Neural Networks 22 (2009) 348–361

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Learning and generation of goal-directed arm reaching from scratch

Hiroyuki Kambara [a,b,*], Kyoungsik Kim [b,c], Duk Shin [d], Makoto Sato [a], Yasuharu Koike [a,b]

[a] Tokyo Institute of Technology, Precision and Intelligence Laboratory, Yokohama, 226-8503, Japan
[b] Japan Science and Technology Agency CREST, Saitama, 332-0012, Japan
[c] Tokyo Institute of Technology, Department of Computational Intelligence and Systems Science, Yokohama, 226-8502, Japan
[d] Toyota Central R&D Labs., Inc., Aichi, 480-1192, Japan

ARTICLE INFO

Article history:
Received 20 September 2007
Received in revised form 14 July 2008
Accepted 18 November 2008

Keywords:
Reaching
Reinforcement learning
Feedback-error-learning
Internal model
Trajectory planning

ABSTRACT

In this paper, we propose a computational model for arm reaching control and learning. Our model describes not only the mechanism of motor control but also that of learning. Although several motor control models have been proposed to explain the control mechanism underlying well-trained arm reaching movements, it has not been fully considered how the central nervous system (CNS) learns to control our body. One of the great abilities of the CNS is that it can learn by itself how to control our body to execute required tasks. Our model is designed to improve the performance of control in a trial-and-error manner which is commonly seen in human's motor skill learning. In this paper, we focus on a reaching task in the sagittal plane and show that our model can learn and generate accurate reaching toward various target points without prior knowledge of arm dynamics. Furthermore, by comparing the movement trajectories with those made by human subjects, we show that our model can reproduce human-like reaching motions without specifying desired trajectories.

1. Introduction

When we move our hand from one point to another, the hand paths tend to gently curve and the hand speed profiles are bell-shaped (Abend, Bizzi, & Morasso, 1982; Atkeson & Hollerback, 1985; Uno, Kawato, & Suzuki, 1989). Since humans show these highly stereotyped trajectories among an infinite number of possible ones, it has been suggested that the central nervous system (CNS) is optimizing arm movements so as to minimize some kind of cost function (Flash & Hogan, 1985; Harris & Wolpert, 1998; Uno et al., 1989). Cost functions specify movement-related variables that should be minimized during or after the movement. Meanwhile, several computational control models have been proposed to explain the way the CNS generates a set of motor commands that could minimize cost functions (Flash, 1987; Gribble, Ostry, Sanguineti, & Laboissiere, 1998; Hogan, 1984; Miyamoto, Nakano, Wolpert, & Kawato, 2004; Todorov & Jordan, 2002; Wada & Kawato, 1993). The hand trajectories predicted by these models are in strong agreement with experimental data. The purpose of these models, however, is to predict well-learned reaching movements themselves and not to describe the process of learning. In order to reproduce the movements, the control models were designed using detailed knowledge about the dynamics of musculoskeletal systems.

The purpose of this paper is to propose a motor control model that can learn the control law for reaching movements while actually controlling the arm. Let us call this type of model a "motor control-learning model". From observing infants' inaccurate and jerky motions (Konczak & Dichgans, 1997; Zaal, Daigle, Gottlieb, & Thelen, 1999), the motor skill to generate accurate and smooth adult-like movements seem to be acquired through motor learning performed in our daily life. However, this kind of learning is not as simple as general supervised learning problems. Since there is no explicit "teacher" that can provide the CNS with correct motor commands, the CNS has to learn how to control the body in a trial-and-error manner, through interaction with the environment.

Reinforcement learning has attracted much attention as a self-learning paradigm for acquiring optimal control strategy through trial-and-error (Sutton & Barto, 1998). In particular, the actor-critic method, one of the major frameworks for the temporal difference learning, has been proposed as a model of learning in the basal ganglia (Barto, 1995; Doya, 1999). We adopt the actor-critic method (Doya, 2000) in order to acquire a feedback controller for multi-joint reaching movements. Although we are not the first to apply the actor-critic method to a reaching task, the previous model only explained a reaching movement toward one particular target (Izawa, Kondo, & Ito, 2004). In our daily life, we are not

* Corresponding author at: Tokyo Institute of Technology, Precision and Intelligence Laboratory, Yokohama, 226-8503, Japan. Tel.: +81 45 924 5054; fax: +81 45 924 5016.
E-mail address: hkambara@hi.pi.titech.ac.jp (H. Kambara).

0893-6080/$ – see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.neunet.2008.11.004

always reaching to the same target. The CNS should be learning how to generate reaching movements toward various targets in the workspace. However, it is difficult to realize various movements with high accuracy using a single feedback controller. Since the gravitational force acting on the arm depends on the posture of the arm, the force required to hold the hand at the target varies with the target position. Furthermore, the magnitude of muscle tension varies with the posture of the arm even if a level command signal is sent to the muscle. For these reasons, there is no guarantee that a single feedback controller trained for a particular target would generate accurate reaching movements to other targets.

Here we introduce an additional controller called an inverse statics model, which supports the feedback controller in generating reaching movements toward various targets. It handles the static component of the inverse dynamics of the arm. That is, it transforms a desired position (or posture) into a set of motor commands that leads the hand to the desired position and holds it there. Note that the arm converges to a certain equilibrium posture when a constant set of motor commands is sent to the muscles because of the spring-like properties of the musculoskeletal system (Feldman, 1966). If the inverse statics model is trained properly, it can compensate for the static forces (e.g. gravity) at the target point. Therefore, accurate reaching movements toward various target points are realized by combining the inverse statics model and the feedback controller that works moderately well within the workspace. To acquire an accurate inverse statics model in a trial-and-error manner, we adopt the feedback-error-learning scheme (Kawato, Furukawa, & Suzuki, 1987). In this scheme, inverse dynamics (or statics) models of controlled objects are trained by using command outputs of the feedback controller as error signals. This learning scheme was originally proposed as a computational coherent model of cerebellar motor learning (Kawato et al., 1987). The original model, however, did not explain how to acquire the feedback controller for arm movements. In our model, the actor-critic method is introduced to train the feedback controller. Therefore, our model gives a possible solution to the problem of feedback controller design in the feedback-error-learning scheme.

In addition to the feedback controller and the inverse statics model, we introduced a forward dynamics model of the arm into our motor control-learning model. The forward dynamics model is an internal model that predicts a future state of the arm given outgoing motor commands. It has been proposed that the CNS is utilizing the forward dynamics model to internally predict the state of the arm during the control process (Miall & Wolpert, 1996; Wolpert, Miall, & Kawato, 1998). The existence of the forward dynamics model in the CNS is also supported by psychophysical experiments (Bard, Turrell, Fleury, Teasdale, Lamarre, & Martin, 1999; Wolpert, Ghahramani, & Jordan, 1995). The forward dynamics model can be trained in a supervised learning manner since the teaching signal can be obtained from somatosensory feedback. In the literature of automatic control, the strategy to combine system identification with reinforcement learning succeeded in autonomously controlling machines with complex dynamics such as helicopters (Abbeel, Coates, Quigley, & Ng, 2007). In our model, the forward dynamics model is designed to predict the state of the arm at a future time instant so as to compensate for delay of motor output caused by graded development of the muscle force. The predicted future states are then utilized to determine command outputs of the feedback controller.

