A Forward Model at Purkinje Cell Synapses Facilitates Cerebellar Anticipatory Control

Ivan Herreros-Alonso
SPECS lab
Universitat Pompeu Fabra
Barcelona, Spain
ivan.herreros@upf.edu

Xerxes D. Arsiwalla
SPECS lab
Universitat Pompeu Fabra
Barcelona, Spain

Paul F.M.J. Verschure
SPECS, UPF
Catalan Institution of Research
and Advanced Studies (ICREA)
Barcelona, Spain

Abstract

How does our motor system solve the problem of anticipatory control in spite of a wide spectrum of response dynamics from different musculo-skeletal systems, transport delays as well as response latencies throughout the central nervous system? To a great extent, our highly-skilled motor responses are a result of a reactive feedback system, originating in the brain-stem and spinal cord, combined with a feed-forward anticipatory system, that is adaptively fine-tuned by sensory experience and originates in the cerebellum. Based on that interaction we design the counterfactual predictive control (CFPC) architecture, an anticipatory adaptive motor control scheme in which a feed-forward module, based on the cerebellum, steers an error feedback controller with *counterfactual* error signals. Those are signals that trigger reactions as actual errors would, but that do not code for any current or forthcoming errors. In order to determine the optimal learning strategy, we derive a novel learning rule for the feed-forward module that involves an eligibility trace and operates at the synaptic level. In particular, our eligibility trace provides a mechanism beyond co-incidence detection in that it convolves a history of prior synaptic inputs with error signals. In the context of cerebellar physiology, this solution implies that Purkinje cell synapses should generate eligibility traces using a forward model of the system being controlled. From an engineering perspective, CFPC provides a general-purpose anticipatory control architecture equipped with a learning rule that exploits the full dynamics of the closed-loop system.

1 Introduction

Learning and anticipation are central features of cerebellar computation and function (Bastian, 2006): the cerebellum learns from experience and is able to anticipate events, thereby complementing a reactive feedback control by an anticipatory feed-forward one (Hofstoetter et al., 2002; Herreros and Verschure, 2013). This interpretation is based on a series of anticipatory motor behaviors that originate in the cerebellum. For instance, anticipation is a crucial component of acquired behavior in eye-blink conditioning (Gormezano et al., 1983), a trial by trial learning protocol where an initially neutral stimulus such as a tone or a light (the conditioning stimulus, CS) is followed, after a fixed delay, by a noxious one, such as an air puff to the eye (the unconditioned stimulus, US). During early trials, a protective unconditioned response (UR), a blink, occurs reflexively in a feedback manner following the US. After training though, a well-timed anticipatory blink (the conditioned response, CR) precedes the US. Thus, learning results in the (partial) transference from an initial feedback action to an anticipatory (or predictive) feed-forward one. Similar responses occur during anticipatory postural adjustments, which are postural changes that precede voluntary motor movements, such as raising an arm while standing (Massion, 1992). The goal of these anticipatory adjustments is to counteract the postural and equilibrium disturbances that voluntary movements introduce. These

30th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

behaviors can be seen as feedback reactions to events that after learning have been transferred to feed-forward actions anticipating the predicted events.

Anticipatory feed-forward control can yield high performance gains over feedback control whenever the feedback loop exhibits transmission (or transport) delays (Jordan, 1996). However, even if a plant has negligible transmission delays, it may still have sizable inertial latencies. For example, if we apply a force to a visco-elastic plant, its peak velocity will be achieved after a certain delay; i.e. the velocity itself will lag the force. An efficient way to counteract this lag will be to apply forces anticipating changes in the desired velocity. That is, anticipation can be beneficial even when one can act instantaneously on the plant. Given that, here we address two questions: what is the optimal strategy to learn anticipatory actions in a cerebellar-based architecture? and how could it be implemented in the cerebellum?

To answer that we design the counterfactual predictive control (CFPC) scheme, a cerebellar-based adaptive-anticipatory control architecture that learns to anticipate performance errors from experience. The CFPC scheme is motivated from neuro-anatomy and physiology of eye-blink conditioning. It includes a reactive controller, which is an output-error feedback controller that models brain stem reflexes actuating on eyelid muscles, and a feed-forward adaptive component that models the cerebellum and learns to associate its inputs with the error signals driving the reactive controller. With CFPC we propose a generic scheme in which a feed-forward module enhances the performance of a reactive error feedback controller steering it with signals that facilitate anticipation, namely, with *counterfactual errors*. However, within CFPC, even if these counterfactual errors that enable predictive control are learned based on past errors in behavior, they do not reflect any current or forthcoming error in the ongoing behavior.

