An Internal Model for Acquisition and Retention of Motor Learning During Arm Reaching

An Internal Model for Acquisition and Retention of Motor Learning During Arm Reaching

Luca Lonini
luca.lonini@libero.it
Centre for Integrated Research—Biomedical Robotics and Biomicrosystems Lab, Università Campus Bio-Medico, 00128, Rome, Italy

Laura Dipietro
lauradp@mit.edu
Newman Lab for Biomechanics and Human Rehabilitation, Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139, U.S.A.

Loredana Zollo
l.zollo@unicampus.it
Eugenio Guglielmelli
e.guglielmelli@unicampus.it
Centre for Integrated Research—Biomedical Robotics and Biomicrosystems Lab, Università Campus Bio-Medico, 00128, Rome, Italy

Hermano Igo Krebs
hikrebs@mit.edu
Newman Lab for Biomechanics and Human Rehabilitation, Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139, U.S.A.; Burke Medical Research Institute, Department of Neurology and Neuroscience, Weill Medical College of Cornell University, White Plains, NY 10605, U.S.A.; and Department of Neurology, University of Maryland School of Medicine, Baltimore, MD 21201, U.S.A.

Humans have the ability to learn novel motor tasks while manipulating the environment. Several models of motor learning have been proposed in the literature, but few of them address the problem of retention and interference of motor memory. The modular selection and identification for control (MOSAIC) model, originally proposed by Wolpert and Kawato, is one of the most relevant contributions; it suggests a possible strategy on how the human motor control system learns and adapts to novel environments. MOSAIC employs the concept of forward and inverse models. The same group later proposed the hidden Markov model (HMM) MOSAIC, which affords learning multiple tasks. The significant drawback of this second approach is that the HMM must be trained with a complete data set that includes all contexts. Since the number of contexts

or modules is fixed from the onset, this approach does not afford incremental learning of new tasks. In this letter, we present an alternative architecture to overcome this problem, based on a nonparametric regression algorithm, named locally weighted projection regression (LWPR). This network structure develops according to the contexts allowing incremental training. Of notice, interaction force is used to disambiguate among different contexts. We demonstrate the capability of this alternative architecture with a simulated 2 degree-of-freedom representation of the human arm that learns to interact with three distinct objects, reproducing the same test paradigm of the HMM MOSAIC. After learning the dynamics of the three objects, the LWPR network successfully learns to compensate for a novel velocity-dependent force field. Equally important, it retains previously acquired knowledge on the interaction with the three objects. Thus, this architecture allows both incremental learning of new tasks and retention of previously acquired knowledge, a feature of human motor learning and memory.

1 Introduction

Humans are able to learn a large repertoire of motor behaviors and manipulate objects with different dynamic properties. Also, they exhibit capabilities of fast adaptation to different environmental situations and show high reactivity in fast movements. Several studies, including experiments on the adaptation of unimpaired subjects to novel force fields, led to the concept of an internal model (e.g., Ito, 1970; Shadmehr & Mussa-Ivaldi, 1994; Flanagan, Wolpert, & Johansson, 2001; Flanagan, Vetter, Johansson, & Wolpert, 2003; Franklin, Osu, Burdet, Kawato, & Milner, 2003). Internal models are thought to be neural representations of human body and of the interaction environment, generated by the central nervous system and continuously evolving during learning (Craik, 1943). They may represent an efficient mechanism to build and solve complex computational issues of motor control (Miall, Weir, Wolpert, & Stein, 1993; Miall & Wolpert, 1996; Wolpert & Ghahramani, 2000). In particular, they may play a fundamental role in generating anticipatory actions and performing fast and coordinated movements (Kawato, 1999) while maintaining low stiffness (Gomi & Kawato, 1996). It was hypothesized that the cerebellum acquires multiple internal models in order to adapt to different environmental situations, and each model is tuned to a specific context (Wolpert & Kawato, 1998). Evidence was provided through studies on the cerebellum based on fMRI (Imamizu, Kuroda, Miyauch, Yoshioka, & Kawato, 2003).

Several computational solutions found in the robotics and neuroscience literature try to implement human learning of multiple tasks by generating multiple internal models (Gomi & Kawato, 1993; Narendra & Balakrishnan, 1997; Wolpert & Kawato, 1998; Haruno, Wolpert, & Kawato, 2001; Doya, Samejima, Katagiri, & Kawato, 2002; Petkos & Vijayakumar, 2007). Most of the approaches require defining in advance the number of internal models needed to perform different tasks; the context is often inferred using a Bayesian probabilistic model in order to select the proper internal model for the particular context (Wolpert & Kawato, 1998; Haruno et al., 2001; Doya et al., 2002; Petkos & Vijayakumar, 2007). Furthermore, retention and interference in motor memory remain unsolved issues. A biologically inspired model that accounts for these issues could be used to analyze and improve motor training procedures. As we are exploring the common traits between motor learning and motor recovery from stroke, such models could be used to improve and validate neurorehabilitation therapy protocols.

Modular selection and identification for control (MOSAIC) is a computational model that copes with the problem of learning multiple motor tasks under a biological framework (Wolpert & Kawato, 1998). It proposes the notion of paired internal forward and inverse models for human motor learning and control, thus suggesting a possible strategy used to learn and adapt to different dynamic conditions. The forward and inverse models act respectively as predictors and controllers. Each forward and inverse model pair is coupled in a module, where each module specializes in a different context (task). Two implementations were described for the responsibility function, which is used to select among modules: one was based on the soft-max likelihood function, which was proposed in the original version of the MOSAIC (Wolpert & Kawato, 1998), and another was based on a hidden Markov model (HMM), which was proposed in an extended version of the MOSAIC (Haruno et al., 2001).