In the present study, we apply our motor control-learning model to a point-to-point reaching task in the sagittal plane. By simulating the learning process of the reaching task, we show that our model can accurately control the arm to reach toward various target points without assuming prior knowledge of the arm dynamics. In addition, we compare reaching movements simulated by our model with those of human subjects, and show that our model can reproduce features of both hand path and speed profile in human reaching movements without planning desired trajectories.

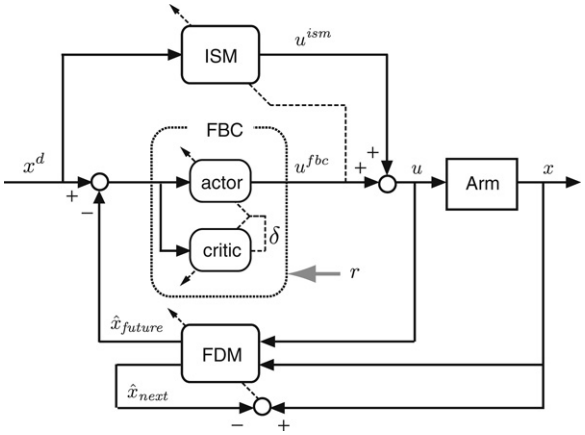


Fig. 1. The architecture of motor control-learning model: the model has three main modules, Inverse Statics Model (ISM), Feedback Controller (FBC), and Forward Dynamics Model (FDM). The FBC is composed of actor and critic units, which correspond to a controller and value function estimator respectively in the actor-critic method. The ISM generates a feed-forward motor command $\boldsymbol{u}^{ism}$ that shifts the equilibrium state of the arm to the desired state $\boldsymbol{x}^d$. On the other hand, the FBC generates a feedback motor command $\boldsymbol{u}^{fbc}$ that reduces the error between the desired state $\boldsymbol{x}^d$ and the future state $\hat{\boldsymbol{x}}_{future}$ predicted by the FDM. The error signal for the ISM is the feedback motor command $\boldsymbol{u}^{fbc}$. Meanwhile, the teaching signal for the FDM is the state of the arm $\boldsymbol{x}$ observed at next time instant. The FBC is trained by the actor-critic method so as to maximize the cumulative reward r. The temporal difference error δ related to the reward r is used as the reinforcer and error signal for the actor and critic units, respectively.

2. Motor control-learning model

Fig. 1 illustrates architecture of the motor control-learning model for a reaching task. The model consists of three main modules, inverse statics model (ISM), feedback controller (FBC), and forward dynamics model (FDM). The FBC is composed of actor and critic units, which correspond to a controller and value function estimator respectively in the actor-critic method.

At the beginning of each trial, a target point of reaching is given as a desired state $\boldsymbol{x}^d$ to the model. This $\boldsymbol{x}^d$ is kept constant at the target point throughout the trial. The ISM receives $\boldsymbol{x}^d$ as an input and generates a time-invariant motor command $\boldsymbol{u}^{ism}$. If the ISM were trained correctly, $\boldsymbol{u}^{ism}$ shifts the equilibrium of the arm to the target point. On the other hand, at time t, the FBC receives a state error between desired state $\boldsymbol{x}^d$ and future state $\hat{\boldsymbol{x}}_{future}(t-\Delta t)$ predicted by the FDM Δt second before time t. The FBC, then, transforms the state error into a feedback motor command $\boldsymbol{u}^{fbc}(t)$. The sum of $\boldsymbol{u}^{ism}$ and $\boldsymbol{u}^{fbc}(t)$ is sent to the arm as a total motor command $\boldsymbol{u}(t)$. Based on the total motor command $\boldsymbol{u}(t)$ and the state $\boldsymbol{x}(t)$, the FDM predicts next state $\hat{\boldsymbol{x}}_{next}(t)$ and also future state $\hat{\boldsymbol{x}}_{future}(t)$.

The three modules improve their performance in the following way. A teaching signal for FDM's prediction $\hat{\boldsymbol{x}}_{next}(t)$ is given by observing the actual state at time $t+\Delta t$. Therefore, the FDM can be trained in normal supervised learning manner, in which the error signal is determined as

$$E_{fdm}(t) = \boldsymbol{x}(t+\Delta t) - \hat{\boldsymbol{x}}_{next}(t). \tag{1}$$

On the other hand, the ISM is trained with the feedback-error-learning scheme in which the error signal for ISM's output $\boldsymbol{u}^{ism}$ is FDM's output, that is,

$$E_{ism}(t) = \boldsymbol{u}^{fbc}(t). \tag{2}$$

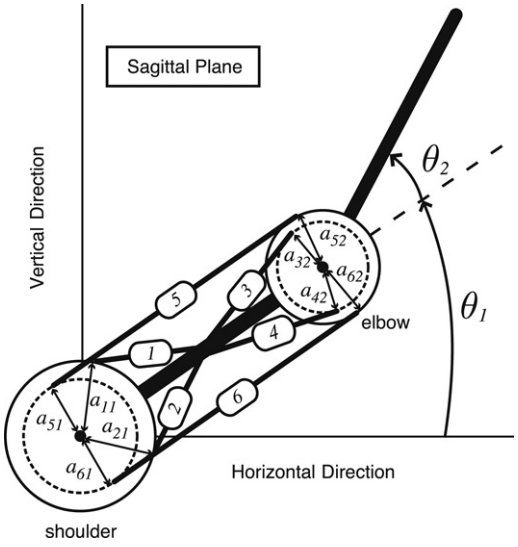


Fig. 2. Two link arm model with two joints and six muscles: θ_1 and θ_2 are the angles of shoulder and elbow joints, respectively. Six muscles numbered from 1 to 6 are shoulder flexor, shoulder extensor, elbow flexor, elbow extensor, double-joints flexor and double-joints extensor, respectively. $a_{i,j}$ is the moment arm of the ith muscle torque against the jth joint.

Finally, the FBC is trained with the actor-critic method. The signal used to improve both actor and critic units is TD (temporal difference) error δ. The value of δ is determined from a reward signal r and state-value of the state error $(\boldsymbol{x}^d - \hat{\boldsymbol{x}}_{future})$ estimated by the critic unit. The TD error δ serves as an error signal for the critic unit. Meanwhile, it serves as a reinforcement signal for the actor unit (the details are described in Section 3).

3. Mathematical description of arm and motor control-learning model

We present here detailed mathematical descriptions of multi-joint arm model and the motor control-learning model shown in the previous section.

3.1. Arm model

We adopted an arm model composed of two-link arm with two joints and six muscles as shown in Fig. 2. Two joints correspond to shoulder and elbow joints. Six muscles are two pairs of mono-articular muscles acting on shoulder and elbow exclusively and one pair of bi-articular muscles acting on both joints.

