2013 IEEE/RSJ International Conference on
Intelligent Robots and Systems (IROS)
November 3-7, 2013. Tokyo, Japan

Speed generalization capabilities of a cerebellar model on a rapid navigation task

Ivan Herreros [‡], Giovanni Maffei [‡], Santiago Brandi [‡], Martí Sánchez-Fibla [‡] and Paul F.M.J. Verschure [‡,§]
{ivan.herreros,giovanni.maffei,santiago.brandi,marti.sanchez,paul.verschure}@upf.edu

Abstract— **The cerebellum is a brain structure necessary for skilled motor behaviour and has a well understood and repetitive architecture. Such an architecture inspired the Marr-Albus-Ito theory of cerebellar learning, that provides an explanation for the acquisition of motor skills by the cerebellum. Numerous computational models inspired in such a theory have already been employed in robotic tasks. Here we look into one of the suggested roles of the cerebellum, the replacement of reflexes by anticipatory actions and we apply it to a robot navigation task. The acquisition of anticipatory actions has been thoroughly studied in the field of classical conditioning. Of particular interest is the so-called CS-intensity effect, an effect that links the rapidity of execution of an anticipatory protective action, the Conditioned Response (CR), to the intensity of a predictive signal, the Conditioning Stimulus (CS). We propose that the CS-intensity effect implements a built-in sensory-motor contingency that allows to carry over a skill learned in a safe and easy context, e.g., turning at slow velocity, to a more difficult one, e.g., a turning at a faster speed. We demonstrate this hypothesis in a series of experiments where a robot has to navigate a track that has a turn. We show that after being trained at a slow velocity, by means of the CS-intensity effect, the cerebellar controller modulates the turning such that its onset anticipates as the robot speed increases. Ultimately, through incremental learning, this generalization allows the robot to learn to navigate the track at its maximum speed.**

I. INTRODUCTION

Since the first theories of cerebellar learning were proposed [1], [2], computational models of the cerebellum have been applied to robotic control tasks [3]. The aim of such implementations has been not only theoretical, to validate the functionality of cerebellar theories [4], [5], [6], [7], but also practical, since computational models of the cerebellum perform competitively in some domains (e.g. bipedal walking [8], robotic arm control [9]). Competitive applications are often obtained with models that abstract away from the cerebellar physiology. Such models reproduce 1) the overall cerebellar architecture at the level of the information pathways and 2) the error-based learning rule. However poor attention is devoted to the fine-grained dynamics of the cerebellar computation. We believe that as much as the robotic community has gained from the implementation of control systems blueprinted from the cerebellar architecture, it will gain from reproducing the dynamics of the real cerebellum. Here we look into the dynamics of anticipatory reflexes controlled by the cerebellum as studied in classical conditioning. More concretely, we look into the CS-intensity effect, that links the speed and amplitude of execution of an anticipatory reflex to the intensity of the cue signal [10], [11]. We show that the CS-intensity effect implements a sensory-motor contingency that allows generalizing a learned motor skill along different speeds of execution. Finally, we demonstrate that this generalization enables to safely learn the rapid execution of a motor action, namely, to perform a turn at high speed avoiding collisions.

In classical conditioning the experimenter sets up a contingency between a neutral Conditioning Stimulus (CS) and a noxious Unconditioned Stimulus (US) such that the CS becomes a predictor for an upcoming threat [12]. Initially, the US triggers an innate and reflex-like protective Uncoditioned Response (UR) that after training will be preceded by a similar anticipatory action, i.e., the Conditioned Respone (CR) . For instance, in eye-blink conditioning a usual setup has a tone preceding a mild electric shock to the peri-orbital area of an animal by a time interval below a second [13] that elicits a protective eyeblink. After a number of repetitions of such paired $CS - US$ presentations, animals develop an anticipatory blink, the CR, aiming to mitigate the harm caused by the US.

Here we have translated the delay eyeblink paradigm to a collision avoidance task. In such a task, a robot has to traverse a track avoiding collisions. The track contains a single turn, that is preceded by a series of stripes on the ground. The robot is equipped with sensors through which it detects the marks on the ground (CS) and the proximity to the walls (US). The proximity signal triggers first a reactive turn (UR_t) and, over a certain threshold, a reactive braking (UR_b). Initially, the robot is tested at a velocity for which the reactive control safely avoids collisions even though the close proximity to the wall forces the robot to brake. Thus, firstly, cerebellar learning will be expressed as an anticipatory and smooth turn, the CR, preventing the robot from reducing the velocity.

