The cerebellum plays a role in association with functions of those structures: the brain stem visual system provides the visual data, the cerebral cortex activates saccade generation, and the superior colliculus is assumed to contain preprogrammed saccades for certain distances and directions. Once the saccade is triggered, the eye accelerates rapidly to the maximum velocity, exceeding 500°/sec at mid-trajectory, and then decelerates to bring the fovea accurately to the target. This quick movement does not allow the oculomotor system to utilize visual feedback accurately locating a visual target in the central retinal region. In this sense, a saccade must be

guided by a mechanism predicting its goal on the basis of information available before initiation of an eye movement. The predictive control for a saccade is of the same nature as feedforward control of the VOR in the sense that both must operate without feedback. This explanation from the control point of view as well as lesion experiments show the involvement of the cerebellum, it seems that without a cerebellum, no precise eye movements are possible.

*Smooth pursuit eye movements* continuously follow a moving target with a high acuity velocity of the target. As smooth pursuit eye movements are voluntary, it is assumed that the central command for this movement arises in the cerebral cortex. According to [13], smooth pursuit eye movements is related to the cerebrotocerebellar communication rather than to cerebellar sidepaths for reflexive pathways. It has been shown, that the activities of Purkinje cells interacting in smooth pursuit eye movements are extracting information from visual, oculomotor and vestibular signals. These signals convey information about target velocity relative to eye velocity, eye velocity relative to head velocity and head velocity in space. There is also the possibility, that the cerebellum interacts at different stages of the information processing, e.g. for avoiding overshoot due to activation of VOR, OKR and COR.

**3.3.3.2. Voluntary Limb Movements.** Considering the influence of the cerebellum in voluntary limb control, the studies of Gordon Holmes [15] are very interesting: Holmes carefully catalogued the effects of gunshot injuries in the cerebellar region. In his works he clearly shows that it is responsible for fine limb motion control. He gives several comparisons which show the effects of cerebellar lesions, and according to him, the cerebellum is responsible for subconscious trajectory generation and execution. He gives an interesting example: One of his patients who had a lesion in the right cerebellar hemisphere of the cerebrotocerebellum mentioned: "The movements of my left arm are done subconsciously, but I have to think out each movement of the right [affected] arm. I come to a dead stop in turning and have to think before I start again" [15]. Eccles et al. [24] has proposed that the cerebellum spares us this mental task: a general command can be given by the cerebral cortex, that leaves the specific details of the execution of movement to be carried out by subcortical, notably cerebellar, mechanisms.

### 3.4. Complexity of the Control Situation for the Cerebellum

The control situation in the human body is more complex than in robot system due to two facts: first, the human joints are very difficult to control due to the various anatomic structures which interact in any movement; second, communication in the central nervous system is considerably slower than in technical control systems.

**3.4.1. The Human Mechanism.** The human body is a framework of jointed bones and other segments. The human musculoskeletal system contains more than 600 muscles and 206 bones in a mature body. Actuation of the skeleton comes from large bundles of muscle fibers connected via a seamless transition of tendons and ligaments. Altogether 37 degrees of freedom are important for the human body, though it contains even more degrees of freedom due to flexibilities and degrees of freedom having minor importance [25].

Human joints are very complex and up to now not fully understood. Many anatomic structures around the joint contribute either actively (i.e., muscle contractile elements) or passively (i.e., capsules, ligaments, tendons, tissues, non-contractile portions of muscles) to the intersegmental resultant forces or moments [26]. Various active muscle elements interact in a simple joint movement, e.g., in the human knee altogether 47 muscle elements are involved in movements [27]. The active as well as passive muscle elements are very complex: They have a hysteresis which means that the muscle force generated for the same muscle extension and applied external force is different depending on the direction of movement [28]; the muscle stiffnesses are time-varying, highly nonlinear, complex curves [29]. Muscle length, external forces and forces generated by the muscle as well as contraction or extension velocity form a three dimensional, highly nonlinear relationship [30].

**3.4.2. Communication.** Another disadvantage for a controller of the human "robot" system are the delays between action and reaction. While a robot controller one can operate in few milliseconds using sensory and modern information processing, the biological controller cerebellum cannot react that fast.

The pathway from brain to the hand has a delay of up to 60 ms [31]. For the toe, another 20 ms have to be added. Although some information processing already occurs in the spinal cord, the human controller

has to use substantial predictive control, compared to a technical system. To demonstrate how efficient a biological control system has to be, we evaluate a simple example. Assume our plant is a first-order system with a time constant  $T = 60$  ms (which is approximately the latency time constant of skeletal muscles [32], although the muscle plant is more complex since it is a nonlinear second or higher order system), our. The used controller is a PID controller and we assume communication delays  $T_t = 127$  ms (similar to biological proprioceptive lag [6]). According to Schmidt [33], the stability of such a system is only given if change in movement occurs at a frequency lower than  $f_{krit} = f(T_t/T)$ . For  $0 < T_t/T < 10$ , it can be approximated by  $f_{krit} \approx 1/(2T_t) \approx 4$  Hz. This is too slow for all practical purposes of limb movement. Therefore, there has to be a more advanced solution for motion control embedded in the cerebellum.

### 3.5. Models of Cerebellar Interaction in Motor Control

According to [13], there are three models of cerebellar interaction with extra-cerebellar systems: the (a) Sidepath model, the (b) Loop model, and the (c) Lateral interaction model. Each of these models resulted from the research of the described motor functions.

**3.5.1. Sidepath Model.** As simple example of the sidepath model is the previously described vestibulo-ocular reflex (VOR). The reflex is activated by a head movement, which activates the control line itself as well as the flocculus-cerebellum. The cerebellum generates an output, which is given together with the control line onto the vestibular nucleus. The output of this nucleus is now used by the motor control neurons for activation of the eye muscles. The retina error signal is propagated back through the inferior olive into the cerebellum.

Similar to this model of cerebellar interaction, we can derive the ones for tecto-reticulospinal pathway, cortico-rubrospinal pathway and interstitiospinal pathways. This model is stable for applications having huge delays since it does not include any feedback control circuits.

It can be schematized as in Fig. 4, where  $G_1$  defines the vestibular organ or the spinal cord depending on the controlled reflex.  $G_2$  is the controlled system while  $S$  is the symbol for the cerebellum. The system transfer function can be described by

$$T = (G_1 - S)G_2. \quad (13)$$

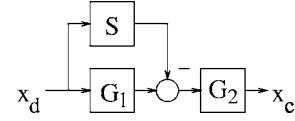


Figure 4. The sidepath model;  $G_1$  is the spinal cord controller,  $G_2$  is the controlled system, while  $S$  is the cerebellum.

Hence, if the transfer function of the sidepath  $S$  is changed, the one of the overall control system subsequently changes as well. Note, that if  $S = G_1$ ,  $T$  will be zero. Ito [13] points out, that  $T$  may have varied values for gating the output; by reducing  $S$  below  $G_1$ , it switches the output on. This gating model is supported by the observation that a bundle of mossy fibers afferents which excites a specific motor neuron group will also inhibit the group of motor neurons of the antagonist muscle.