The HMM is a probabilistic model of context transitions. Contrary to the soft-max likelihood function, the HMM performs this switching role quite well (Haruno et al., 2001). However, because the HMM estimates the probability of switching from one module to another (transition probabilities), the HMM MOSAIC needs to be trained on the complete task space (manipulated objects). This implies that incremental learning cannot be accomplished (Rabiner, 1989). The number of modules is fixed and set at the beginning of the training; objects (tasks) lying outside the polyhedral defined by the objects used for training cannot subsequently be added to the model (Haruno et al., 2001). This is a major limitation of the HMM MOSAIC since humans can learn novel tasks in an orderly fashion. This limitation precludes us from employing the HMM MOSAIC in neurorehabilitation to model motor recovery from stroke. For subjects recovering from stroke, the ability to learn different motor tasks in an orderly fashion is crucial (Krebs, Hogan, Aisen, & Volpe, 1998; Hogan et al., 2006; Krebs et al., 2007).

To extend the HMM MOSAIC to such circumstances requires developing new methods that afford dynamic allocation of hidden states. In fact, the machine learning community has been developing such novel HMM methods (e.g., Beal, Ghahramani, & Rasmussen, 2002), but so far all attempts employ heavy sampling methods, which are not suitable for motor control. We are also exploring alternative approaches to model this feature of motor learning. Here, instead of using a model-switching strategy, with a predefined number of multiple internal models, we report on an approach that allows a single internal model to handle multiple contexts and affords incremental learning. In contrast to the other approaches in the literature, our approach employs the interaction force to directly infer the context. We use a locally weighted projection regression (LWPR) network (Vijayakumar, D'Souza, & Schaal, 2005) to implement the inverse model. LWPR networks belong to the class of locally weighted learning algorithms (Atkeson, Moore, & Schaal, 1997). These algorithms exploit spatially localized linear models to approximate nonlinear functions and have been successfully used for incremental learning in robotic platforms (Schaal, Atkeson, & Vijayakumar, 2002; Bacciu, Zollo, Guglielmelli, Leoni, & Starita, 2004; Zollo et al., 2008). A salient feature of the LWPR network is its reduced computational burden, which is accomplished by projecting the network inputs onto orthogonal directions using a partial least squares (PLS) algorithm (Geladi & Kowalski, 1986) and performing regressions in the reduced space (Vijayakumar et al., 2005). Such a method allows incremental learning as long as inputs to the network are spatially separated in the "input domain." We use the interaction force between the hand and the environment to perform such a separation in the input variables space.

Our aim is to learn different tasks without limiting a priori the number of tasks and minimizing interference with the previously learned tasks. We demonstrate that our LWPR-based architecture is capable of learning to manipulate the same three objects trained in the HMM MOSAIC literature (Haruno et al., 2001), each presented in a separate training session, and after completing such training, learning a new task (compensating for the perturbations generated by a velocity-dependent force field) while retaining the previously acquired tasks. In other words, this architecture is capable of acquiring and retaining new motor tasks presented in an orderly fashion, thus providing a more realistic model for human lifelong learning capability and for the neurorehabilitation process.

Several studies have investigated arm-reaching movements of unimpaired subjects exposed to a robot-generated velocity-dependent perturbation force (e.g., Shadmehr & Mussa-Ivaldi, 1994; Krebs, Brashers-Krug, et al., 1998; Osu, Hirai, Yoshioka, & Kawato, 2004; Davidson & Wolpert, 2004). These studies demonstrated that after an adequate number of trials, subjects were able to produce forces that counteracted the forces applied by the robot. In our case, the forces produced by the simulated force field define a novel task that the LWPR architecture has to learn to manipulate.

2 LWPR-Based Control Architecture

The LWPR-based architecture is shown in Figure 1. It consists of a proportional derivative (PD) controller, which generates a feedback torque τ_{FB}

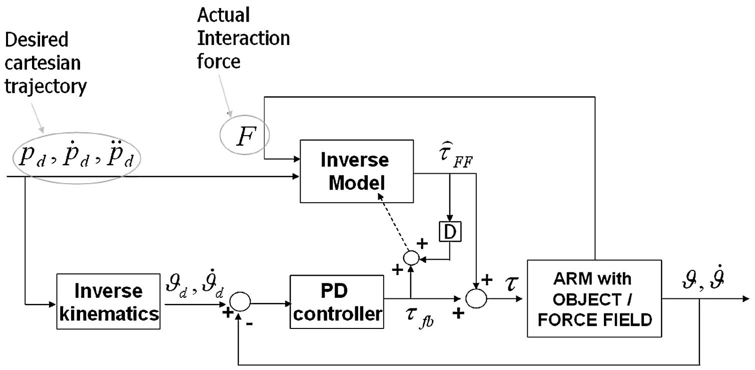

Figure 1: Proposed LWPR-based architecture. It consists of an inverse model and a PD feedback controller. The inverse model is implemented using the LWPR algorithm. Its inputs are the desired Cartesian trajectory in terms of position, velocity, and accelerations and the interaction force F acting at the hand. The output is the feedforward torque ($\hat{\tau}_{FF}$). The PD feedback controller inputs are the actual and desired angular positions and velocities (θ, θ_d and $\dot{\theta}, \dot{\theta}_d$), which are derived from the Cartesian trajectory by means of analytical inverse kinematics. The output is the feedback torque τ_{fb}. The feedback and the feedforward motor commands are combined to generate the training signal according to the feedback error learning strategy. The block labeled D stands for a sample unit delay. The force input F is used to learn an inverse dynamic model of the arm plus the environment.

proportional to the difference between the desired and actual joint trajectory, and an adaptive inverse model of the arm, whose output is a feedforward torque $\hat{\tau}_{\text{FF}}$. The trajectory is planned in the Cartesian space, and the corresponding joint angles are computed by inverse kinematics. This choice does not limit the validity of our approach. In the case of redundant systems, redundancy can be managed by the selection of appropriate algorithms (Baillieul & Martin, 1990).[1] We train an inverse model that inputs the desired Cartesian trajectory and the interaction forces between the hand and the environment to generate the proper torques to compensate for these external forces while following the desired trajectory. The learning scheme should be able to learn novel data presented in an orderly fashion while minimizing interference and retaining previously learned tasks. Such data account for the different tasks experienced according to the interaction force level.