Our rationale is to apply a sensory motor contingency like the CS-intensity effect to generalize the turn learned at the slower and *safer* velocity to a faster and *dangerous* one, i.e., a velocity for which the reactive controller would not prevent the robot collision. Ultimately, this would allow the robot to learn to navigate the track at higher velocities. In order to map the speed into the intensity of the CS, we use the derivative of the visual signal, i.e., aa analogous to simple optic flow singal. Likewise a faster robot will experience a

Work supported by eSMC FP7-ICT- 270212.

[‡] SPECS, Technology Department, Universitat Pompeu Fabra, Carrer de Roc Boronat 138, 08018 Barcelona, Spain.

[§] ICREA, Institució Catalana de Recerca i Estudis Avançats, Passeig Lluís Companys 23, 08010 Barcelona more intense (albeit shorter) CS.

We use an adaptive filter model of the cerebellum with a de-correlation learning rule [14]. In accordance with this model, the signal generated by the visual input is decomposed on a series of components with different temporal profiles, we call them cortical bases. The cerebellar learning consists then in finding the weights of the linear combination that maps the response of cortical bases into a correct output. In our case, a correct output is the one producing an anticipatory turn that either allows the robot to navigate the track without reducing the velocity or, if this is not possible, to simply stay within the track boundaries avoiding collision.

According to this scheme, when the CS-intensity is increased, the variation of the CR learned at a given intensity depends on the response of the cortical bases to the new CS. For instance, if doubling the intensity of the stimulus only doubles the response of the cortical bases, but does not alter their temporal profile, the cerebellar output will only be scaled in amplitude but not shifted in time. Thus, in practical terms, the reponse of cortical bases responses has to be nonlinear. To achieve this, we implement in our model the signal transduction mechanism that is applied in the input stage of the cerebellar cortex, i.e., at the granular layer [15]. Namely, we assume that the granule cells, the cells that code the cortical bases, act as linear thresholded filters [16] and that their output results from the interaction of a fast excitatory component minus a slow inhibitory one [17].

To summarize, we propose that the CS-intensity effect allows to generalize a sensory motor association learned at one speed of execution to different onesand that this generalization can help a robot to master the execution of skilled behaviour at high speeds through incremental training. In order to reproduce the CS-intensity effect in the computational model of the cerebellum, we generate the cortical bases mimicking the fast excitation followed by slow inhibition signal transduction of the cerebellar cortex. Finally, we validate this proposal with a series of robotic experiments where a robot navigates a track at gradually increasing velocities.

II. METHODS

A. Computational architecture

The computational architecture implemented two layers of control. The reactive controller maps the proximity signals at both sides of the robot (US_l and US_r) into reflex-like avoidance turns (UR_{tl} and UR_{tr}) and braking actions (UR_{bl} and UR_{br}). In addition, an adaptive controller, implemented as a cerebellar analysis-synthesis adaptive filter controller ([14]), will eventually acquire an anticipatory turn after the CS and ahead of the UR. In these experiments, since the tracks included just a single right turn, we only implemented a controller to anticipate the turn to the right. However, we mounted reactive controllers in both sides, to ensure that the robot stayed on the track even when the anticipatory turn was overshooting.

