Brief Papers

Cerebellar Input Configuration Toward Object Model Abstraction in Manipulation Tasks

Niceto R. Luque, Jesus A. Garrido, Richard R. Carrillo, Olivier J.-M.D. Coenen and Eduardo Ros

***Abstract*— It is widely assumed that the cerebellum is one of the main nervous centers involved in correcting and refining planned movement and accounting for disturbances occurring during movement, for instance, due to the manipulation of objects which affect the kinematics and dynamics of the *robot-arm plant model*. In this brief, we evaluate a way in which a cerebellar-like structure can store a model in the granular and molecular layers. Furthermore, we study how its microstructure and input representations (context labels and sensorimotor signals) can efficiently support model abstraction toward delivering accurate corrective torque values for increasing precision during different-object manipulation. We also describe how the explicit (object-related input labels) and implicit state input representations (sensorimotor signals) complement each other to better handle different models and allow interpolation between two already stored models. This facilitates accurate corrections during manipulations of new objects taking advantage of already stored models.**

***Index Terms*— Adaptive, biological control system, cerebellum architecture, learning, robot, spiking neuron.**

I. Introduction

In the framework of a control task, many successful approaches which use different kinds of "learning" (adaptation mechanisms) in the control loop have been developed: reinforcement learning [1], where systems can learn to optimize their behavior making use of rewards and punishments, genetic algorithms [2], where control systems are evolved over many generations mimicking the process of natural evolution, recurrent artificial neural networks [3], and also, recently approaches based on biologically realistic spiking neural networks (SNNs) [4], [5]. Most of the works focused on SNNs addressing issues such as computational complexity and real-time feasibility [6], biologically plausible models of different complexity [7], effects of biological learning rules [8], etc. This brief represents a multidisciplinary research effort in which SNNs adopting a cerebellar-like neural topology are used with biologically plausible neural models. We evaluate how the topology of a biological neuronal circuit is specifically related with its potential functionality, starting from electrophysiological recordings (validating the cell models) to a proposed biologically plausible spiking control solution.

More concretely, in this brief, we describe how a SNN mimicking a cerebellar micro-structure allows an internal corrective model abstraction. By adopting a cerebellar-like network, we explore how different sensor representations can be efficiently used for a corrective model abstraction corresponding to different manipulated objects. When a new object is manipulated and the system detects that significant trajectory errors are being obtained, the abstracted internal model adapts itself to match the new model (kinematic and dynamic modifications of a base arm plant model). The stored models to be used can be selected by explicit object-related input signals (as specific input patterns generated for instance from the visual sensory pathway) or implicit signals (such as a haptic feedback). This can be seen as a "cognitive engine" that abstracts the inherent object features through perception-action loops and relates them with other incidental properties, such as color, shape, etc. The cognition process that relates both properties is important because it allows the inference of inherent properties just by activating explicit perceived primitives making possible to build up models of the environment that describe how it will "react" when interacting with it.

In the framework of a robot control task, manipulating objects that significantly affect the base kinematic and dynamic model with bio-inspired schemes is an open issue [5], [9], [10]. Biology seems to have developed (evolved) a scalable control system capable of abstracting new models in an incremental way in real time. This requires a smart model abstraction engine which is believed to be largely based on the cerebellum [11]. State-of-the-art simulation tools [12] and also hardware platforms [13], [14] allow cell-based simulation of nervous centers of certain complexity in the framework of biologically relevant tasks. This allows addressing studies in which the function and structure of nervous centers are conjointly evaluated to better understand how the system operation is based on cell and network properties.

The working hypothesis and methodology of this brief can be briefly described as follows:

1) we address a biologically relevant task which consists in an accurate manipulation of objects which affect a base (kinematic and dynamic) model of the base plant using low power actuators;

Manuscript received May 10, 2011; revised May 11, 2011; accepted May 11, 2011. Date of publication June 23, 2011; date of current version August 3, 2011. This work was supported in part by the European Union Sensorimotor Structuring of Perception and Action for emergent Cognition Project, under Grant IST 028056 and the national projects, MULTIVISION under Project TIC-3873 and ITREBA Project TIC-5060.

N. R. Luque, J. A. Garrido, and E. Ros are with the Department of Computer Architecture and Technology, University of Granada, Granada 18071, Spain (e-mail: nluque@atc.ugr.es; jgarrido@atc.ugr.es; eduardo@atc.ugr.es).

R. R. Carrillo is with the Department of Computer Architecture and Electronics, University of Almeria, Almeria 4120, Spain (e-mail: rcarrillo@atc.ugr.es).

O. J.-M. D. Coenen is with the Intelligent Systems Research Center, University of Ulster, Londonderry, Ireland (e-mail: olivier@oliviercoenen.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNN.2011.2156809

2) we define and implement a spiking-neuron-based cerebellum model to evaluate how different properties of the cerebellar model affect the functional performance of the system.

II. Material and Methods

A. Experimental Setup. Interfacing the Cerebellum Model with a Robot

1) Robot Plant: For the robot plant simulation, we have implemented an interface between the simulator of the light-weight-robot (LWR) developed at DLR [15] and the control loop including the cerebellum module. The LWR robot is a 7-DOF arm composed of revolute joints. For the sake of simplicity, in our experiments, we used the first (we will refer to it as q_1), third (q_2) and fifth joints (q_3), keeping the other ones fixed, limiting the number of degrees of freedom.

2) Training Trajectory: The described cerebellar model has been tested on a task of smooth pursuit, in a similar way to the one adopted by other authors [16]: a target moves along a repeated trajectory, which is a composition of sinusoidal components (this represents the desired trajectory). In previous works, we evaluated a simpler cerebellar model in a target-reaching task [5], [10] and a simple smooth pursuit task. After these preliminary works, in this brief, we study context switching capability, generalization-interpolation capability, and different input sensorimotor representations.

