Cerebellar Learning for Control of a Two-Link Arm in Muscle Space *

Andrew H. Fagg[1] Nathan Sitkoff[1] Andrew G. Barto[1]
James C. Houk [2]

{fagg | sitkoff | barto}@cs.umass.edu
[1]Department of Computer Science
University of Massachusetts
Amherst, MA 01003-4610

houk@casbah.acns.nwu.edu
[2]Department of Physiology
Northwestern University School of Medicine
Chicago, IL 60611-3008

Abstract

Biological control systems have long been studied as possible inspiration for the construction of robotic controllers. The cerebellum is known to be involved in the production and learning of smooth, coordinated movements. In this paper, we present a model of cerebellar control of a muscle-actuated, two-link, planar arm. The model learns in a trial-and-error fashion to produce bursts of muscle activity that accurately bring the arm to a specified target. When the cerebellum fails to bring the arm to the target, an extra-cerebellar module performs low-quality corrective movements, from which the cerebellum may update its program. In learning to perform the task, the cerebellum constructs an implicit inverse model of the plant. This model uses a combination of delayed sensory signals and recently-generated motor commands to compute the new output motor signal.

1 Introduction

Modern robotics research is concerned with the control of complex plants. Such plants exhibit non-trivial dynamics and potentially long feedback delays. However, in order to be successful, many control techniques require accurate models of both the plant and the environment with which the plant interacts.

In biology, both actuator and plant are in a constant state of flux, and behave in a complex and non-linear fashion. Delays of the sensory-motor loops are typically measured in tens or hundreds of milliseconds. Millions of years of evolution have developed biological controllers that are very good at controlling these systems. These controllers do not rely on high-quality, pre-defined models of the plant. Rather, the control algorithm is tuned incrementally through experience with the environment. It is not surprising, then, that we should turn to biology for inspiration.

The primate cerebellum is known to be involved in the learning and production of coordinated movements [13]. Two classes of inputs carry information into the cerebellum: the *mossy fibers* (MFs) and the *climbing fibers* (CFs). The mossy fibers provide both plant state and contextual information [4]. The CFs, on the other hand, are thought to provide information that reflect errors in recently generated movements [6]. This information is used to adjust the programs encoded by the cerebellum.

This paper proposes a controller for a 2-DOF, 6-muscle simulated arm which is based on the structure of the primate cerebellum. This controller does not rely on a high quality, built-in model of the plant, but rather acquires an implicit model through experience. In addition, we do not assume the existence of a reference trajectory on which control or learning may be based. Finally, control signals are not in the form of torques to be applied to the joints, but instead control is performed directly in *muscle space*.

2 System Architecture

The plant used in our experiments is a simulated, two degree-of-freedom, planar arm (Figure 1). Actuation of the arm is performed by three pairs of opposing muscles. Two pairs of muscles individually actuate the shoulder and elbow joints, while the third pair actuates both joints simultaneously. Each muscle behaves as a non-linear visco-elastic (spring) element. Control of the arm is achieved by setting the rest length of each muscle.

The system architecture, outlined in Figure 2, consists of two control modules (the *extra-cerebellar* (EC) and *cerebellar* modules), a spinal/muscle processing system, the arm plant, and a module responsible for estimating movement errors (representing the *inferior olive*).

The learning controller (the cerebellar module) is constructed from an array of *adjustable pattern generators* (APGs) [2, 3], each of which drives a single muscle. These APGs combine the current target position with delayed sensory and motor efference signals to produce motor commands that are intended to bring

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

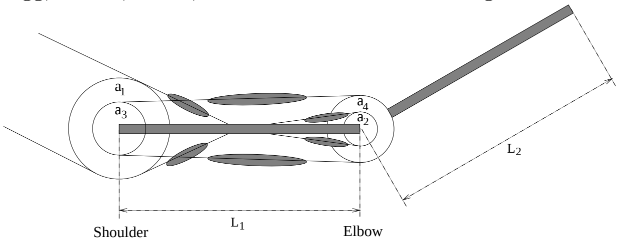

Figure 1: A 2 DOF planar arm. The six muscles are attached in such a way that four of them induce torques about single joints; the remaining two simultaneously actuate both joints (Figure adapted from [7]).

the arm to the specified goal. In cases where the arm does not reach this goal, the static EC module is activated. This module produces a short, constant burst of activity in muscle space that brings the arm closer to the goal.