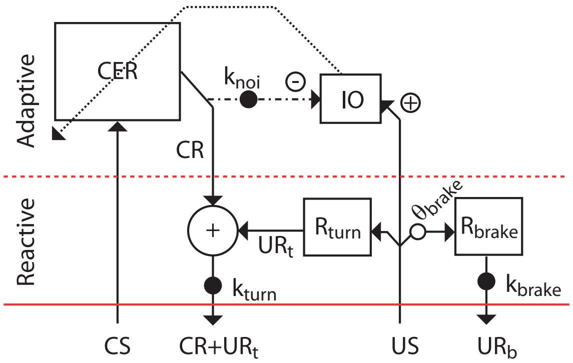

Fig. 1. Computational architecture with the reactive and adaptive controllers. CS, conditioning stimulus (visual input); US, unconditioned stimulus (proximity signal); CR, conditioned response (anticipatory turn); $\mathrm{UR_t}$, unconditioned response (reactive turn); $\mathrm{UR_b}$, unconditioned response (reactive brake); CER, cerebellar controller; $\mathrm{R_{turn}}$, reactive turn controller; $\mathrm{R_{brake}}$, reactive brake controller; θ_{brake}, threshold for braking; $\mathrm{k_{noi}}$, gain of the Nucleo-olivary inhibition; $\mathrm{k_{turn}}$, gain of the turning command; $\mathrm{k_{brake}}$, gain of the braking command.

B. Cerebellar model

The neurobiological assumptions underpinning this model have been decribed elsewhere [4]. We just highlight that this model implements the Nucleo-Olivary Inhibition (NOI), through which the cerebellum can compare its output signal with the sensory signal carrying the US information, and adjust its mapping until the mismatch is minimized. Additionally, here we add a slow inhibitory components to the computation of each basis computation, that allows for more precise responses.

To generate the signal of the cortical bases we produce two components for each basis: a fast excitatory and a slow inhibitory one. Each component consists of a double exponential convolution. The time constants of the convolutions for the excitatory and inhibitory component are randomly drawn from two flat probability distributions ranging from 0.05 to 0.1 seconds, and from 0.2 to 5.5 seconds, respectively. These values are significantly over the physiological time constants of the synaptic currents found in the cerebellar cortex, but they are appropriated for the current experimental setup. The value obtained after the two convolutions is then thresholded and scaled for each basis. The whole computation is formalized as follows:

$$\begin{aligned} e_j^r(t) &= \gamma_j^r e_j^r(t-1) + CS(t) \\ e_j^d(t) &= \gamma_j^d e_j^d(t-1) + e_j^r(t) \\ e_j(t) &= \sigma_j [e_j^d(t) - \theta_j]^+ \end{aligned}$$

where j indexes a particular basis. e_j^r and e_j^d compute a convolution and, informally, each one governs the rise and decay of the e_j basis, respectively. They are controlled by the persistence factors γ_j^r and γ_j^d, which generate the appropriated exponential decay. The third equation adds a nonlinearity (a threshold θ_j) that in the current implementation is critical for obtaining the response latency modulation by the CS intensity (note that $[x]^+ = \max(x, 0)$). I.e., without such a threshold the computational model would act as a linear filter. For each cortical basis an inhibitory component, i_j is generated with the exact same process, only using time constants 10 times larger.

The final value of the basis, p_j, is computed as follows:

$$p_j(t) = [e_j(t) - i_j(t)]^+$$

The output of the cerebellar controller is given by:

$$CR(t) = [\mathbf{p}(t)^T \mathbf{w}(t)]^+$$

where $\mathbf{w}(t)$ is the vector of weights and $\mathbf{p}(t)$, the vector of bases.

The weights are updated using the de-correlation learning rule:

$$\Delta w_j(t) = \beta \; E(t) \; p_j(t - \delta)$$

where β is the learning rate and $E(t)$ is the error signal, computed by the inferior olive output (see below). δ provides the latency of the NOI. The value of δ determines how much the adaptive action anticipates the reactive one and has to exceed the feedack delay [18]. In our experiments we used a β of 0.01 and a δ of 1.0 s. We estimated that the feedback delay in our system was in the order of 0.2 seconds.

Finally, the error signal for the cerebellar system is computed as the difference between the scaled cerebellar output and the US signal as follows:

$$E(t) = US(t) - k_{noi} CR(t - \delta) \tag{1}$$

The factor scaling the CR is the gain of the NOI, k_{noi}. Note that through this computation, the error signal for the cerebellum is suppressed before the sensory US is completely prevented, for which a fraction of the initial reactive response, UR, still prevails after training. The amplitude of this residual UR depends on the k_{noi} value and has functional implications (see [4]). In short, with a k_{noi} of 1 the final amplitude of both actions would be similar whereas with a k_{noi} equal to 0 the adaptive response would completely replace the reactive one. In this setup we use a value of 0.4, obtaining a CR that is bigger than the UR at the end of training.