In addition to eye-blink conditioning and postural adjustments, the interaction between reactive and cerebellar-dependent acquired anticipatory behavior has also been studied in paradigms such as visually-guided smooth pursuit eye movements (Lisberger, 1987). All these paradigms can be abstracted as tasks in which the same predictive stimuli and disturbance or reference signal are repeatedly experienced. In accordance to that, we operate our control scheme in trial-by-trial (batch) mode. With that, we derive a learning rule for anticipatory control that modifies the well-known least-mean-squares/Widrow-Hoff rule with an eligibility trace. More specifically, our model predicts that to facilitate learning, parallel fibers to Purkinje cell synapses implement a forward model that generates an eligibility trace. Finally, to stress that CFPC is not specific to eye-blink conditioning, we demonstrate its application with a smooth pursuit task.

2 Methods

2.1 Cerebellar Model
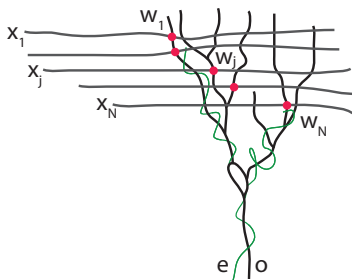
Figure 1: Anatomical scheme of a Cerebellar Purkinje cell. The x_j denote parallel fiber inputs to Purkinje synapses (in red) with weights w_j. o denotes the output of the Purkinje cell. The error signal e, through the climbing fibers (in green), modulates synaptic weights.

We follow the simplifying approach of modeling the cerebellum as a linear adaptive filter, while focusing on computations at the level of the Purkinje cells, which are the main output cells of the cerebellar cortex (Fujita, 1982; Dean et al., 2010). Over the mossy fibers, the cerebellum receives a wide range of inputs. Those inputs reach Purkinke cells via parallel fibers (Fig. 1), that cross

2

dendritic trees of Purkinje cells in a ratio of up to 1.5×10^6 parallel fiber synapses per cell (Eccles et al., 1967). We denote the signal carried by a particular fiber as x_j, $j \in [1, G]$, with G equal to the total number of inputs fibers. These inputs from the mossy/parallel fiber pathway carry contextual information (interoceptive or exteroceptive) that allows the Purkinje cell to generate a functional output. We refer to these inputs as *cortical bases*, indicating that they are localized at the cerebellar cortex and that they provide a repertoire of states and inputs that the cerebellum combines to generate its output o. As we will develop a discrete time analysis of the system, we use n to indicate time (or time-step). The output of the cerebellum at any time point n results from a weighted sum of those cortical bases. w_j indicates the weight or synaptic efficacy associated with the fiber j. Thus, we have $\boldsymbol{x}[n] = [x_1[n], \ldots, x_G[n]]^{\mathsf{T}}$ and $\boldsymbol{w}[n] = [w_1[n], \ldots, w_G[n]]^{\mathsf{T}}$ (where the transpose, T, indicates that $\boldsymbol{x}[n]$ and $\boldsymbol{w}[n]$ are column vectors) containing the set of inputs and synaptic weights at time n, respectively, which determine the output of the cerebellum according to

$$o[n] = \boldsymbol{x}[n]^{\mathsf{T}} \boldsymbol{w}[n] \tag{1}$$

The adaptive feed-forward control of the cerebellum stems from updating the weights according to a rule of the form

$$\Delta w_j[n+1] = f(x_j[n], \ldots, x_j[1], e[n], \Theta) \tag{2}$$

where Θ denotes global parameters of the learning rule; $x_j[n], \ldots, x_j[1]$, the history of its pre-synaptic inputs of synapse j; and $e[n]$, an error signal that is the same for all synapses, corresponding to the difference between the desired, r, and the actual output, y, of the controlled plant. Note that in drawing an analogy with the eye-blink conditioning paradigm, we use the simplifying convention of considering the noxious stimulus (the air-puff) as a reference, r, that indicates that the eyelids should close; the closure of the eyelid as the output of the plant, y; and the sensory response to the noxious stimulus as an error, e, that encodes the difference between the desired, r, and the actual eyelid closures, y. Given this, we advance a new learning rule, f, that achieves optimal performance in the context of eye-blink conditioning and other cerebellar learning paradigms.