[1] For redundant systems like humans, analytical inverse solutions to the kinematics do not exist normally. Thus, differential and numerical solutions need to be employed.

We illustrate our approach with a simulated arm made up of two rigid links moving in the horizontal plane, without any friction. The dynamic equation of the arm can be written as follows:

$$I(\theta)\ddot{\theta} + C(\theta, \dot{\theta})\dot{\theta} = \tau - J^T F, \tag{2.1}$$

where:

- $\theta, \dot{\theta}, \ddot{\theta}$: respectively, angular position, velocity, and acceleration of the arm (2 × 1)
- I : inertia matrix (2 × 2)
- C: centripetal and Coriolis matrix (2 × 2)
- τ: control torque vector (2 × 1)
- J^{T} : transpose of the Jacobian matrix (2 × 2)
- F: interaction forces vector acting on the arm in Cartesian space (2 × 1)

The inverse model is implemented using the LWPR network (Vijayakumar et al., 2005), which approximates a nonlinear function (i.e., the dynamic model) by means of piecewise linear regressions. Each linear model approximates the nonlinear function within a region of domain determined by a gaussian kernel called a receptive field (RF). The RF assigns a weight to an input data vector x according to the following function:

$$w_k = e^{\left(-\frac{1}{2}(x-c_k)^T D_k (x-c_k)\right)}, \tag{2.2}$$

where w_k is the value of the kth gaussian kernel and c_k the center of the RF. The matrix D_k (distance metric) determines the size and shape of the RF, thus defining the region of validity of the kth linear model. Each model gives its individual prediction according to the following linear relationship:

$$\begin{aligned} \hat{y}_k &= (x - c_k)^T b_k + b_{0,k} = \tilde{x}^T \beta_k \\ \tilde{x} &= ((x - c_k)^T, 1)^T, \quad \beta_k = (b_k, b_{o,k}), \end{aligned} \tag{2.3}$$

where b_k and $b_{0,k}$ are the parameters of the kth linear model (i.e., the regression coefficients). Linear models are added on an as-needed basis: whenever input data do not activate any existing RFs above a threshold, a new linear model is added, with the corresponding RF centered at that point. In our simulation, this threshold is set to 0.1. The algorithm then tunes the parameters D_k and β_k for each RF using nonparametric regression techniques (Schaal & Atkeson, 1998; Vijayakumar et al., 2005). The output $\hat{y}$ of the network is given by the weighted sum of the individual predictions $\hat{y}_k$,

$$\hat{y} = \frac{\sum_{k=1}^{K} \hat{y}_k w_k}{\sum_{k=1}^{K} w_k}, \tag{2.4}$$

where K is the number of linear models that have been allocated.

Table 1: The Characteristics of the Three Objects.

Object	M (Kg)	B (N s/m)	K (N/m)
1	1	2	8
2	5	7	4
3	8	3	1

To approximate the mapping described in equation 2.1, the network inputs are the desired Cartesian trajectory and interaction forces at the hand, and the outputs are the feedforward torques. The inverse model is trained using a feedback error learning strategy. In this approach, the feedback controller is exploited to convert trajectory errors into the motor command space, thus providing an error signal to train a neural network (Kawato, 1990). The LWPR algorithm requires a target signal, and since the target motor commands are not available, the following pseudo-target signal (Shibata & Schaal, 2001) is adopted:

$$\tau_{FF}(t-1) = \hat{\tau}_{FF}(t-1) + \tau_{FB}(t), \tag{2.5}$$

where τ_{FF} is the network target (i.e., the feedforward torque), $\hat{\tau}_{FF}$ is the network output, and τ_{FB} is the feedback controller output.

3 Model Testing and Validation

3.1 External Forces. Arm simulations are carried out first with the same three objects described in the HMM MOSAIC (Haruno et al., 2001), followed by reaching movements within a velocity-dependent force field (Shadmehr & Mussa-Ivaldi, 1994). The interaction force when interacting with the three objects is described by

$$\begin{aligned} F &= M\ddot{p} + B\dot{p} + Kp \\ p &= [x \quad y]^T, \end{aligned} \tag{3.1}$$

where M is the 2×2 mass matrix, B is the 2×2 damping matrix, and K is the 2×2 stiffness matrix. These equal value diagonal matrices are exactly the same as described in Haruno et al. (2001). The sole difference is that we allow the interaction forces to actuate on both x and y axes. Table 1 summarizes the parameters for the three objects.

Because it has been used extensively in previous motor learning research, we chose an experimental design similar to that of Shadmehr and Mussa-Ivaldi (1994). The perturbation forces were velocity dependent, a conservative force field ($\vec{F}^T \cdot \vec{V} = 0$) according to the following relations

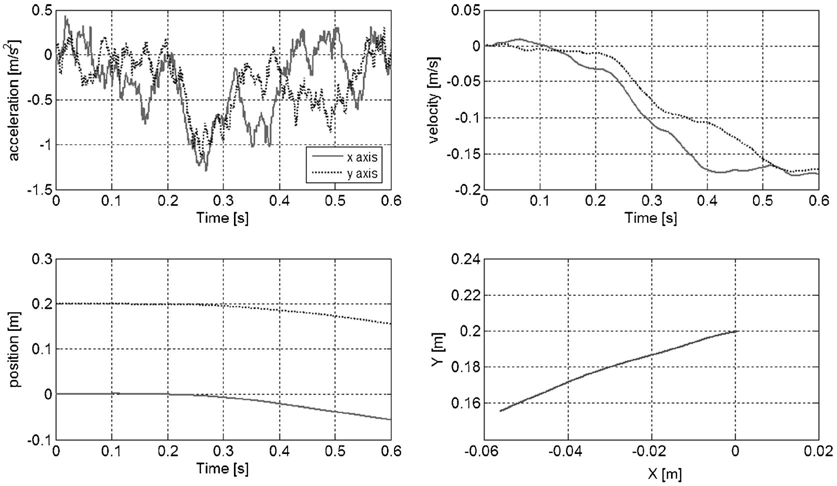

Figure 2: Desired acceleration, velocity, and position profiles on the x and y axis generated by an Ornstein-Uhlenbeck process. The bottom right plot shows the resulting hand trajectory.