Using the sidepath model different types of control can be achieved [34]: In the VOR all the reflex between head and eye motion is clearly straightforward. Therefore, the transfer function should optimally only adapt head motion on eye velocity, i.e.,  $T = 1$ . This leads to the understanding that  $(G_1 - S) = -1/G_2$ . The cerebellar sidepath and the cerebral motor cortex together form an inverse model of the eye muscular system, see Fig. 5. In voluntary movement, there is feedback inside the reflex. Therefore, the sidepath interacts from the outside into the reflex control loop.

Three types of control have been proposed as function of the cerebellum in the sidepath model: adaptive control, predictive control and multi-variable control. In adaptive control the cerebellum adjusts the characteristics of the system in response to changes of its environment, the VOR seems to be an example of this type of control. In predictive control, an appropriate future output is predicted from the current input, which takes place in saccadic eye movements.

The sidepath model has been extended even on the concept of “motor programming”. A motor program is defined as a set of muscle commands that are structured before a movement sequence before a movement sequence begins, and that allow the entire sequence to be carried out uninfluenced by peripheral feedback.

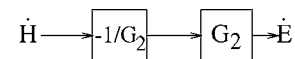


Figure 5. The sidepath model for the VOR circuit;  $G_1$  and  $S$  form an inverse system to the controlled object  $G_2$ .

There are sophisticated motor programs like playing piano as well as simple, specific ones for coordinated movement of eyes and head. It is likely that such motor programs are created in the cerebral cortex as a rough plan and subsequently refined by the cerebellum.

**3.5.2. Loop Model.** Several situations cannot be described using the sidepath-model, the most prominent are rhythmic discharges like in stepping or scratching and the OKR, but also voluntary movement is often considered to be integrated into a closed-loop system.

It is suggested that a closed-loop composed of the spinal ascending pathways, the cerebellum, and the brain stem descending pathways may be superimposed on a segmental motor neuron group [13]. A similar loop involves prospinal neuron groups in the cervical spinal cord which is also under influence of the cerebellum.

Closed feedback control loops consisting of a controlled object  $G$  and a controller are the common solution in traditional robot control. This type of loop is very improbable in cerebellar control since the delays in biological systems are large, gains are low, and as we have seen before such a systems becomes instable at higher velocities if it does not contain a perfect controller. Therefore [13] proposes that the cerebellum includes a model  $G'$  of the dynamic system  $G$  as well as a controller  $H'$  as shown in Fig. 6. The transfer function of this system can be described by

$$T = \frac{G}{(1 + G'H')} \quad (14)$$

and if  $G' = G$  and  $H' = H$ , it becomes equal to the transfer function of the engineering closed-loop (see Fig. 6). It can be hypothesized, that the cerebellum provides an internal loop which simulates the dynamics of a system in closed mode. According to [13], this model can be used for explaining the learning process of voluntary limb control. When performing an

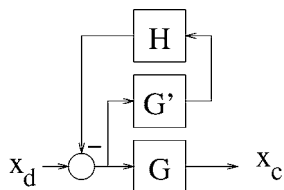


Figure 6. The loop model;  $G$  is the controlled system,  $G'$  stores a forward model of the controlled system, while  $H$  defines an inverse model.

unskilled voluntary movement, high errors will be produced. As experience is gained, the movements become more refined and as the learning process progresses, our cerebellum becomes similar to a sidepath controller which will provide a correct output at all times. Unfortunately, this hypothesis is difficult to test and therefore it is only a working model.

**3.5.3. Lateral Interaction Model.** Ito [13] points out that there is even a third possibility: a corticonuclear microcomplex could cross over two or more separate control systems, such that it can provide lateral interaction of influences from one control system to others. This model is not yet supported by neural studies, but it is plausible from the anatomical interconnection point of view. Lateral interaction could for example be used to prevent sensory input from interfering into voluntary movements. It does not imply improvements of dynamic characteristics, but could improve the control systems performance by adjusting its environmental conditions.

### 3.6. Theories of Internal Cerebellar Function

We will now discuss several *conceptual models of cerebellar function*. We will at first discuss what kind of a model the cerebellum might contain, and subsequently we will describe several approaches introduced in the last fifty years of cerebellar research.

**3.6.1. Classification of Cerebellar Models.** According to [35] cerebellar models can be classified into three groups: models which contain an inverse model of the mammal body dynamics; models which contain a forward model, i.e., a simulation of the controlled system, and combinations of both.

**3.6.1.1. Inverse Models.** An inverse model is given the desired state of the system and will subsequently produce a motor command, which brings the system in this state. It is often assumed that the cerebellum acquires an inverse model of the system; Wolpert et al. [36] give biological reasons for this assumptions. Still, there is reason to concern: Learning an inverse model is generally a difficult task due to the fact that a motor command error is not available. If the motor error could be obtained without long-term memories, an inverse model would not be necessary.

**3.6.1.2. Forward Model.** A forward dynamic model of a system predicts the next state while being given the current state as well as the last motor command. This of course means that there has to be a controller chained to this model which can generate a motor command using the prediction as well as the given desired state. According to [36] there is a variety of conceptual ideas which lead to the hypothesis that the cerebellum contains a forward model. One example is the Smith-predictor hypothesis described below.

**3.6.1.3. Paired Forward and Inverse Models.** In [36], the authors speculate that the cerebellum contains forward as well as inverse models. Unlike in the other two hypotheses, they assume that there is a huge amount of modules which contain inverse as well forward models. Those are considered to be interacting to generate behaviors. They give three reasons for this assumption: First, the control situations in the real world are essentially modular. The forward model would predict the next state and activate the responsible inverse model for each situation. Second, single modules can learn without de-learning already learned modules. Finally, situations which have not yet been encountered can be solved using the experience of several surrounding modules.

### **3.6.2. Conceptual Models of Cerebellar Function.**

We will now introduce a few important conceptual models on cerebellar function. Research in this direction started in the late 1950s with the *timing organ theory*. It subsequently got a big push due to the introduction of the *Marr-Albus theory*. In the last decades, other ideas like viewing the *cerebellum as a sensory organ* or the *microcomplex theory* have been introduced.

**3.6.2.1. The Timing Organ Theory.** The first mechanistic model to appear was proposed by Braitenberg and Atwood [37] and Braitenberg [38]. These authors interpreted the cerebellum as timing device, they argued that most coordinated movements actions require a fine timing and that the cerebellum would provide a common internal clock. Thus the parallel fibers were primarily seen as delay lines in their theory. As Arbib [39] points out, timing does not require a clock in the conventional sense. The timing organ theory is beyond the scope of this review but is treated in the paper by Rhodes and Bullock this issue.

**3.6.2.2. Marr-Albus Theory.** The other two early models by Marr [40] and Albus [41] view the cerebel-

lum as a context-driven learning pattern recognition system. They suggested that the parallel fibers only transmit information which is weighted by Purkinje cell-parallel fiber synapses and determines whether the Purkinje cell fires if the inputs are excited above a firing level. They described the Purkinje cell as a single learning device, i.e., a perceptron. It is a generator of elemental movement triggered by the parallel fiber input and learned due to climbing fiber error signals. Their models resulted into the computational cerebellar model CMAC.

**3.6.2.3. Microcomplex Theory.** The Marr-Albus theory focuses on the cerebellar cortex and associates each Purkinje cell with an elemental movement—it does not consider that the cerebellar cortex is part of a cerebellar system and that this system is part of the central nervous system. In the microcomplex theory, things are treated in a more global way. A small area of the cerebellar cortex receives input from climbing fibers from a small group of inferior olive neurons and in turn sends its output to a small group of deep nuclear neurons. It is assumed that this area is an integrated system, a *microzone* or cerebellar corticonuclear microcomplex [13].