The contributions of the two control modules are combined in the spinal/muscle system, which transforms muscle space signals into joint torques. Motor commands for opposing muscles reciprocally inhibit one-another (transform S). Then, multiple muscles induce torques about individual joints, as described by the transform A.

Errors in cerebellar-generated movements are assessed by the *inferior olive* (IO), and are measured in muscle space. This is accomplished indirectly by observation of muscle length changes in response to the corrective motor commands generated by the EC module. The IO-generated signals are then used to update the APGs contained within the cerebellar module.

2.1 Arm Model

The plant is a typical planar, 2-DOF arm, whose dynamics are described as follows:

$$M(\theta)\ddot{\theta} + C(\theta,\dot{\theta})\dot{\theta} = \tau(\theta,\dot{\theta},u), \qquad (1)$$

where:

$$M(\theta) = \begin{pmatrix} I_1 + I_2 + M_2 L_1^2 + 2M_2 L_1 L_{g2}\cos(\theta_2) & I_2 + M_2 L_1^2 L_{g2}\cos(\theta_2) \\ I_2 + M_2 L_1^2 L_{g2}\cos(\theta_2) & I_2 \end{pmatrix}, \qquad (2)$$

$$C(\theta,\dot{\theta}) = M_2 L_1 L_{g2}\sin(\theta_2)\begin{pmatrix} -2\dot{\theta}_2 & -\dot{\theta}_2 \\ \dot{\theta}_1 & 0 \end{pmatrix} - \begin{pmatrix} D1 & 0 \\ 0 & D2 \end{pmatrix}, \qquad (3)$$

and u is the descending motor command.

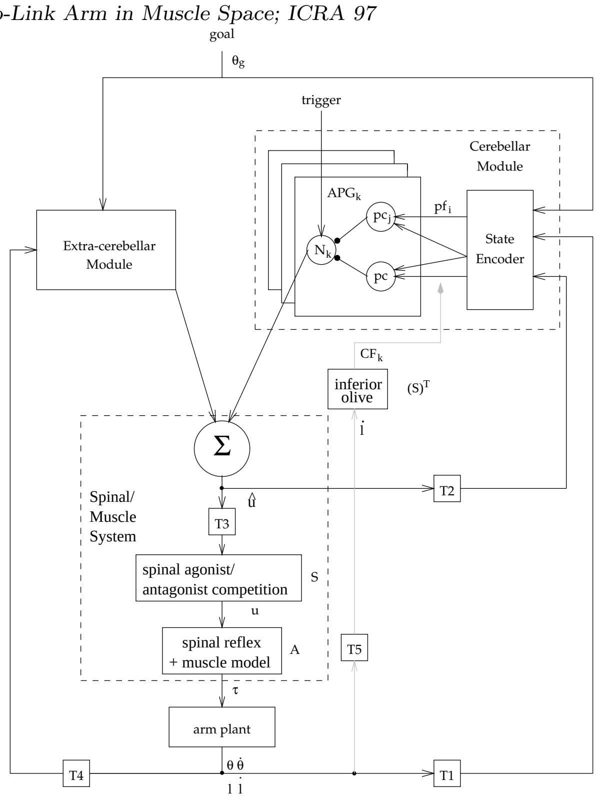

Figure 2: The System Architecture. Two control modules (cerebellar and extra-cerebellar systems) produce signals in muscle space. These signals are transformed by the spinal/muscle system into torque commands that drive the arm plant. Movement errors are detected by the inferior olive, which are used to update the cerebellar control program.

Links 1 and 2 correspond to the upper and lower arm, respectively. The arm parameters, M (mass of each link), L (link length) L_g (center of gravity), I (moment of inertia), and Di (joint friction) are adapted from [7].

2.2 Muscle Model

The muscles are modeled as visco-elastic (spring) elements, which pull, but do not push. The state of each muscle is a function of its current length (l), the rate of change in length ($\dot{l}$), and the muscle rest length, $l_r(u)$. The tension produced by a single muscle is defined according to:

$$T(l,\dot{l},u) = f[Q(l - l_r(u)) + D\,\dot{l}^{\frac{1}{5}}], \qquad (4)$$

where $f()$ is a threshold-linear function:

$$f(x) = \begin{cases} 0 & x < 0 \\ x & otherwise, \end{cases} \tag{5}$$

Q is the spring constant, and D is the gain of the non-linear damping factor.