2.2 Cerebellar Control Architecture
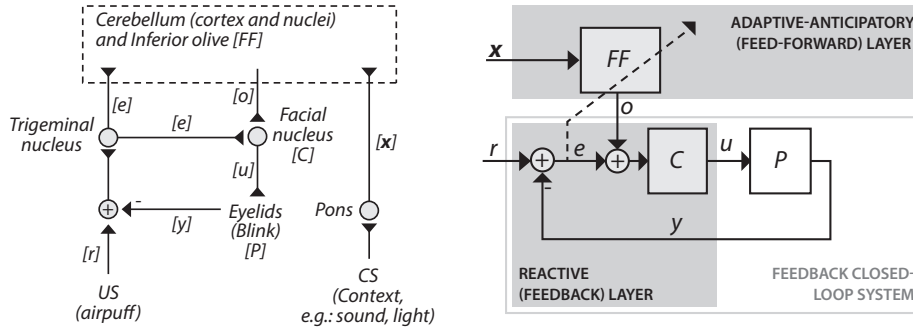
Figure 2: Neuroanatomy of eye-blink conditioning and the CFPC architecture. *Left*: Mapping of signals to anatomical structures in eye-blink conditioning (De Zeeuw and Yeo, 2005); regular arrows indicate external inputs and outputs, arrows with inverted heads indicate neural pathways. *Right*: CFPC architecture. Note that the feedback controller, C, and the feed-forward module, FF, belong to the control architecture, while the plant, P, denotes an object controlled. Other abbreviations: r, reference signal; y, plant's output; e, output error; $\mathbf{x}$, basis signals; o, feed-forward signal; and u, motor command.

We embed the adaptive filter cerebellar module in a layered control architecture, namely the CFPC architecture, based on the interaction between brain stem motor nuclei driving motor reflexes and the cerebellum, such as the one established between the cerebellar microcircuit responsible for conditioned responses and the brain stem reflex circuitry that produces unconditioned eye-blinks (Hesslow and Yeo, 2002) (Fig. 2 *left*). Note that in our interpretation of this anatomy we assume that cerebellar output, o, feeds the lower reflex controller (Fig. 2 *right*). Put in control theory terms, within the CFPC scheme an adaptive feed-forward layer supplements a negative feedback controller steering it with feed-forward signals.

3

Our architecture uses a single-input single-output negative-feedback controller. The controller receives as input the output error $e = r - y$. For the derivation of the learning algorithm, we assume that both plant and controller are linear and time-invariant (LTI) systems. Importantly, the feedback controller and the plant form a reactive closed-loop system, that mathematically can be seen as a system that maps the reference, r, into the plant's output, y. A feed-forward layer that contains the above-mentioned cerebellar model provides the negative feedback controller with an additional input signal, o. We refer to o as a *counter-factual* error signal, since although it *mechanistically* drives the negative feedback controller analogously to an error signal it is not an *actual* error. The counterfactual error is generated by the feed-forward module that receives an output error, e, as its teaching signal. Notably, from the point of view of the reactive layer closed-loop system, o can also be interpreted as a signal that offsets r. In other words, even if r remains the reference that sets the target of behavior, $r + o$ functions as the *effective* reference that drives the closed-loop system.

3 Results

3.1 Derivation of the gradient descent update rule for the cerebellar control architecture

We apply the CFPC architecture defined in the previous section to a task that consists in following a finite reference signal $\mathbf{r} \in \mathbb{R}^N$ that is repeated trial-by-trial. To analyze this system, we use the discrete time formalism and assume that all components are linear time-invariant (LTI). Given this, both reactive controller and plant can be lumped together into a closed-loop dynamical system, that can be described with the dynamics $\mathbf{A}$, input $\mathbf{B}$, measurement $\mathbf{C}$ and feed-through $\mathbf{D}$ matrices. In general, these matrices describe how the state of a dynamical system autonomously evolves with time, $\mathbf{A}$; how inputs affect system states, $\mathbf{B}$; how states are mapped into outputs, $\mathbf{C}$; and how inputs instantaneously affect the system's output $\mathbf{D}$ (Astrom and Murray, 2012). As we consider a reference of a finite length N, we can construct the N-by-N transfer matrix $\mathcal{T}$ as follows (Boyd, 2008)

$$\mathcal{T} = \begin{bmatrix} \boldsymbol{D} & 0 & 0 & \dots & 0 \\ \boldsymbol{CB} & \boldsymbol{D} & 0 & \dots & 0 \\ \boldsymbol{CAB} & \boldsymbol{CB} & \boldsymbol{D} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{CA^{N-2}B} & \boldsymbol{CA^{N-3}B} & \boldsymbol{CA^{N-4}B} & \dots & \boldsymbol{D} \end{bmatrix}$$

With this transfer matrix we can map any given reference $\mathbf{r}$ into an output $\mathbf{y}_r$ using $\mathbf{y}_r = \mathcal{T}\mathbf{r}$, obtaining what would have been the complete output trajectory of the plant on an entirely feedback-driven trial. Note that the first column of $\mathcal{T}$ contains the impulse response curve of the closed-loop system, while the rest of the columns are obtained shifting that impulse response down. Therefore, we can build the transfer matrix $\mathcal{T}$ either in a model-based manner, deriving the state-space characterization of the closed-loop system, or in measurement-based manner, measuring the impulse response curve. Additionally, note that $(\mathbf{I} - \mathcal{T})\mathbf{r}$ yields the error of the feedback control in following the reference, a signal which we denote with $\mathbf{e}_0$.