**3.6.2.4. The Cerebellum as a Sensory Organ.** Paulin [42] challenged the view of the cerebellum as motor controller and coordinator assumed by the previous theories. By investigating a variety of species, he established the hypotheses that structure of cerebellum of a species is depends on the complexity of its sensory input. He concludes that the cerebellum stores a forward model which tracks sensory data and predicts a filtered current state of the system.

**3.6.2.5. The Cerebellum as a Sensory-Motor Organ.** Another view was proposed by Miall et al. [8]. These authors see the cerebellum as a sensory organ as well as a motor organ. They assume that the cerebellum acts as a Smith-Predictor, building forward models of the skeletomuscular system (cf. [43]). It is an example for the loop model, it contains a forward model which predicts current as well as future state variables which are not available due to the , thus overcoming long delays.

## **3.7. Theories on Cerebellar Learning**

The regular structure of the cerebellum has allowed for a detailed insight in its functionality, including the learning process. We know that the cerebellum learns

sequences of voluntary movements as well as motor programs, but also adapts to external influences. Still, there is a controversy on the role of the climbing fiber.

**3.7.1. The Climbing Fiber.** A largely accepted hypothesis on cerebellar learning was proposed in the Marr-Albus theory: parallel fiber synapses on Purkinje cells are weakened if the Purkinje cells fire when the climbing is active. This adaption process is called *long-term depression LTD*; if a similar activity of the parallel fibers occur, the firing probability of the Purkinje cell will be lower. In 1982, Ito et al. [44] verified this hypothesis by demonstrating that simultaneous stimulation of parallel fibers and climbing fibers results into LTD of their synapses.

However, if only LTD would take place, the responses of the Purkinje cells would eventually reduce to zero. Therefore LTD, which assumed to take place in many regions of the brain, is usually accompanied by an opposite process, *long-term potentiation LTP* [45]. Still, some authors believe that a characteristic of cerebellar learning is the absence of LTP [46], yet that it might have some other relaxation process.

In the Marr-Albus theory, the climbing fiber signal is interpreted as an error signal. This hypothesis is often used in computational cerebellar models, but not generally accepted by biological researchers. According to [47] climbing fiber signals do have several properties which cannot be explained by the error signal hypotheses and [48] note that the function of the climbing fiber is still a mystery. Ebner and Bloedel [49] have done experiments which showed that some Purkinje cell synapses were excited or inhibited randomly in response to climbing fiber input. Some authors also believe that the climbing fiber serves as an internal timing signal. Still, other experiments show that there is a correlation between motor error and climbing fiber activity [50, 51].

Despite those uncertainties, three facts on cerebellar learning are largely accepted:

- The climbing fibers serve as a teaching input.
- Learning in the cerebellum takes place by simultaneous activation of parallel and climbing fibers at the same Purkinje cell.
- The cerebellum learns models of the skeletomuscular system [8, 52].

**3.7.2. Learning Architectures of the Cerebellum.** In the previous section, we have discussed the function of the climbing fiber, but not how it actually achieves its

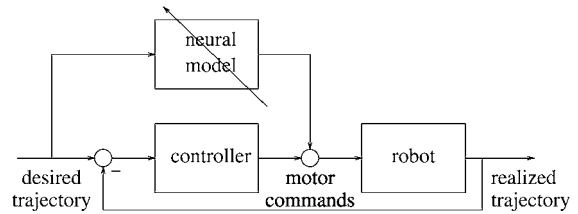


Figure 7. Feedback-error learning. A feedback controller achieves stability while the cerebellum (or other neural model) generates a feedforward motor command. The output of the feedback-error signal serves as a training signal which will converge to zero, if the neural model is learning correctly.

role. For this aspects, two learning-architectures have been proposed, feedback-error learning and central pattern learning.

**3.7.2.1. Feedback-Error Learning.** Feedback-error learning architecture is inspired from the sidepath model [53, 54]. It is assumed that the spinal cord performs feedback control on the muscles and that the cerebellum stores a predictive inverse model which generates the feedforward motor commands. The learning of the cerebellum arises due to the motor torque generated by the feedback controller (see Fig. 7).

Although feedback-error-learning can be proven to be globally stable, this must not be the case locally. Due to internal dynamics it can become locally instable and the learning scheme often takes the neural model away from a close to optimal solution and it will take a large amount of samples to reach this point again. A simple example for this is a piecewise linear system with a large stiction at zero velocity. It can be approximated fast using supervised learning, but will diverge frequently if feedback error learning is applied. It will even de-learn a close to perfect approximation. The second important point against applying this learning scheme is the difficulty to understand what the neural model does when it is not converging. We do not always understand whether our architecture is wrong, we do have numerical errors or whether it was due to bad initial parameters.

**3.7.2.2. Central Pattern Learning.** Houk [55] assumes that the cerebellum is not directly embedded in the control loop. It is assumed to tune the red nucleus which is assumed to be central pattern generator and generates the motor commands. Learning is viewed as distributed in many sites, not just in the cerebellum, but also in the entire premotor networks. The cerebellum is only seen as a setter of equilibrium points—by

setting the equilibrium point of a system to a different position, it generates a movement which is controlled without cerebellar interaction.

#### 4. Computational Models of the Cerebellum

Since the development of the Marr-Albus model and the computational Model CMAC, there have been many more cerebellar models. In, e.g. [56] and [57] an overview of many of those models can be found. Unfortunately, several of those models are not *computational* models but *conceptual* ones. In order to obtain an overview on scalar cerebellar control, we will review: the (1) Cerebellar Model Articulation Controller CMAC [41], the (2) Adjustable Pattern Generator APG [58], and the (3) “Fairly Obvious Extension” FOX. Other models, such as Kawato’s Internal Models [52] and Schweighofer’s Limb Movement Models [60] are not taken into account in this discourse.

Each model will be described in respect to at least three aspects: How was it obtained from cerebellar physiology? What is being computed? How does it learn?

##### 4.1. The Cerebellar Model Articulation Controller CMAC

One of the first well-known computational models of the cerebellum is the *Cerebellar Model Arithmetic Controller CMAC* introduced by Albus [41]. The algorithm was based on Albus’ understanding of the cerebellum, but it was not proposed as a biologically plausible model [61].

Having been successfully used in various robot control tasks, CMAC has received ample attention in the computing literature. Miller et al. [62] and Lang [63] and many other authors have published advanced CMAC applications, such as control of industrial robot arms [63], hand-eye-systems [64], and biped walking [65]. Being extensively mathematically analyzed [62], it has become well accepted by control engineers.