An input signal of the arm model is a motor command vector $\boldsymbol{u} = (u_1, \ldots, u_6)^{\mathrm{T}}$. Each $u_i \in [0, 1]$ determines the activation level of the ith muscle. We introduced biological noise in motor command that increases with the mean level of motor command. Such signal-dependent noise is often modeled as a white noise with zero mean and variance of ku^2 (Harris & Wolpert, 1998). The motor command including the noise is calculated as

$$u_i^{noise}(t) = u_i(t) + w_i(t) \quad (i = 1, 2, \ldots, 6) \tag{3}$$

where $w_i(t) \sim N(0, k^{sdn} u_i^2(t))$ with a coefficient $k^{sdn} = 0.01$.

In neurophysiological studies, it is known that muscle force changes smoothly in time and it can be well predicted by low-pass filtering the neural impulse with a second-order filter (Mannard & Stein, 1973). To implement such a property in the arm model, we applied a second-order low-pass filter to $\boldsymbol{u}^{noise}$ as

$$\tilde{u}_i(t) = \int_0^t u_i^{noise}(s)\, h(t-s)\, \mathrm{d}s \tag{4}$$

where function h is an impulse response of the filter given by

$$h(z) = \frac{1}{\kappa_1 - \kappa_2}\left(\exp\left(-\frac{z}{\kappa_1}\right) - \exp\left(-\frac{z}{\kappa_2}\right)\right). \tag{5}$$

Here κ_1 and κ_2 are time constants and we set them as 92.6 ms and 60.5 ms, respectively. The values of two time constants were used for reconstructing muscle tension from EMG (electromyograph) signal in human (Koike & Kawato, 1995).

The muscle force $\boldsymbol{T}$ is then determined from the filtered motor command $\tilde{\boldsymbol{u}}$, muscle length $\boldsymbol{L}$, and contraction velocity $\dot{\boldsymbol{L}}$ as

$$T_i = K_i(\tilde{u}_i)\{L_i - L_i^{rest}(\tilde{u}_i)\} + B_i(\tilde{u}_i)\dot{L}_i \tag{6}$$

where $\boldsymbol{K}(\tilde{\boldsymbol{u}})$, $\boldsymbol{B}(\tilde{\boldsymbol{u}})$, and $\boldsymbol{L}^{rest}(\tilde{\boldsymbol{u}})$ denote muscle stiffness, viscosity, and rest length, respectively. This model is called the Kelvin–Voight model (Ozkaya & Nordin, 1999). We assume that $\boldsymbol{K}(\tilde{\boldsymbol{u}})$, $\boldsymbol{B}(\tilde{\boldsymbol{u}})$, and $\boldsymbol{L}^{rest}(\tilde{\boldsymbol{u}})$ are linear functions of the filtered motor command $\tilde{\boldsymbol{u}}$,

$$\begin{aligned} K_i(\tilde{u}_i) &= k_{0i} + k_{1i}\tilde{u}_i \\ B_i(\tilde{u}_i) &= b_{0i} + b_{1i}\tilde{u}_i \\ L_i^{rest}(\tilde{u}_i) &= l_{0i}^{rest} + l_{1i}^{rest}\tilde{u}_i \end{aligned} \tag{7}$$

where k_{0i}, b_{0i}, and l_{0i}^{rest} are intrinsic elasticity, viscosity, and rest length of the ith muscle, respectively. Also, k_{1i}, b_{1i}, and l_{1i}^{rest} represent the variation rate of elasticity, viscosity, and rest length, respectively. We also assume that the values of moment arms are constant. As a consequence, the muscle length $\boldsymbol{L}$ is described as a linear function of the joint angles $\boldsymbol{\theta}$, given by

$$L_i = l_{0i} - \sum_{j=1}^{2} A_{i,j}\theta_j \tag{8}$$

where l_{0i} is the length of the ith muscle when the joint angle $\boldsymbol{\theta} = \boldsymbol{0}$, and $A_{i,j}$ is the (i, j) element of the moment arm matrix A given by

$$A = \begin{pmatrix} a_{11} & -a_{21} & 0 & 0 & a_{51} & -a_{61} \\ 0 & 0 & a_{32} & -a_{42} & a_{52} & -a_{62} \end{pmatrix}^{\mathrm{T}}. \tag{9}$$

The joint torque $\boldsymbol{\tau}$ is then determined as

$$\boldsymbol{\tau} = A^{\mathrm{T}}\boldsymbol{T}. \tag{10}$$

The dynamic equations of 2-link arm moving within the sagittal plane are shown in Appendix A. The values of all parameters used in the arm model are presented in Appendices A and B.

3.2. State and motor command

The state variable $\boldsymbol{x}$ shown in Fig. 1 is composed of joint angles and angular velocities, and represented as $\boldsymbol{x} = (\theta_1, \theta_2, \dot{\theta}_1, \dot{\theta}_2)^{\mathrm{T}}$, where θ_1 and θ_2 are shoulder and elbow joint angles in Fig. 2, respectively. The desired state $\boldsymbol{x}^d$ has the same dimension as the state $\boldsymbol{x}$, but its angular velocity components are always set 0, that is, $\boldsymbol{x}^d = (\theta_1^{trg}, \theta_2^{trg}, 0, 0)^{\mathrm{T}}$, where θ_1^{trg} and θ_2^{trg} are the angles of shoulder and elbow joint at the target position, respectively. Since the arm has two degrees of freedom, target joint angles can be uniquely determined from target positions of the hand in the sagittal plane.

Each element of the feed-forward motor command $\boldsymbol{u}^{ism}$ shown in Fig. 1 is set between 0 and 1. On the other hand, each element of the feedback motor command $\boldsymbol{u}^{fbc}$ is set between -0.5 and 0.5. The total motor command $\boldsymbol{u}$ sent to the arm is then given by

$$u_i = \begin{cases} 0 & \text{for } u_i^{ism} + u_i^{fbc} < 0 \\ 1 & \text{for } u_i^{ism} + u_i^{fbc} > 1 \\ u_i^{ism} + u_i^{fbc} & \text{otherwise.} \end{cases} \tag{11}$$

We show in Section 3.3 the equations determining the values of $\boldsymbol{u}^{ism}$ and $\boldsymbol{u}^{fbc}$.

3.3. Networks

To implement the three modules FBC, ISM, and FDM in our motor control-learning model, we adopt artificial neural networks. The FBC is composed of actor and critic units, and each unit is implemented by a NRBF (normalized gaussian radial basis function) network (Bugmann, 1998). The input signal to both networks is a deviation between the desired state $\boldsymbol{x}^d$ and the predicted future state $\hat{\boldsymbol{x}}^{future}$. Let us denote this deviation as $\boldsymbol{q} \equiv \boldsymbol{x}^d - \hat{\boldsymbol{x}}^{future}$. In the actor-critic method, the actor unit learns a control law $\boldsymbol{\mu}(\boldsymbol{q})$, and the critic unit learns a value function $V(\boldsymbol{q})$. The output of the critic unit is given by

$$V(\boldsymbol{q}) = \sum_{k=1}^{N} w_k^C b_k(\boldsymbol{q}) \tag{12}$$

where $b_k(\boldsymbol{q})$ is the kth basis function, N denotes the total number of basis functions, and w_k^C is the weight parameter. The value of the kth basis function is given by

$$b_k(\boldsymbol{q}) = \frac{\exp\left(-\|\boldsymbol{S}_k(\boldsymbol{q}-\boldsymbol{c}_k)\|^2\right)}{\sum_{n=1}^{N}\exp\left(-\|\boldsymbol{S}_n(\boldsymbol{q}-\boldsymbol{c}_n)\|^2\right)} \tag{13}$$

where the vector $\boldsymbol{c}_k$ defines the center of the kth basis function, and the matrix $\boldsymbol{S}_k$ determines the shape of the kth basis function. The feedback motor command $\boldsymbol{u}^{fbc}$ is determined by the output of the actor unit as

$$u_i^{fbc}(\boldsymbol{q}) = g\left(\mu_i(\boldsymbol{q}) + \sigma n_i\right) - 0.5$$
$$\mu_i(\boldsymbol{q}) = \sum_{k=1}^{N} w_{i,k}^A b_k(\boldsymbol{q}). \tag{14}$$

