Cerebellar Spiking Engine: Towards Objet Model Abstraction in Manipulation

N. R. Luque, J. A. Garrido, R. R. Carrillo, E. Ros

***Abstract*—This paper presents how a plausible cerebellum-like architecture can abstract corrective models in the framework of a robot control task when manipulating objects that significantly affect the dynamics of the system. The presented scheme is adequate to control non-stiff-joint robots with low-power actuators which involve controlling systems with high inertial components. We evaluate the way in which the cerebellum stores a model in the granule layer, how its microstructure can efficiently abstract models and deliver accurate corrective torques for increasing precision during object manipulation. Particularly we study how input sensory-motor representations can enhance model abstraction capabilities during accurate movements, making use of explicit (model-related input labels) and implicit model representations (sensory signals).**

Finally we focus on how our cerebellum model (using a temporal correlation kernel) properly deals with transmission delays in sensory-motor pathways.

***Keywords*—Spiking Neuron, Cerebellum, Adaptive, Simulation, Learning, Robot, Biological Control Systems**

I. Introduction

Controlling fast non-stiff-joint robots accurately with low power actuators is a difficult task which involves high inertia. Biological systems are in fact non-stiff-joint "plants" driven with relatively low-power actuators. These biological systems have developed smart "model abstraction" capabilities through evolution. In this way, the control commands of biological systems are generated taking into account the "plant model" (for instance arm + object). In the framework of accurate control with a large number of degrees of freedom (DOFs), extracting efficiently models from explorative manipulation, storing them without interference with other previously acquired ones and retrieving these models accurately for each case are capabilities in which biological systems are still beyond current cutting edge control technology. With regards to machine learning approaches, specific models are being developed to address these hard tasks [1-4].

If the control scheme is based on accurate kinematic and dynamic models, and the dynamics of the plant changes (manipulating tools and objects will modify the base model), this will lead to significant distortions along the desired trajectory, affecting the final achieved accuracy [5-7]. Therefore, these systems are required to be adaptive to tune the corrective models for specific object or tool manipulation [8].

The cerebellum seems to play a crucial role in this model abstraction task [9]. But how this is supported by actual network topologies, cell models and adaptation properties is an open issue. We have addressed the study of how this model abstraction task can be achieved with a cerebellum-like architecture based on spiking neurons. Lately, last generation simulation tools [10-14] allow the definition and simulation of nervous centers of certain complexity in the framework of biologically-relevant tasks. This allows addressing studies in which function and structure of nervous centers are conjointly evaluated to better understand how the system operation is based on cell and network properties.

In previous works we evaluated a cerebellar model in reaching point task [15] and a simple smooth pursuit task [16]. In this paper we evaluate how an input configuration encoding inherent propioception signals along the trajectory and also inputs from other sensory-systems (such as vision) are conjointly used in an object manipulation task.

We also study how an adaptive cerebellum-like module using a basic temporal-correlation kernel (including long-term depression (LTD) and long-term potentiation (LTP) at parallel fiber-Purkinje cells synapses) can build corrective models to compensate deviations in the robot trajectory when the dynamics of the controlled plant is altered and also can deal properly with transmission delays in sensory-motor pathways.

Manuscript received February 5, 2010. This work has been supported by the EU project SENSOPAC (IST-028056) and the national grant DINAM-VISION (DPI2007-61683).

N. R Luque and J. A. Garrido are PhD students at the University of Granada (Periodista Miguel Saucedo s/n Granada (Spain)) in the Department of Computer Architecture and Technology. (E-mail: nluque@atc.ugr.es and jgarrido@atc.ugr.es).

R. R Carrillo is a PhD researcher of the University of Almeria (Ctra. Sacramento s/n La Cañada de San Urbano Almeria (Spain)) in the Department of Computer Architecture and Electronics. (E-mail: rcarrillo@atc.ugr.es).

E. Ros is professor at the Univerity of Granada (Periodista Miguel Saucedo s/n Granada (Spain)) in the Department of Computer Architecture and Technology. (E-mail: eros@atc.ugr.es).

978-1-4244-8126-2/10/$26.00 ©2010 IEEE

I. Methods

For extensive spiking network simulations we have further developed and used an advanced event-driven simulator based on lookup tables (EDLUT) [10] [15]. For the robot plant simulation we have implemented an interface between the simulator of the LWR (Light-Weight-Robot developed at DLR [17]) and the control loop. In fact, in this application field it is of interest to develop control schemes for relatively fast movements at reasonable accuracy and which adapt when controlling different tools. For the sake of simplicity, in our experiments we use a simulator of this robot in which we have fixed some joints to reduce the number of actual joints to three, limiting the number of degrees of freedom.

A. *Arm simulator*

We have integrated the robot plant simulator of the LWR (Light-Weight-Robot developed at DLR [17]) with the afterwards described control loop. The simulated-robot-plant physical characteristics can be dynamically modified to match different contexts. The LWR robot is a 7-DOF arm composed of revolute joints. In our experiments we used the first (we will refer to it as q_1), second (q_2) and fourth joint (q_3) and keep the others fixed. The task for the experiments with LWR robot is to follow an eight-like trajectory. This is a rather standard benchmarking trajectory. The inverse kinematics is computed to obtain a smooth trajectory in joint angle space. The trajectories of individual joints have enough variation so that a sufficiently rich movement will be executed. This trajectory allows studying the platform dynamics [18] [19].

This robot is specially suited for interactions with humans in unstructured, daily environments. In these scenarios, it is of specific interest the use of robots based on low-power actuators in order to reduce the danger for humans in case of malfunctioning. Furthermore, the accuracy in position and trajectory is not fully exploitable because of the uncertainty in the actual positions of the robot itself and the unstructured scenarios.