Let $\mathbf{o} \in \mathbb{R}^N$ be the entire feed-forward signal for a given trial. Given commutativity, we can consider that from the point of view of the closed-loop system o is added directly to the reference $\mathbf{r}$, (Fig. 2 *right*). In that case, we can use $\mathbf{y} = \mathcal{T}(\mathbf{r} + \mathbf{o})$ to obtain the output of the closed-loop system when it is driven by both the reference and the feed-forward signal. The feed-forward module only outputs linear combinations of a set of bases. Let $\mathbf{X} \in \mathbb{R}^{N \times G}$ be a matrix with the content of the G bases during all the N time steps of a trial. The feed-forward signal becomes $\mathbf{o} = \mathbf{X}\mathbf{w}$, where $\mathbf{w} \in \mathbb{R}^G$ contains the mixing weights. Hence, the output of the plant given a particular $\mathbf{w}$ becomes $\mathbf{y} = \mathcal{T}(\mathbf{r} + \mathbf{X}\mathbf{w})$.

We implement learning as the process of adjusting the weights $\mathbf{w}$ of the feed-forward module in a trial-by-trial manner. At each trial the same reference signal, $\mathbf{r}$, and bases, $\mathbf{X}$, are repeated. Through learning we want to converge to the optimal weight vector $\mathbf{w}^*$ defined as

$$\mathbf{w}^* = \arg\min_w c(\mathbf{w}) = \arg\min_w \frac{1}{2}\mathbf{e}^\mathsf{T}\mathbf{e} = \arg\min_w \frac{1}{2}(\mathbf{r} - \mathcal{T}(\mathbf{r} + \mathbf{X}\mathbf{w}))^\mathsf{T}(\mathbf{r} - \mathcal{T}(\mathbf{r} + \mathbf{X}\mathbf{w})) \quad (3)$$

where c indicates the objective function to minimize, namely the L_2 norm or sum of squared errors. With the substitution $\tilde{\mathbf{X}} = \mathcal{T}\mathbf{X}$ and using $\mathbf{e}_0 = (\mathbf{I} - \mathcal{T})\mathbf{r}$, the minimization problem can be cast as a

4

canonical linear least-squares problem:

$$\mathbf{w}^* = \arg\min_w \frac{1}{2}(\mathbf{e}_0 - \tilde{\mathbf{X}}\mathbf{w})^\intercal(\mathbf{e}_0 - \tilde{\mathbf{X}}\mathbf{w}) \tag{4}$$

One the one hand, this allows to directly find the least squares solution for $\mathbf{w}^*$, that is, $\mathbf{w}^* = \tilde{\mathbf{X}}^\dagger\mathbf{e}_0$, where $\dagger$ denotes the Moore-Penrose pseudo-inverse. On the other hand, and more interestingly, with $\mathbf{w}[k]$ being the weights at trial k and having $\mathbf{e}[k] = \mathbf{e}_0 - \tilde{\mathbf{X}}\mathbf{w}[k]$, we can obtain the gradient of the error function at trial k with relation to w as follows:

$$\nabla_w c = -\tilde{\mathbf{X}}^\intercal\mathbf{e}[k] = -\mathbf{X}^\intercal\mathcal{T}^\intercal\mathbf{e}[k]$$

Thus, setting η as a properly scaled learning rate (the only global parameter Θ of the rule), we can derive the following gradient descent strategy for the update of the weights between trials:

$$\mathbf{w}[k+1] = \mathbf{w}[k] + \eta\mathbf{X}^\intercal\mathcal{T}^\intercal\mathbf{e}[k] \tag{5}$$

This solves for the learning rule f in eq. 2. Note that f is consistent with both the cerebellar anatomy (Fig. 2*left*) and the control architecture (Fig. 2*right*) in that the feed-forward module/cerebellum only requires two signals to update its weights/synaptic efficacies: the basis inputs, $\mathbf{X}$, and error signal, $\mathbf{e}$.