Here, $\mu_i(\boldsymbol{q})$ is the ith output of the actor unit's network and $w_{i,k}^A$ is the weight parameter. The function $g(y)$ in Eq. (14) is the sigmoid function of y, and used to limit the value of the feedback motor command within the range from -0.5 to 0.5. Also, n_i is the white noise for motor command exploration, and σ determines the magnitude of the noise given by

$$\sigma = \sigma_0 \exp(-V(\boldsymbol{q})) \tag{15}$$

where σ_0 is a constant parameter.

In order to update the values of the weight parameters w_k^C and $w_{i,k}^A$, we used the continuous-time version of TD error (Doya, 2000) given by

$$\delta(t) = r(t) - V(\boldsymbol{q}(t)) + \gamma \dot{V}(\boldsymbol{q}(t)) \tag{16}$$

where $r(t)$ is the reward signal explained in Section 3.4, and γ is the time constant for discounting future rewards. At each time t, the weight parameters in the network of the critic unit are updated by using the rule

$$\dot{w}_k^C = \eta^C \delta(t) e_k^C(t) \tag{17}$$

where η^C is the learning rate and $e_k^C(t)$ is the eligibility trace given by

$$e_k^C(t) = \int_0^t \exp\left(-\frac{t-s}{\lambda}\right) b_k(\boldsymbol{q}(s))\, \mathrm{d}s \tag{18}$$

where λ is the decay factor for the eligibility trace. The weight parameters in the network of the actor unit are updated by using the rule

$$\dot{w}_{i,k}^A = \eta^A \delta(t) e_{i,k}^A(t) \tag{19}$$

where η^A is the learning rate and $e_{i,k}^A(t)$ is the eligibility trace given by

$$e_{i,k}^A(t) = \int_0^t h(t-s) d_{i,k}^A(s)\, \mathrm{d}s$$
$$d_{i,k}^A(s) = \sigma(s) n_i(s) b_k(\boldsymbol{q}(s)). \tag{20}$$

Here the function $h(z)$ is the same function as the impulse response of the second order low-pass filter used for filtering motor command (Eq. (5)).

The ISM module is also implemented by a NRBF network. The input to the ISM is the desired state $\boldsymbol{x}^d$, and the ith element of ISM's output is given by

$$u_i^{ism} = g\left(\sum_{k=1}^{M} v_{i,k} c_k(\boldsymbol{x}^d)\right) \tag{21}$$

where $c_k(\boldsymbol{x}^d)$, M and $v_{i,k}$ are the kth basis function, the total number of basis functions and the weight parameter, respectively. We adopt the feedback-error-learning scheme (Kawato et al., 1987) to train the network of the ISM. In this learning scheme, feedback controller's output is used as an error signal for feed-forward controller's output. The updating rule for the weight parameters in the network of the ISM is given by

$$\dot{v}_{i,k} = \eta^I \tilde{u}_i^{fbc}(t) c_k(\boldsymbol{x}^d) \tag{22}$$

$$\tilde{u}_i^{fbc}(t) = \int_0^t h(t-s) u_i^{fbc}(s)\, \mathrm{d}s \tag{23}$$

where η^I is the learning rate and $\tilde{u}_i^{fbc}(t)$ is the filtered feedback motor command. The function $h(z)$ is the impulse response of the low-pass filter defined by Eq. (5). Note that the error signal $\tilde{u}_i^{fbc}(t)$ includes dynamical terms of the inverse dynamics of the arm, since $\tilde{u}_i^{fbc}(t)$ is determined not only from the positional deviation but also from its time derivative. Therefore, when the arm is moving, it does not provide accurate error information to the ISM. To train the ISM properly, there must be a time period that the arm is held around the target posture. Therefore, we set the total time of each training trial long enough to include the holding period after the arm reached around the target.

Finally, the FDM module is implemented by a three-layer feed-forward neural network. The inputs to the network at time t are the current state $\boldsymbol{x}(t)$ and the filtered motor command $\tilde{\boldsymbol{u}}(t)$. The network's output $\Delta\hat{\boldsymbol{x}}(t)$ is a prediction of an amount of the state change made within the period from time t to $t + \Delta t$. Here Δt is a time step of the simulation, and we set it as $\Delta t = 0.01$ s. The predicted next state $\hat{\boldsymbol{x}}_{next}(t)$ is then given by

$$\hat{\boldsymbol{x}}_{next}(t) = \boldsymbol{x}(t) + \Delta\hat{\boldsymbol{x}}(t). \tag{24}$$

The future state $\hat{\boldsymbol{x}}_{future}(t)$, a predicted state of the arm at $\alpha\Delta t$ ahead, is determined by iterating above procedure α times. During the iteration, the filtered motor command inputted into the FDM is kept constant at $\tilde{\boldsymbol{u}}(t)$. We set $\alpha = 12$ and thus the FDM predicts the state of the arm at 0.12 s ahead as the future state. The reason for setting $\alpha = 12$ comes from the fact that the delay time in step response of the low-pass filter used to smooth motor command (Eq. (4)) is about 0.127 s. A teaching signal for the predicted next state $\hat{\boldsymbol{x}}_{next}(t)$ is the actual state $\boldsymbol{x}(t + \Delta t)$ observed at time $t + \Delta t$ and the weight parameters in the network is updated by the backpropagation method.

3.4. Reward

The reward signal r in Fig. 1 is determined by two components, r_d and r_u. The first component r_d is a reward for reducing the distance between the target point and the hand position of future state of the arm predicted by the FDM. Let us name this distance "future hand distance" and denote it as $\hat{d}$. Then r_d at time t is represented as

$$r_d(t) = k_d \left(\exp\left(-\frac{\hat{d}(t)^2}{\sigma_d^2} \right) - 0.5 \right) \tag{25}$$

where k_d and σ_d are positive constant parameters. We set them as $k_d = 1$ and $\sigma_d = 0.06$ m. The magnitude of r_d increases with decrease of the future hand distance $\hat{d}$. The second component r_u is the penalty for energy consumption, and its value is determined from the filtered motor command $\tilde{\boldsymbol{u}}$ as

$$r_u(t) = k_u \sum_{i=1}^{6} \tilde{u}_i(t)^2 \tag{26}$$

where k_u is a positive constant parameter, and we set it as $k_u = 0.1$.