(Krebs, Brashers-Krug, et al., 1998):

$$\begin{bmatrix} F_x \\ F_y \end{bmatrix} = \begin{bmatrix} 0 & B_c \\ -B_c & 0 \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix}, \tag{3.2}$$

where B_c is a coefficient equal to 40 N s/m; the velocity in x-y direction $(\dot{x}, \dot{y})$ is given in m/s; and the forces in the x-y direction ($\mathrm{F_x}$, $\mathrm{F_y}$) are in newton, with the force field actuating in a clockwise fashion.

3.2 Desired Trajectory. To allow proper comparison, we employed the same desired trajectories employed by Haruno et al. (2001) in the simulations of the HMM MOSAIC model. Trajectories are generated by an Ornstein-Uhlenbeck (OU) process (Uhlenbeck & Ornstein, 1930). A stochastic process X_t is an OU process if it is stationary, gaussian, Markovian, and continuous in probability. It can be described by the following linear stochastic differential equation:

$$dX_t = -\rho(X_t - \mu)\,dt + \sigma \cdot dW_t, \tag{3.3}$$

where W_t is Brownian motion with a unit variance parameter and μ, ρ, and σ are constants. From equation 3.3, a desired stochastic acceleration profile is generated for each axis. Figure 2 shows the desired hand accelerations, velocities, and positions and the resulting hand path.

In the case of the force field, the desired trajectory is a reaching movement. Human reaching movements tend to be straight in Cartesian space and have a typical bell-shaped velocity profile (Flash & Hogan, 1985; Morasso, 1981). The hand trajectory is obtained by minimizing the jerk, which leads to the following general solution for each coordinate:

$$x(t) = a_o + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5, \tag{3.4}$$

where the constants a_i are found by imposing boundary conditions on positions, velocities, and accelerations. We consider the boundary conditions as zero velocity and acceleration at the start and end of movement.

3.3 Training. To resemble human learning that occurs in orderly fashion, we included four distinct training sessions. During the first three training sessions, the simulated human arm was holding one of the three HMM MOSAIC objects (i.e., first object 1, then object 2, and finally object 3) and was commanded with a desired trajectory generated by the OU process. The resulting network was then trained during a reaching arm movement in the presence of the velocity-dependent force field. The LWPR network adds linear models and corresponding RFs as needed during execution of the movement.

3.4 Stability. One of the major drawbacks of LWPR is the sensitivity to the initial conditions (Vijayakumar et al., 2005), especially to the initialization of the distance metric matrix D_k (see equation 2.2). In our case, to ensure convergence of the learning algorithm, this initialization and the PD gains must be carefully selected. Also, the network inputs and outputs must be normalized according to their physical ranges of variation. Table 2 shows the selected ranges for our case.

At the beginning of each training session, the distance metric matrix D_k associated with each RF was initialized with the inverse of the variance of the input data (Schaal et al., 2002). The matrix D_k is chosen as diagonal and is of the following form:

$$\begin{aligned} D_k &= h \begin{pmatrix} d_1 & 0 & \cdots & 0 \\ 0 & d_2 & \cdots & 0 \\ \vdots & & \ddots & \\ 0 & & & d_8 \end{pmatrix} \\ d_i &= \frac{1}{\sigma_i^2}, \end{aligned} \tag{3.5}$$

where σ_i^2 is the variance of the ith input and h is a scalar coefficient equal to 0.5. Here the subscript i goes from 1 to 8 since the input vector has eight

Table 2: Ranges of Variation for Variable Normalization for the LWPR Network.

x	[−0.25,0.25] m
y	[0,0.6] m
$\dot{x}$	[−2,2] m/s
$\dot{y}$	[−2,2] m/s
$\ddot{x}$	[−3,3] m/s
$\ddot{y}$	[−3,3] m/s
F_x	[−10,10] N
F_y	[−10,10] N
τ_{sho}	[−10,10] Nm
τ_{elb}	[−10,10] Nm

Note: Each variable is normalized to ensure convergence of the learning algorithm.

entries (six components for the desired Cartesian trajectory, velocity, and acceleration and two components for the external force). This matrix defines the initial shape of the RF activation function. The shape of an RF defines the region of validity of the corresponding linear model, according to the value of the gaussian kernel (see equation 2.2). By initializing the matrix D_k with the inverse of the variance of the training input data, we are assuming that some prior information is known as to the distribution of these data. This is especially useful when dealing with data that have different distribution among the training sessions, as is the case of the interaction forces in the four tasks.

The PD gains of the feedback controller must also be initialized properly. In Nakanishi and Schaal (2004), sufficient conditions were provided for choosing the feedback gains in order to ensure stability of the feedback error learning scheme. The approach was based on the strict positive realness (SPR) of the error dynamics and was demonstrated for single input–single output systems. In general, small values allow large deviations from the reference trajectory, thus producing an inaccurate training signal, while high values lead to instability. The feedback control torque τ_{FB} is computed in the joint space as follows:

$$\begin{aligned} \tau_{FB} &= K_p\tilde{\theta} + K_d\dot{\tilde{\theta}} \\ \tilde{\theta} &= \theta_d - \theta, \end{aligned} \tag{3.6}$$

where K_p and K_d are the 2 × 2 matrices of the proportional and derivative gains and regulate the arm stiffness and damping during training and θ_d and θ is the 2 × 1 vector of the desired and actual angular position, respectively.

Table 3: nMSE for Each Task.

Task	nMSE PD Alone	nMSE PD + Inverse Model	nMSE PD + Inverse model After FF	ΔnMSE
OBJ 1	0.1402	0.0265	0.0263	−0.0002
OBJ 2	0.2247	0.0455	0.0481	0.0026
OBJ 3	0.3440	0.0490	0.0843	0.0353
Force Field	0.0011	7.16E-08		

Notes: ΔnMSE is the difference between the nMSE after and before learning the FF task. Negative values indicate improved performance.