C. Experimental setup

The setup consists of an epuck robot [19] and a Mixed-Reality Robot Arena (MRRA) [20]. The robot navigates a track back-projected onto the table displaying the CS signal as a series of green stripes. A tracking system captures the position and direction of the robot in order to compute the sensory data, proximity and visual signals. These virtual sensory signals are provided to the controller system, that then issues the appropriated motor commands to the epuck robot via a bluetooth connection. In summary this setup mixes the constraints of controlling a real robot with the versatility of generating a virtual scenario.

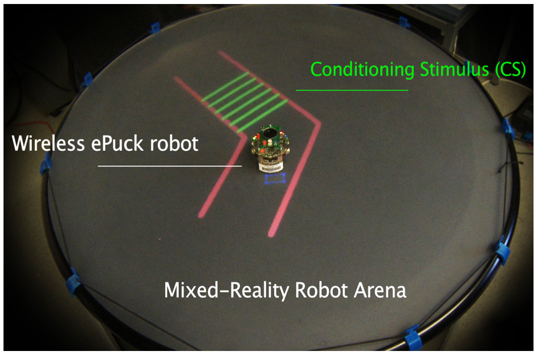

Fig. 2. Experimental setup. Mixed reality environment with the back-projected arena and the physical robot (e-puck).

Stimuli: The robot is equipped with *virtual* sensors. Namely, a camera and proximity sensors. The camera allows to detect the green stripes displayed on the ground of the track. From this input, we compute the proportion of the visual field occupied by CS stimulus (the green stripes). The CS is then computed as the differential of the previous signal. With this procedure, the intensity of the CS signal is linked to the speed of the robot, since the instantenous variation in the visual field is proportional to the robot velocity. The proximity sensors are mounted at each side, frontally and 15 degrees away from the forward direction. They have a range of 6 cm, and are normalized such that at maximum proximity their value is 1.

Motor commands: The motor of each wheel is controlled independently, with a signal that blends the output of all controllers as follows:

$$\begin{aligned} M_l = & \; M_{init} + k_{turn}(CR + UR_{tl} - UR_{tr}) \\ & -k_{brake}(UR_{bl} + UR_{br}) \\ M_r = & \; M_{init} + k_{turn}(CR - UR_{tl} + UR_{tr}) \\ & -k_{brake}(UR_{bl} + UR_{br}) \end{aligned}$$

M_l and M_r denote the left and right commands, respectively. M_{init} sets the initial velocity, that is maintained constant for each trial. We use values ranging from 8 cm/s (corresponding to a M_{init} equal to 20 units) to 20 cm/s (corresponding to a M_{init} equal to 50 units). The reactive turns (UR_{tl} and UR_{tr}), that convey the same value as their contra-lateral US signals, are added to the anticipatory turn (CR) and multiplied by the motor gain for the turn k_{turn}. Each brake command is computed as follows:

$$UR_{bl} = (1 - \theta_{brake})[(US_r - \theta_{brake})]^+$$

In short, each UR_b is computed as the proximity signal exceeding the θ_{brake} threshold, normalized. Both braking actions are then added and multiplied by the corresponding motor gain (k_{brake}). In our experiment we set k_{turn} and k_{brake} to values of 8 and 20, respectively, and set the braking threshold to 0.5, what corresponds approximately to a 3 cm distance from the wall.

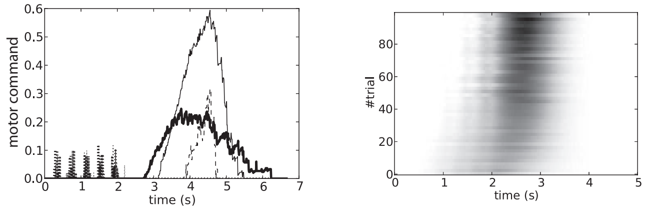

Fig. 3. Evolution of the responses. (*Left*). Reactive commands at the first trial (turning [thin line] and braking [thin dashed line]) and at last trial (reactive turning [thick solid line], there is no braking at the last trial). The cue signals are deisplayed near the onset of the trial [grey dotted lines]. (*Right*). Evolution of the adaptive command. Darker color indicates higher amplitude of the response.