The rest length of the muscle, $l_r(u)$, is a linear function of the motor command:

$$l_r(u) = l_0 - ru, \tag{6}$$

where l_0 is the rest length of the muscle when a 0-motor command is given. Increases in the motor command (u) result in a decrease in the muscle rest length.

The non-linear damping term, which models the effects of the spinal muscle stretch reflex [14, 2], is formulated such that damping is relatively high for low velocities, but as velocities increase, the term becomes less significant. This has the effect of creating a *stickiness* in the muscle's behavior, which allows high-velocity movements without the ringing generally associated with a purely linear model.

Muscle tensions induce torques about joints according to:

$$\tau(\theta, \dot{\theta}, u) = A(\theta)^T T(l, \dot{l}, u), \tag{7}$$

where the matrix $A(\theta)$ describes the moment arms for each muscle. In this model, however, the moment arms are assumed to be independent of θ:

$$A^T = \begin{pmatrix} a_1 & -a_1 & 0 & 0 & a_3 & -a_3 \\ 0 & 0 & a_2 & -a_2 & a_4 & -a_4 \end{pmatrix}. \tag{8}$$

Finally, muscle lengths are related to joint angles according to:

$$l = l_m - A\theta, \tag{9}$$

where l_m is the vector of muscle lengths when $\theta = (0, 0)$.

2.3 Spinal Processing

In addition to the muscle stretch reflex already discussed (and incorporated as part of the muscle model), spinal processing is viewed as performing one additional computational step: the reciprocal modulation of the agonist/antagonist motor commands.

This effect is modeled using reciprocal inhibition with a threshold:

$$u = f(S\, \hat{u}), \tag{10}$$

where

$$S = \begin{pmatrix} 1 & -0.5 & 0 & 0 & 0 & 0 \\ -0.5 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -0.5 & 0 & 0 \\ 0 & 0 & -0.5 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -0.5 \\ 0 & 0 & 0 & 0 & -0.5 & 1 \end{pmatrix}, \tag{11}$$

and $\hat{u}$ is the combined motor commands from the cerebellar and EC modules. In addition, in S, each pair of columns, $2i$ and $2i + 1$, correspond to agonist/antagonist muscle pairs.

2.4 Cerebellar Module

In the primate brain, *mossy fibers* (MFs) carry plant state, motor efference, and other contextual signals into the cerebellum [4]. These fibers impinge on *granule cells*, whose axons give rise to *parallel fibers* (PFs). Through the combination of inputs from multiple classes of mossy fibers and local inhibitory interneurons, the granule cells are thought to provide a sparse expansive encoding of the incoming state information [1].

In our model, a sparse, expansive encoding is computed by the *state encoder* (Figure 2) using a *CMAC* (Cerebellar Model Arithmetic Computer) [1, 10]. This coding scheme makes use of multiple tilings over the state space. A single tiling partitions the space defined by several state variables (e.g. θ_e, $\dot{\theta}_e$, and $\hat{u}_1$) into discrete, non-overlapping rectangular volumes (tiles). When the system state falls into a particular tile, the tile is given an activation level of 1, and all others within the tiling are set to 0. Like Albus, we imagine that each tile corresponds to a single granule cell [1].

Due to the large number of state variables, it is impractical for a single tiling to effectively partition the entire space. Thus, a number of tilings are used in parallel, each making use of different subsets of state variables.

The large number of parallel fibers converge on a much smaller set of *Purkinje cells* (PCs):

$$PC_j \;=\; g(\sum_i PF_i \cdot w_{ij}), \tag{12}$$

where PC_j is the cell's firing rate, and w_{ij} is the (variable) strength of the synapse from PF_i to PC_j. $g()$ is defined as a threshold-ramp function:

$$g(x; \rho, \bar{\rho}, \lambda, \bar{\lambda}) = \begin{cases} 0 & x < \lambda \\ \rho + (\bar{\rho} - \rho)\frac{x - \lambda}{\bar{\lambda} - \lambda} & \lambda \le x \le \bar{\lambda} \\ \bar{\rho} & otherwise. \end{cases} \tag{13}$$

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 APG_k is computed as follows:

$$u_{C\,k} = 1 - \frac{1}{|L_k|} \sum_{j \in L_k} PC_j, \tag{14}$$

where L_k is the set of Purkinje cells that are within APG k.