For object 1, the following values (Berniker, 2000) are used:

$$K_p = \begin{pmatrix} 20 & 40 \\ 40 & 50 \end{pmatrix}$$
$$K_d = \begin{pmatrix} 2 & 4 \\ 4 & 5 \end{pmatrix}. \tag{3.7}$$

For objects 2 and 3, the gains are increased by a factor of 2 and 4, respectively. These values for the gains were derived heuristically. After learning, the gains were set again to their base values (i.e., the values adopted for object 1). This is not unlike human behavior during adaptation to novel dynamics. At the first stages, we co-contract to compensate for external disturbances, and later, as we learn to manipulate the task dynamics, we reduce the stiffness and generate a compensatory force to the external disturbance (Franklin et al., 2003; Milner & Franklin, 2005). For training on the velocity-dependent force field, we employed the same K_p and K_d used for object 1.

4 Results

Table 3 contains values of the normalized mean square error (nMSE) for each training session. The nMSE is defined as the mean square error divided by the variance of the target data values. The first two numerical columns show the nMSE when using the PD controller alone or combined with the trained inverse model, respectively. As expected, the nMSE is always lower when using the PD control combined with the inverse model. In the third column, the nMSE is reported for the tasks performed with the three objects after learning the force field (FF) task. A measure of the interference during learning is provided by ΔnMSE in the last column, which is the difference between the nMSE after and before learning with the FF task.

Figure 3 shows the training results of the inverse model for movements lasting 0.6 seconds, while the arm holds each of the three different objects.

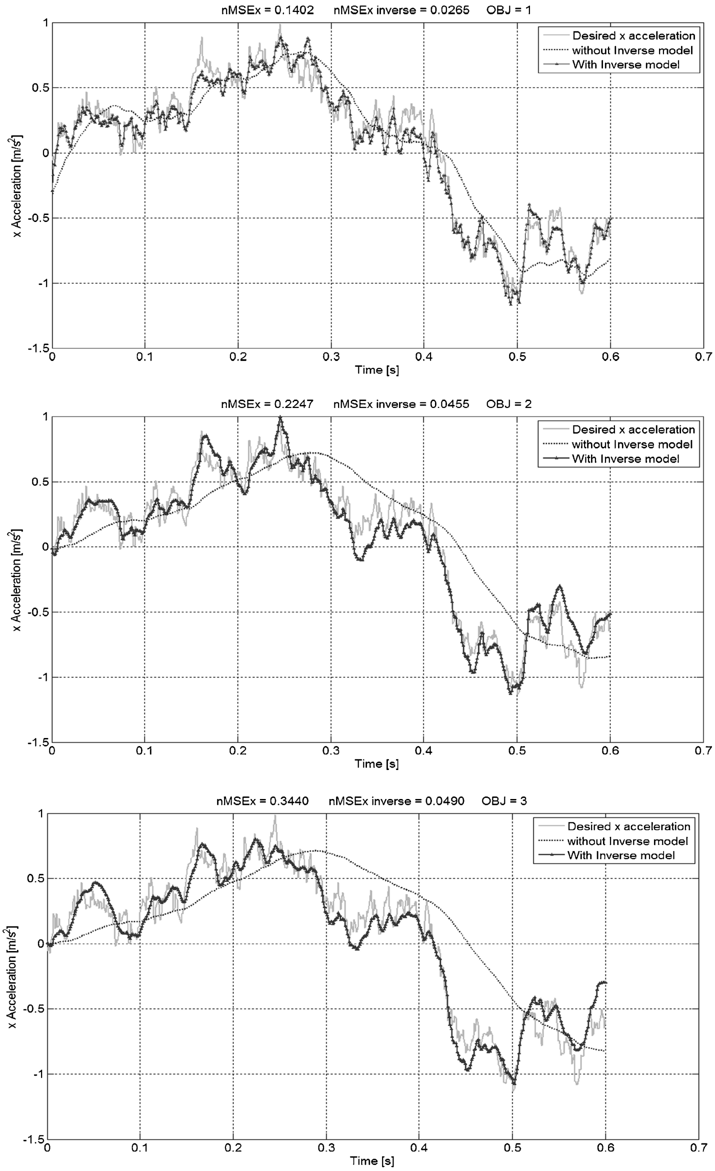

Figure 3: Training with three distinct objects. The figure shows the desired x-axis acceleration (solid gray line) and actual acceleration when using the PD controller (dotted line) and when using both the PD and the trained inverse model (dark line). Each plot from top to bottom corresponds to the arm holding one of the three distinct objects, and each plot lists the normalized mean square error (nMSE) for the PD alone case and the combined PD and inverse model.

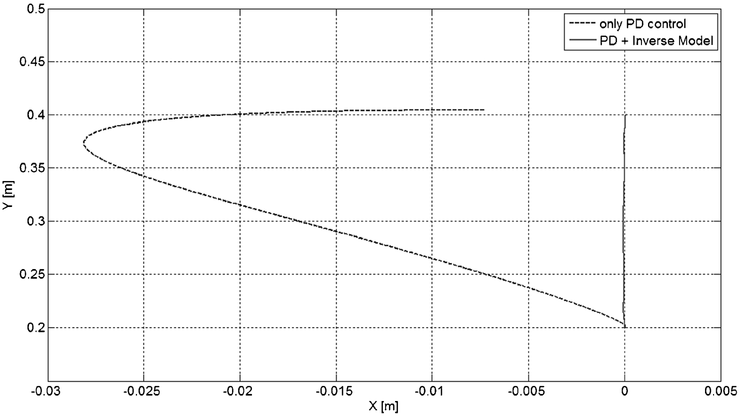

Figure 4: Hand trace during reaching and a force field perturbation. The figure shows the hand trajectory when using the PD controller alone (dotted line) and after training the inverse model (solid line). Note the hook in the untrained case (PD controller alone).

Each plot shows the desired hand acceleration and the actual accelerations when using the PD controller alone or combined with the inverse model. Despite the sharpness of the desired acceleration profile, the inverse model allows tracking of the acceleration reference, compensating for the interaction forces exerted by the objects. For example, in the case of object 3, the nMSE drops from 0.344 to only 0.049 when the PD controller is used in conjunction with the inverse model.