The total reward r is then given by

$$r(t) = r_d(t) - r_u(t). \tag{27}$$

4. Simulation of motor learning in reaching task

In order to assess the reliability of our motor control-learning model, we simulated the learning process of a point-to-point reaching task in the sagittal plane. Based on results of the simulation, we demonstrate how the motor control system, generating reasonable reaching movements, is formed by our model.

4.1. Conditions of simulation

The whole learning process consists of 100,000 trials. At the beginning of each trial, the target angles of shoulder and elbow joints are determined randomly within the range $-100 \leq \theta_1^{trg} \leq -20$ deg and $30 \leq \theta_2^{trg} \leq 110$ deg, respectively. The initial joint angles $\theta_1(0)$ and $\theta_2(0)$ are also determined randomly within the same range as that of target angles. Meanwhile, the initial angular velocities $\dot{\theta}_1(0)$ and $\dot{\theta}_2(0)$ are both set 0 deg/s. The total duration of each trial is set 2 s and each trial terminates when 2 s has passed or the arm state gets out the pre-determined state space. The state of the arm is updated at every Δt (=0.01 s) with fourth-order Runge–Kutta method. The motor command signal is also updated at every time step Δt. The weight parameters in the networks and the eligibility traces are also updated at every Δt by applying the update rules described in Section 3.3 with the Euler integration.

The NRBF networks of both actor and critic units have $11 \times 11 \times 9 \times 9$ gaussian basis functions located on a grid with an even interval in each dimension of the input space ($-90 \leq q_1, q_2 \leq 90$ deg, $-720 \leq q_3, q_4 \leq 720$ deg/s). On the other hand, $17 \times 17 \times 1 \times 1$ gaussian basis functions in the NRBF network of the ISM were located on a grid with an even interval in each dimension of the input space ($-100 \leq x_1^d \leq -20$ deg, $30 \leq x_2^d \leq 110$ deg, $x_3^d = x_4^d = 0$ deg/s). We set 20 nodes in the hidden layer of the neural network for the FDM. Before starting the learning process, values of the weight parameters in the networks of the actor and critic units are randomly determined so as to distribute uniformly within 0 and 1. Those of the FDM network are also randomly determined within the range between 0 and 0.1. On the other hand, values of the weight parameters in the ISM network are all set at 0. The learning rates for the actor unit, critic unit, ISM, and FDM are set at 5, 5, 0.5, and 0.1, respectively. We set other constant parameters mentioned in Section 3.3 as $\sigma_0 = 1$, $\gamma = 1$, and $\lambda = 0.3$. Note that the weight parameters in the networks are initialized only at the beginning of the first trial and updated through whole learning process. This means that although the training situation differs from trial to trial, the same motor control system is continuously trained. The skill acquisition process usually experienced by infant or adult is this kind of learning process.

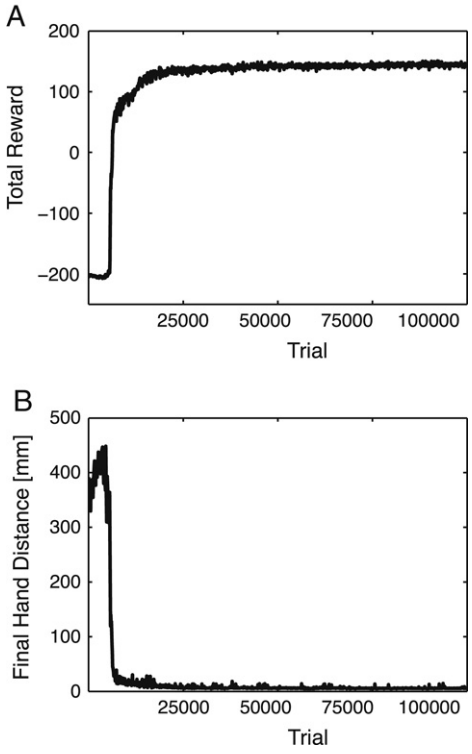


Fig. 3. Learning performance: (A) Total reward signal gained in each trial. (B) Distance between the target and the hand position at the last moment of each trial. We added $-(2 - t_l)/\Delta t$ to the total reward for the trial that did not last for 2 s (t_l and Δt are the time that the state of the arm get out the state space and the step time in the simulation, respectively). The data plotted in the figure are the average over successive 100 trials.

4.2. Results

4.2.1. Learning performance

The total reward signals gained in each trial and the hand distance at the last moment of each trial are plotted against trial number in Fig. 3. Here the hand distance denotes a distance between the target and the hand position, and let us call the hand distance at the end of each trial as *final hand distance*. As the number of trials increases, the total reward approaches a certain high value and the *final hand distance* comes close to zero. From this result, we can say that the motor control-learning model succeeded in improving the performance of reaching through training trials.

4.2.2. Improvements in ISM, FBC, and FDM

Let us see how the abilities of the three modules, ISM, FBC, and FDM, improved during the learning process. Fig. 4 shows the changes in command outputs of the ISM and the FBC, and state prediction error of the FDM. We plot, from the top of the figure, (A) ISM's output $\boldsymbol{u}^{ism}$ at the beginning of each trial, (B) FDM's output $\boldsymbol{u}^{fbc}(\boldsymbol{q})$ when the predicted state corresponds to the target state, and (C) the average errors in FDM's prediction within each trial. Note that the predicted future state rarely corresponds to

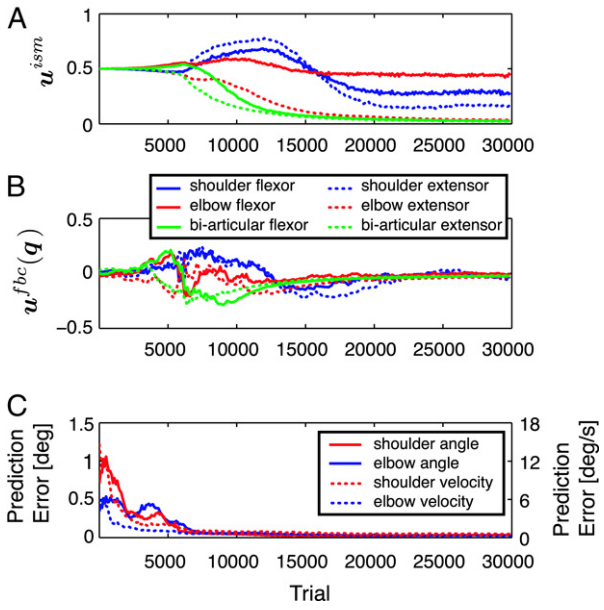


Fig. 4. Improvements in the ISM, FBC, and FDM: (A) The output of the ISM in each trial. (B) The output of the FBC when the predicted future state of the arm corresponds to the target state, that is, $\boldsymbol{q} = 0$. (C) Average error of FDM's prediction in each trial. The plots are only made for the range from 1st to 30,000th trial and each of them is the average over successive 100 trials. Notable changes were not found after the 30,000th trial. Note that the data plotted in (B) were obtained by inputting $\boldsymbol{q} = \mathbf{0}$ into the FBC at the interval between one trial and the next.

the target state during each trial. Therefore, we obtained the data plotted on (B) by inputting $\boldsymbol{q} = 0$ to the FBC at the interval between one trial and the next. Since the FDM was trained in a supervised learning manner in which the teaching signals were given by observing the state of the arm, its prediction error kept reducing from the beginning of the learning process (see Fig. 4(C)). The ISM was also trained with supervised learning in the meaning that the error signals were given by the FBC. However, unlike the FDM, its outputs were unchanged in the early stage of the learning process (see Fig. 4(A)). This is because the FBC, whose outputs were used as the error signals to train the ISM, needed thousands of training trials to acquire decent performance. The output of the ISM began to change since the FBC started to generate biased motor commands needed to hold the arm against the gravitational force. ISM's training progressed as the number of trial increases, and FDM's output against $\boldsymbol{q} = 0$, the error signal for the ISM when the predicted future state corresponds to the target state, came close to zero (see Fig. 4(B)).