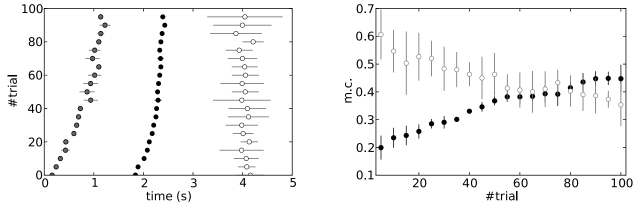

Fig. 5. Quantification of the responses during training. (*Left*). Timings relative to the CS onset. Onset (gray markers) and peak (black) of the adaptive response and peak of the reactive one (empty marker). Mean and standard deviations computed every five trials. (*Right*). Maximum amplitudes of both responses.

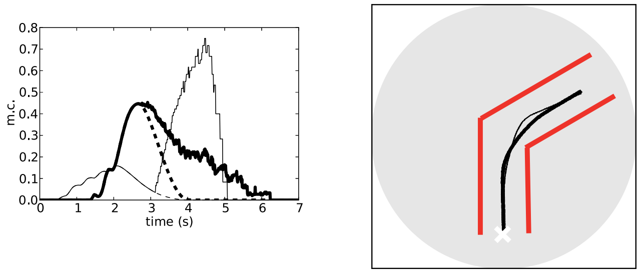

Fig. 4. Integration of the reactive and adaptive responses. (*Left*). Total command in an early trial (thin lines) and at the end of learning (thick lines). The dashed lines separate the adaptive (below) and reactive components of the response (above). (*Right*). Trajectory of the robot for the same two trials

III. RESULTS

A. *Experiment 1*

The goal of this experiment is to train the robot to acquire a predictive turn allowing it to traverse the track without decreasing the initial velocity. The training session lasts 100 trials. We set up an initial velocity of 8 cm/s. This speed is sufficiently slow to prevent the robot from hitting the wall under sole reactive control. However, even at this slow velocity the proximity signal reaches the braking level (Fig. 3 *left*). As training progresses we observe the acquisition of a predictive turn that slowly grows in amplitude and becomes more accurately timed (Fig. 3 *right*). Note that at the end of the training the reactive response is not completely erased: the proximity signal still grows over the threshold causing a reactive turn, but stays below the braking threshold (Fig. 3 *left*). Thus, once successfully trained, the robot balances the predictive and the reactive actions such that it can traverse the track as fast as possible preserving the initial speed.

Concerning the relative timing of both actions, δ_{noi} determines how much the adaptive response will anticipate the reactive one (Fig. 5). We set up this parameter to 1 s, obtaining an optimal merge of both actions (Fig. 4). Namely, the adaptive response peaks at the onset of the reactive one, which is the textbook definition of adaptive timing in the classical eye-blink conditioning paradigm. This merge results in a final trajectory where the robot displays a single turn different from the trajectory during the early trials, where both turns can be singled out (Fig. 4).

B. *Experiment 2*

With the second experiment we assess how well the response acquired at the initial safe velocity generalizes to faster velocities. For this, we applied the cerebellar controller that we trained in the previous experiment and we froze its memory by setting the learning rate to 0. Afterwards, we increased the initial velocity every five trials by a step of 0.8 cm/s. In this way, at the trained velocity the cerebellar controller outputs the same response acquired during the training, but the response at higher velocities depends on how the increased intensity of the CS is translated into the response.

We first evaluate the experiment in behavioral terms. For this we test which is the maximum speed that the robot can achieve before colliding with the walls. We observe that after having only been trained at a velocity of 8 cm/s, the robot can safely navigate the track at speeds up to 17.6 cm/s. Note that the highest safe speed with sole reactive control was of 14.4 cm/s (see supplementary video). Therefore, even though the robot cannot learn from scratch to traverse the track at a speed of 17.6 cm/s, because it will crash with the wall in the first trial, it can traverse the track at such a velocity if it is initially trained at the *safe* velocity of 8 cm/s. This suggests that, in principle, we can use this controller to train the robot optimally navigate this track at the speed of 17.6 if we first train it at 8 cm/s.