##### 4.1.1. From Cerebellar Anatomy to CMAC Structure.

The idea has its origins in the BOXES approach proposed by Michie and Chambers [66]. BOXES is the simplest form of function approximation. For  $n$  variables there is an  $n$ -dimensional hypercube where the function values are being stored. Basically it is just a table lookup process. BOXES suffers from the curse of dimensionality [67]: if each variable can

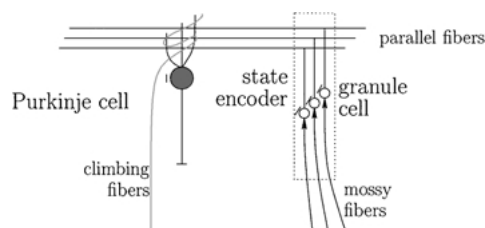


Figure 8. The Cerebellar Model Articulation Controller CMAC mapped on the cells of the cerebellum. It consists out of a single Purkinje cell as well as a state encoder having a huge amount of granule cells.

be discretized into  $D$  different steps, the hypercube has to store  $D^n$  function values. For simplification, Albus assumed that the output of the mossy fiber is the discretized input state of the controller. Therefore the granule cells just map this discretized input onto several granule cells which are modelled as on-off switches. If a granule cell is on, it fires on its parallel fiber. The process up to this point happens in the state encoder. Since several granule cells fire at the same time, the generalization of the CMAC network takes place in this state encoder (Fig. 8).

The learning signal is provided by the climbing fibers; in this model the error to the desired value given by a supervisor performs as a learning signal [68].

**4.1.2. Computation and Learning in CMAC.** The operation of the Albus CMAC can most easily be described in terms of a large set of overlapping, multi-dimensional receptive fields with finite boundaries. Any input vector falls within the range of some of the local receptive fields and outside of the range of the other receptive fields. The response of the CMAC neural network to a given input is determined by the average of the responses of the receptive fields excited by that input, and is not affected by the other receptive fields. Similarly, neural network training for a given input vector affects the adjustable parameters of the excited receptive fields, but does not those of the other receptive fields.

The organization of the receptive fields of a typical Albus CMAC neural network with a two-dimensional input space is as follows. The total collection of receptive fields is divided into  $C$  subsets, referred to as “layers” (the layers represent parallel  $n$ -dimensional hyperspaces for a network with  $n$  inputs). The receptive fields in each of the layers have rectangular boundaries and are organized so as to span the input space with overlap. Any input vector excites one receptive field



from each layer, for a total of  $C$  excited receptive fields for any input. Each of the receptive fields layers is identical in organization, but each layer is offset relative to the others in the input hyperspace. The overlap of the receptive fields produces input *generalization*, while the offset of the adjacent layers of receptive fields produces input *quantization*. The ratio of the width of each receptive field (input generalization) to the offset between adjacent layers of receptive fields (input quantization) must be equal to  $C$  for all dimensions of the input space. The integer parameter  $C$  is referred to as the generalization parameter. This organization of the receptive fields guarantees that only a fixed number  $C$  of receptive fields is excited by any input. However, the total number of receptive fields required to span the input space can still be large for many practical problems [61].

Each receptive field in the Albus CMAC is assumed to be an on-off type of entity. If a receptive field is excited, its response is equal to the magnitude of a single adjustable weight specific to that receptive field. If a receptive field is not excited, its response is zero. The CMAC output is thus the average of the adjustable weights of the excited receptive fields. If nearby points in the input space excite the same receptive fields, they produce the same output value.

The output only changes when the input crosses one of the receptive field boundaries. The Albus CMAC neural network thus produces piece-wise constant outputs. The implementation of the Albus CMAC proceeds as follows:

**4.1.2.1. Identify the  $C$  Receptive Fields/Granule Cells Excited by the Input.** The first step is to normalize and quantize the input vector  $\mathbf{S} = (s_1, s_2, \dots, s_n)^T$  to

$$\mathbf{S}' = \left( \left[ \frac{s_1 - s_{1,\min}}{s_{1,\max} - s_{1,\min}} \right], \left[ \frac{s_2 - s_{2,\min}}{s_{2,\max} - s_{2,\min}} \right], \dots, \left[ \frac{s_n - s_{n,\min}}{s_{n,\max} - s_{n,\min}} \right] \right)^T \quad (15)$$

using the maximal and minimal values of  $s_i$ . The next step is to form the vector addresses  $\mathbf{A}_i$  of the  $C$  receptive fields which contain input point  $\mathbf{S}'$  (i.e., the names of the  $C$  activated granule cells):

$$\mathbf{A}_i = \begin{bmatrix} a_{i1} \\ a_{i2} \\ \vdots \\ a_{iC} \end{bmatrix} = \begin{bmatrix} s'_1 - (s'_1 - i) \mod C \\ s'_2 - (s'_2 - i) \mod C \\ \vdots \\ s'_C - (s'_C - i) \mod C \end{bmatrix} \quad (16)$$

for  $i = 1, 2, \dots, C$ . Since the number of receptive fields can be quite large, they are hashed onto the granule cells:

$$a'_i = \left( \sum_{j=1}^C T_{j,(a_{ij} \mod R_j)} \right) \mod M. \quad (17)$$

$T_j$  represents a table of uniformly distributed random values with  $R_j$  total table entries.  $M$  is the number of granule cells available.

**4.1.2.2. Compute the Output.** Using the  $a'_i$  it is easy to find the adjustable weights for those activated  $C$  receptive fields in a pool of stored weights. The output of the CMAC network is the average of the  $C$  adjustable weights:

$$y(\mathbf{S}) = \frac{1}{C} \sum_{j=1}^C W_{a'_j} \quad (18)$$

**4.1.2.3. Learn.** CMAC is trained by a supervised training using the Widrow-Hoff-Rule. The trainer provides inputs  $\mathbf{S}$  and desired outputs  $y_d(\mathbf{S})$ . The weight adjustment  $\Delta W$  is given by

$$\Delta W = \beta(y_d(\mathbf{S}) - y(\mathbf{S})) \quad (19)$$

where  $\beta$  is the learning rate for the CMAC network.

## 4.2. Improvements and Extensions of CMAC

The CMAC has been thoroughly investigated in various publications. There have been many contributions to the improvement of the CMAC algorithm, which include the (1) improvement of organization of receptive fields [69], (2) receptive field sensitivity functions [68], (3) collision resistance for hashing [68], (4) optimal weight smoothing [70], and (5) weight magnitude normalization [68].

### 4.2.1. Improvement of Organization of Receptive Fields.

The receptive fields are hypercubes of side  $C$  in the normalized input space. All  $C$  receptive fields are located inside any region of the hypercube. They are optimally organized if the  $C$  receptive fields are uniformly distributed. Any algorithm which satisfies this criteria qualifies as a CMAC mapping. Unfortunately the Albus-CMAC mapping is non-uniformly distributed. Parks and Militzer [69] and Miller and Glanz [68] researched to improve this problem. While [69] solve this

problem by exhaustive search, [68] have found practical rules as fast solution. They define a displacement vector

$$\mathbf{D} = (d_1, d_2, \dots, d_n)^T \quad (20)$$

and obtain the address in the hypercube by

$$\mathbf{A}_i = \begin{bmatrix} a_{i1} \\ a_{i2} \\ \vdots \\ a_{iC} \end{bmatrix} = \begin{bmatrix} s'_1 - (s'_1 - id_1) \pmod{C} \\ s'_2 - (s'_2 - id_2) \pmod{C} \\ \vdots \\ s'_C - (s'_C - id_n) \pmod{C} \end{bmatrix} \quad (21)$$

for  $i = 1, 2, \dots, C$ . In these terms, the conventional CMAC would be defined by a lattice displacement vector of  $\mathbf{D}_{\text{Albus}} = (1, 1, \dots, 1)^T$ . Miller and Glanz [68] suggest that one should select  $\mathbf{D}_{\text{Miller}} = (1, 3, 5, \dots, 2n + 1)^T$  as displacement vector.