4.2.3. Reaching motions

Next, let us see how the reaching motion changed during the learning process. Fig. 5 illustrates reaching motions simulated with sets of weight parameters just after 1000th, 5000th, 10,000th, and 100,000th trial. The tangential velocity profiles of the hand are also plotted under the illustrations of the arm's motions. We set initial and target states as $\boldsymbol{\theta}(0) = (-40, 80, 0, 0)$ and $\boldsymbol{\theta}^{trg} = (-80, 40, 0, 0)$ for these demonstrations. At the early stage of the learning process (1000th trial), the elbow joint was extended and the arm got away from the target (Fig. 5(A)). As a result, the arm got out of the work space and the reaching ended up in failure. The reaching at the 5000th trial also failed. However, considerable improvement can be seen in the behavioral output. Unlike the 1000th trial, the arm once moved toward the target as seen in Fig. 5(B). Although the arm did not stop at the target, it remained within the work space. As the number of trials increases up to 10,000, the motor control-learning model became able to stop the arm at the target and hold it there as seen in Fig. 5(C). Although the velocity profile is almost bell-shaped, there is a small bump around $t = 0.5$ s, indicating a lack of smooth deceleration of the arm. This small bump in the velocity profile seems to result from a small corrective motion that is seen around the target point. At the final stage of the learning process (100,000th trial), no corrective movement is observed in arm's motion and the velocity profile became a smooth bell-shape typically observed in point-to-point reaching movements of adult human (Fig. 5(D)).

4.2.4. Dependence of movement accuracy on the target position

To see how accurately the motor control-learning model became able to reach the targets through training trials, we simulated reaching movements toward 900 different targets. We used a set of weight parameters just after 100,000th trial to implement the control system. As a measure of the accuracy of reaching movement, we adopted the *final hand distance*, the distance between the target point and the hand position at the last moment of each trial. For each of the 900 targets, ten reaching movements starting from ten randomly chosen initial states were simulated. The average value of ten *final hand distances* against the same target is converted into color grade and shown on the position of corresponding target to represent the accuracy of reaching as the color map (Fig. 6). For almost all of the targets, the hand reached the points within 10 mm around the target points. The average and standard deviation of the *final hand distances* among all targets are 3.37 mm and 1.57 mm, respectively. Therefore, it can be said that our model succeeded in achieving highly accurate reaching through the successful learning process.

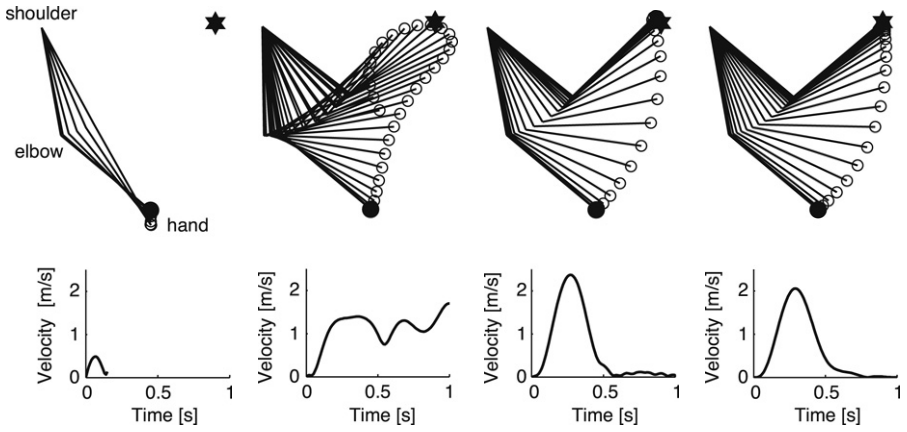
4.3. Essential role of ISM in reaching control and learning

As we mentioned in the introduction of this paper, the ISM seems necessary for accurate reaching movements toward various target points, since it can provide the target-dependent static force required to hold the hand at the target. However, one might think that the FBC learned with the actor-critic method would be good enough for the reaching task. In order to show the importance of the ISM in arm reaching control and learning, we simulated the learning process of the reaching task with a motor control-learning model that has almost the same architecture as proposed one except for not including the ISM. To avoid the motor command being negative, we made the feedback motor command $\boldsymbol{u}^{fbc}$ to take the value within 0 to 1 by altering Eq. (14) to $u_i^{fbc}(\boldsymbol{q}) = g(\mu_i(\boldsymbol{q}) + \sigma n_i)$, where function $g()$ is the sigmoid function. The total motor command $\boldsymbol{u}$ is then equal to $\boldsymbol{u}^{fbc}$. By comparing the results of the learning processes simulated by proposed model and the model without the ISM, we demonstrate how the ISM affects the learning process of the reaching task.

4.3.1. Learning performance

We simulated ten learning processes for both models. The values of initial weight parameters and the seed given to a random number generator varied within ten learning processes. All conditions for the simulations were the same as those described in Section 4.1.

To illustrate the time course of each of ten learning processes, we plotted the total reward signal gained in each trial in Fig. 7. The patterns of the growth of total reward signal differ between the learning processes simulated by our model and the model without the ISM. Transitions of the total reward signal followed almost the same course in ten learning processes simulated by our model. In addition, the learning convergence is faster and much steeper compared to the learning processes simulated by the model without the ISM. By contrast, without the ISM, the time



(A) 1000th trial. (B) 5000th trial. (C) 10,000th trial. (D) 100,000th trial.

Fig. 5. Reaching motions during learning process. Upper and lower plots in each of (A)–(D) illustrate arm's motion and tangential velocity profile of the hand from time 0 to 1 s, respectively. White circles in the upper plots denote the locations of the hand during each trial. On the other hand, black circles and hexagrams denote initial positions of the hand and target positions, respectively. The joint angles at the initial and target positions in all of the trials are set as $(\theta_1(0), \theta_2(0)) = (-80, 40)$ and $(\theta_1^{trg}, \theta_2^{trg}) = (-40, 80)$, respectively.