B. *Control scheme*

There has been a wide range of cerebellar motor-control-system approaches proposed in the literature.

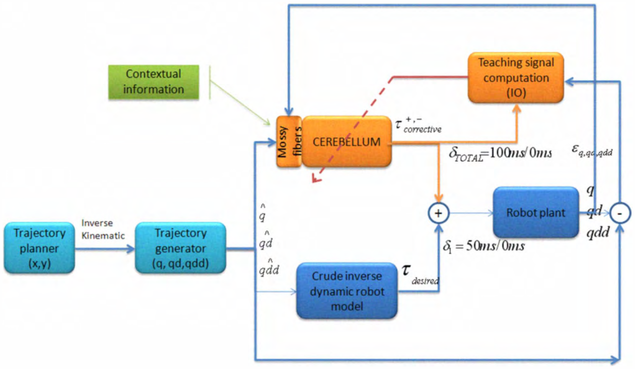

Fig. 1. ***Control loop***. The adaptive cerebellar model delivers add-on corrective actions to compensate for deviations in the base dynamic and kinematic plant model when manipulating objects.

This is a very active research field (for a review the reader is referred to [20]).

Lee Miller [21] proposes a cerebellar control system based on a predictive signal (supplied by the cerebellum) with the aim of giving a progressive and proper motor control commands. According to this theory we developed our cerebellar control-loop model, a feedback cerebellar control.

In our control loop (see Fig 1) the desired arm states (robot end-effector position at each time) are generated by the *trajectory planner* to follow the desired trajectory (smooth pursuit). This trajectory in Cartesian coordinates is translated into joint coordinates (positions (q), velocities (qd) and accelerations (qdd)) by the *trajectory generator* (a crude inverse kinematic model representing the output of motor cortex and other motor areas) in our experiment the robot follows the trajectory described in expression (1).

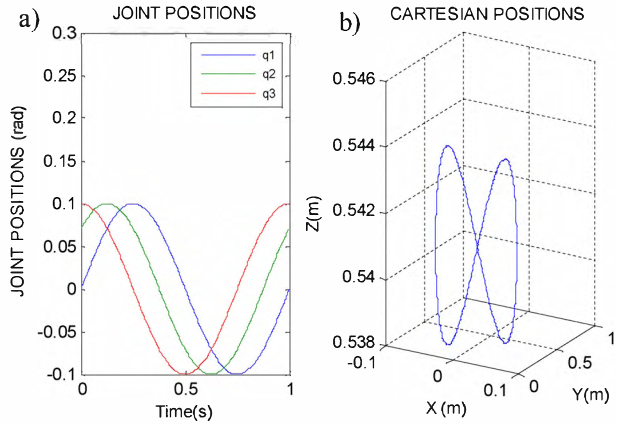

Fig. 2. ***Three-joint periodic trajectories describing 8-shape movements*** a) Angular coordinates of each joint of the LWR robot b) 3D view of the robot end-effector trajectory in Cartesian coordinates.

$$q_1 = A \sin(\pi t) + C_1 \qquad (1.A)$$
$$q_2 = A \sin(\pi t + \theta) + C_2 \qquad (1.B)$$
$$q_3 = A \sin(\pi t + 2\theta) + C_3 \qquad (1.C)$$

These desired arm states in joint coordinates are used at each time step to compute a crude torque commands (*crude inverse dynamic robot model*). They are also used together with the *contextual information* (related to the manipulated object) as input to cerebellum which produces the predictive corrective commands ($\tau_{corrective}$) which are added to these crude torque commands ($\tau_{desired}$).

Total torque is delayed (on account of the biological motor pathways) and supplied to the *robot plant*. The difference between the actual robot trajectory and the desired one is also delayed and used by the *teaching signal computation* module to calculate the inferior olive (IO) cerebellum input signal. This signal will be used by the cerebellum to adapt its output as described in *Learning Process* section.

Industrial applications with high latency sensors require the use of standard techniques to explicitly compensate these delays. Nevertheless these techniques seem to have a poor performance in this kind of problems [22]. In biological systems it is thought that the cerebellum solves this problem. The cerebellum model described here compensates the sensory-motor delay by means of the temporal-correlation kernel (see *Results* and *Learning Process* section).

C. Cerebellum-arm Interface

The analog signals representing the cerebellum input (desired arm position, velocity and contextual information. See *Cerebellum Model* section) are continuously translated into mossy fiber (MF) activity. They are divided into groups and each group encodes an input variable. Each MF of a group is associated to a specific value range of an input variable (receptive field). A MF is modeled as a leaky integrate-and-fire neuron whose input current is calculated using overlapping radial basis functions (RBF) which account for the receptive fields. Therefore, a MF fires when the input variable takes a value covered by its corresponding RBF. Thus, each group of MF covers the total working range of state variables (spatio-temporal population coding).

D. Spiking neuron simulation

The cerebellum module consists of a network which contains a considerable amount of spiking neurons (See *Cerebellum Model* section). To simulate this network efficiently we have opted to use EDLUT simulator. This application software is an event-driven simulator which allows the fast network simulation of relatively-complex neural models through an innovative method: Each neural model included in the network (usually defined by a set of differential equations which govern the neural state) is simulated for every possible neural state and the consequent evolution of each neural state variable is stored in lookup tables. Then, when a simulation of a network containing these models is done, it can be performed without requiring a computational-costly numerical procedure for solving the differential equations defining the neural model.