2.5 Extra-Cerebellar Module

For a given movement, the cerebellar control module may generate a pattern of muscle activity that does not bring the arm to rest at the target position. In these cases, it is the role of the hard-wired EC module to produce one or more corrective movements. It is not necessary, however, that a perfect correction be produced. Rather, the minimal constraints are that 1) the corrective movement makes reasonable positive progress toward the goal, and 2) the movement generally does not overshoot the goal.

In the model, corrective movements are generated by short, constant bursts of muscle activation. A crude activation pattern, u_{EC}, is computed as the arm comes to rest at a point far from the goal:

$$u_{EC} = f((A^T S)^+ \, J(\theta_g - \theta)), \tag{15}$$

where J is a square diagonal matrix consisting of scaling terms, $\theta_g - \theta$ is the current deviation of the arm from the goal, and the "$^+$" operator is the pseudo-inverse.

The descending muscle motor command is a combination of the cerebellar and EC signals:

$$\hat{u} = gate \cdot u_{EC} + u_C, \tag{16}$$

where *gate* is a binary indicator of the participation of the EC system. This gate is only opened when the arm has begun to slow at a point that is far from the goal (far being more than 1 cm). Once opened, the gate remains in this state for a fixed period of time (50 ms in our experiments) before closing.

3 Learning Algorithm

When the cerebellum does not bring the arm to the target, requiring the extra-cerebellar module to generate a corrective movement, 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 issue, referred to as the *credit assignment problem* 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 is that of dealing with the significant delays between the time that the cerebellum generates a motor signal, and the time that it receives corrective information related to that signal.

3.1 Structural Credit Assignment

In biological systems, the inferior olive is known to be the source of CF inputs into the cerebellum, which are thought to carry some form of error information related to the current movement [6]. Studies in cat show that when a limb is passively moved by an external force, CFs fire in response to the proprioceptive inputs from the the limb [5]. In addition, the CFs are differentially responsive to the direction of limb movement. During active movements (such as stepping), some CF responses are silenced except at the very end of movement, where certain perturbations are likely to occur (e.g. due to misanticipation of the height of the step).

In the model, CFs deliver error information to the cerebellum's APGs (there is one CF for each APG). The CF behavior draws in two key ways from the behavior observed in cat. First, from the perspective of the cerebellar-generated movement, the corrective movement made by the EC module is seen essentially as passive (or unexpected). Thus, CF signals only deviate from their background firing rate when corrective movements are initiated. Second, the increase (or decrease) in CF firing probability is a function of the rate of stretch of the muscles. Each CF is maximally responsive to stretches in the muscles that are activated by the corresponding APG (we will refer to this as the *direction* of the APG). In other words, the corrective movement is spatially coded in the array of CFs.

The firing probability of the array of CFs, PR, is computed according to:

$$PR = g(-(S^T \, \dot{l})). \tag{17}$$

Positive errors (movements along the same direction as an APG) result in an increase in the corresponding CF firing probability (from the baseline), whereas negative errors (in the opposite direction as the APG) result in a decrease in firing probability.

3.2 Temporal Credit Assignment

The temporal credit assignment problem results from certain delays in the model. First, there is a significant delay between the time when a PF becomes active and when the effects of its activation may be

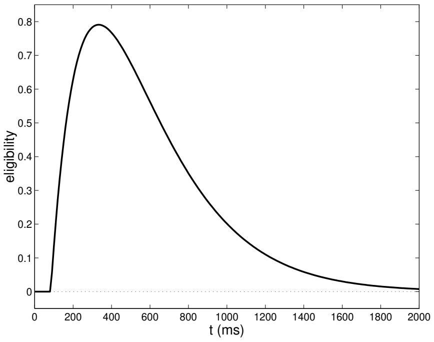

Figure 3: Eligibility trace $\bar{e}_{ij}$ in response to the transient activation of parallel fiber i.

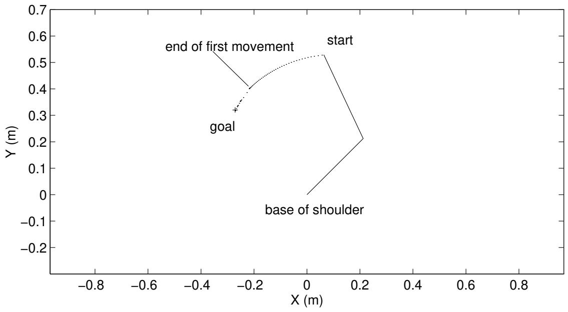

Figure 4: Plant behavior on trial 13. The initial and target positions are indicated, as is end-point of the first (cerebellar-generated) movement.

sensed by the inferior olive. Second, the time between the activation of a PF and the generation of a corrective movement (which gives rise to CF signals) can vary by several hundred milliseconds.