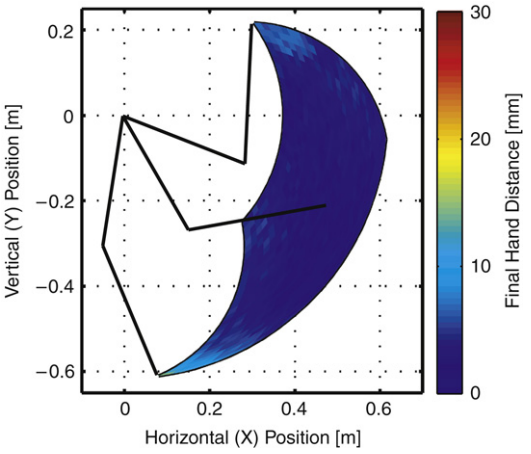


Fig. 6. Accuracy of reaching movements toward 900 targets: The *final hand distance*, the distance between target and hand position at the end of reaching movement, is displayed on the target position in the color map. The origin is set at the shoulder position and three stick figures of the arm are superimposed so as to illustrate the relative size of target area against the arm. All reaching movements are simulated using a set of weight parameters acquired with 100,000 training trials. The average and standard deviation of the 900 *final hand distances* are 3.37 mm and 1.56 mm, respectively.

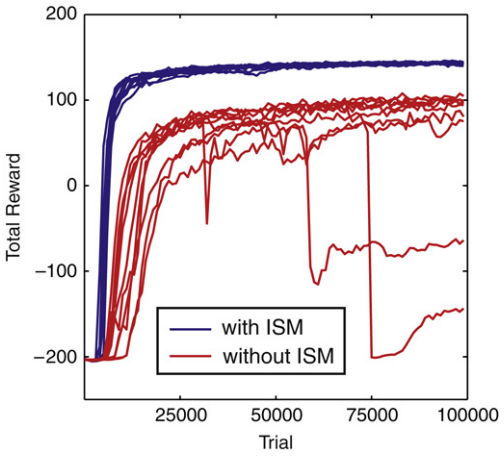


Fig. 7. Learning performance with and without the ISM: Total reward signal gained in each trial in the learning processes simulated by our model (blue lines) and the model without the ISM (red lines). Each of the data plotted in the figure is the average over successive 1000 trials.

course of total reward signals varies from one process to another and two of ten learning processes even ended up in failure. The values of total reward signal at the end of the learning processes were also affected by the existence of the ISM. The average values are 143.0 for ten learning processes with the ISM but 92.1 for eight successful learning processes without the ISM.

4.3.2. Accuracy of reaching movement

We also investigated the effect of the ISM on the accuracy of reaching movements. The accuracy of the reaching movements without the ISM was analyzed in the same way we did in Section 4.2.4. Reaching movements toward 900 different target points were simulated with the motor control-learning model without the ISM. To implement the model, we used a set of weight parameters acquired by the learning process that ended up with highest total reward signal among ten learning processes without the ISM.

Fig. 8 shows the *final hand distances* of the reaching movements against 900 target points. The average and standard deviation of the *final hand distances* among all targets became 11.27 mm and

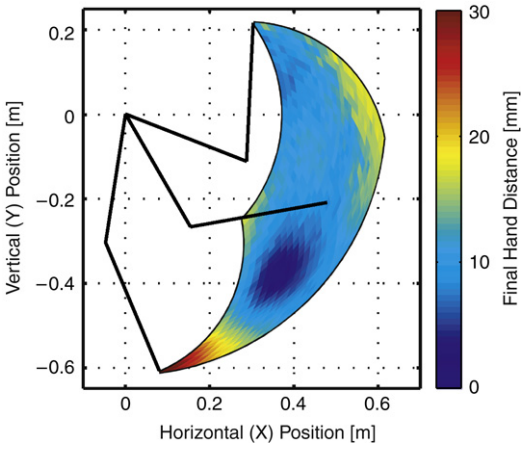


Fig. 8. Accuracy of reaching movements toward 900 targets without the ISM: The *final hand distances* against 900 different targets are displayed on the target position in the color map. All reaching movements are simulated using weight parameters set acquired by the motor control-learning model without the ISM. The average and standard deviation of the 900 *final hand distances* are 11.27 mm and 4.06 mm, respectively.

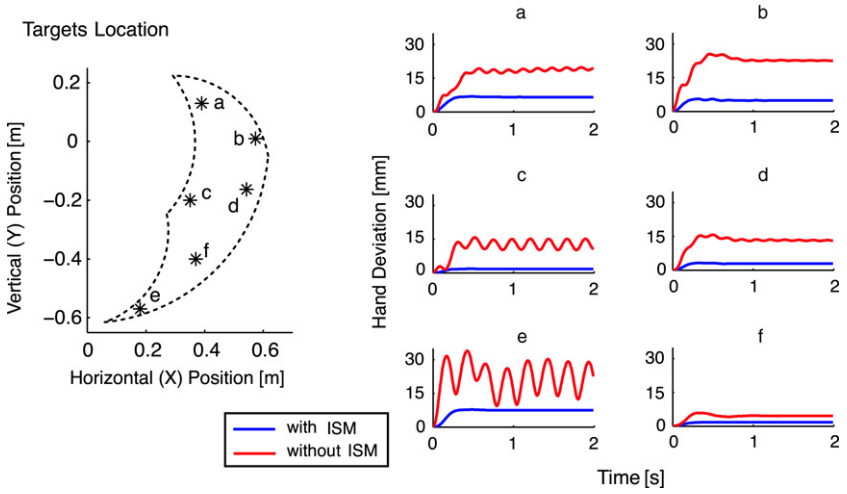


Fig. 9. Posture control task with and without the ISM: The distance between each target and the hand during posture control tasks simulated by the control model with the ISM (blue lines) and without the ISM (red lines). Asterisk markers on the left side figure are the target points. The area bounded by the dashed line is the target area shown in Figs. 6 and 8.

4.06 mm, respectively, while those are 3.37 mm and 1.56 mm with our model. The reaching movements without the ISM are not as accurate as those generated by our model. Furthermore, the *final hand distances* are more sensitive on the target positions compared with those in Fig. 6. The lack of the ISM, which compensates target-dependent static forces, reduced the accuracy of reaching movements and made it more sensitive on the location of target.

In addition, the ISM seems to be useful to keep the hand from trembling around target points. Fig. 9 illustrates temporal variations of hand deviation from target points during the posture control task. Here, we simulated a posture control task by starting reaching movements from the target. When the arm was controlled without the ISM, the hand trembled around the target points in some cases. On the other hand, our model made the hand stay at the point close to the target in any case. These results demonstrate the essential role of the ISM in effective learning and precise and stable control of arm reaching movements in the sagittal plane.

5. Reaching movements of human subject and simulation

In this section, to investigate whether our model can reproduce human-like motions, we compare hand trajectories of point-to-point reaching movements simulated by our model with those of human subjects. We also compare them with the trajectories generated by the "minimum-variance model" (Harris & Wolpert, 1998), that is one of today's most popular motion planning models.