**4.2.2. Receptive Field Sensitivity Functions.** Miller & Glanz [68] have also investigated CMAC networks with graded receptive field sensitivity functions instead of the all-or-none of Albus' CMAC. The CMAC output is influenced more by receptive fields for which the input vector is near the center of the active range, and less by receptive fields for which the input is near the borders of the active field. If one maps this function on the cerebellum this would mean, that the firing frequency of the granule cells is not only on-off but varies depending on the input. The CMAC computations and learning have to be modified for this step. First a normalized, non-discretized variation of the input vector is created:

$$\mathbf{S}' = \left( \frac{s_1 - s_{1,\min}}{s_{1,\max} - s_{1,\min}}, \frac{s_2 - s_{2,\min}}{s_{2,\max} - s_{2,\min}}, \dots, \frac{s_n - s_{n,\min}}{s_{n,\max} - s_{n,\min}} \right)^T. \quad (22)$$

For any arbitrary input point in the normalized space, the minimum distance  $\delta_i$  to any face of the receptive field  $i$  is then given by

$$\delta_i = \min(s''_i - a_{ij} + 0.5, C - s''_i + a_{ij} - 0.5) \quad (23)$$

for  $j = 1, 2, \dots, n$ . In this case,  $\delta_i = 0$  corresponds to any point on a face of the receptive field, while  $\delta_i = C/2$  corresponds to the single point in the center of the receptive field.  $\delta_i$  Varies linearly along any linear path

from any point on a face of the receptive field to the point of the center. Now a sensitivity function is defined by  $f(\delta_i) = \delta_i$  which is good enough for all practical cases [68]. We now have a new computation function

$$y(\mathbf{S}) = \frac{\sum_{i=1}^C \delta_i W_{A'_i}}{\sum_{i=1}^C \delta_i} \quad (24)$$

and a new learning rule

$$\Delta W = \beta[y_d(\mathbf{S}) - y(\mathbf{S})] \delta_i \frac{\sum_{j=1}^C \delta_j}{\sum_{j=1}^C \delta_j^2} \quad (25)$$

**4.2.3. Collision Resistance for Hashing.** If  $a'_n = a'_m$  occurs for  $\mathbf{A}_n \neq \mathbf{A}_m$ , we have encountered a hashing collision. This means that two different points in the generalized input space are hashed on the same point. It has the effect that training adjustments to two distinct and possibly distant receptive fields affect the same adjustable weights. In Albus' CMAC hashing collisions are ignored. According to [68] this is often reasonable, since the CMAC outputs are averages over several receptive fields (there is no specific desired response for any single receptive field), and since the goal is often to *approximate* a target function rather than to reproduce it exactly.

Hashing collisions slow down training convergence but do usually allow us to find a satisfactory solution. Of course, if hashing collisions are too frequent, this statement does not hold. The probability of having no hashing collisions when training a single pair of data approximately equals

$$P_{\text{nocollision}} = \left(1 - \frac{M_u}{M}\right)^C \quad (26)$$

where  $M_u/M$  represents the fraction of the available weight storage that has been affected by previous training. Due to exponential growth for higher values of the generalization parameter  $C$ , the probability of collision-free training is low, unless the utilization of available storage is very low.

Collision-free hashing generally involves storing the identifier of the receptive field, the address  $A_i$ , along with each adjustable weight or weight vector, so that collisions can be detected and avoided. Unfortunately, this means that there is a huge amount of storage required for this method which would make the hashing itself obsolete. Collision-resistant hashing can provide

a compromise by storing a pseudo-random hash tag, derived from the receptive field virtual address  $A_i$ , along with each adjustable weight. Collisions can then be detected and avoided reliably but not certainly by comparing the stored hash tag with the value derived from the new address. If the tags do not match, the CMAC weight memory is searched sequentially until either a tag match or an unallocated location (blank tag) is found. The tags are again generated using the a random number table. According to [68] the probability is now approximately

$$P_{\text{nocollision}} = \left(1 - \frac{M_u}{kM}\right)^C \quad (27)$$

where  $k$  represents the dynamic range of the hash tag. Already for  $k=2$ , the probability of not having any collision at all is significantly lower. For large values of  $k$  the exponential growth with  $C$  is not such a big problem.

#### 4.3. The Adjustable Pattern Generator APG

The *Adjustable Pattern Generator model* APG was introduced by Houk and others in [55, 58, 71–77] and several other publications. The term “Adjustable Pattern Generator” was coined since the model can generate a burst command with adjustable intensity and duration. The APG has only been applied in a few control situations, e.g., a single muscle system or two-link robot arm having six muscles.

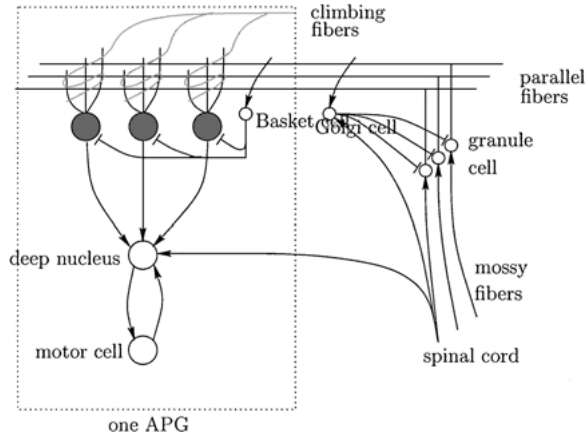


Figure 9. A state-encoder and a single adjustable pattern generator APG mapped on the cells of the cerebellum.

#### 4.3.1. From Cerebellar Anatomy to APG Structure.

The APG is based on the same understanding of the mossy fiber-granule cell-parallel fiber structure as CMAC. In all of the above mentioned papers, the authors use the CMAC state encoder as described by Albus.

Still the learning module, the APG, is really different from CMAC and in some regards much more like the cerebellum. In the APG a nucleus cell, connected to a motor cell in a feedback circuit, is inhibited by a group of Purkinje cells. The synapses of those Purkinje cells learn by a reinforcement given by one climbing fiber for each APG. The climbing fiber gives an external error signal (Fig. 9).

#### 4.3.2. Computation and learning in APG

**4.3.2.1. Compute the Output.** In the primate brain, mossy fibers (MF) carry plant state, motor efference, and other contextual signals into cerebellum [74]. These fibers impinge on granule cells, whose axons excite the parallel fibers (PF). For this encoding the CMAC state encoder is used by Houk et al. The improved version of the state encoder described in the CMAC section can also be used for APG. This allows the APG to work better than in its original version. The large number of parallel fibers converge on a much smaller set of Purkinje cells (PC). The Purkinje cell activation is determined by the activated parallel fibers and the weight of the connecting synapse. This process is described by

$$PC_j = g\left(\sum PF_i w_{ij}\right) \quad (28)$$

where  $PC_j$  is the cells firing rate and  $w_{ij}$  is the changing strength of the synapse from  $PF_i$  to  $PC_j$ .  $g(\cdot)$  is defined as a threshold-ramp function:

$$g(x) = \begin{cases} 0 & x < \lambda \\ \rho + (\bar{\rho} - \rho) \frac{x - \lambda}{\bar{\lambda} - \lambda} & \lambda \leq x \leq \bar{\lambda} \\ \bar{\rho} & \text{otherwise.} \end{cases} \quad (29)$$

$\rho$  is the minimum value of the Purkinje cell if is activated,  $\bar{\rho}$  the maximum value.  $\lambda$  and  $\bar{\lambda}$  are the lower and upper thresholds of the ramp function.