We use an 8-like trajectory defined by in (1). The trajectories of the individual joints have enough variation so that a sufficiently rich movement is executed allowing dynamic robot arm features to be revealed [17]. A multi-joint movement is more complex from a mechanical standpoint than a summed combination of single-jointed movements. This is due to the interaction torque values generated by one linkage moving on another. In this framework, the cerebellum role on the control task becomes more complex

$$q1 = A\sin\pi t,\ q2 = A\sin(\pi t + \theta),\ q3 = A\sin(\pi t + 2\theta). \tag{1}$$

3) Control Loop: It is largely assumed that the cerebellum plays a major role in motor control [18]–[20]. Based on this hypothesis, a wide range of cerebellar motor-control-system approaches have been proposed in the literature (for a review, the reader is referred to [21]). The central nervous system (CNS) executes three relevant tasks. The desired trajectory computation in visual coordinates, the task-space coordinate translation into body coordinates, and finally, the motor command generation. As in [22], in this brief, we have adopted the feedback-error learning (FEL) scheme in order to deal with variations in the dynamics of the robot-arm [22] in connection with a crude inverse dynamic model. But in contrast with the adaptation modules used by Miller *et al.* [20], we use a biologically plausible neural model as described in the next section. Using FEL, the association cortex supplies the motor cortex with the desired trajectory in body coordinates, where the motor command is generated using an inverse dynamic arm model.

Kawato *et al.* [22] relate different components of the control scheme with the biological counterpart. As described in [22], the spinocerebellum-magnocellular red nucleus system provides an accurate internal neural model of the dynamics of the musculoskeletal system which is learned by sensing the result of the movement. The cerebrocerebellum-parvocellular red nucleus system provides a crude internal neural model of the inverse-dynamics of the musculoskeletal system which is acquired while monitoring the desired trajectory.

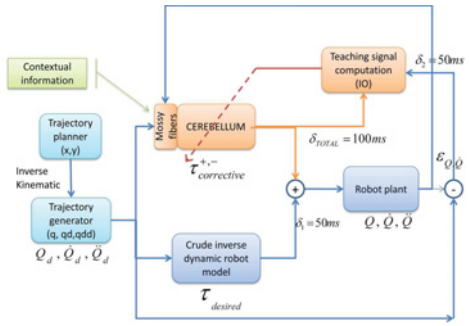

Fig. 1. Control loop.

The crude inverse dynamic model and the dynamical model work together by means of updating the motor command and predicting possible errors in the movement. As illustrated in Fig. 1, the cerebellar pathways are structured in a feedforward architecture, in which only information about sensory consequences of incorrect commands is obtained (i.e., the difference between actual and desired joint positions of the arm). We developed our cerebellar-based control loop according to this model as illustrated in Fig. 1.

We have also built a module to translate a small set of analog signals into a sparse cell-based spike-timing representation (spatio-temporal population coding). They encode the arm's desired and actual states (position and velocity) as well as contextual information. This module has been implemented using a set of mossy fibers (MFs) with specific receptive fields covering the working range of the different state variables.

B. Cerebellum Model

For extensive spiking network simulations, we have further developed and used an advanced event-driven simulator based on lookup Tables EDLUT [23]. EDLUT is an open-source tool [5] which accelerates the simulation of SNNs by compiling the dynamic response of pre-defined cell models into lookup tables before the actual network simulation. The proposed cerebellar architecture (Fig. 2) consists of the following layers:

1) MFs: MFs carry both contextual information and sensory joint information. A MF is modeled by a leaky I & F neuron, whose input current is calculated using overlapping radial basis functions as receptive fields in the value space of the input signals.

2) Granular Layer (1500 Cells): This layer represents a simplified cerebellar granular layer. The information given by MFs is transformed into a sparse representation in the granule layer [24]. Each granular cell (GR) has four excitatory input connections: three of them from randomly chosen joint-related MF groups and another one from a context-related MF. Parallel fibers (PFs) are the output of this layer.

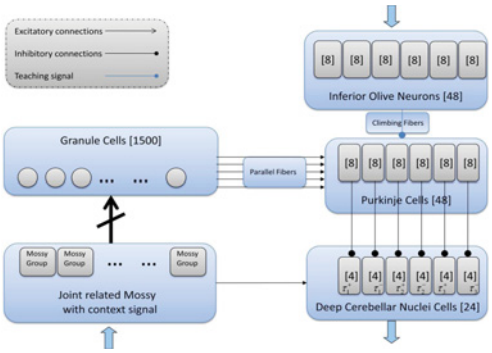

Fig. 2. Cerebellum configuration inputs encoding the movement (desired arm states, actual sensorimotor signals and context-related signals) are sent (upward arrow) through the PFs. Error-related inputs are sent (upper downward arrow) through the CFs. Outputs are provided by the DCN cells (lower downward arrow).

3) Climbing Fibers (CFs) (48 CFs): This layer is composed of six groups of eight *CF*s each. It carries the IO output which encodes teaching spike trains (related to the error) for the supervised learning in the PF–PC connections.

4) Purkinje Cells (PC) (48 Cells): They are divided into six groups of eight cells. Each GR is connected to 80% of the PCs. Each PC receives a single teaching signal from a CF. PF–PC synaptic conductances are modified by the learning mechanism during the training process.

5) Deep Cerebellar Nuclei Cells (DCN) (24 Cells): The cerebellum model output is generated by six groups of these cells. The corrective torque value of each joint is encoded by a couple of these groups, one group is dedicated to compensate positive errors (agonist) and the other one is dedicated to compensate negative errors (antagonist). Each neuron group in the DCN receives excitation from every MF cell and inhibition from the two corresponding PCs. In this way, the subcircuit PC–DCN–IO is organized in six microzones, three of them for joint positive corrections (one per joint) and the other three of them for joint negative corrections (one per joint). The DCN outputs are added as corrective activity in the control loop.