3.2 $\mathcal{T}^\intercal$ facilitates a synaptic eligibility trace

The standard least mean squares (LMS) rule (also known as Widrow-Hoff or decorrelation learning rule) can be represented in its batch version as $\mathbf{w}[k+1] = \mathbf{w}[k] + \eta\mathbf{X}^\intercal\mathbf{e}[k]$. Hence, the only difference between the batch LMS rule and the one we have derived is the insertion of the matrix factor $\mathcal{T}^\intercal$. Now we will show how this factor acts as a filter that computes an eligibility trace at each weight/synapse. Note that the update of a single weight, according Eq. 5 becomes

$$w_j[k+1] = w_j[k] + \eta\mathbf{x}_j^\intercal\mathcal{T}^\intercal\mathbf{e}[k] \tag{6}$$

where $\mathbf{x}_j$ contains the sequence of values of the cortical basis j during the entire trial. This can be rewritten as

$$w_j[k+1] = w_j[k] + \eta\mathbf{h}_j^\intercal\mathbf{e}[k] \tag{7}$$

with $\mathbf{h}_j \equiv \mathcal{T}\mathbf{x}_j$. The above inner product can be expressed as a sum of scalar products

$$w_j[k+1] = w_j[k] + \eta\sum_{n=1}^{N}\mathbf{h}_j[n]\mathbf{e}[k,n] \tag{8}$$

where n indexes the within trial time-step. Note that $\mathbf{e}[k]$ in Eq. 7 refers to the whole error signal at trial k whereas $\mathbf{e}[k,n]$ in Eq. 8 refers to the error value in the n-th time-step of the trial k. It is now clear that each $\mathbf{h}_j[n]$ weighs how much an error arriving at time n should modify the weight w_j, which is precisely the role of an eligibility trace. Note that since $\mathcal{T}$ contains in its columns/rows shifted repetitions of the impulse response curve of the closed-loop system, the eligibility trace codes at any time n, the convolution of the sequence of previous inputs with the impulse-response curve of the reactive layer closed-loop. Indeed, in each synapse, the eligibility trace is generated by a forward model of the closed-loop system that is exclusively driven by the basis signal.

Consequently, our main result is that by deriving a gradient descent algorithm for the CFPC cerebellar control architecture we have obtained an exact definition of the suitable eligibility trace. That definition guarantees that the set of weights/synaptic efficacies are updated in a locally optimal manner in the weights' space.

3.3 On-line gradient descent algorithm

The trial-by-trial formulation above allowed for a straightforward derivation of the (batch) gradient descent algorithm. As it lumped together all computations occurring in a same trial, it accounted for time within the trial implicitly rather than explicitly: one-dimensional time-signals were mapped onto points in a high-dimensional space. However, after having established the gradient descent algorithm, we can implement the same rule in an on-line manner, dropping the repetitiveness assumption inherent to trial-by-trial learning and performing all computations locally in time. Each weight/synapse must

5

have a process associated to it that outputs the eligibility trace. That process passes the incoming (unweighted) basis signal through a (forward) model of the closed-loop as follows:

$$\begin{aligned} \boldsymbol{s}_j[n+1] &= \boldsymbol{A}\boldsymbol{s}_j[n] + \boldsymbol{B}x_j[n] \\ h_j[n] &= \boldsymbol{C}\boldsymbol{s}_j[n] + \boldsymbol{D}x_j[n] \end{aligned}$$

where matrices $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$ and $\boldsymbol{D}$ refer to the closed-loop system (they are the same matrices that we used to define the transfer matrix $\mathcal{T}$), and $\boldsymbol{s}_j[n]$ is the state vector of the forward model of the synapse j at time-step n. In practice, each "synaptic" forward model computes what would have been the effect of having driven the closed-loop system with each basis signal alone. Given the superposition principle, the outcome of that computation can also be interpreted as saying that $h_j[n]$ indicates what would have been the displacement over the current output of the plant, $y[n]$, achieved feeding the closed-loop system with the basis signal x_j. The process of weight update is completed as follows:

$$w_j[n+1] = w_j[n] + \eta h_j[n]e[n] \tag{9}$$

At each time step n, the error signal $e[n]$ is multiplied by the current value of the eligibility trace $h_j[n]$, scaled by the learning rate η, and subtracted to the current weight $w_j[n]$. Therefore whereas the contribution of each basis to the output of the adaptive filter depends only on its current value and weight, the change in weight depends on the current and past values passed through a forward model of the closed-loop dynamics.