A set of Purkinje cells, in turn, provide inhibitory signals to a single cerebellar nuclear cell. It is the activation of the nuclear cell that determines the output of the APG. The control signal output by a single APG is

computed as

$$u_C = 1 - \frac{1}{|L|} \sum_{j \in L} PC_j \quad (30)$$

Hereby  $L$  denominates the set of Purkinje cells in APG.

**4.3.2.2. Learning.** When the cerebellar module is active, its output is evaluated by a quality function which is the input of the climbing fiber CF. The learning algorithm must determine which of the PF-to-PC synapses must be updated in order to improve the cerebellum's movement generation performance. This is called the *credit assignment problem*. It consists of two components. The *structural credit assignment problem* asks which of the synapses must be updated in order to improve the cerebellum's movement generation performance. The *temporal credit assignment problem* deals with the significant delays between the time the cerebellum generates a motor signal and the time that the cerebellum generates a motor signal, and the time that it receives corrective information related to that signal.

While CMAC differentiates between a learning and a recall phase, the APG is an on-line learning robot controller. This means that it is constantly learning while controlling a system.

**4.3.2.3. Structural Credit Assignment Problem.** In APG, the climbing fibers CF deliver error information to the cerebellum. Positive errors result in an increase in the corresponding CF firing probability, whereas negative errors result in a decrease. Since the firing frequency of the climbing fibers is very low, Barto et al. propose that the climbing fibers can only be *on* or *off*. Since this feature is only necessary for biological plausibility but very inefficient from the engineering point of view, our experiments show that it is of better use to take the numerical value of the firing frequency.

**4.3.2.4. Temporal Credit Assignment Problem.** The temporal credit assignment problem results from certain delays in the model. There is a significant delay between the time when a PF becomes active and when the effects of its activation may be sensed by the inferior olive. In APG this problem is addressed by the use of eligibility traces on synapses. This trace is memory for recent synaptic activity, it marks the synapse as being eligible for potential update if inputs arrive from

the inferior olive. It is computed by

$$\frac{de_{ij}}{dt} = -\frac{-e_{ij} + PF_i}{\tau_{\text{elig}}} \quad (31)$$

and

$$\frac{d\hat{e}_{ij}}{dt} = -\frac{-e_{ij} + \hat{e}_{ij}}{\tau_{\text{elig}}}. \quad (32)$$

$e_{ij}$  and  $\hat{e}_{ij}$  are called the first and second eligibility terms, and  $\tau_{\text{elig}}$  is the memory time constant. The potential plasticity of a synapse is computed by

$$\bar{e}_{ij} = f(3\hat{e}_{ij} - e_{ij}) \quad (33)$$

with  $f(\cdot)$  defined by

$$f(x) = \begin{cases} x & \text{if } x > 0 \\ 0 & \text{otherwise.} \end{cases} \quad (34)$$

The learning rule has two components: a strong learning by long-term depression (LTD) and a much weaker form of long-term potentiation (LTP). The LTD is defined by  $-\alpha g(\bar{e}_{ij})(CF - CF_{\text{base}})$ , while LTP is computed by  $+\beta \eta_{ij} PC_j$ , where  $\eta_{ij}$  denotes a change of  $PC_j$ . Hereby  $\alpha$  and  $\beta$  are positive parameters with  $\alpha \gg \beta$ . The resulting learning rule is

$$\Delta w_{ij} = -\alpha g(\bar{e}_{ij})(CF - CF_{\text{base}}) + \beta \eta_{ij} PC_j. \quad (35)$$

#### 4.4. The "Fairly Obvious Extension" FOX

The FOX controller was developed by Russel Smith and published in his PhD-thesis [59]. Like APG it is based on CMAC and, like APG, it is an on-line learning controller. Apparently the term "Fairly Obvious Extension" was used since the improvements in the model over CMAC are biologically realistic as well as sound from an engineering point of view. Smith [59] gives several impressive simulated and hardware applications of this model, e.g., a humanoid robot.

The FOX controller is again based on the same understanding of cerebellar structure as CMAC. The major difference is that the climbing fiber carries error information which is filtered and used for weight modification due to eligibilities as in APG. From the biological point of view, the FOX controller does not bring in any further information than APG though it is more realistic than CMAC. Still there are some technical features which make the FOX controller interesting:

first, it uses vectorized instead of scalar eligibilities. Secondly, it uses the linearizations of the system matrix  $A(t)$  and the input transformation matrix  $B(t)$  for eligibility update. This allows a decoupling of the state variables in the learning process. Still, it also acts as a generalizing lookup table like CMAC and APG and should suffer from the same dimensionality problems.

#### 4.5. Discussion

Cerebellar models with a state encoder such as CMAC, APG, or FOX are inspired by the enormous number of granule cells in the human cerebellar cortex. Consequently, it is often suggested that the state space of the controlled object may be covered by small boxes in which only a small number of granule cells fire. A simple thought experiment (as in [52]) shows how insufficient this approach is: Let us assume that we have a robot system which has ten degrees of freedom (less than the human body). Assuming a state-free system we have 30 state variables, one each for position, velocity, and acceleration. Furthermore, there exist forces between all these degrees of freedom due to Coriolis and centrifugal forces. Therefore the state encoder of the entire system cannot be split up into several small state-encoders of the single degrees of freedom. Even if our state-space is only split into ten areas per variable—which is insufficient for real-world control applications—a number of  $10^{30}$  neurons is required for coarse coding the inputs of our system. This number is neither realistic from the engineering point of view nor from the biological point of view. Therefore, a state-encoder approach never suffice for large-scale (or scalable) applications.

## 5. Conclusion

We have shown elsewhere [78] that the models discussed in this section cannot cope with a set of basic cerebellum control-benchmarks [79]. Also, the scalability discussion (Section 4.5) shows that many models do not scale well, and are unlikely candidates for the control of complex dynamic structures.

A different approach (MPFIM), which overcomes the scalability problem, has been proposed by Wolpert and Kawato [35] (see also [78]). Although currently only used for simple problems, the proposed system has its limitations but seems a plausible candidate for cerebellar-based control.

Clearly, the path towards a serious use of cerebellar models for robot control has not yet been seriously investigated. To date the question of whether the use of cerebellar models for robot control is a valid approach remains unanswered.