This computational efficiency permits this powerful simulation tool to be applied in complex tasks involving robot-control where massive spiking cerebellar architectures are evaluated. In comparison with others simulation tools, EDLUT[10-11] is intermediate between the very detailed simulators, such as NEURON[12] or GENESIS[13] and the event-driven simulation schemes based on simple analytically described cell dynamics such as SpikeNET[14]. Different neuron models can be used (for a comparative study see [23]. We have chosen the neuron model described in the next sections because it captures certain biologically plausible features [Table1]

E. Neural Models

The simulated spiking network is composed by two different cell types.

The used cell models are a modified version of the spike-response model (SRM) [24-25] with decaying synaptic conductances. Thus, the neuron models account for dynamic synaptic conductance rather than simply for constant current flows, providing an improved description over simpler I&F models.

The synaptic conductance has been modelled as decaying exponential functions triggered by input spikes as stated by the expressions (2):

$$g_{exc}(t)=\begin{cases} 0 & , \quad t<t_0 \\ g_{exc}(t_0)\cdot e^{-(t-t_0)/\tau_{exc}} & , \quad t\ge t_0 \end{cases} \qquad (2A)$$

$$g_{inh}(t)=\begin{cases} 0 & , \quad t<t_0 \\ g_{inh}(t_0)\cdot e^{-(t-t_0)/\tau_{inh}} & , \quad t\ge t_0 \end{cases} \qquad (2B)$$

Where g_{exc} and g_{inh} represent the excitatory and inhibitory synaptic conductance. τ_{exc} and τ_{inh} represent the correponding synaptic time constants. This exponential representation has several advantages. First, it is an effective representation of realistic synaptic conductances (the improvement in accuracy from the next most complex representation, a double-exponential function, is hardly worthwhile [10]). Secondly, each synaptic conductance type requires only a single state variable, because synaptic inputs through several synapses of the same type can simply be summed recursively when updating the total conductance if they have the same time constants. Therefore, when an input spike is received at time t, for example, through an excitatory synapse, its corresponding conductance is updated as described in expression (3):

$$g_{exc(post-spike)}(t)=G_{exc,j}+g_{exc(pre-spike)}(t) \qquad (3)$$

($G_{exc,j}$ is the weight of synapse *j*; a similar relation holds for inhibitory synapses).

In our simulations, the synaptic parameters have been chosen to represent excitatory AMPA-receptor-mediated synapse time constants and inhibitory GABAergic synapse time constants of cerebellar granule cells [26-29]. Note that different synaptic connections in different cells might have different parameters.

The membrane potential (V_m) at time *t*, is defined by the differential equation (4):

$$C_m\frac{dV_m}{dt}=g_{exc}(t)(E_{exc}-V_m)+g_{inh}(t)(E_{inh}-V_m)+G_{res}(E_{rest}-V_m) \quad (4)$$

Where the conductances $g_{exc}(t)$ and $g_{inh}(t)$ integrate all the contributions received through individual synapses, G_{rest} represents the resting conductance and E_{exc}, E_{inh} and E_{rest} represent the corresponding reversal potentials.

Equation (4) is amenable to numerical analysis. In this way, we can calculate V_m, g_{exc} and g_{inh} for a given time after a previous neural state or input spike. The firing time (t_f) is the time when the membrane potential (V_m) reaches

TABLE I
PARAMETERS OF THE DIFFERENT CELL TYPES
USED IN SIMULATED NETWORK.

Parameter	Granule Cell	Purkinje Cell
Refractory period	1ms	2ms
Membrane capacitance	2pF	400pF
Total excitatory peak conductance	1nS*100	1.3nS*175000*10%
Total inhibitory peak conductance	1nS*200	3nS*150
Threshold	-40mV	-52mV
Resting potential	-70mV	-70mV
Resting conductance	0.2nS	16nS
Resting time constant (τ_m)	10ms	25ms
Excitatory-synapse time constant (τ_{exc})	0.5ms	0.5ms
Inhibitory-synapse time constant (τ_{inh})	10ms	1.6ms

Parameters obtained from the following papers: GrC [30-34] and PC [35-38].

the firing threshold (V_{th}) ; an output spike is emitted. It can be calculated from the membrane potential evolution.

Table 1 shows the equation parameters corresponding to the two neural models used in the simulated cerebellum.

F. Cerebellum model

The proposed cerebellar generic architecture is shown in Fig. 3. This cerebellum model is composed by the following layers:

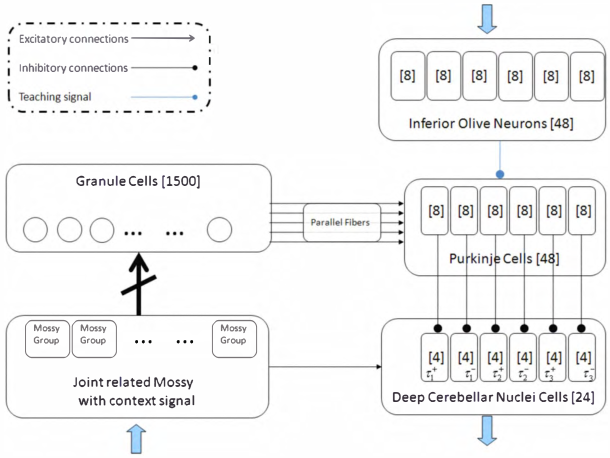

Fig. 3. ***Cerebellum model diagram.*** Inputs encoding the movement (desired and actual arm states) are sent (upward arrow) through the mossy fibers to the granular layer. These inputs encode the desired and actual position and velocity of each joint along the trajectory and also contex-related information. Inputs encoding the error are sent (upper downward arrow) through the inferior olive (IO). Cerebellar outputs are provided by the deep-cerebellar-nuclei cells (DCN) (lower downward arrow). The DCN collects activity from the mossy fibers (excitatory inputs) and the Purkinje cells (inhibitory inputs). The outputs of the DCN are added as corrective torque in the control loop of Fig.1

Mossy fibers (MF): This layer carries explicit contextual information and the desired and actual robot joint position and velocity. (For an explanation of the translation from this information into cell activity, see *Cerebellum-arm Interface* section).