The MFs encode input representation in a rather specific way and the granular cells integrate information from different MFs. These characteristics partially embed functional roles of the inhibitory loop driven by the Golgi cells. Therefore, although Golgi cells have not been explicitly included, part of their functional roles has been integrated into the system.

C. Learning Process

It is well known that the cerebellum not only learns sequences of pre-defined voluntary movements but also adapts itself to external influences. This behavior seems to be very difficult to analyze, but the cerebellum presents a regular structured architecture that facilitates the study of how learning may take place in our context-driven scenario using this topology.

Although there seems to be an adaptation process at many sites within the cerebellar structure [25], the main synaptic adaptation driven by teaching or temporal signals (from the IO) seems to take place at the PF–PC synapses. We have adopted a plasticity mechanism that drives the modification of the PF–PC synapses in the cerebellar model, based on the concept of "eligibility trace" [16]. This trace aims to relate spikes from IO error-related activity and the previous activity of the PF that is supposed to have generated this error signal. The eligibility trace idea stems from experimental evidence showing that a spike in the CF afferent to a PC is more likely to depress a PF–PC synapse if the corresponding PF has been firing between 50 and 200 ms before the CF spike arrives at the PC [16], [26]. This is indicated in (3) [5] where the integration kernel $k(t)$ is defined in (2). A marginal peak occurs in the learning rule (around 450 ms after spike arrival) due to the event-driven simulation scheme (mathematical expression based on exponential functions modulating a periodic kernel, this presents a little hump), its impact in the global learning amount is negligible (4%) and can be considered as non-specific noise. In comparison with the previous learning schemes [5], [10] in similar cerebellum structures, this one allows to shift the maximum peak independently from the peak width. Therefore, we can tune the control loops to different sensorimotor delays and can narrow the maximum peak to allow more specific learning.

We have used a simplified spiking cerebellar neural network with spike-timing dependent plasticity (STDP). This plasticity has been implemented including long-term depression (LTD) and long-term potentiation (LTP) mechanisms in the following.

1) LTD produces a synaptic efficacy decrease when a spike from the IO reaches a PC. The IO output activity is interpreted as an error signal [18], [21], triggering a weight depression mechanism in synapses (PF–PC connections) depending on the received activity from the PFs. To calculate this amount of decrease, this previous activity is convolved with an integral kernel as defined by (2). Different expressions can be used for the learning rule [8]. This kernel mainly takes into account all the PF spikes which arrived 100 ms before the IO spike to overcome the effect of transmission delays of this range on sensory and motor signals (see Fig. 1). After this mechanism is repetitively activated, when the same pattern of PFs appears, the PC will not fire, in such a way, they will not inhibit its corresponding DCN cells [16], [27].
2) LTP produces a fixed increase in synaptic efficacy each time a spike arrives through a PF at the corresponding PC as defined by (2). For the sake of synaptic conductance equilibrium, LTD is accompanied by the opposite process (LTP), which takes place at this same synaptic site [28].

$$\begin{aligned} LTD: \forall i, \Delta w_i &= -\int_{-\infty}^{IOspiketime} k\left(t - t_{IOspike}\right) \\ LTP: \Delta w_i &= \alpha \end{aligned} \tag{2}$$

$$k(t) = e^{-(t-tpost\,synaptic\,spike)} \times \sin\left(t - t_{post\,synaptic\,spike}\right)^{20}. \tag{3}$$

These two learning rule components need to be tuned complementing each other to be able to efficiently reduce

action errors in the framework of a control task. In biological systems, sensors (for instance, skin sensors and proprioceptors) are not directly connected to the cerebellum. They pass through the cuneate nucleus [29] and other centers along the sensory pathway where signals seem to be efficiently organized to better address the cerebellar processing engine. In this brief, we investigate how the cerebellum model can take advantage of different cerebellar input representations during object manipulation.

D. Mossy Layer Configuration in the Cerebellar Model

Different mossy layer configuration models have been proposed in order to improve the cerebellum storage capability.

1) *Base "Desired-proprioceptive configuration."* It consists of 120 joint-related fibers, the MF layer has been divided into six groups of 20 fibers, three groups of fibers encoding joint positions (one group per joint) and the other three, encoding joint velocities.
2) Encoding Approach (EC) model. *Explicit Context* EC (16 context-related fibers plus 120 joint-related fibers): This mossy layer configuration uses the base desired-proprioceptive configuration adding 16 context-related fibers. The contextual information is coded by two groups of eight fibers. An external signal (related to the "label or any captured property" of the object, for instance assuming information captured through visual sensory system) feeds these dedicated-eight-grouped fibers.
3) IC model. *Implicit Context* EC (240 joint-related fibers): The MF layer consists of 12 groups of 20 fibers and delivers the actual and desired joint velocity and position information. It uses the base-desired proprioceptive configuration and adds three groups of fibers encoding actual joint positions and other three groups encoding actual joint velocities. The implicit contextual information is conveyed using these six groups of fibers. The actual position and velocity "helps" the cerebellum to recognize where and how far from the ideal (desired) situation it is. These deviations implicitly encode a "context-like" representation based on sensorimotor complexes.
4) EC & IC. *Explicit and Implicit Context* encoding approach (16 context-related fibers plus 240 joint-related fibers): It uses the base desired proprioceptive and incorporates also IC and EC architectural specifications. Thus, this MF layer is a combination of the EC and IC models described above.

The main aim of searching a proper mossy layer configuration is to exploit the capability of the granule layer for generating a sequence of active neuron populations without recurrence. This sequence is able to efficiently represent the passage of time (representation of different time passages are related with different input signals). Our system takes advantage of this spatiotemporal discrimination of input signals for learning different contexts.