We address these problems through the use of an *eligibility trace* on the synapse [8, 12]. This trace acts as a memory of recent synaptic activity. The memory marks the synapse as being *eligible* for potential update in the event that later training inputs arrive from the inferior olive.

In our model, we make use of a second-order eligibility trace [11], whose time-course is shown in Figure 3. We compute this trace as follows:

$$\tau_{elig} \frac{de_{ij}}{dt} = -e_{ij} + PF_i \ , \qquad (18)$$

$$\tau_{elig} \frac{d\hat{e}_{ij}}{dt} = -\hat{e}_{ij} + e_{ij} \ , \qquad (19)$$

where e_{ij} and $\hat{e}_{ij}$ are the first and second-order eligibility terms, and τ_{elig} is the memory time constant.

The potential plasticity of a synapse, $\bar{e}_{ij}$, is defined as a function of both the first- and second-order eligibility terms:

$$\bar{e}_{ij} = f(3\ \hat{e}_{ij} - e_{ij}). \qquad (20)$$

The eligibility trace has the advantage that the memory of PF activation is temporally blurred over a long period of time. As we will see, this implies that CF error signals do not have to be carefully timed relative to PF activity in order for a useful program to be learned.

Finally, updates to the PF-to-PC synaptic strength require the coincidence of eligibility and corresponding CF input:

$$\Delta w_{ij} = -\alpha\ g(\bar{e}_{ij})\ (CF_k - CF_{base}), \qquad (21)$$

where CF_k is the climbing fiber that projects to APG_k (which contains PC_j), and whose binary state is selected as a function of its probability of firing (PR_k). α is the learning rate, and CF_{base} is the background firing rate of the climbing fiber.

In cases where the corrective movement is aligned with APG_k, PR_k will be near 1. Thus, on average, the marked synapses from the PFs to the PCs within APG_k will be decremented. The next time that the same movement is executed, the PCs of APG_k will be less active, resulting in a higher level of activity for APG_k. On the other hand, if the corrective movement is in exactly the opposite direction as APG_k, PR_k will be near 0, the synapse strengths will be incremented, and PC activity will increase, resulting in a decrease in APG activity. In essence, the trajectory that results will be a blend of the original and corrective movements.

4 Simulation Results

In this section, the behavior of the model is demonstrated during learning of a single point-to-point movement. Each of the six APGs consists of eight PCs. The *state encoder* is made up of 120 separate tilings. Each tiling receives input from between 3 and 5 state variables, which are drawn from: θ, θ_g, *error* (i.e $\theta_g - \theta$), $\dot{\theta}$, and $\hat{u}$ (the motor efference copy).

The sensory afference delay to the state encoder (T1 in Figure 2) is 50 ms; motor efference delays (T2) range from 15 to 75 ms; the descending motor command delay (T3) is 50 ms; the sensory afference delay to the EC module (T4) is 50 ms; and inferior olive delay (T5) is 20 ms. A simulation timestep of 5 ms is used.

At the beginning of a trial, the arm is placed in an initial starting configuration (*start* in Figure 4), and the target location is presented to the controller