This input layer has been modelled according to an *Explicit and Implicit Context Encoding Approach* (16 fibers encode the context information and 240 fibers encode the joint-related information). The fibers in charge of the joint-related information encoding have been divided into 12 groups of 20 fibers: 3 groups encode desired joint positions (1 group per joint), other 3 groups encode desired joint velocities (1 group per joint), other 3 groups encode actual joint positions and the remaining 3 groups encode actual joint velocities. The explicit contextual information is encoded by 2 groups of 8 fibers (see Fig. 4).

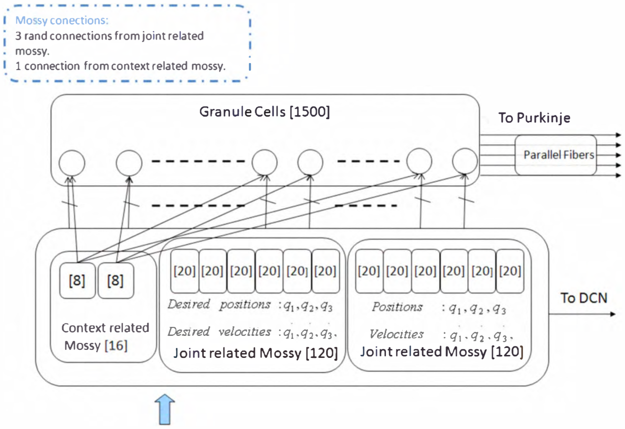
Mossy conections:
3 rand connections from joint related mossy.
1 connection from context related mossy.
Granule Cells [1500]
To Purkinje
Parallel Fibers
[8] [8]
Context related Mossy [16]
[20] [20] [20] [20] [20] [20]
Desired positions : q_1, q_2, q_3
Desired velocities : $\dot{q}_1, \dot{q}_2, \dot{q}_3$
Joint related Mossy [120]
[20] [20] [20] [20] [20] [20]
Positions : q_1, q_2, q_3
Velocities : $\dot{q}_1, \dot{q}_2, \dot{q}_3$
Joint related Mossy [120]
To DCN

Fig. 4. ***Granular layer model.*** Explicit and implicit context encoding approach. Each granule cell receives an explicit context signal and three randomly-chosen mossy fibers from current and desired positions and velocities

Granular layer (GR) (1500 cells): A simplified (i.e. notice the lack of Golgi cells and interneurons) granular layer of the cerebellum has been designed with the purpose of preprocessing the input of the Purkinje cells (PC). The information supplied by MF is transformed into a sparse representation that facilitates discrimination of very similar inputs [39]. In this large GR layer, each cell receives four excitatory connections. Three connections from randomly-chosen joint-related MF groups and the other one from a context-related MF. Parallel fibers (PF) are the output of this layer. The abstracted corrective models are learned and stored in the weights of the PF-PC connections.

Climbing fibers (CF) (48 fibers): This layer is composed of 6 groups of 8 climbing fibers. It carries the error signals coming from IO PCs. This IO output encodes through spike trains the teaching signal used for supervised learning in the PF-PC connections.

Purkinje cells (PC) (48 cells): They are divided into 6 groups of 8 cells. Each GR cell is connected to 80 per cent of the PCs. They also receive the teaching signals from the CF.

Deep–cerebellar-nuclei cells (DCN) (24 cells): The cerebellum model output is generated by 6 groups of these cells whose activity provides corrective torques for the specified arm state. The corrective torque of each joint is encoded by a couple of these groups, one group is dedicated to generate positive torques (agonist) and the other one is dedicated to generate negative ones (antagonist). Each neuron group in the DCN receives excitation from every

MF and inhibition from the two corresponding PCs. In this way, the sub circuit PC-DCN-IO is organized in six microzones; three of them for positive joint torques (one per joint) and three for negative joint torques (one per joint).

1) *Learning Process*

The used spiking cerebellar model has been outfitted with STDP (spike-timing-dependent plasticity). Although it is thought that there exists synaptic plasticity at several sites within the cerebellar cortex [40-41], in our model we concentrate the plasticity at the PF-PC synapses, since this seems to be the main synaptic plasticity site driven by teaching or temporal signals (coming from the IO). Therefore, in our model the abstracted corrective models are stored at the PF-PC synapses. The conductance of these synapses is set to an initial medium value (15nS) at the beginning of the simulation, and is modified through the plasticity along the training process each time [42]. This synaptic plasticity is composed of two mechanisms: Long-term potentiation (LTP) and long-term depression (LTD). For the LPD mechanism, the received IO output activity is interpreted as an error signal [16] [43-45] [20]. Each IO-cell spike triggers a weight depression at the PF-PC synapse which received activity from the corresponding PF. The amount of synaptic conductance decreased depends on when the PF spikes arrived at the PC. That is, when an IO spike is received, the past PF activity is convolved with an integral kernel (defined by expression 4) and the corresponding conductance is modified as stated by expression 5A. Different integral kernels can be used to account for the past activity [25]. We have opted for this kernel expression since it can be efficiently implemented in an event-driven simulation [10] and because it considers only the past activity at a certain time interval. Since it relates this past input PF activity with present IO error activity, the delay introduced by transmission pathways of the cerebellum MF input can be overcome by this introduced signal timing difference. After this mechanism is repetitively activated, some PC output is reduced for the learned input patterns, in such way; this PC will not inhibit its corresponding DCN cells, producing cerebellar output [46-47]. To compensate the synaptic conductance depression, LTD is accompanied by the opposite process at the same synapses: LTP. [15][40-42]. When an input spike arrives at the PC through a PF, the corresponding synapse conductance is increased by a fixed amount, as stated by expression 5B.