As indicated in Section II-B, afferent MFs are randomly connected to granule cells, on average, four MFs [30] per

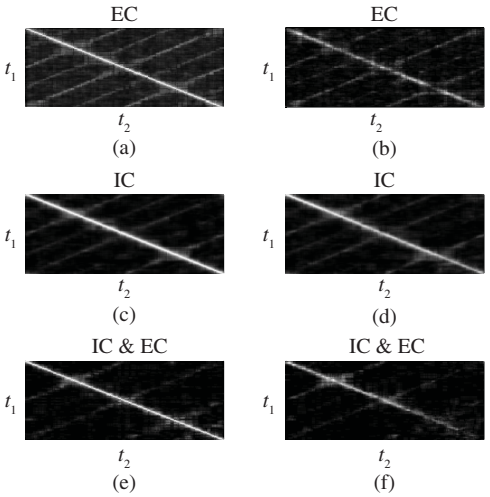

Fig. 3. Similarity indices for a spatiotemporal activity between two activity patterns using EC, IC, and IC & EC configurations. The values of indices are represented in gray scale; black 0, white 1. (a), (c), (e) Left side panels show a white diagonal band indicating a proper generation of a time-varying granular activity population. EC presents a mean gray value of 0.18, IC leads to a mean gray level of 0.101, and IC & EC leads to a mean gray value of 0.074. (b), (d), (f) Right side panels show similarity indices for two contexts. The darker the matrix is, the better uncorrelated activity patterns are. EC presents a mean gray value of 0.024, IC leads to a mean gray value of 0.096, and IC & EC achieves 0.044.

granule cell. When an input signal pattern arrives at the MFs, a spatiotemporal activity pattern is generated and the population of active neurons in the granule layer changes in time according to this received input. In order to evaluate the non-recurrence in this activation train, the following correlation function (4) is used [31]:

$$C(t_1, t_2) = \frac{\sum_i f_i(t_1) f_i(t_2)}{\sqrt{\sum_i f_i^2(t_1)} \sqrt{\sum_i f_i^2(t_2)}} \tag{4}$$

where f_i corresponds to the instantaneous frequency of the n_i neuron (frequency measured within a 20-ms time window). The numerator calculates the inner product of the population vector of active neurons at times t_1 and t_2, and the denominator normalizes the vector length. $C(t_1, t_2)$ takes values from 0 to 1, 0 if two vectors are complementary, 1 if two vectors are identical. To facilitate the production of accurate corrective terms, different input signals shall generate different spatiotemporal activity patterns. The following correlation function is used to evaluate this point as indicated in (5):

$$C(t_1, t_2) = \frac{\sum_i f_i^{(1)}(t_1) f_i^{(2)}(t_2)}{\sqrt{\sum_i f_i^{(1)2}(t_1)} \sqrt{\sum_i f_i^{(2)2}(t_2)}} \tag{5}$$

where $f_i^{(1)}$ and $f_i^{(2)}$ denote the activities of the n_i neuron at time t under different input signals (1 and 2, respectively).

The left panels in Fig. 3(a), (c) and (e) shows the similarity index using a $t_1 \times t_2$ matrix within the active granular population at t_1 and t_2. A wide white band, surrounding the main diagonal, points out that the index decreases monotonically

as the distance $[t_1–t_2]$ increases. That means a one-to-one correspondence between the active neuron population and time. This implies that a dynamically active neuron activity changing can represent the passage of time.

The right panels in Fig. 3(b), (d) and (f) show how different input signals can be discriminated by different activity patterns. The values of the similarity index are small suggesting that the two represented activity patterns are independent of the other one. Actual and desired entries of the IC configuration vary during time leading to a richer codification within a single context, while EC only uses desired entries varying along the trajectory execution. On the other hand EC gives a better granular activity codification between contexts by using its specific contextual signals. IC has no specific entries helping to distinguish activity patterns when using two different contexts. IC & EC takes advantages from both configurations, it uses the context and the position/velocity entries to produce a better time-varying granular activity population.

E. Experimental Methods

We have carried out several experiments to evaluate the capability of cerebellar architecture to select and abstract models using different cerebellar topologies. In these experiments, objects which significantly affect the dynamics and kinematics of the base plant model have been manipulated to evaluate the performance of different cerebellar configurations. Finally, we have also studied how interpolation/generalization can be naturally done for different plant + object models which have not been used during the training process. We divided the experiments into the following groups.

1) Cerebellar input configuration including only context-related signals (and desired arm states) (EC).
2) Cerebellar input configuration including only sensorimotor representation (IC) (i.e., desired and actual arm states).
3) Cerebellar input configuration including conjointly sensorimotor and context-related signals (IC & EC).

For this purpose, we have used a set of benchmark trajectories that we repeat in each iteration and evaluate how learning adapts the GR-PC weights to tune accurate corrective actions in the control loop (Fig. 1).

F. Quantitative Performance Evaluation

The learning process performance is characterized by using three estimates calculated from the mean absolute error (MAE) curve. The accuracy gain estimates the error reduction rate comparing the accuracy before and after learning. This estimate helps to interpret the adaptation capability of the cerebellum when manipulating different objects, provided that the initial error is different

$$Accuracy\ Gain = MAE_{initial} - \left[\frac{1}{n}\sum_{i=0}^{n} MAE_{(final-i)}\right]; \quad n = 30. \quad (6)$$

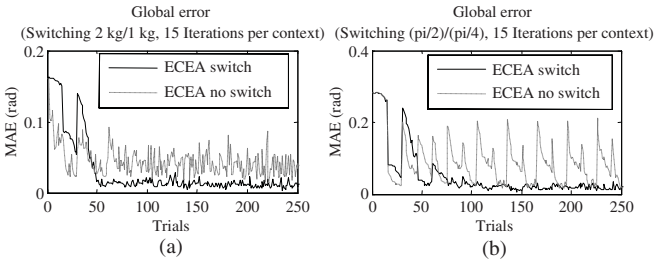

Fig. 4. Multi-context simulation with changes in dynamics and kinematics using EC cerebellar input. Each sample represents the MAE evolution (sum of error at different joints) for a trajectory execution (trial) during learning with no context-related signals and with explicit context-related signals. (a) Manipulating two different loads with and without context signals. Explicit context signals reduce 68.31% the final average error and 70.73% the final standard deviation. (b) Equivalent end-segment of the arm has been rotated in certain angles $\pi/2$ and $\pi/4$. The corrective torque values should compensate these different deviations in each context (with and without activated context signals). Explicit context switching signals reduce 62.04% the final average error and 26% the final standard deviation.