3.4 Simulation of a visually-guided smooth pursuit task

We demonstrate the CFPC approach in an example of a visual smooth pursuit task in which the eyes have to track a target moving on a screen. Even though the simulation does not capture all the complexity of a smooth pursuit task, it illustrates our anticipatory control strategy. We model the plant (eye and ocular muscles) with a two-dimensional linear filter that maps motor commands into angular positions. Our model is an extension of the model in (Porrill and Dean, 2007), even though in that work the plant was considered in the context of the vestibulo-ocular reflex. In particular, we use a chain of two leaky integrators: a slow integrator with a relaxation constant of 100 ms drives the eyes back to the rest position; the second integrator, with a fast time constant of 3 ms ensures that the change in position does not occur instantaneously. To this basic plant, we add a reactive control layer modeled as a proportional-integral (PI) error-feedback controller, with proportional gain k_p and integral gain k_i. The control loop includes a 50 ms delay in the error feedback, to account for both the actuation and the sensing latency. We choose gains such that reactive tracking lags the target by approximately 100 ms. This gives $k_p = 20$ and $k_i = 100$. To complete the anticipatory and adaptive control architecture, the closed-loop system is supplemented by the feed-forward module.
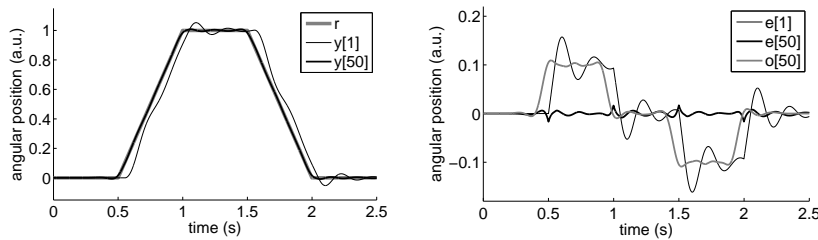
Figure 3: Behavior of the system. Left: Reference ($\mathbf{r}$) and output of the system before ($\mathbf{y}[1]$) and after learning ($\mathbf{y}[50]$). Right: Error before $\mathbf{e}[1]$ and after learning $\mathbf{e}[50]$ and output acquired by cerebellar/feed-forward component ($\mathbf{o}[50]$)

The architecture implementing the forward model-based gradient descent algorithm is applied to a task structured in trials of 2.5 sec duration. Within each trial, a target remains still at the center of the visual scene for a duration 0.5 sec, next it moves rightwards for 0.5 sec with constant velocity, remains still for 0.5 sec and repeats the sequence of movements in reverse, returning to the center. The cerebellar component receives 20 Gaussian basis signals ($\mathbf{X}$) whose receptive fields are defined in the temporal domain, relative to trial onset, with a width (standard-deviation) of 50 ms and spaced by 100 ms. The whole system is simulated using a 1 ms time-step. To construct the matrix $\mathcal{T}$ we computed closed-loop system impulse response.

6

At the first trial, before any learning, the output of the plant lags the reference signal by approximately 100 ms converging to the position only when the target remains still for about 300 ms (Fig. 3 *left*). As a result of learning, the plant's behavior shifts from a reactive to an anticipatory mode, being able to track the reference without any delay. Indeed, the error that is sizable during the target displacement before learning, almost completely disappears by the 50th trial (Fig. 3 *right*). That cancellation results from learning the weights that generate a feed-forward predictive signal that leads the changes in the reference signal (onsets and offsets of target movements) by approximately 100 ms (Fig. 3 *right*). Indeed, convergence of the algorithm is remarkably fast and by trial 7 it has almost converged to the optimal solution (Fig. 4).
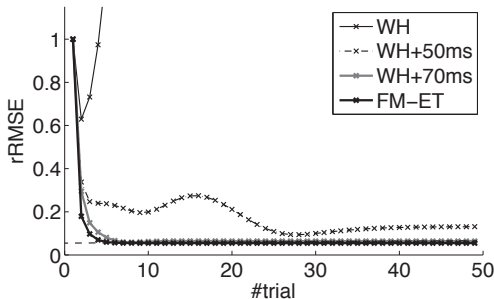
Figure 4: Performance achieved with different learning rules. Representative learning curves of the forward model-based eligibility trace gradient descent (FM-ET), the simple Widrow-Hoff (WH) and the Widrow-Hoff algorithm with a delta-eligibility trace matched to error feedback delay (WH+50 ms) or with an eligibility trace exceeding that delay by 20 ms (WH+70 ms). Error is quantified as the relative root mean-squared error (rRMSE), scaled proportionally to the error in the first trial. Error of the optimal solution, obtained with $\boldsymbol{w}^* = (\mathcal{T}\boldsymbol{X})^\dagger \boldsymbol{e}_0$, is indicated with a dashed line.

To assess how much our forward-model-based eligibility trace contributes to performance, we test three alternative algorithms. In both cases we employ the same control architecture, changing the plasticity rule such that we either use no eligibility trace, thus implementing the basic Widrow-Hoff learning rule, or use the Widrow-Hoff rule extended with a delta-function eligibility trace that matches the latency of the error feedback (50 ms) or slightly exceeds it (70 ms). Performance with the basic WH model worsens rapidly whereas performance with the WH learning rule using a "pure delay" eligibility trace matched to the transport delay improves but not as fast as with the forward-model-based eligibility trace (Fig. 4). Indeed, in this case, the best strategy for implementing a delayed delta eligibility trace is setting a delay exceeding the transport delay by around 20 ms, thus matching the peak of the impulse response. In that case, the system performs almost as good as with the forward-model eligibility trace (70 ms). This last result implies that, even though the literature usually emphasizes the role of transport delays, eligibility traces also account for response lags due to intrinsic dynamics of the plant.