## References

1. S. Arimoto and F. Miyazaki, "Stability and robustness of PID feedback control for robot manipulators of sensory capability," in *Proceedings of the First International Symposium on Robotics Research*, edited by M. Brady and R.P. Paul, MIT Press: Cambridge MA, pp. 783–801, 1984.
2. C.-P. Chou and B. Hannaford, "Measurement and modeling of McKibben pneumatic artificial muscles," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 1, pp. 90–102, 1996.
3. M. Katayama and M. Kawato, "Learning trajectory and force control of an artificial muscle arm by parallel-hierarchical neural network model," in *Advances in Neural Information Processing Systems*, edited by R. Lippmann, J. Moody, and D. Touretzky, Morgan Kaufmann: San Mateo, vol. 3, pp. 436–442, 1991.
4. T. Hesselroth, K. Sarkar, P. van der Smagt, and K. Schulten, "Neural network control of a pneumatic robot arm," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 24, no. 1, pp. 28–38, 1994.
5. P. van der Smagt, F. Groen, and K. Schulten, "Analysis and control of a rubbertuator arm," *Biological Cybernetics*, vol. 75, no. 5, pp. 433–440, 1996.
6. H.A. Talebi, K. Khorasani, and R.V. Patel, "Neural network based control schemes for flexible-link manipulators: Simulations and experiments," *Neural Networks*, vol. 11, no. 7/8, pp. 1357–1377, 1998.
7. K.J. Cole and J.H. Abbs, "Kinematic and electromyographic responses to perturbation of a rapid grasp," *Journal of Neurophysiology*, vol. 57, pp. 1498–1510, 1987.
8. R.C. Miall, D.J. Weir, D.M. Wolpert, and J.F. Stein, "Is the cerebellum a Smith predictor?" *Journal of Motor Behavior*, vol. 25, pp. 203–216, 1993.
9. N. Hogan, "Mechanical impedance of single- and multi-articular systems," in *Multiple Muscle Systems: Biomechanics and Movement Organization* edited by J.M. Winters and S.L.-Y. Woo, Springer-Verlag: New York, pp. 149–163, 1990.
10. T. Yoshikawa, *Foundations of robotics*. MIT Press: Cambridge, MA, 1990.
11. J.J. Slotine and W. Li, "On the adaptive control of robot manipulators," *International Journal of Robotics Research*, vol. 6, no. 3, pp. 49–59, 1987.
12. C. Ghez and S. Fahn, "The cerebellum," in *Principles of Neural Science*, edited by E.R. Kandel and J.H. Schwartz, Elsevier Science Publishing: Amsterdam, pp. 502–522, 1985.
13. M. Ito, *The Cerebellum and Neural Control*, Raven Press: New York, 1984.
14. J. Dichgans and H.C. Diener, "Clinical evidence for functional compartmentalization of the cerebellum," in *Cerebellar Function*, edited by J.R. Bloedel, Springer Verlag: Berlin, pp. 127–147, 1985.
15. G. Holmes, "The cerebellum of man," *Brain*, vol. 62, pp. 1–30, 1939.

16. M. Glickstein, "Cerebellar agenesis," *Brain*, vol. 117, pp. 1209–1212, 1994.
17. M. Föhle and V. Braitenberg, "Some quantitative aspects of cerebellar anatomy as guide to speculation on cerebellar functions," in *Cerebellar Function*, edited by J.R.B. et al., Springer Verlag: Berlin, pp. 186–200, 1985.
18. C.R. Noback, *The Human Nervous System*, 2nd edn., McGraw-Hill: New York, 1981.
19. J.D. Keeler, "A dynamical system view of cerebellar function," *Physica D*, vol. 42, pp. 386–410, 1990.
20. G. Chauvet, "Habituation rules for a theory of the cerebellar cortex," *Biological Cybernetics*, vol. 55, pp. 201–209, 1986.
21. A.C. Guyton, *Structure and Function of the Nervous System*, W. B. Saunders Company, 1972.
22. Checkley, *The Brain*, 1999.
23. D.W. Molavi, *Neuroscience Tutorial*, 1997.
24. J.C. Eccles, M. Ito, and J. Szentagothai, *The Cerebellum as a Neuronal Machine*, Springer-Verlag: Berlin, 1967.
25. M. Rosheim, *Robot Evolution: The Development of Anthroprobotics*, John Wiley and Sons: New York, 1994.
26. M. Vrahas, R. Brand, T. Brown, and J. Andrews, "Contribution of passive tissues to the intersegmental moments of the hip," *Journal of Biomechanics*, pp. 357–362, 1990.
27. R. Brand, R. Crowninshield, C. Wittstock, D. Petersen, C. Clark, and F.V. Krieken, "A model of lower extremity muscular anatomy," *Transaction of the ASME*, pp. 302–320, 1982.
28. J. Mansour and M. Audu, "The passive elastic moment at the knee and its influence on human gait," *Journal of Biomechanics*, pp. 369–373, 1986.
29. P. Weiss, I. Hunter, and R. Kearney, "Human ankle joint stiffness over the full range of muscle activation levels," *Journal of Biomechanics*, pp. 539–544, 1988.
30. R. Woittiez, P. Huijting, H. Boom, and R. Rozendal, "A three-dimensional muscle model: A quantified relation between form and function of skeletal muscles," *Journal of Morphology*, pp. 95–113, 1984.
31. J.H. Abbs and K.J. Cole, "Neural mechanisms of motor equivalence and goal achievement," in *Higher Brain Functions: Recent Explorations of the Brain's Emergent Properties*, edited by S.P. Wise, John Wiley & Sons: New York, pp. 15–43, 1987.
32. R. Klinke and S. Silbernagl, *Lehrbuch für physiologie*, Georg Time Verlag.
33. G. Schmidt, *Regelungstechnik 1*, Skript der Technischen Universität München.
34. M. Ito, "A new physiological concept on cerebellum," *Rev. Neuol. (Paris)*, pp. 564–569, 1990.
35. D. Wolpert and M. Kawato, "Multiple paired forward and inverse models for motor control," *Neural Networks*, vol. 11, pp. 1317–1329, 1998.
36. D. Wolpert, C. Miall, and M. Kawato, "Internal models in the cerebellum," *Trends in Cognitive Sciences*, vol. 2, pp. 338–347, 1998.
37. V. Braitenberg and R.P. Atwood, "Morphological observations on the cerebellar cortex," *Journal of Comparative Neurology*, vol. 109, pp. 1–34, 1958.
38. V. Braitenberg, "Functional interpretation of cerebellar histology," *Nature*, vol. 190, pp. 539–540, 1961.
39. M.A. Arbib, P. Irdi, and J. Szentagothai, *Neural Organization: Structure, Function, and Dynamics*, MIT Press: Cambridge, MA, pp. 262–302, 1998.
40. D. Marr, "A theory of cerebellar cortex," *Journal of Physiology*, vol. 202, pp. 437–470, 1969.
41. J. Albus, "A theory of cerebellar function," *Mathematical Bioscience*, vol. 10, pp. 25–61, 1971.
42. M.G. Paulin, "The role of the cerebellum in motor control and perception," *Brain, Behavior, and Evolution*, vol. 31, pp. 39–50, 1993.
43. P. van der Smagt and G. Hirzinger, "The cerebellum as computed torque model," in *Fourth International Conference on Knowledge-Based Intelligent Engineering Systems & Allied Technologies*, IEEE, pp. 760–763, 2000.
44. M. Ito, M. Sakurai, and P. Tongroach, "Climbing fiber induced depression of both mossy fiber responsiveness and glutamate sensitivity of cerebellar Purkinje cells," *Journal of Physiology*, vol. 324, pp. 113–134, 1982.
45. T.V.P. Bliss and G. Collingridge, "A synaptic model of memory: Long-term potentiation in the hippocampus," *Nature*, vol. 361, pp. 31–39, 1993.
46. A. Artola and W. Singer, "Long-term depression of excitatory synaptic transmission and its relationship to long-term potentiation," *Trends in Neuroscience*, vol. 16, pp. 480–487, 1993.
47. E. De Schutter, "A new functional role for cerebellar long term depression," *Progress in Brain Research*, vol. 114, pp. 529–542, 1997.
48. J.I. Simpson, D.R. Wylie, and C.I. De Zeeuw, "On climbing fiber signals and their consequence(s)," *Behavioral and Brain Sciences*, vol. 19, pp. 384–398, 1996.
49. Ebner and Bloedel, "Neural organization: Structure, function, and dynamics," pp. 262–302, 1984.
50. M. Kawato and H. Gomi, "A computational model of four regions of the cerebellum based on feedback-error learning," *Biological Cybernetics*, vol. 68, pp. 95–103, 1992.
51. P.F.C. Gilbert and W.T. Thach, "Purkinje cell activity during learning," *Brain Research*, vol. 128, pp. 309–328, 1977.
52. M. Kawato, "Cerebellum and motor control," in *The Handbook of Brain Theory and Neural Networks*, edited by M.A. Arbib, MIT Press: Cambridge, MA, pp. 172–178, 1995.
53. M. Kawato, "Feedback-error-learning neural network for supervised motor learning," in *Neural Networks for Sensory and Motor Systems*, edited by R. Eckmiller, Elsevier: Amsterdam, 1990.
54. J.L. Contreras-Vidal, S. Grossberg, and D. Bullock, "A neural model of cerebellar learning for arm movement control: Cortico-spino-cerebellar dynamics," *Learning & Memory*, vol. 3, pp. 475–502, 1997.
55. J.C. Houk, "Cooperative control of limb movements by the motor cortex, brainstem, and cerebellum," in *Models of Brain Function*, edited by R.M.J. Cotterill, Cambridge University Press: Cambridge, pp. 309–325, 1989.
56. J.C. Houk, J.T. Buckingham, and A.G. Barto, "Models of the cerebellum and motor learning," *Behavioral and Brain Sciences*, vol. 19, pp. 368–383, 1996.
57. P. van der Smagt, "Cerebellar control of robot arms," *Connection Science*, vol. 10, pp. 301–320, 1998.
58. J.C. Houk, J.T. Buckingham, and A.G. Barto, "Models of the cerebellum and motor learning," *Behavioral and Brain Sciences*, vol. 19, no. 3, pp. 368–383, 1996.
59. R. Smith, "Intelligent motion control with an artificial cerebellum," unpublished doctoral dissertation, University of Auckland,