The final error (average error over the last 30 trials)

$$Final\ Error = \left[\frac{1}{n}\sum_{i=0}^{n} MAE_{(final-i)}\right]; \quad n = 30. \quad (7)$$

The final error stability (standard deviation over the last 30 movement trials)

$$Final\ Error\ Stability = \left[\frac{1}{n}\sum_{i=0}^{n} \sigma\left(MAE_{(final-i)}\right)\right]; \quad n = 30. \quad (8)$$

III. Experimental Results

A. EC Cerebellar Input

The explicit context EC uses a set of MFs to explicitly identify the context, assuming that they carry information provided by other areas of the CNS (such as vision which helps to identify the correct model to be used) or even cognitive signals. Therefore, a specific group of context-based MFs become active when the corresponding context is present. In this way, when a certain context becomes active, a GR population is pre-sensitized due to the specific context-related signals. We have randomly combined the sensor signals (desired position and velocity) of the different joints and the context-related signals (in the MF to GR connections) allowing granule cells to receive inputs from different randomly selected MFs (at the network-topology definition stage). In order to explicitly evaluate the capability of these signals to separate neural populations for different object models, each granule cell has four synaptic input connections: three random MF entries which deliver joint-related information and one MF which delivers context-related signals. In this case, we have evaluated the capability of the cerebellum model to efficiently use these context-related signals to learn to separate models when manipulating objects of different weights or different kinematics (deformation in the robot-plant end-segment) (Fig. 4).

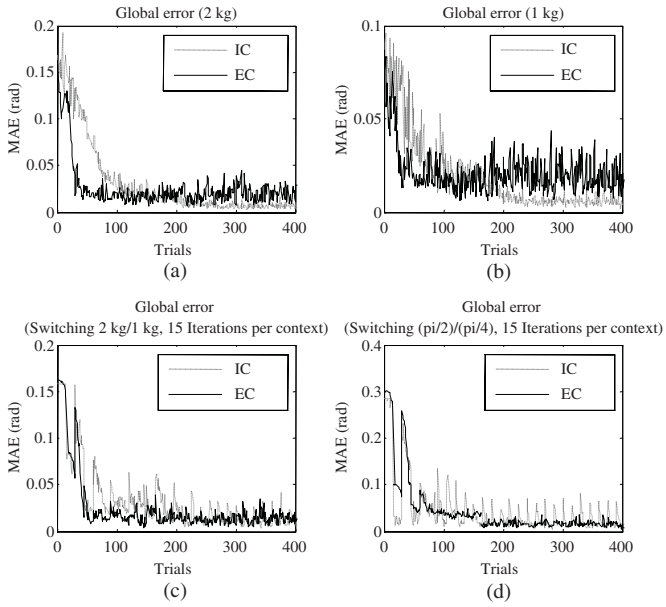

Fig. 5. Single-context simulation using EC and IC cerebellar input and multi-context simulation with changes in kinematics and dynamics using IC cerebellar input. (a) and (b) Manipulation of objects of different loads (2 kg/1 kg) without context signals. Each sample represents the MAE for a trajectory execution (trial). (c) MAE evolution during the learning process in EC and IC with dynamics-changing contexts. Two contexts with different loads are manipulated, switching every 15 trials. (d) MAE evolution for EC and IC configurations and two contexts with different ending deformation (kinematics change), switching every 15 trials.

B. IC Cerebellar Input

In this section, we define an implicit context encoding approach (IC), where no context-identifying signals are used. The sensor signals (actual position and velocity of the robot) implicitly encode (through MFs) the context during object manipulation. We have randomly combined the sensor signals (position and velocity) of the different joints (in the MF to GR connections) allowing granule cells to receive four inputs from different randomly selected MFs. The context models are distributed along cell populations. These cell populations are dynamically changing during the learning process (because the actual trajectory changes as corrective torque values are learned and integrated). Each time a new context is activated, the specific neural population is tuned due to the slightly different sensorimotor signals during the trajectory execution. The context switching in IC is done automatically and learning is carried out in a non-destructive manner, learned contexts are not destroyed (Fig. 5).The fact that IC transitions do not need explicit contextual information may indicate that this configuration allows interpolation between different learned contexts. This capability is explored by making the cerebellum learn two contexts alternately and then, presenting a new intermediate context (Fig. 6).

As shown in Fig. 5, although EC has a faster convergence speed, IC presents a lower final error (0.007 rad. average final error in IC against 0.018 rad. in EC) and a more stable behavior (0.002 rad. of standard deviation in IC against 0.006 rad. of standard deviation in EC) after the learning process.