$$k(t) = e^{-(t - t_{post\ synaptic\ spike})} \sin(t - t_{post\ synaptic\ spike})^{20} \qquad (4)$$

$$LTD: \forall i, \Delta w_i = - \int_{-\infty}^{IOspiketime} k(t - t_{IOspike}) \delta_{GRspike-i}(t)\, dt \qquad (5A)$$

$$LTP: \Delta w_i = \alpha \qquad (5B)$$

G. *Measurements*

The learning performance is characterized using three estimates which are calculated from the evolution of the mean absolute error (MAE) of the three robot joint coordinates along the executed trajectory (trial) during the learning process:

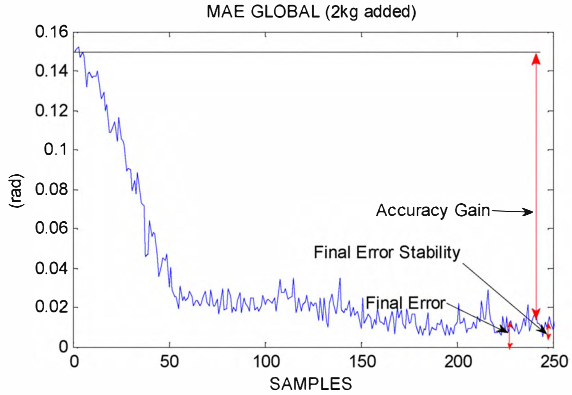

Fig. 5. ***MAE-based estimators.*** Evolution of the mean absolute error (MAE) of the robot joint coordinates in radians. During the learning process of an 8-like trajectory the MAE decreases until a final value range. The learning performance is evaluated using three estimates extracted from the MAE evolution: 1) Accuracy gain, 2) Final error and 3) Final error stability

1) Accuracy gain (estimates the error reduction rate comparing the error at the beginning of the learning process and at the end). This estimate is important when manipulating different objects since the initial error for each one may be different.

2) Final error (average error of the last 30 movement trials).

3) Final error stability (standard deviation of the last 30 movement trials).

We have carried out 250 trials. During this process, the obtained error in each trajectory execution decreases until it reaches a final stable value. The obtained mean absolute error (MAE) of a single complete training process is shown in Fig. 5. We obtain the performance estimates defined before (accuracy gain, final error and final error stability). We have used these performance estimates to characterize the adaptation mechanism capability.

II. Results

A. Learning dynamic models

We have carried out a set of experiments to study the capability of the cerebellar architecture to abstract different corrective models when the dynamics of the plant is modified by manipulating different objects.

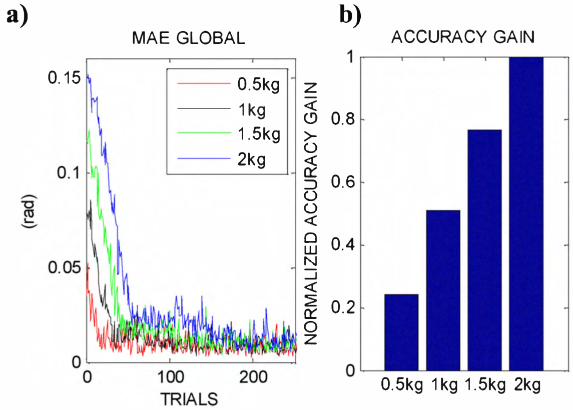

Fig. 6. ***Learning Performance when manipulating different objects (0.5kg, 1kg, 1.5kg and 2kg) during a 250-trial leaning processes***. a) MAE evolution during the whole learning process. b) Accuracy gain estimate achieved for each manipulated object. The different initial error for each manipulated object is revealed by this estimate.

The performance results of the followed trajectory have been evaluated during 250 trajectory executions (trials). Fig.6a shows the MAE evolution (each point represents the error along the 8-like trajectory) for each training process manipulating four objects attached at the end of the last segment of the robot arm (effector): 0.5kg, 1kg, 1.5kg and 2kg object. Fig. 6b shows the performance in terms of accuracy gain for the same experiment. Under normal conditions, without extra mass added to the end of the effector, the "*crude inverse dynamic robot model*" module calculates rough motor commands to control the robot plant. Under altered dynamics conditions, in contrast, the motor commands are very inaccurate to compensate for the new undergone forces (inertia, etc.), and this leads to distortions in the performed trajectories. During repeated trials, cerebellar model is able to learn/abstract a corrective dynamic model for each manipulated object and supplies the corrective motor torques. Therefore, the error is more significantly reduced when the model of the "*crude inverse dynamic robot model*" module differs markedly from the dynamic model of the plant (robot + manipulated object) since the cerebellum produces higher corrective forces.