Now, we examine the output of the cerebellar controller to assess whether the navigation of the track at higher velocities is achieved, 1) because the exact response that was learned at the initial training speed is triggered at higher velocities but *still* facilitates the turning, or 2) because the learned response is generalized in congruence with the requirements of the increased velocity. Note that as the velocity increases, the interval between the CS and the collision shortens and that, given the dynamics of the motor plant, the amplitude of the turning command also increases (i.e., reproducing the same curvature at a higher speed requires a higher control signal). Thus, to generalize from the previous learned motor command, the cerebellum has, 1) to anticipate the adaptive response such that the robot turns earlier and 2) increase the amplitude of the motor command. We recall here that the speed of the robot is implicitly coded in the intensity of the CS.

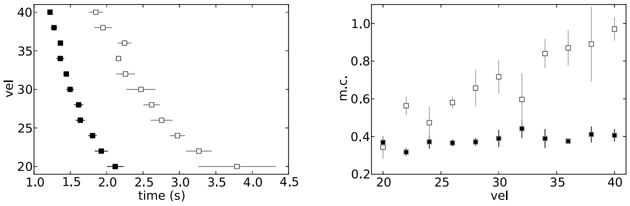

Fig. 6. Quantification of the responses during generalization. (*Left*). Timings relative to the CS onset. Peak of the adaptive response (black) and peak of the reactive one (empty marker). Mean and standard deviations computed every five trials. (*Right*). Maximum amplitudes of both responses.

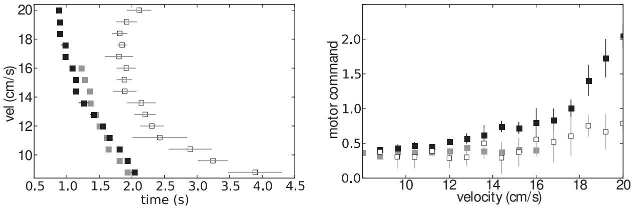

Fig. 7. Quantification of the responses during continuous learning. (*Left*). Timings relative to the CS onset. Peak of the adaptive response (black) and of the reactive one (empty marker). For comparison, the peak of the CRs obtained in the previous experiment are also displayed (gray). Mean and standard deviations computed every five trials. (*Right*). Maximum amplitudes of both responses.

We obtain that as the velocity increases (as the intensity of the CS stimulus increases) the timing of the adaptive response is anticipated. The anticipation is such that at all the velocities tested the adaptive response still peaks ahead of the reactive one (Fig. 6 *left*). The adaptive turn remains anticipatory even when the increase of speed triggers an earlier reactive turn. However, we do not obtain an increase in the amplitude of the response together with the increased velocity (Fig. 6 *right*). This lack of generalization of the amplitude may be the reason why, even though the learned response remains anticipatory, at some point it becomes insufficient to keep the robot on the track.

C. Experiment 3

Finally, we want to find out whether the controller allows the robot to traverse the track at its maximum velocity (20 cm/s). For this we run an experiment in which the robot is incrementally trained at higher velocities. As in the previous experiment we depart from a cerebellar controller already trained during 100 trials at the initial velocity of 8 cm/s. Whenever the robot navigates the track without braking for 5 consecutive trials or after performing 10 trials at the same velocity, we increase the velocity by a step of 0.8 cm/s. Our hypothesis is that, since the controller generalizes the response learned at a slower velocity to a higher one, learning to perform a correct turn at an increased speed would only require a fine-tuning of the initial acquired CR.

We obtain that the robot is able to navigate the track at the maximum velocity, even though it cannot avoid braking for speeds above 17.6 cm/s. To assess how much the generalization from slower to higher velocities facilitates the learning, we compare the results from experiment 2, that were generalized from a single initially trained velocity, with the commands learned in this experiment, where the learning proceeds at each velocity step, assuming that a major similarity implies a better generalization. We observe that having learned to perform the turn at the velocity of 8 cm/s, the controller correctly generalizes the timing at the velocities up to 17.6 cm/s, i.e., there is no systematic difference between the timing of the generalized and the learned commands (Fig. 7 *left*). However, both commands are very different in amplitude (Fig. 7 *right*). As our controller did not reproduce the amplitude component of the generalization, the learned commands have a bigger amplitude than the generalized ones. Finally, besides delaying the collision with the wall, that in the current experiment did not occur even at the maximum velocity of the robot, learning a more precise response allows the robot to navigate the track faster (Fig 8).