Figure 4 shows the results of the subsequent training of the same network on arm reaching movements exposed to the perturbation force field. The reaching movement lasts for 1.2 s and covers 20 cm of displacement. The figure shows a top view of the hand trace before and after training the inverse model. Note the clockwise "hook" similar to experimental data on unimpaired subjects during initial exposure to the perturbation force field (Shadmehr & Mussa-Ivaldi, 1994; Krebs, Brashers-Krug, et al., 1998). After training the inverse model, the hand path is close to a straight line, owing to the torque compensation due to the inverse model. The nMSE between desired and actual position decreases from a value of 0.0011 to a value of 7.16E-08, when using the inverse model, showing an almost perfect compensation of the perturbing forces.

Learning is performed online; namely, linear models are added during arm movements. The maximum number of epochs is set to 5, since no reduction of the nMSE is obtained with further learning. At the end of the training sessions, 80 linear models are allocated.

Figure 5 shows the retention results. After learning the objects task and subsequently the force field task, the network was tested again on the objects

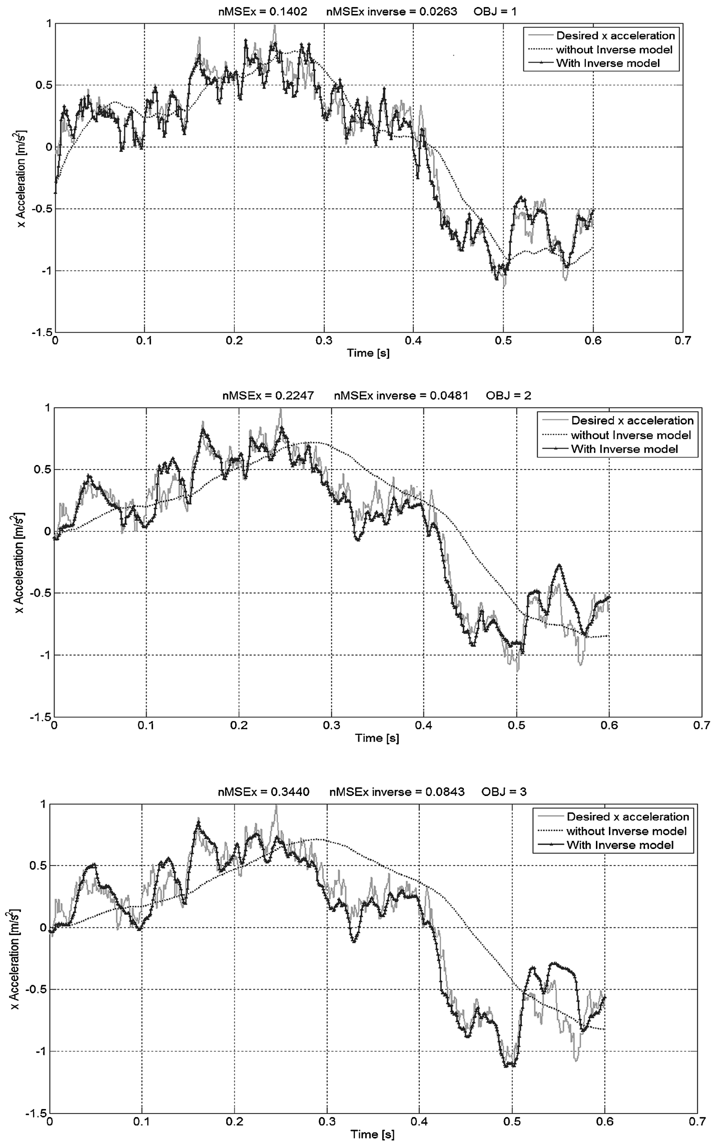

Figure 5: Retention of learning. The figure shows the retention and forgetfulness of the previously learned dynamics of the three objects. After training the inverse model on the perturbation force field, we tested the network for retention, and as can be seen, the LWPR inverse model is still able to compensate for the forces exerted by each object. Of note, the nMSE for the inverse model reveals poorer performance than in Figure 3, which is in line with another characteristic of human motor learning, acquisition, and retention.

task. Again each plot corresponds to the arm holding one of the three objects. The network successfully acquires the ability to compensate for the novel velocity-dependent force field, and of note, it retains previously acquired knowledge on the interaction with the three objects. As shown in Table 3, the nMSE is always notably below the values obtained with the PD controller alone. In the worst case corresponding to object 3, the nMSE increases from 0.0490 to 0.0843, which is far smaller than the 0.344 value obtained for the PD controller alone and might indicate forgetfulness. For the best case, object 1′s nMSE slightly decreased after learning the force field task, meaning that no negative interference occurred. Thus, the proposed architecture shows features of motor learning, including forgetfulness and retention.

5 Discussion

Acquisition and retention of multiple internal models is a general computational approach to cope with the problem of learning multiple tasks and quickly adapt to different contexts (Wolpert & Kawato, 1998; Haruno et al., 2001; Doya et al., 2002; Petkos & Vijayakumar, 2007). Most of the control architectures proposed in the literature share the approach of learning multiple models and using a probabilistic model to switch or mix the outputs in order to obtain motor commands suitable for each context. To our knowledge, they do not allow incremental acquisition and retention of multiple motor tasks.

MOSAIC is an important representative model of human motor learning and adaptation to different contexts. MOSAIC has a modular architecture made of a fixed number of modules that should be able to learn multiple different tasks. Through a soft-max function or on a later version, an HMM, MOSAIC is able to learn and switch between multiple tasks. Of note, the HMM MOSAIC was introduced later to improve switching between modules with respect to the soft-max version, especially when the manipulated object changes infrequently (every 200 ms or more). There are two major drawbacks in these schemes. First, when a fixed number of modules is used, the architecture cannot learn additional novel contexts. In other words, the MOSAIC model is not capable of subsequently adding new objects whose properties lie outside the tetrahedron defined by the a priori trained modules. Second, although the HMM MOSAIC model is able to learn and correctly select among the a priori defined objects, training of the HMM requires simultaneous training of all states. In order to learn transitions between the states (number of objects), objects must change during the movement. In other words, the model must learn all the contexts at the same time. While this handicap might not invalidate this model for some applications, retention is a key feature of motor learning. Humans do not retrain the entire repertoire of previously learned tasks when learning a new task. They are capable of learning new tasks presented an in orderly fashion.