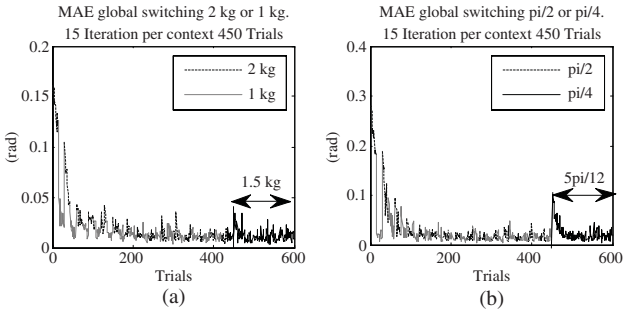

Fig. 6. Multi-context simulation with changes in kinematics and dynamics using IC cerebellar input. Interpolation capability. (a) After 450 trials of 15 iterations per context (2 kg/1 kg added alternatively to the robot arm), a new 1.5 kg context is presented to the cerebellum. (b) After 450 trials of 15 iterations per context (the end-segment of the robot arm includes different rotations: $\pi/2$ and $\pi/4$ angles alternatively), a new $5\pi/12$ context is presented to the cerebellum.

Accuracy gain, final error average and final standard deviation are similar in IC and EC. EC develops a better inter-context transition. Comparing EC with IC in a dynamic context switching experiment, we obtain context switching error discontinuities 47.6% larger and a standard deviation 24.7% higher in the EC explicitly canceling context switching signals than in the IC configuration [Figs. 4(a) versus 5(c)]. This highlights the importance of actual sensorimotor signals efficiently used in the IC configuration, compared to EC which only used desired states during manipulation.

Finally, comparing EC with IC in a kinematic context switching experiment, we obtain context switching error discontinuities 32.85% larger and a final standard deviation 16.71% higher in the EC without activating context switching signals than in the IC configuration [Figs. 4(b) versus 5(d) explicit context signals are efficiently used in EC configuration].

C. IC Plus EC Cerebellar Input

In this section, we evaluate how the previous EC and IC input representations are complementary. In this case, the cerebellar architecture includes both inputs. The MFs arriving in the cerebellum encode the desired states, the actual states (positions and velocities), and also, context signals which identify the current contexts.

In Fig. 7(c) IC & EC uses the pre-learned synaptic weights obtained in previous contexts to deal with a new payload. Nevertheless, sensorimotor state signals feeding MFs drive fast to a new contextual adaptation. The kinematics interpolation is not efficient [Fig. 7(d)], interpolation across kinematics changes is not an easy task (not linear).

IC & EC configuration also becomes robust against incongruent external context-related signals (for instance, extracted from vision). As shown in Fig. 7(e), during each epoch, the external context signal changes do not match the actual object switching (i.e., the external context signal does not remain constant while manipulating a 2 kg object and it does not do it either when using a 1 kg object). Thus, *context* 1 value in the first 2 kg-415-trial-context equals A and *context* 2 value in the first 1 kg-15-trial-contexts equals B.

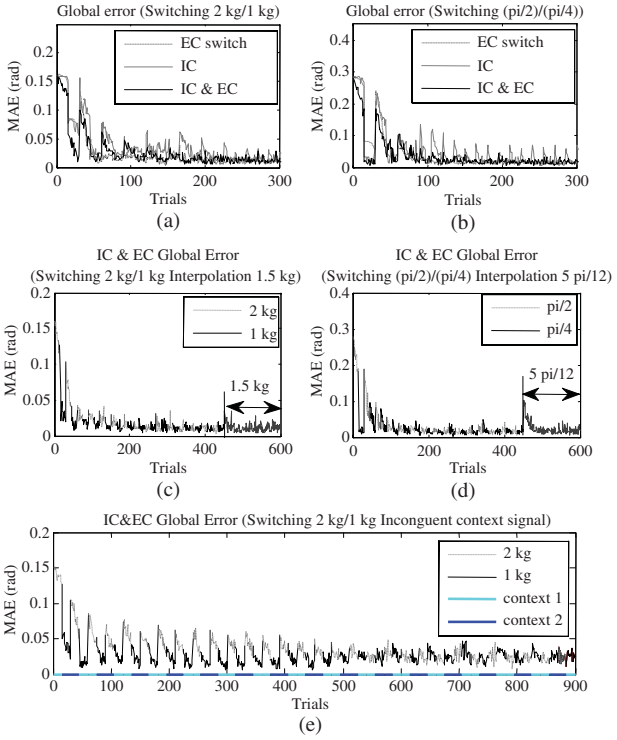

Fig. 7. Multi-context simulation with changes in kinematics and dynamics using EC, IC, and IC & EC cerebellar input. Interpolation of a new context and robustness against incongruent contextual input signals. (a) Dynamics correction task with different loads in the robot arm. (b) Kinematics correction task with different deviations in the end-segment of the robot arm. (c) 1.5 kg load is fixed to the end-segment of the robot. (d) $5\pi/12$ rotation in the end-segment of the robot is presented. (e) IC & EC configuration is able to avoid non-congruent contextual signals. Context-related input signals are indicated with highlighted colors in the x-axis of the plot.

In the following 15-trial-context switching trials, values A and B are interchanged. The incoming external contextual information is not congruent but, thanks to sensorimotor state signals (actual position and velocity of IC configuration), the cerebellum is able to deal with these "misleading" external signals.

IV. Conclusion

We have proposed a new simple biologically plausible cerebellar module which can abstract models of manipulated objects that significantly affect the initial dynamics and also kinematics of the plant (arm + object), providing corrective torque values toward more accurate movements. The results are obtained from object manipulation experiments. This new cerebellar approach, with two representations, receiving context-related inputs (EC) and actual sensory robot signals (IC) encoding the context during the experiments, has been studied. The IC & EC cerebellar configuration takes advantage of both configurations which complement each other. Smoother inter-context transitions are achieved at a fast convergence speed. It allows the interpolation of new contexts (different loads under manipulation) based on previously acquired models. Moreover, a good learning curve profile in long-term epochs can be achieved and finally, the capability of "overcoming" misleading external contextual information, making this cerebellar configuration robust against incongruent representations (Fig. 7), is remarkable. Furthermore, the results obtained with this kind of cerebellar architecture are coherent with the experiments [32], [33]. Therefore, when both representations congruently encode the context, they shall complement each other, while when they are incongruent, they interfere with each other. This is so because in the implemented cerebellar architecture, context classification and model abstraction tasks are carried out in a distributed manner. No pre-classification process is executed to disambiguate incongruent context identification. In our approach, we have also evaluated how sensorimotor representation can overcome incongruent incidental context-related signals (i.e., sensorimotor representation dominating a context-related incongruent signal).