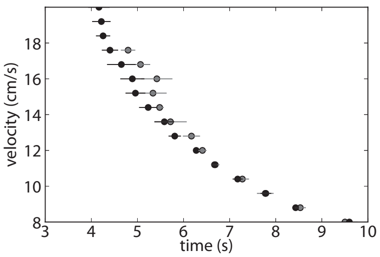

Fig. 8. Time to navigate the track at different velocities with continuous learning (dark) or only training at the slowest speed (gray). Mean and standard deviation of the last five trials in each condition.

In conclusion, we observe that the correct generalization of the timing of the CR to higher velocities allows the robot to learn to navigate the track at its maximum velocity.

IV. CONCLUSIONS

We have presented a control architecture inspired by the cerebellum that allows a robot to navigate a track avoiding collisions. Such controller learns to transform a purely reactive avoidance response into a more complex response that includes both anticipatory and reactive components, i.e., the CR and the UR, respectively. We have also shown that even if the robot is only trained at the slowest velocity, the generalized response is still adaptive for higher velocities, with the CR correctly anticipating the US onset. In addition, we show that this generalization facilitates the process of learning to navigate the track at velocities much higher that the ones that could be safely handled by the reactive control alone.

To achieve this result we have extended previously existing computational models of the cerebellum [4], [14] with a method for generating the cortical bases inspired by the cerebellar physiology. More concretely, here we added two features that mimic the computation of the cerebellar granule cells: the interplay between fast excitation and slow inhibition [17] and the non-linearity of their responses [16]. The first modification allows the system to acquire precise responses and the second, the non-linearity of the bases, namely, the addition of a threshold to the currents, is necessary for achieving the modulation of the CR latency by the intensity of the CS. Since in our set-up the intensity of the CS is linked to the velocity of the robot, this means that as the velocity of the robot increased the latency of the response was correctly advanced.

This addition allowed us to explore the generalization of an avoidance action over different speeds of execution, a feature not studied in other previous studies that used cerebellar controllers for collision avoidance [7], [5].

With this controller the reactive turn, the UR, is not completely suppressed by the CR. The amount of the final response transferred to the adaptive controller is determined by the gain of the NOI (k_{noi}) ([4]). The residual UR is the only means for the controller architecture to know that the CR is necessary. For instance, if we remove the US of the current set-up (removing the turn while keeping the CS) this controller would gradually extinguish the CR. Thus thanks to the comparison between cerebellar output and sensory input performed via the NOI this controller can manage the acquisition and extinction of an anticipatory adaptive reflex in a totally autonomous way.

The proposed controller architecture is not task specific and, in general, it could be applied to scenarios having a feedback signal that has to be kept under a certain safety level. An interesting suggestion is that this type of cotroller could be involved safe limb control in soft robots [21]. In that case, the cerebellar controller would take care of avoiding the limbs to too strongly collide with the robot's own limbs or external objects. The major difference then in that case is that the signal playing the role of the CS in be internally generated, reflecting the robot's intent to move move the limb.

Regarding the delay of the NOI, some questions remain open. First, how can this parameter be learned? And secondly, can it also be modulated? In partucular, the second question has a functional relevance in our scenario, because even though the CS-CR interval is learned by the system, the CR-UR is fixed. In other words, the CR always anticipates de UR by the same fixed time interval, even if it it would be more effective for this time interval to be adjusted according to the velocity.

Thus, to conclude we have presented an controller for the acquisition of anticipatory reflexes that besides being completely autonomous, it includes a built-in sensory motor contingency that modulates the timing of the protective action according to the intensity of the predictive cue. For the first time we have implemented this controller with a real robot. The next step will be to apply this controller with more complex motor plants to validate whether this built-in sensory motor contingency facilitates learning the rapid execution of motor actions when the dynamics are more complex.

V. ACKNOWLEDGMENTS

This work was supported by eSMC FP7-ICT- 270212.

REFERENCES

[1] J. Albus, "A theory of cerebellar function," *Mathematical Biosciences*, vol. 10, no. 1-2, pp. 25–61, 1971.

[2] D. Marr, "A theory of cerebellar cortex," *The journal of physiology*, vol. 202, no. 2, p. 437, 1969.