We proposed learning of a single inverse model that is subject to different interaction forces using a nonparametric regression network, the LWPR. This network is based on local linear approximators and is capable of minimizing interference with previously learned tasks, provided that the new task occupies a different space (i.e., activates different RFs). As with other approaches in the literature (Haruno et al., 2001; Doya et al., 2002), our approach mixes the output of different linear models to approximate the target function (see equation 2.4). However to our knowledge, for the first time, we propose the use of an additional input of the interaction force to infer the context directly. We do not need to estimate the context through a probabilistic model. Other studies have shown that we learn both the inverse dynamic model of arm and the interaction forces to control the kinematics of manipulated objects (Dingwell, Mah, & Mussa-Ivaldi, 2002). In addition, we use the haptic information given by the interaction force during learning of a skilled task (Huang, Gillespie, & Kuo, 2006).

As shown in the simulations, the proposed architecture allows learning different tasks, such as manipulation of three distinct objects and compensation of perturbation forces, in an orderly fashion, and it is able to retain learning. Furthermore, in contrast to other methods (Haruno et al., 2001), the inverse model is not required at onset. The only information required in advance is an estimate of the variance of the input data for each task. Of note, an unexpected limitation of the proposed algorithm to model human motor learning is its fast convergence. Only a limited number of repetitions are usually sufficient for the LWPR network to learn each task (five epochs in our object learning example). For the perturbation force field, the hand trace was almost straight after the second trial as compared to the 160 movements experimentally observed in humans (Krebs, Brashers-Krug, et al., 1998).

6 Conclusion

We presented a novel model for motor control of the human arm movement based on a nonparametric regression network, the LWPR. This approach, though using a simple control scheme, showed its ability to acquire and retain learning of different tasks presented in an orderly fashion. By training an inverse model with additional input, the interaction force, the network is capable of discriminating among different contexts. Moreover, the LWPR network benefits from local approximation, which allows retention of previously learned tasks. At this stage, one can only speculate that such a computational scheme might be highly advantageous in many different situations. For example during locomotion, the interaction forces during the swing and stance phases are quite distinct, as are the body dynamics. Likewise when handling a tool, the initial reaching arm movement to the work space is unconstrained, while the actual tool manipulation is highly constrained and possibly requires tool stabilization. In conclusion, to the best of our knowledge, our model provides significant advantages over previous models of motor learning of multiple tasks. That is, our model not only addresses key deficiencies of past efforts including serial acquisition of learning and retention, which are key features of human motor learning; it may also herald new insights and a deeper understanding of the motor system, with important implications for basic research in neuroscience and translational applications to neurorehabilitation and advanced robotics.

Acknowledgments

This work was supported in part by the FP6-EU-IST-FET (NEUROBOTICS-IST-2003-001917) research program, by NICHD-NCMRR Grant 1 R01-HD045343, the NYSCORE, and the INTERLINK-MOTHER project funded by the Italian Ministry of University and Research. We acknowledge a very fruitful exchange with the reviewers and look forward to continuing this dialogue. H.I.K is a co-inventor in MIT-held patents for the robotic devices used to treat patients with neurological deficits. He holds equity positions in Interactive Motion Technologies, the company that manufactures this type of technology under license to MIT.

References

Atkeson, C. G., Moore, A. W., & Schaal, S. (1997). Locally weighted learning. *Artificial Intelligence Review*, *11*(1:5), 11–73.

Bacciu, D., Zollo, L., Guglielmelli, E., Leoni, F., & Starita, A. (2004). A RLWPR network for learning the internal model of an anthropomorphic robot arm. In *Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems* (Vol. 1, pp. 260–265). N.p.

Baillieul, J., & Martin, D. P. (1990). Resolution of kinematic redundancy. *Proceedings of Symposia in Applied Mathematics, American Mathematical Society*, *41*, 49–89.

Beal, M. J., Ghahramani, Z., & Rasmussen, C. E. (2002). The infinite hidden Markov model. In T. G. Dietterich, S. Becker, & Z. Ghahramani (Eds.), *Advances in neural information processing systems*, *14* (pp. 577–584). Cambridge, MA: MIT Press.

Berniker, M. (2000). *A biologically motivated paradigm for heuristic motor control in multiple contexts*. Unpublished master's thesis, Massachusetts Institute of Technology.

Craik, K. J. W. (1943). *The nature of explanation*. Cambridge: Cambridge University Press.

Davidson, P. R., & Wolpert, D. M. (2004). Scaling down motor memories: De-adaptation after motor learning. *Neuroscience Letters*, *370*(2–3), 102–107.

Dingwell, J. B., Mah, C. D., & Mussa-Ivaldi, F. A. (2002). Manipulating objects with internal degrees of freedom: Evidence for model-based control. *Journal of Neurophysiology*, *88*(1), 222–235.

Doya, K., Samejima, K., Katagiri, K., & Kawato, M. (2002). Multiple model-based reinforcement learning. *Neural Comput.*, *14*(6), 1347–1369.

Flanagan, J. R., Vetter, P., Johansson, R. S., & Wolpert, D. M. (2003). Prediction precedes control in motor learning. *Current Biology*, *13*(2), 146–150.

Flanagan, J. R., Wolpert, D. M., & Johansson, R. S. (2001). Sensorimotor prediction and memory in object manipulation. *Canadian Journal of Experimental Psychology*, *55*(2), 89–97.

Flash, T., & Hogan, N. (1985). The coordination of arm movements: An experimentally confirmed mathematical model. *Journal of Neuroscience*, *4*, 1688–1703.