B. Context switching between two dynamic models. Non-destructive learning.

To assess the capability of the cerebellar architecture to abstract and store different corrective models simultaneously, we have conducted an experiment in which the dynamics of the plant is changed during the learning process each 15 trials. We alternate between two different contexts: manipulating a 1kg object and a 2kg object. The context-related cerebellar input is supplied with different signals in each context to enable the cerebellum to differentiate both contexts allowing different models (contexts) to be abstracted and retrieved efficiently in a non-destructive manner delivering corrective actions in the framework of a control task. In Fig 7. It is shown that the learning is done in a non-destructive manner since once the final error for each context is reached, this error value is maintained stable when the context changes (therefore not destroying the previous-context model). We have obtained a trade-off between the accuracy and the capability to abstract different models with low interference. The accuracy of the abstracted models is increased if the cerebellar sensory input signals are prioritised dedicating more encoding and computing resources along the processing pathway (MF-GR-PC-DCN cells). This characteristic relies on the separation capability of the granular layer for sensory signals related to different contexts. Alternatively to a markedly-separated representation of the sensory signals, we have explored how specific context-related cerebellar input signals can be prioritized requiring lower amounts of dedicated resources (input fibers and connections) to each abstracted model.

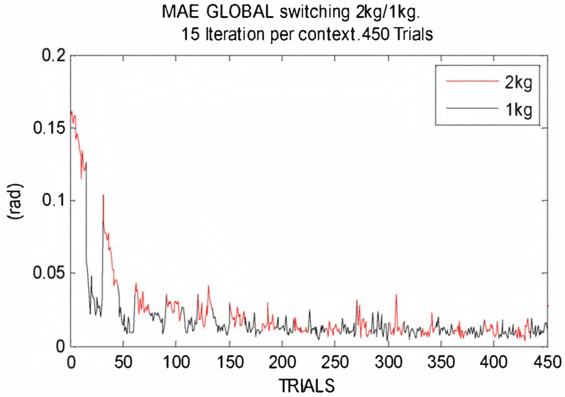

Fig. 7. ***Non-destructive learning in a context switching scenario.*** The dynamics of the plant is alternately changed between two contexts. In the first context, the end-segment of the robot arm is loaded with a 2kg object and in the second one with a 1kg object.

C. Dealing with sensory-motor signal delays

Since biological systems must cope with significant sensory-motor delays (around 100 ms) [48] and it is thought that the cerebellum solves this problem [22], the cerebellar model described here is conceived to deal with these sensory-motor delays. To evaluate the capability of the cerebellar architecture to overcome these delays, we have run five 250-trial learning processes using different sensory-motor delays and manipulating a 2kg object. In each simulation, the integral-kernel peak (see expression 4) is adjusted to match the corresponding sensory-motor delay.

Therefore even if the sensory-related signals (errors) arrive delayed at the cerebellum, the learning mechanism considers the corresponding past input. Fig 8a shows the MAE evolution during the five simulations and Fig 8b shows the achieved final error and final error stability for the same simulations. As it is shown, the delay value does not affect to a large extent the obtained performance in long term. The final average error is nearly constant in these experiments.

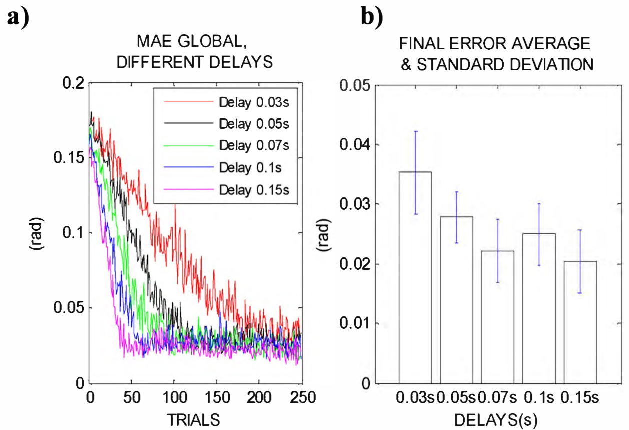

Fig. 8. ***Learning with different sensory-motor delays (from 30ms to 150ms). The robot is manipulating a 2kg object.*** a) MAE evolution b) Final error and final error stability for each delay.

However, when using small delays, the error-reduction speed is slower than when using long delays. For small delays the integral kernel has a sharper shape. The sharper the integral kernel shape is, the more precise the learning becomes. On the other hand, this drives us to a slower synaptic weights adaptation. For long delays the incoming signals are distinguished less precisely and more connection weights are affected by the same input signal, therefore, a smoother and wider kernel shape allows the learning to achieve good results more quickly.

III. Conclusion

This work focuses on studying how a cerebellar adaptive module operating together with a crude inverse dynamics model can effectively provide corrective torques to compensate deviations in the dynamics of a base plant model (due to object manipulation). We have evaluated how a new temporal-correlation kernel driving an error-related LTD and a compensatory LTP component (complementing each other) can achieve effective adaptation of the corrective cerebellar output. We have shown how the temporal-correlation kernel can be adjusted to overcome the sensory-motor delays.

We have also evaluated how this cerebellar module can abstract models corresponding to manipulated objects that significantly affect the dynamics of the plant (arm+object), providing corrective torques for more accurate movements. The cerebellar model includes two new proposed state input representations encoding context-specific inputs (ECEA) and current sensory signal encoding the immediate state during the experiment (ICEA). These context-encoding strategies complement each other. Furthermore, the results obtained with this kind of cerebellar structure are coherent with the experiments done with human experimentation [49-51].

As future work, we will study the scalability in joints of this cerebellar configuration (using multiple “microcomplexes”), its robustness against noise in mossy fibers entries and inferior olive, its behavior in the framework of different control loops and kinematics changes due to object manipulation. The results of this paper aim to advance in building the bridge between neurophysiologists and I.T.

Acknowledgment

This work has been supported by the EU grant SENSOPAC (IST 028056) and the national projects DINAM-VISION (DPI2007-61683) and MULTIVISION (TIC-3873). We would like to acknowledge the support and constructive discussions of Patrick van der Smagt, Holger Urbanek and Olivier J.-M. D. Coenen.