- Department of Electrical Engineering, 1999.
60. N. Schweighofer, "Computational models of the cerebellum in the adaptive control of movements," unpublished doctoral dissertation, University of Southern California.
  61. M. Brown and C. Harris, *Neurofuzzy Adaptive Modelling and Control*, Prentice Hall: Englewood cliffs, NJ, 1994.
  62. W.T. Miller, F.H. Glanz, and L.G. Kraft, "CMAC: An associative neural network alternative to backpropagation," in *Proceedings of the IEEE, Special Issue on Neural Networks*, vol. 78, pp. 1561–1567, 1990.
  63. M. Lang, "A real-time implementation of a neural-network controller for industrial robotics," unpublished doctoral dissertation, Institute of Aerospace Studies, University of Toronto, 1997.
  64. R. Hewes and W. Miller, "Practical demonstration of a learning control system for a five axis industrial robot," in *Proceedings SPIE Intelligent Robots and Computer Vision*, vol. 1002, 1988.
  65. W. Miller and A. Kun, "Dynamic balance of a biped walking robot," in *Neural Systems for Robotics*, edited by P. van der Smagt and O. Omidvar, Academic Press: San Diego, CA, pp. 17–35, 1997.
  66. D. Michie and R.A. Chambers, "BOXES: An experiment in adaptive control," in *Machine Intelligence*, edited by E. Dale and D. Michie, Oliver and Boyd: Edinburgh, vol. 2, pp. 137–152, 1968.
  67. R. Bellman, *Dynamic Programming*, Princeton University Press: Princeton, NJ, 1957.
  68. W.T. Miller and F.H. Glanz, *The University of New Hampshire Implementation of the Cerebellar Model Arithmetic Computer CMAC*, Internet-Pages.
  69. P.C. Parks and J. Miltzer, "A comparison of five algorithms for the training of CMAC memories for learning control systems," *Automatica*, vol. 28, no. 5, pp. 1027–1035, 1992.
  70. L.G. Kraft, "Optimized weight smoothing for CMAC neural networks," in *Dlr Technical Report 515-97-28* edited by P. van der Smagt, pp. 8–10, 1997.
  71. J.C. Houk, S.P. Singh, C. Fisher, and A.G. Barto, "An adaptive sensorimotor network simulation inspired by the anatomy and physiology of the cerebellum," in *Neural Networks for Control*, edited by W. Miller, R. Sutton, and P. Werbos, MIT Press: Cambridge, MA, pp. 301–348, 1990.
  72. A.G. Barto, J.T. Buckingham, and J.C. Houk, "A predictive switching model of cerebellar movement control," in *Advances in Neural Information Processing Systems*, vol. 8, edited by D.S. Touretzky, M.C. Mozer, and M.E. Hasselmo, MIT Press: Boston, MA, pp. 138–144, 1995.
  73. A.H. Fagg, N. Sitkoff, A.G. Barto, and J.C. Houk, "Cerebellar learning for control of a two-link arm in muscle space," in *Proceedings of IEEE Conference on Robotics and Automation*, pp. 6–12, 1996.
  74. A.H. Fagg, N. Sitkoff, A.G. Barto, and J.C. Houk, "A model of cerebellar learning for control of arm movements using muscle synergies," in *Proceedings of IEEE Symposium on Computational Intelligence in Robotics and Automation*, pp. 2638–2644, 1997.
  75. A.H. Fagg, N. Sitkoff, A.G. Barto, and J.C. Houk, "A computational model of cerebellar learning for limb control," in *Proceedings of the Spring Meeting on the Neural Control of Movement*, 1997.
  76. A.H. Fagg, A.G. Barto, and J.C. Houk, "Learning to reach via corrective movements," *Tenth Yale Workshop on Adaptive and Learning Systems*, 1998.
  77. A.G. Barto, A.H. Fagg, N. Sitkoff, and J.C. Houk, "A cerebellar model of timing and prediction in the control of reaching," *Neural Computation*, vol. 11, no. 3, pp. 565–594, 1999.
  78. J. Peters and R. Riener, "A real-time model of the human knee for a virtual orthopaedic trainer," in print.
  79. P. van der Smagt, "Benchmarking cerebellar control," *Robotics and Autonomous Systems*, vol. 32, pp. 237–251, 2000.
  80. G. Hirzinger, "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*, 1996.



**Jan Peters** received his Diplom-Informatiker degree from the Fernuniversitaet Hagen, Germany in 2000 and his Diplom-Ingenieur degree in 2001 from the Technical University of Munich.



**Patrick van der Smagt** received his Ph.D. in 1995 in Mathematics and Computer Science at the University of Amsterdam. In 1995 he joined the Institute of Robotics and Mechatronics of the German Aerospace Center. His research interests include cerebellar based control of manipulators and hands, as well as biomorphic limb design. URL <http://www.robotic.de/Smagt/>