Franklin, D. W., Osu, R., Burdet, E., Kawato, M., & Milner, T. E. (2003). Adaptation to stable and unstable dynamics achieved by combined impedance control and inverse dynamics model. *Journal of Neurophysiology*, *90*, 3270–3282.

Geladi, P., & Kowalski, B. (1986). Partial least squares regression: A tutorial. *Analytica Chimica Acta*, *185*, 1–17.

Gomi, H., & Kawato, M. (1993). Recognition of manipulated objects by motor learning with modular architecture networks. *Neural Networks*, *6*, 485–497.

Gomi, H., & Kawato, M. (1996) Equilibrium-point control hypothesis examined by measured arm stiffness during multijoint movement. *Science,* 272, 117–120.

Haruno, M., Wolpert, D. M., & Kawato, M. (2001). MOSAIC model for sensorimotor learning and control. *Neural Comput.*, *13*, 2201–2220.

Hogan, N., Krebs, H. I., Rohrer, B., Palazzolo, J. J., Dipietro, L., Fasoli, S. E., et al. (2006). Motions or muscles? Some behavioral factors underlying robotic assistance of motor recovery. *J. Rehabil. Res. Dev.*, *43*(5), 605–618.

Huang, F. C., Gillespie, R. B., & Kuo, A. D. (2006). Human adaptation to interaction forces in visuomotor coordination. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, *14*(3), 390–397.

Imamizu, H., Kuroda, T., Miyauch, S., Yoshioka, T., & Kawato, M. (2003). Modular organization of internal models of tools in the human cerebellum. *Proceedings of the National Academy of Sciences USA*, *100*, 5461–5466.

Ito, M. (1970). Neurophysiological aspects of the cerebellar motor control system. *International Journal of Neurology*, *7*(2), 162–176.

Kawato, M. (1990). Feedback-error-learning neural network for supervised motor learning. In R. Eckmiller (Ed.), *Advanced neural computers* (pp. 365–372). Amsterdam: North-Holland.

Kawato, M. (1999). Internal models for motor control and trajectory planning. *Current Opinion in Neurobiology*, *9*, 718–727.

Krebs, H. I., Brashers-Krug, T., Rauch, S. L., Savage, C. R., Hogan, N., Rubin, R. H., et al. (1998). Robot-aided functional imaging: Application to a motor learning study. *Human Brain Mapping*, *6*(1), 59–72.

Krebs, H. I., Hogan, N., Aisen, M. L., & Volpe, B. T. (1998). Robot-aided neurorehabilitation. *IEEE Transactions on Rehabilitation Engineering*, *6*(1), 75–87.

Krebs, H. I., Volpe, B. T., Williams, D., Celestino, J., Charles, S. K., Lynch, D., et al. (2007). Robot-aided neurorehabilitation: A robot for wrist rehabilitation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, *15*(3), 327–335.

Miall, R. C., Weir, D. J., Wolpert, D. M., & Stein, J. F. (1993). Is the cerebellum a Smith predictor? *Journal of Motor Behavior*, *25*(3), 203–216.

Miall, R. C., & Wolpert, D. M. (1996). Forward models for physiological motor control. *Neural Networks*, *9*(8), 1265–1279.

Milner, T. E., & Franklin, D. W. (2005). Impedance control and internal model use during the initial stage of adaptation to novel dynamics in humans. *Journal of Physiology*, *567*(2), 651–664.

Morasso, P. (1981). Spatial control of arm movements. *Experimental Brain Research*, *42*, 223–227.

Nakanishi, J., & Schaal, S. (2004). Feedback error learning and nonlinear adaptive control. *Neural Networks*, *17*(10), 1453–1465.

Narendra, K. S., & Balakrishnan, J. (1997). Adaptive control using multiple models. *IEEE Transactions on Automatic Control*, *42*(2), 171–187.

Osu, R., Hirai, S., Yoshioka, T., & Kawato, M. (2004). Random presentation enables subjects to adapt to two opposing forces on the hand. *Nature Neuroscience*, *7*(2), 111–112.

Petkos, G., & Vijayakumar, S. (2007). Context estimation and learning control through latent variable extraction: From discrete to continuous contexts. In *Proceedings of the 2007 International Conference on Robotics and Automation (ICRA07)* (pp. 2117–2123). N.p.

Rabiner, L. R. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, *77*(2), 257–286.

Schaal, S., & Atkeson, C. G. (1998). Constructive incremental learning from only local information. *Neural Comput.*, *10*(8), 2047–2084.

Schaal, S., Atkeson, C. G., & Vijayakumar, S. (2002). Scalable techniques from nonparametric statistics for real time robot learning. *Applied Intelligence*, *17*(1), 49–60.

Shadmehr, R., & Mussa-Ivaldi, F. A. (1994). Adaptive representation of dynamics during learning of a motor task. *Journal of Neuroscience*, *14*(5), 3208–3224.

Shibata, T., & Schaal, S. (2001). Biomimetic gaze stabilization based on feedback-error-learning with nonparametric regression networks. *Neural Networks*, *14*(2), 201–216.

Uhlenbeck, G. E., & Ornstein, L. S. (1930). On the theory of Brownian motion. *Phys. Rev.*, *36*, 823–841.

Vijayakumar, S., D'Souza, A., & Schaal, S. (2005). Incremental online learning in high dimensions. *Neural Comput.*, *17*, 2602–2634.

Wolpert, D. M., & Ghahramani, Z. (2000). Computational principles of movement neuroscience. *Nature Neuroscience*, *3*, 1212–1217.

Wolpert, D. M., & Kawato, M. (1998). Multiple paired forward and inverse model for motor control. *Neural Networks*, *11*, 1317–1329.

Zollo, L., Eskiizmirliler, S., Teti, G., Laschi, C., Burnod, Y., Guglielmelli, E., et al. (2008). An anthropomorphic robotic platform for progressive and adaptive sensorimotor learning. *International Journal of Advanced Robotics*, *22*, 91–118.

Received March 3, 2008; accepted January 6, 2009.