References

[1] S. Vijayakumar, A. D’Souza, S. Schaal, “Incremental Online Learning in High Dimensions”. *Neural Computation*, vol. 17, no. 12, pp. 2602-2634, 2005.

[2] C. K. Williams, and C. Rasmussen,” Gaussian processes for regression” *in M. Touretzky&Hasselmo (Eds), Advances in neural information processing systems*, ch. 8. Cambridge, MA: MIT Press 1996.

[3] A. J. Smola, and B. Schölkopf,”A tutorial on support vector regression” *in NEUROCOLT Tech. Rep NC-TR-98-030*. London: Royal Holloway College 1998.

[4] Z. Ghahramani, and M. Beal, “Variational inference for Bayesian mixtures of factor analysers”, *in S. A. Solla, T, K. Leen, K.-R. Müller (Eds), Advances in neural information processing systems* vol. 12 pp. 449-455, Cambridge, MA: MIT Press 2000.

[5] M. Kawato, and D. M. Wolpert.” Internal models for motor control” *in Novartis Foundation Symposium,* vol: 218, pp. 291-307.1998.

[6] A. Haith, S. Vijayakumar**,”**Implications of different classes of sensorimotor disturbance for cerebellar-based motor learning models***”.in Biological Cybernetics archive, Biological Cybernetics***, vol. 100, no.1, pp: 81-95. 2009.

[7] G. Petkos, and S. Vijayakumar, “Load estimation and control using learned dynamics models”. *IEEE InternationalConference on Intelligent Robots and Systems.*2007**.**

[8] D. M. Wolpert, R.C. Miall, and M. Kawato,”Internal models in the cerebellum”. *Trends in Cognitive Sciences,* vol. 2, no. 9, pp. 338-347.1998.

[9] R. C. Miall,”TheCerebellum, Predictive Control and Motor Coordination” in *Novartis Foundation Symposium.* Sensory Guidance of Movement, vol. 218, pp. 272-290.2007.

[10] E. Ros ,R. R. Carrillo, E. M. Ortigosa., B. Barbour, and R. Agís, “Event-Driven Simulation Scheme for Spiking Neural Networks Using Lookup Tables to Characterize Neuronal Dynamics”. *Neural Computation,* vol. 18, pp. 2959-2993.2006.

[11] EDLUT 2010. [Online]. Available: http://edlut.googlecode.com

[12] NEURON 2010.[Online].Available: http://neuron.duke.edu

[13] GENESIS 2010.[Online].Availablehttp://www.genesis-sim.org/BABEL

[14] SpikeNET2010.[Online].Available: http://cerco.ups-tlse.fr/~delorme/spikenet/SpikeNET.html

[15] R. R. Carrillo, E. Ros, C. Boucheny, and O. J.-M. D. Coenen, “A real-time spiking cerebellum model for learning robot control”. *Biosystems*, vol. 94, no. 1-2, pp. 18-27. 2008.

[16] R. R. Carrillo, E. Ros, B. Barbourb, C. Boucheny, and O. J.-M. D. Coenen,”Event-driven simulation of neural population synchronization

facilitated by electrical coupling”. *Biosystems*, vol. 87, no. 2-3, pp: 275-280.2007.

[17] J.Butterfaß, M. Grebenstein, H. Liu, and G. Hirzinger, “DLR-Hand II: next generation of a dextrous robot hand”. *IEEE International Conference on Robotics and Automation,* pp. 109-14. 2001.

[18] G. Petckos, S. Vijayakumar. “Load estimation and control using learned dynamics models” *IEEE International Conference on Intelligent Robots and Systems* (IROS '07).2007.

[19] E. Burdet, Codourey A. “Evaluation of parametric and nonparametric nonlinear adaptive controllers”. Robotica. Cambridge University Press, vol. 16, pp. 59-73, 1998.

[20] M. Ito,”Cerebellar circuitry as a neuronal machine”.*Progress in Neurobiology*, vol. 78, pp. 272-303. 2006.

[21] L. E. Miller, R. N. Holdefer, J.C. Houk, “The role of the cerebellum in modulating voluntary limb movement commands”. *Archives Italiennes de Biologie*, vol. 140, no. 3, pp. 175-83. 2002.

[22] S. Massaquoi and J. J. E.Slotine, “The Intermediate Cerebellum may function as a Wave Variable Processor," *Neuroscience Letters*, vol. 215. 1996.

[23] E. M. Izhikevich, “Simple Model of Spiking Neurons” IEEE Transactions on neural networks, vol. 14, no. 6, Nov. 2003.

[24] J. Feng “ Is the integrate-and-fire model good enough?”. *A review. Neural Networks*, vol.14, pp. 955-975. 2001.

[25] W. Gerstner, and W. Kistler, “Spiking neuron models: Singleneurons, populations, plasticity*”. Cambridge: Cambridge University*. 2002.

[26] Z. Nusser, S. CullCandy, and M. Farrant, “Differences in synaptic GABA(A) receptor number underlie variation in GABA mini amplitude.” *Neuron,* vol. 19, pp. 697-709. 1997.

[27] *D. J. Rossi, and M. Haman*n, “Spillover-mediated transmission at inhibitory synapses promoted by high affinity alpha(6) subunit GABA(A) receptors and glomerular geometry.” *Neuron* vol. 20, pp. 783-795. 1998.

[28] R. A. Silver, D. Colquhoun, S. G. CullCandy, and B. Edmonds, “Deactivation and desensitization of non-NMDA receptors in patches and the time course of EPSCs in rat cerebellar granule cells.” *Journal of Physiology – London,* vol. 493, pp. 167-173. 1996.