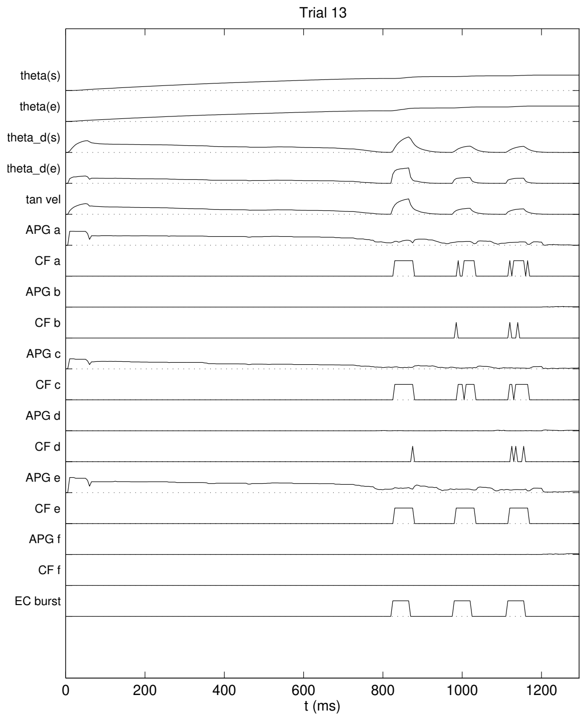

Figure 5: Plant/Neural behavior on trial 13. **theta(s)** *and* **theta(e)** *correspond to the joint angles of the shoulder and elbow, respectively.* **theta_d(s|e)** *represent the joint velocities. For each APG, two traces are shown: the APG activity and the corresponding CF activity. The APGs correspond to the following muscles: shoulder agonist (a) and antagonist (b), elbow ag/ant pair (c and d), and the biarticulate ag/ant pair (e and f).* **EC_burst** *indicates when the EC system generated a corrective movement.*

(*goal* in the Figure). The movement is then triggered, and proceeds under cerebellar control until the module turns off all descending motor commands, and the arm comes to rest. If the arm has stopped moving at a position that is not near the target, the EC system is triggered, which initiates a corrective arm movement. The trial terminates if the arm reaches the target position or the EC system has generated 9 bursts without successfully reaching the target. A new trial is then initiated, with a resetting of the arm position and the selection of a new target position. In the experiment presented below, however, only a single start/target pair is used.

Figures 4 and 5 illustrate the behavior of the plant and the cerebellar control module during the 16th learning trial. The movement produced by the cerebellar module has a duration of 800 ms, and brings

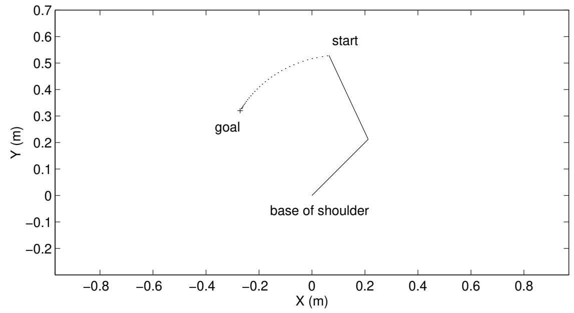

Figure 6: Final plant behavior (trial 27). The cerebellar control module generates the entire movement to the target.

the arm approximately 70% of the distance to the target. The primary contributors to this movement are the APGs that activate the elbow agonist (*APGa* in Figure 5) and the biarticulate agonist muscle (*APGe*), with a less significant role played by the shoulder agonist (*APGc*).

Subsequent movements are due to the bursts generated by the EC system. In generating the first corrective movement, the EC system recruits the APGs driving all three agonist muscles.This is indicated indirectly by the high level of activity of the corresponding climbing fibers (*CFa*, *CFc*, and *CFe*), and by the low activity of the opposing CFs.

During learning, these CF signals cause the original cerebellar-generated movement to be blended with the corrective movements. As seen in Figure 7, this ultimately leads to an increase in the initial burst in all three agonist muscles.

By trial 27, the cerebellar module has learned to generate an appropriate motor pattern for reaching the target (Figures 6 and 7), and no longer requires the aid of the EC module to reach the target. The biarticulate and shoulder agonist APGs (*APGe* and *APGa*) are the primary contributors to this final motor program. Comparing Figures 4 and 6, note that even though the EC module generates a trajectory made up of three discrete hops, the final trajectory produced by the cerebellar module does not reflect this behavior. This happens because the cerebellum does not simply append corrective movements onto its original movement, but rather blends the EC-generated movement into its own.