In a classical machine learning approach, disambiguation is usually explicitly done through a classification module (decision making) that can be tuned to adopt a winner-takes-all strategy and leads to a single context model to be recalled even in this incongruent context representation. In biological systems, this kind of pre-classification (disambiguation) mechanisms may be processed in other nervous centers, although it may reduce the interpolation and generalization capabilities of the cerebellar model presented.

References

[1] A. G. Barto and M. Sridhar, "Recent advances in hierarchical reinforcement learning," *Discr. Event Dyn. Syst.*, vol. 13, no. 4, pp. 341–79, 2003.

[2] D. A. Linkens and H. O. Nyongesa, "Learning systems in intelligent control: An appraisal of fuzzy, neural and genetic algorithm control applications," *IEEE Proc. Control Theory Appl.*, vol. 143, no. 4, pp. 367–386, Jul. 1996.

[3] K. J. Hunta, D. Sbarbaroa, R. Z. Bikowskia, and P. J. Gawthropa, "Neural networks for control systems—A survey," *Automatica*, vol. 28, no. 6, pp. 1083–112, Nov. 1992.

[4] S. Ghosh-Dastidar and H. Adeli, "Spiking neural networks," *Int. J. Neural Syst.*, vol. 19, no. 4, pp. 295–308, 2009.

[5] R. R. Carrillo, E. E. Ros, C. Boucheny, and O. Coenen, "A real-time spiking cerebellum model for learning robot control," *Biosystems*, vol. 94, nos. 1–2, pp. 18–27, Oct.–Nov. 2008.

[6] M. J. Pearson, A. G. Pipe, B. Mitchinson, K. Gurney, C. Melhuish, I. Gilhespy, and M. Nibouche, "Implementing spiking neural networks for real-time signal-processing and control applications: A model-validated FPGA approach," *IEEE Trans. Neural Netw.*, vol. 18, no. 5, pp. 1472–1487, Sep. 2007.

[7] E. M. Izhikevich, "Simple model of spiking neurons," *IEEE Trans. Neural Netw.*, vol. 14, no. 6, pp. 1569–1572, Nov. 2003.

[8] W. Gerstner and W. Kistler, *Spiking Neuron Models: Single Neurons, Populations, Plasticity*. Cambridge, U.K.: Cambridge Univ. Press, 2002.

[9] M. A. Arbib, G. Metta, and P. van der Smagt, "Neurorobotics: From vision to action," in *Handbook of Robotics*. New York: Springer-Verlag, 2008, ch. 62, pp. 1453–1475.

[10] C. Boucheny, R. R. Carrillo, E. Ros, and O. J.-M. D. Coenen, *Real-Time Spiking Neural Network: An Adaptive Cerebellar Model* (Lecture Notes in Computer Science), vol. 3512. New York: Springer-Verlag, 2005, pp. 136–144.

[11] D. M. Wolpert, R. C. Miall, and M. Kawato, "Internal models in the cerebellum," *Trends Cog. Sci.*, vol. 2, no. 9, pp. 338–347, Sep. 1998.

[12] R. Brette, M. Rudolph, T. Carnevale, M. Hines, D. Beeman, J. M. Bower, M. Diesmann, A. Morrison, P. H. Goodman, and F. C. Harris, "Simulation of networks of spiking neurons: A review of tools and strategies," *J. Comput. Neurosci.*, vol. 23, no. 3, pp. 349–398, 2007.

[13] E. Ros, E. M. Ortigosa, R. Agís, R. Carrillo, and M. Arnold, "Real-time computing platform for spiking neurons (RT-spike)," *IEEE Trans. Neural Netw.*, vol. 17, no. 4, pp. 1050–1063, Jul. 2006.