[29] S. Tia, J. F. Wang, N. Kotchabhakdi, and S. Vicini, “Developmental changes of inhibitory synaptic currents in cerebellar granule neurons: Role of GABA(A) receptor alpha 6 subunit. “*Journal of Neuroscience* vol. 16, pp. 3630-3640.1996.

[30] E. D’Angelo E, G. Defilippi, P. Rossi, and V. Taglietti,” Synaptic excitation of individual rat cerebellar granule cells in situ: evidence for the role of NMDA receptors.” *Journal of Physiology,* vol . 482, pp. 397-413. 1995.

[31] E. D’Angelo, T. Nieus, A. Maffei, S. Armano, P. Rossi, et al. “ Theta-frequency bursting and resonance in cerebellar granule cells: experimental evidence and modeling of a slow K+-dependent mechanism.” *Journal of Neuroscience,* vol. 21 pp. 759-770. 2001.

[32] T. Nieus,E. Sola, J. Mapelli, E. Saftenku, P. Rossi, et al. ” LTP regulates burst initiation and frequency at mossy fiber-granule cell synapses of rat cerebellum: Experimental observations and theoretical predictions”. *Journal of neurophysiology,* vol. 95, pp. 686-699.2006.

[33] E. D’Angelo, P. Rossi, and V. Taglietti, “Different proportions of N-Methyl-D-Aspartate and Non-N-Methyl-D-Aspartate receptor currents at the mossy fiber granule cell synapse of developing rat cerebellum.” *Neuroscience*, vol. 53, pp. 121-130.1993.

[34] E. D’Angelo, G. Defilippi, P. Rossi, and V. Taglietti, “Ionic mechanism of electroresponsiveness in cerebellar granule cells implicates the action of a persistent sodium current”. *Journal of Neurophysiology*, vol. 80, pp. 493-503.1998.

[35] D. Jaeger, “No parallel fiber volleys in the cerebellar cortex: evidence from cross-correlation analysis between Purkinje cells in a computer model and in recordings from anesthetized rats”. *Journal of Computational Neuroscience*, vol. 14, pp. 311-327. 2003.

[36] A. Roth, and M. Häusser, “Compartmental models of rat cerebellar Purkinje cells based on simultaneous somatic and dendritic patch-clamp recordings”. *Journal of Physiology*, vol. 535, pp. 445-472. 2001.

[37] S. Solinas, R. Maex, and E. De Schutter, “Synchronization of Purkinje cell pairs along the parallel fibre axis: a model.” *Neurocomputing* vol.52-54, pp. 97-102.2003

[38] D. Jaeger, E. De Schutter, E, and J. Bower, “The role of synaptic and voltage-gated currents in the control *of Purkinje cell spiki*ng: a modeling study”. *Journal of Neuroscience*, vol. 17, pp. 91-106. 1997.

[39] E. D’Angelo, T. Nieus, M. Bezzi, A. Arleo and O.J.-M. D. Coenen, “Modeling Synaptic Transmission and Quantifying Information Transfer in the Granular Layer of the Cerebellum”. *Lecture Notes in Computer Science, Springer-Verlag Publisher,* vol. 3512, pp. 107-13. 2005.

[40] T. V. Bliss, T. Lømo, T. “Long-lasting potentiation of synaptic transmission in the dentate area of the anaesthetized rabbit following stimulation of the perforant path”. *Journal of Physiology,* vol. 232, pp. 331-356. 1973.

[41] C. D. Hansel, D. J. Linden, and E. D'Angelo,”Beyond Parallel Fiber LTD: The Diversity of Synaptic and Non-Synaptic Plasticity in the Cerebellum”. *Nature Neuroscience*, vol. 4, pp. 467-475. 2001

[42] G. Bi, and M. Poo, “Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type”. Journal of Neuroscience, vol. 18, pp. 10464-10472. 1998.

[43] D. Marr,” A theory of cerebellar cortex”. *Journal of Physiology*, vol. 202, pp. 437-470.1969.

[44] J. S. Albus,”A theory of cerebellar function “. *Mathematical Bioscience,* vol. 10. 1971.

[45] M. Ito, “The cerebellum and neural control “. *New York, Raven Press.*1984.

[46] M. Ito, M. Sakurai, and P. Tongroach,” Climbing fiber induced depression of both mossy fiber responsiveness and glutamate sensitivity of cerebellar Purkinje cells”. *Journal of Physiology*, vol. 324, pp. 113-134. 1982.

[47] R. E. Kettner, S. Mahamud, H. Leung, N. Sittko, J. C. Houk, B. W. Peterson, and A. G. Barto, “Prediction of complex two-dimensional trajectories by a cerebellar model of smooth pursuit eye movement”. *Journal of Neurophysiology,* vol. 77, no. 4, pp. 2115-2130. 1997.

[48] J. L. Raymond, and S. G. Lisberger,”Neural learning rules for the vestibulo-ocular reflex.” *Journal of Neuroscience, vol.* 18 no. 21, pp. 9112-9129. 1998.

[49] A. Hamilton, D. W. Joyce, J. R. Flanagan, C. D. Frith, and D. M. Wolpert, “Kinematic cues in perceptual weight judgment and their origins in box lifting”. *Psychological Research,* vol. 71, no. 1, pp. 13-21. 2007.

[50] A. A. Ahmed, D. M. Wolpert, and J. R. Flanagan, ”Flexible representations of dynamics are used in object manipulation “. *Current Biology,* vol. 18, no. 10, pp. 1-6. 2008.

[51] D. A. Braun, A. Aertsen, D. M. Wolpert, and C. Mehring,” Motor task variation induces structural learning”. Current Biology, vol. 19, no. 4, pp. 352-357. 2009.