The cerebellar controller is capable of learning single target movements within about 20-30 trials of experience. In addition, the cerebellum's ability to store and appropriately generalize over a set of targets has been tested. In these experiments, the start and target

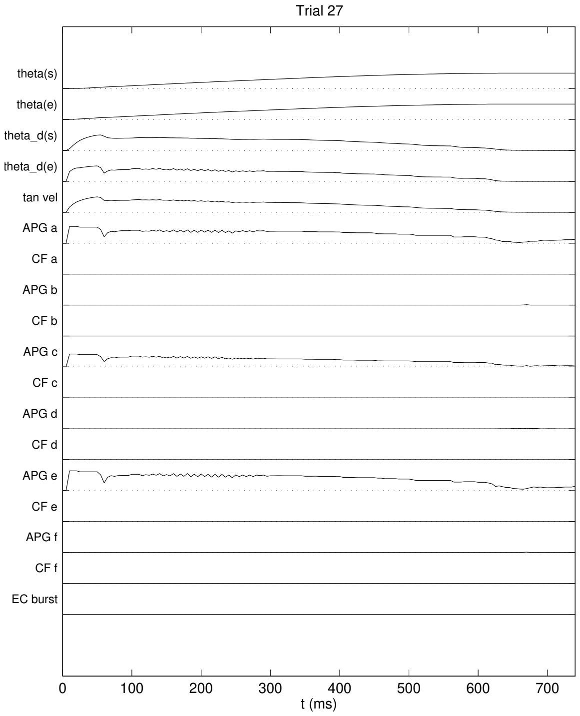

Figure 7: Plant/Neural behavior on trial 27. Trace labels are the same as in Figure 5. The shoulder (a) and biarticulate (e) agonist APGs contribute the most to producing the movement.

positions for a given trial are selected from a uniform distribution over an 80×80 degree range of the joint space. After 10000 learning trials, the cerebellar controller produces movements with an average of 0.7 cm error in the endpoint position (average movement distances are 28 cm). Furthermore, the controller hits the target on 86% of the trials (within 1 cm of the goal), and reaches to within 2 cm of the goal on 99% of the trials.

5 Conclusions and Future Work

A biologically-inspired model of cerebellar control of a 2-DOF planar arm has been presented. In the design of the control system, we have deviated in several ways from more traditional control methods.

First, our model attempts to exploit the non-linearities of the plant, instead of suppressing them. From the point of view of higher control centers, non-linear damping (provided by spinal reflex circuits) causes the muscle to "stick" at low velocities, while still allowing for rapid contractions. This property allows for fast arm movements without ringing near a target location (a problem that is faced with linear systems).

Second, the output of our cerebellar controller is not torque to be applied to a set of joints. Rather, its output falls within a more abstract actuator space. On the engineering side, this allows us to consider multiple actuators affecting individual joints, and multiple joints being driven by a single actuator – without having to rely on a high-quality model of these dependencies. From the biological perspective, this approach is also more satisfying. If one assumes that joint torques are the currency of output from the cerebellum, then it becomes necessary to assume that lower-level mechanisms are performing the complex transformation from joint torque space into muscle space. However, it is precisely the structure of the cerebellum that makes it very well suited to learning such a transformation. In taking advantage of this property, it is not necessary to directly represent joint torques at all.

Finally, our model does not make use of *reference trajectories*, as has been assumed in other models of cerebellar control [7, 11]. More specifically, there is little physiological evidence to support the idea that another subsystem provides to the cerebellum a detailed kinematic plan – a plan which happens to satisfy certain *dynamic* optimality criteria. Rather, we see such a higher-level system as providing a target position and possibly some constraints about the desired path (e.g. a few via points). The actual trajectory taken by the arm, then, arises from the interaction between the controller and the natural dynamics of the arm.

The learning problems addressed by our cerebellar model fall into two categories: structural and temporal credit assignment. The structural credit assignment problem is solved 1) by marking active PF-to-PC synapses as eligible for future update, and 2) by using a spatially-coded CF activity pattern to select the correct subset of marked synapses. The CF activity pattern is a function of proprioceptive inputs that result from the generation of corrective movements. The error-corrective module (the EC controller) produces low quality corrective movements only when the cerebellum is unable to bring the arm to the target location.

The error-corrective teacher concept is one that we feel is fairly general. Consider a set of reflexes that are designed to respond to unexpected events (e.g. not reaching a goal position, unexpected muscle pulls, collisions with the environment, or slipping of a grasp). By "eavesdropping" on these reflexes (via proprioceptive or other sensory inputs), the inferior olive can provide the cerebellum with a signed error-corrective signal. With this information, the cerebellum can learn to produce motor outputs similar to those generated by the reflexes, but in an anticipatory manner – correcting the movement before the reflex-generating conditions can occur. However, the cerebellum does not simply generate the equivalent of the reflex movement just prior to the reflex-triggering event. Rather, the reflexive motor signal is blended with motor signals that have been generated by the cerebellum over the previous several hundred milliseconds.