[14] M. J. Pearson, A. G. Pipe, B. Mitchinson, K. Gurney, C. Melhuish, I. Gilhespy, and M. Nibouche, "Implementing spiking neural networks for real-time signal-processing and control applications: A model-validated FPGA approach," *IEEE Trans. Neural Netw.*, vol. 18, no. 5, pp. 1472–1487, Sep. 2007.
[15] J. Butterfaß, M. Grebenstein, H. Liu, and G. Hirzinger, "DLR-hand II: Next generation of a dextrous robot hand," in *Proc. IEEE Int. Conf. Robot. Autom.*, vol. 1. 2001, pp. 109–114.
[16] R. E. Kettner, S. Mahamud, H.-C. Leung, N. Sitkoff, J. C. Houk, B. W. Peterson, and A. G. Barto, "Prediction of complex 2-D trajectories by a cerebellar model of smooth pursuit eye movement," *J. Neurophysiol.*, vol. 77, no. 4, pp. 2115–2130, Apr. 1997.
[17] H. Hoffmann, G. Petckos, S. Bitzer, and S. Vijayakumar, "Sensor-assisted adaptive motor control under continuously varying context," in *Proc. Int. Conf. Inf. Control, Autom. Robot.*, Angers, France, 2007, pp. 1–8.
[18] M. Ito, *The Cerebellum and Neural Control*. New York: Raven, 1984.
[19] E. Todorov, "Optimality principles in sensorimotor control (review)," *Nat. Neurosci.*, vol. 7, pp. 907–915, Aug. 2004.
[20] L. E. Miller, R. N. Holdefer, and J. C. Houk, "The role of the cerebellum in modulating voluntary limb movement commands," *Arch. Ital. Biol.*, vol. 140, no. 3, pp. 175–183, 2002.
[21] M. Ito, "Cerebellar circuitry as a neuronal machine," *Prog. Neurobiol.*, vol. 78, nos. 3–5, pp. 272–303, Feb.–Apr. 2006.
[22] M. Kawato, K. Furukawa, and R. Suzuki, "A hierarchical neural-network model for control and learning of voluntary movement," *Biol. Cybern.*, vol. 57, no. 3, pp. 169–185, 1987.
[23] E. Ros, 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 Comput.*, vol. 18, no. 12, pp. 2959–2993, Dec. 2006.
[24] 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), vol. 3512. Berlin, Germany: Springer-Verlag, 2005, pp. 107–113.
[25] 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," *Nat. Neurosci.*, vol. 4, no. 5, pp. 467–475, May 2001.
[26] R. R. Llinás, "Inferior olive oscillation as the temporal basis for motricity and oscillatory reset as the basis for motor error correction," *Neuroscience*, vol. 162, no. 3, pp. 797–804, Sep. 2009.
[27] M. Ito, M. Sakurai, and P. Tongroach, "Climbing fibre induced depression of both mossy fibre responsiveness and glutamate sensitivity of cerebellar Purkinje cells," *J. Physiol.*, vol. 324, pp. 113–134, Jan. 1982.
[28] R. R. Carrillo, E. Ros, B. Barbour, C. Boucheny, and O. J.-M. D. Coenen, "Event-driven simulation of neural population synchronization facilitated by electrical coupling," *Biosystems*, vol. 87, nos. 2–3, pp. 275–280, Feb. 2007.
[29] P. E. Roland, "Sensory feedback to the cerebral cortex during voluntary movement in man," *Behav. Brain Sci.*, vol. 1, no. 1, pp. 129–171, 1978.
[30] E. Mugnaini, R. L. Atluri, and J. C. Houk, "Fine structure of granular layer in turtle cerebellum with emphasis on large glomeruli," *J. Neurophysiol.*, vol. 37, no. 1, pp. 1–29, Jan. 1974.
[31] T. Yamazaki and S. Tanaka, "The cerebellum as a liquid state machine," *Neural Netw.*, vol. 20, no. 3, pp. 290–297, Apr. 2007.
[32] A. F. Hamilton, D. W. Joyce, J. R. Flanagan, C. D. Frith, and D. M. Wolpert, "Kinematic cues in perceptual weight judgement and their origins in box lifting," *Physiol. Res.*, vol. 71, no. 1, pp. 13–21, 2007.
[33] A. A. Ahmed, D. M. Wolpert, and J. R. Flanagan, "Flexible representations of dynamics are used in object manipulation," *Current Biol.*, vol. 18, no. 10, pp. 763–768, May 2008.

Adaptive Neural Output Feedback Controller Design with Reduced-Order Observer for a Class of Uncertain Nonlinear SISO Systems

Yan-Jun Liu, Shao-Cheng Tong, Dan Wang, Tie-Shan Li, and C. L. Philip Chen, *Fellow, IEEE*

***Abstract*— An adaptive output feedback control is studied for uncertain nonlinear single-input–single-output systems with partial unmeasured states. In the scheme, a reduced-order observer (ROO) is designed to estimate those unmeasured states. By employing radial basis function neural networks and incorporating the ROO into a new backstepping design, an adaptive output feedback controller is constructively developed. A prominent advantage is its ability to balance the control action between the state feedback and the output feedback. In addition, the scheme can be still implemented when all the states are not available. The stability of the closed-loop system is guaranteed in the sense that all the signals are semiglobal uniformly ultimately bounded and the system output tracks the reference signal to a bounded compact set. A simulation example is given to validate the effectiveness of the proposed scheme.**

***Index Terms*— Adaptive neural control, nonlinear systems, output feedback control, reduced-order observer.**

I. Introduction

During the past two decades, the adaptive control of uncertain systems has attracted much attention. Several typical results have been designed in [1]–[3]. A main restriction in these results is that the uncertainties are required to satisfy linearly parameterized conditions with known functions. But this assumption is difficult to be ensured in practice.

In the recent years, many researchers have devoted much effort to deal with the problem of adaptive tracking control for nonlinear systems with completely unknown functions. By using the approximation property of the neural network (NN) or the fuzzy logic systems, several elegant adaptive control strategies have been proposed in [4]–[16] for uncertain nonlinear systems. However, a major constraint in these results is that the system state variables are assumed to be measurable. If the system states are unavailable, these results cannot be applied in practice. In the last decade, much progress has been

Manuscript received October 24, 2009; revised December 12, 2010; accepted June 6, 2011. Date of publication July 12, 2011; date of current version August 3, 2011. This work was supported in part by the National Natural Science Foundation of China, under Grant 61074014, Grant 61074017, and Grant 60874056. The National Fundamental Research 973 Program of China, under Grant 2011CB302801, Macau Science and Technology Development Fund, under Grant 008/2010/A1, and the Natural Science Fund of Liaoning Province, under Grant 20102095.

Y.-J. Liu and S.-C. Tong are with the School of Sciences, Liaoning University of Technology, Jinzhou 121001, China (e-mail: liuyanjun@live.com; jztsc@sohu.com).

D. Wang is with Marine Engineering College, Dalian Maritime University, Dalian, Liaoning 116026, China (e-mail: dwangdl@gmail.com).

T.-S. Li is with Navigation College, Dalian Maritime University, Dalian, Liaoning 116026, China (e-mail: tieshanli@126.com).

C. L. P. Chen is with the Faculty of Science and Technology, University of Macau, Macau, China (e-mail: philip.chen@ieee.org).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNN.2011.2159865