To summarize our results, we have shown with a basic simulation of a visual smooth pursuit task that generating the eligibility trace by means of a forward model ensures convergence to the optimal solution and accelerates learning by guaranteeing that it follows a gradient descent.

4 Discussion

In this paper we have introduced a novel formulation of cerebellar anticipatory control, consistent with experimental evidence, in which a forward model has emerged naturally at the level of Purkinje cell synapses. From a machine learning perspective, we have also provided an optimality argument for the derivation of an eligibility trace, a construct that was often thought of in more heuristic terms as a mechanism to bridge time-delays (Barto et al., 1983; Shibata and Schaal, 2001; McKinstry et al., 2006).

The first seminal works of cerebellar computational models emphasized its role as an associative memory (Marr, 1969; Albus, 1971). Later, the cerebellum was investigates as a device processing correlated time signals(Fujita, 1982; Kawato et al., 1987; Dean et al., 2010). In this latter framework,

7

the use of the computational concept of an eligibility trace emerged as a heuristic construct that allowed to compensate for transmission delays in the circuit(Kettner et al., 1997; Shibata and Schaal, 2001; Porrill and Dean, 2007), which introduced lags in the cross-correlation between signals. Concretely, that was referred to as the problem of *delayed error feedback*, due to which, by the time an error signal reaches a cell, the synapses accountable for that error are no longer the ones currently active, but those that were active at the time when the motor signals that caused the actual error were generated. This view has however neglected the fact that beyond transport delays, response dynamics of physical plants also influence how past pre-synaptic signals could have related to the current output of the plant. Indeed, for a linear plant, the impulse-response function of the plant provides the complete description of how inputs will drive the system, and as such, integrates transmission delays as well as the dynamics of the plant. Recently,

Even though cerebellar microcircuits have been used as models for building control architectures, e.g., the feedback-error learning model (Kawato et al., 1987), our CFPC is novel in that it links the cerebellum to the input of the feedback controller, ensuring that the computational features of the feedback controller are exploited at all times. Within the domain of adaptive control, there are remarkable similarities at the functional level between CFPC and iterative learning control (ILC) (Amann et al., 1996), which is an input design technique for learning optimal control signals in repetitive tasks. The difference between our CFPC and ILC lies in the fact that ILC controllers directly learn a control signal, whereas, the CFPC learns a conterfactual error signal that steers a feedback controller. However the similarity between the two approaches can help for extending CFPC to more complex control tasks.

With our CFPC framework, we have modeled the cerebellar system at a very high level of abstraction: we have not included bio-physical constraints underlying neural computations, obviated known anatomical connections such as the cerebellar nucleo-olivary inhibition (Bengtsson and Hesslow, 2006; Herreros and Verschure, 2013) and made simplifications such as collapsing cerebellar cortex and nuclei into the same computational unit. On the one hand, such a choice of high-level abstraction may indeed be beneficial for deriving general-purpose machine learning or adaptive control algorithms. On the other hand, it is remarkable that in spite of this abstraction our framework makes fine-grained predictions at the micro-level of biological processes. Namely, that in a cerebellar microcircuit (Apps and Garwicz, 2005), the response dynamics of secondary messengers (Wang et al., 2000) regulating plasticity of Purkinje cell synapses to parallel fibers must mimic the dynamics of the motor system being controlled by that cerebellar microcircuit. Notably, the logical consequence of this prediction, that different Purkinje cells should display different plasticity rules according to the system that they control, has been validated recording single Purkinje cells in vivo (Suvrathan et al., 2016).

In conclusion, we find that a normative interpretation of plasticity rules in Purkinje cell synapses emerges from our systems level CFPC computational architecture. That is, in order to generate optimal eligibility traces, synapses must include a forward model of the controlled subsystem. This conclusion, in the broader picture, suggests that synapses are not merely components of multiplicative gains, but rather the loci of complex dynamic computations that are relevant from a functional perspective, both, in terms of optimizing storage capacity (Benna and Fusi, 2016; Lahiri and Ganguli, 2013) and fine-tuning learning rules to behavioral requirements.

Acknowledgments

The research leading to these results has received funding from the European Commission's Horizon 2020 socSMC project (socSMC-641321H2020-FETPROACT-2014) and by the European Research Council's CDAC project (ERC-2013-ADG 341196).

References

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

Amann, N., Owens, D. H., and Rogers, E. (1996). Iterative learning control for discrete-time systems with exponential rate of convergence. *IEE Proceedings-Control Theory and Applications*, 143(2):217–224.