The temporal credit assignment problem results from two factors: 1) the significant hard-wired delay between the time that the cerebellum generates a motor command and the time that its "effects" are seen at the plant; and 2) the variable time delay between the activation of a parallel fiber and the behavioral event that triggers a corrective movement (thus initiating a CF signal). These problems are addressed in this model through the use of a second-order eligibility trace, which maintains a temporally-smeared memory of recently active PF-to-PC synapses.

Finally, the model's ability to rapidly learn an appropriate motor trajectory is the result of the sparse, expansive encoding provided by the granule cell layer (which give rise to the PFs). Such an encoding ensures that two very different movements do not overlap significantly in their representation. This implies that learning for one movement does not tend to interfere with learning for the other.

In continuing work, we are expanding our model in several directions. First, individual APGs are no longer constrained to control single muscles. Rather, we have begun experimentation with the more general case, in which the cerebellum performs control in a *muscle synergy* space [9]. Each synergy differentially projects to multiple muscles, thus providing a large repertoire of available movement directions.

Second, it has been assumed thus far that target locations are specified in joint space, and that this information is available to the EC system. We are experimenting with an algorithm that assesses positional errors in a visual coordinate system, rather than providing these errors directly in joint space.

Finally, we are interested in considering the role of other regions of the primate brain in solving the control problem. Specifically, we view the motor cortex as working in collaboration with the cerebellum to control the limb. In addition, the premotor areas provide a gateway through which visual information may be cast into a motor-centered representation. Finally, the basal ganglia are seen as providing the machinery for the sequencing of discrete steps during performance of a task.

References

[1] J. S. Albus. A theory of cerebellar function. *Mathematical Biosciences*, 10:25–61, 1971.

[2] A. G. Barto, J. T. Buckingham, and J. C. Houk. A predictive switching model of cerebellar movement control. In D. S. Touretsky, M. C. Mozer, and M. E. Hasselmo, editors, *Advances in Neural Information Processing Systems*, volume 8. MIT Press, Cambridge, Massachusetts, 1995.

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

[4] J. R. Bloedel and J. Courville. Cerebellar afferent systems. In J. Brookhart, V. Mountcastle, V. Brooks, and S. Geiger, editors, *Handbook of Physiology, Sect. 1. The Nervous System. Motor Control*. American Physiological Society, Bethesda, MD, 1981.

[5] R. Gellman, A. R. Gibson, and J. C. Houk. Inferior olivary neurons in the awake cat: Detection of contact and passive body displacement. *Journal of Neurophysiology*, 2:185–198, 1985.

[6] M. Ito. A new physiological concept on cerebellum. *Revue Neurologique*, 146(10):564–569, 1990.

[7] M. Katayamma and M. Kawato. Virtual trajectory and stiffness ellipse during multijoint arm movement predicted by neural inverse models. *Biological Cybernetics*, 69:353–362, 1993.

[8] A. H. Klopf. Brain function and adaptive systems - a heterostatic theory. In *Proceedings of the International Conference on Systems, Man, and Cybernetics*, 1974.

[9] L. E. Miller and J. C. Houk. Motor coordinates in primate red nucleus: Preferential relation to muscle activation versus kinematic variables. *Journal of Physiology - London*, 488(2):533–548, 1995.

[10] W. T. Miller, F. H. Glanz, and L. G. Kraft. CMAC: An associative neural network alternative to backpropagation. In C. Lau, editor, *Neural Networks. Theoretical Foundations and Analysis*, pages 233–240. IEEE Press, New York, NY, 1992.

[11] N. Schweighofer. *Computational Models of the Cerebellum in the Adaptive Control of Movements*. PhD thesis, University of Southern California, 1995.

[12] R. S. Sutton and A. G. Barto. Time-derivative models of pavlovian reinforcement. In M. Gabriel and J. Moore, editors, *Learning and Computational Neuroscience: Foundations of Adaptive Networks*, pages 497–537. MIT Press, 1990.

[13] W. T. Thach, H. Goodkin, and J. Keating. The cerebellum and the adaptive coordination of movement. *Annual Review of Neuroscience*, pages 403–442, 1992.

[14] C. H. Wu, J. C. Houk, K. Y. Young, and L. E. Miller. Nonlinear damping of limb motion. *Multiple Muscle Systems: Biomechanics and Movement Organization*, 1990.