[3] J. S. Albus *et al.*, "A new approach to manipulator control: The cerebellar model articulation controller (cmac)," *Journal of dynamic systems, measurement and control*, vol. 97, no. 3, pp. 220–227, 1975.

[4] I. Herreros and P. Verschure, "Nucleo-olivary inhibition balances the interaction between the reactive and adaptive layers in motor control," *Neural Networks*, no. 0, 2013.

[5] J. L. McKinstry, G. M. Edelman, and J. L. Krichmar, "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*, vol. 103, no. 9, pp. 3387–3392, 2006.

[6] N. R. Luque, J. A. Garrido, R. R. Carrillo, S. Tolu, and E. Ros, "Adaptive cerebellar spiking model embedded in the control loop: Context switching and robustness against noise," *International Journal of Neural Systems*, vol. 21, no. 05, pp. 385–401, 2011.

[7] C. Hofstoetter, M. Mintz, and P. F. Verschure, "The cerebellum in action: a simulation and robotics study," *European Journal of Neuroscience*, vol. 16, no. 7, pp. 1361–1376, 2002.

[8] C. Sabourin and O. Bruneau, "Robustness of the dynamic walk of a biped robot subjected to disturbing external forces by using cmac neural networks," *Robotics and Autonomous Systems*, vol. 51, no. 2, pp. 81–99, 2005.

[9] A. H. Fag, N. Sitkoff, J. C. Houk, and A. G. Barto, "Cerebellar Learning for Control of a Two-Link Arm in Muscle Space," *Robotics and Automation, IEEE Intemational Conference*, no. April, pp. 2638–2644, 1997.

[10] P. Svensson, D.-A. Jirenhed, F. Bengtsson, and G. Hesslow, "Effect of conditioned stimulus parameters on timing of conditioned purkinje cell responses," *Journal of neurophysiology*, vol. 103, no. 3, pp. 1329–1336, 2010.

[11] P. Svensson, M. Ivarsson, and G. Hesslow, "Effect of varying the intensity and train frequency of forelimb and cerebellar mossy fiber conditioned stimuli on the latency of conditioned eye-blink responses in decerebrate ferrets." *Learning & Memory*, vol. 4, no. 1, pp. 105–115, 1997.

[12] I. Pavlov and G. Anrep, *Conditioned reflexes*. Dover Pubns, 1927.

[13] I. Gormezano, W. Prokasy, and R. Thompson, *Classical conditioning*. Lawrence Erlbaum, 1987.

[14] P. Dean, J. Porrill, C. Ekerot, and H. Jörntell, "The cerebellar microcircuit as an adaptive filter: experimental and computational evidence," *Nature Reviews Neuroscience*, vol. 11, no. 1, pp. 30–43, 2010.

[15] J. Eccles, M. Ito, and J. Szentágothai, *The cerebellum as a neuronal machine*. Springer Berlin, 1967.

[16] A. Spanne and H. Jörntell, "Processing of multi-dimensional sensorimotor information in the spinal and cerebellar neuronal circuitry: A new hypothesis," *PLoS computational biology*, vol. 9, no. 3, p. e1002979, 2013.

[17] J. Crowley, D. Fioravante, and W. Regehr, "Dynamics of fast and slow inhibition from cerebellar Golgi cells allow flexible control of synaptic integration," *Neuron*, vol. 63, no. 6, pp. 843–853, 2009.

[18] R. Miall, D. Weir, D. Wolpert, and J. Stein, "Is the cerebellum a smith predictor?" *Journal of motor behavior*, vol. 25, pp. 203–203, 1993.

[19] F. Mondada, M. Bonani, X. Raemy, J. Pugh, C. Cianci, A. Klaptocz, S. Magnenat, J.-C. Zufferey, D. Floreano, and A. Martinoli, "The e-puck, a robot designed for education in engineering," pp. 59–65, —2009—.

[20] M. S. Fibla, U. Bernardet, and P. F. Verschure, "Allostatic control for robot behaviour regulation: An extension to path planning," in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*. IEEE, 2010, pp. 1935–1942.

[21] P. Dean, S. Anderson, J. Porrill, and H. Jörntell, "An adaptive-filter model of cerebellar zone c3 as a basis for safe limb control?" *The Journal of physiology*, 2013.