Apps, R. and Garwicz, M. (2005). Anatomical and physiological foundations of cerebellar information processing. *Nature reviews. Neuroscience*, 6(4):297–311.

Astrom, K. J. and Murray, R. M. (2012). *Feedback Systems: An Introduction for Scientists and Engineers*. Princeton university press.

8

Barto, A. G., Sutton, R. S., and Anderson, C. W. (1983). Neuronlike adaptive elements that can solve difficult learning control problems. *IEEE transactions on systems, man, and cybernetics*, SMC-13(5):834–846.

Bastian, A. J. (2006). Learning to predict the future: the cerebellum adapts feedforward movement control. *Current Opinion in Neurobiology*, 16(6):645–649.

Bengtsson, F. and Hesslow, G. (2006). Cerebellar control of the inferior olive. *Cerebellum (London, England)*, 5(1):7–14.

Benna, M. K. and Fusi, S. (2016). Computational principles of synaptic memory consolidation. *Nature neuroscience*.

Boyd, S. (2008). Introduction to linear dynamical systems. Online Lecture Notes.

De Zeeuw, C. I. and Yeo, C. H. (2005). Time and tide in cerebellar memory formation. *Current opinion in neurobiology*, 15(6):667–74.

Dean, P., Porrill, J., Ekerot, C.-F., and Jörntell, H. (2010). The cerebellar microcircuit as an adaptive filter: experimental and computational evidence. *Nature reviews. Neuroscience*, 11(1):30–43.

Eccles, J., Ito, M., and Szentágothai, J. (1967). *The cerebellum as a neuronal machine*. Springer Berlin.

Fujita, M. (1982). Adaptive filter model of the cerebellum. *Biological cybernetics*, 45(3):195–206.

Gormezano, I., Kehoe, E. J., and Marshall, B. S. (1983). Twenty years of classical conditioning with the rabbit.

Herreros, I. and Verschure, P. F. M. J. (2013). Nucleo-olivary inhibition balances the interaction between the reactive and adaptive layers in motor control. *Neural Networks*, 47:64–71.

Hesslow, G. and Yeo, C. H. (2002). The functional anatomy of skeletal conditioning. In *A neuroscientist's guide to classical conditioning*, pages 86–146. Springer.

Hofstoetter, C., Mintz, M., and Verschure, P. F. (2002). The cerebellum in action: a simulation and robotics study. *European Journal of Neuroscience*, 16(7):1361–1376.

Jordan, M. I. (1996). Computational aspects of motor control and motor learning. In *Handbook of perception and action*, volume 2, pages 71–120. Academic Press.

Kawato, M., Furukawa, K., and Suzuki, R. (1987). A hierarchical neural-network model for control and learning of voluntary movement. *Biological Cybernetics*, 57(3):169–185.

Kettner, R. E., Mahamud, S., Leung, H. C., Sitkoff, N., Houk, J. C., Peterson, B. W., and Barto, a. G. (1997). Prediction of complex two-dimensional trajectories by a cerebellar model of smooth pursuit eye movement. *Journal of neurophysiology*, 77:2115–2130.

Lahiri, S. and Ganguli, S. (2013). A memory frontier for complex synapses. In *Advances in neural information processing systems*, pages 1034–1042.

Lisberger, S. (1987). Visual Motion Processing And Sensory-Motor Integration For Smooth Pursuit Eye Movements. *Annual Review of Neuroscience*, 10(1):97–129.

Marr, D. (1969). A theory of cerebellar cortex. *The Journal of physiology*, 202(2):437–470.

Massion, J. (1992). Movement, posture and equilibrium: Interaction and coordination. *Progress in Neurobiology*, 38(1):35–56.

McKinstry, J. L., Edelman, G. M., and Krichmar, J. L. (2006). A cerebellar model for predictive motor control tested in a brain-based device. *Proceedings of the National Academy of Sciences of the United States of America*, 103(9):3387–3392.

Porrill, J. and Dean, P. (2007). Recurrent cerebellar loops simplify adaptive control of redundant and nonlinear motor systems. *Neural computation*, 19(1):170–193.

Shibata, T. and Schaal, S. (2001). Biomimetic smooth pursuit based on fast learning of the target dynamics. In *Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on*, volume 1, pages 278–285. IEEE.

Suvrathan, A., Payne, H. L., and Raymond, J. L. (2016). Timing rules for synaptic plasticity matched to behavioral function. *Neuron*, 92(5):959–967.

Wang, S. S.-H., Denk, W., and Häusser, M. (2000). Coincidence detection in single dendritic spines mediated by calcium release. *Nature neuroscience*, 3(12):1266–1273.

9