5.1. Experiment setup and data processing

Three right-handed male subjects A, B, C (age 24–32) were asked to generate unconstrained point-to-point reaching movements in the sagittal plane with their right arm. The position of the hand was measured by 3-dimensional position measurement equipment (OPTOTRAK 3020, NDI). We set seven points in the sagittal plane (Fig. 10), and chose seven pairs of initial and target points from them. Seven pairs are shown in Table 1. The subjects executed ten reaching movements for each pair. In order to present locations of the initial and target points, we marked seven points on a clear glass and placed it at subject's underarm. The subjects made a fist and an optical marker was attached to the back of their hands. We also marked a dot at the opposite side of the optical marker on their hands and asked them to make reaching movements so as to overlap the dot to the target points marked on

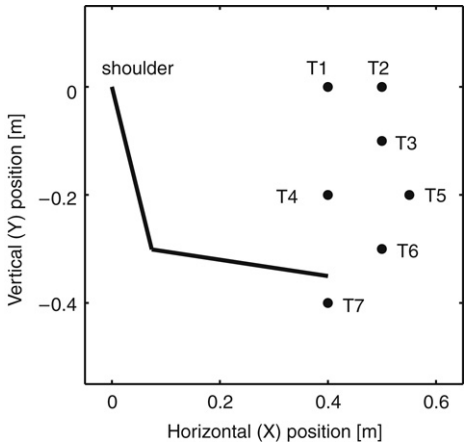


Fig. 10. Location of terminal points for reaching movements in the sagittal plane: The origin is set at the position of the shoulder joint. The positions (X, Y) of seven terminal points are $T1 = (0.4, 0)$, $T2 = (0.5, 0)$, $T3 = (0.5, -0.1)$, $T4 = (0.4, -0.2)$, $T5 = (0.55, -0.2)$, $T6 = (0.5, -0.3)$, $T7 = (0.4, -0.4)$. A stick figure of the arm is superimposed so as to illustrate the relative position of terminal points against the shoulder position.

Table 1
Seven pairs of initial and target points.

Pair index	Initial	Target
a	T5	T1
b	T5	T7
c	T7	T1
d	T2	T7
e	T5	T4
f	T3	T4
g	T6	T4

the clear glass. We asked them to reach as accurately as possible with natural speed and without rotating their wrists. No instruction about the form of hand path or trajectory execution was given to the subjects.

For each of the subjects, we calculated average movement trajectory for each of the seven pairs of initial and target points. The data analysis described below was carried out for each subject. First, positional data of the hand in each movement were digitally low-pass filtered at 10 Hz using fourth-order Butterworth filter. To specify movement duration of each movement, we determined the timing of movement onset and offset using a certain threshold

of hand tangential velocity. We then interpolated the positional data by the spline curve and resampled 100 positional data at equally spaced points between movement onset and offset. The average trajectory of each of the seven pairs was calculated by using the resampled trajectories. We excluded trials whose hand path deviated widely from the average. The average movement duration of each of the seven pairs was used to calculate the tangential velocity of average hand path.

5.2. Simulation setup

To simulate the subjects' reaching using our motor control-learning model, we used a set of weight parameters acquired after 100,000 trials in the learning simulation mentioned in Section 4. For each of the three subjects, seven reaching movements were simulated. In the simulation, initial states of the arm were calculated from the initial hand positions specified in Fig. 10. Since the subjects could not reach exactly to the target points specified in Fig. 10, we determined target states of the arm in the simulation by using the point where subject's average hand path terminates. Note that the only information given to our model to simulate the subjects' movements are the initial and target states of the arm. That is, no information about the position and velocity during the movements was given to our model. Furthermore, the information about the movement duration was not given either.

5.3. Minimum-variance trajectory

The minimum-variance model is one of the most well-known computational models for motor planning of reaching. It claims that reaching movements are planned, in the presence of signal-dependent noise, so as to minimize the variance of hand position over a short post-movement period (Harris & Wolpert, 1998). The model succeeded in reproducing human-like hand trajectories in various reaching tasks in the horizontal plane (Hamilton & Wolpert, 2002; Harris & Wolpert, 1998; Miyamoto et al., 2004). Here we acquired the hand trajectories that minimize the post-movement variance of the reaching movements in the sagittal plane. Initial and target hand positions, and movement durations given to the minimum-variance model are determined from each subject's data. The optimization procedure that we used is almost the same as the one used in the previous work of Harris and Wolpert (1998), except for using the unscented transformation algorithm (Julier, Uhlmann, & Durrant-Whyte, 2004) to derive the variance of hand position during post-movement period. The hand trajectories were parameterized as quintic splines with 7 knots evenly spaced in time. Locations of the knots are initialized so that the quintic splines overlapped with the subjects' trajectories. The post-movement periods were set at 500 ms for all movements. The simplex search method was then used to find optimal knot locations.

The structure of arm model was also almost the same as the one used in the work of Harris and Wolpert (1998), except for the values of link parameters and the existence of gravitational force. Time constants of linear second-order joint actuators are 92.6 ms and 60.5 ms. We set mass, length, center of mass, and inertia of each link at the values used in our simulation (Table 2). We also set shoulder and elbow joint viscosities at 0.52 N m s/rad and 0.33 N m s/rad, respectively. These values are equivalent to the net-viscosities generated by six muscles modeled as Eq. (6), given $\tilde{u}_i = 0.5$ (for all $i = 1, \ldots, 6$).

We programmed the optimization procedure in MATLAB, and verified that the program generates similar results as Harris and Wolpert (1998) when it is applied to reaching movements in the horizontal plane.

5.4. Results

Fig. 11 shows hand trajectories of point-to-point reaching movements executed by the subjects and those simulated by our model. Hand paths and tangential velocity profiles are shown in the figure. Gently curved hand paths were obtained both in the reaching movements of the subjects and our model. For almost all of the movements, the hand paths of the subjects and our model overlap with each other on most parts between the initial and target points. Furthermore, smooth and bell-shaped curves in the velocity profiles of the subjects are reproduced well by our model. Although there are a few movements in which the peak velocities differ between the subjects and our model (movement 'c' and 'd' in (B) and (C)), the hand tangential velocity profiles in the simulation reasonably overlap with the subject's data.

Note that a small bump at the tail of bell-shape relates to a corrective motion around a target point. Such bump can be seen in some of the velocity profiles of our model, while they do not appear in the subjects' data. However, it does not imply that corrective motions exist only in the movements simulated by our model. In fact, the subjects sometimes made corrective motions. The reason why small bumps disappear from the velocity profiles of the subjects is that the movement data during corrective motions were eliminated from the analysis so as to gain the average trajectories. Fig. 12 shows the hand velocity profiles of all of the ten reaching movements between target pair 'b', generated by subject A. Small bumps appear in some of the velocity profiles, indicating the existence of corrective motions in the subject's reaching movements. We have confirmed that these corrective motions can be observed in reaching movements for other target pairs and for other subjects.

The hand trajectories predicted by the minimum-variance model are also plotted in Fig. 11. Some of the hand paths largely deviated from the subjects' data. Upward movements ('a' and 'c') curved largely, while downward movements ('b' and 'd') became a straight line. To investigate which of the two models, our model and the minimum-variance model, makes better prediction of the subjects' trajectories, areas enclosed with hand paths of the subject and those of each model are calculated as a measure of hand path error. In addition, areas enclosed with velocity profiles of subject and each model are calculated as a measure of velocity profile error. Fig. 13 shows the histograms of the hand path errors and velocity profile errors in seven movement patterns. The values shown in the figure are the average data over the three subjects. The hand path errors for our model became smaller in all of the movement patterns, except for pattern 'e'. The velocity profile errors for our model became smaller in four patterns ('a', 'c', 'e', and 'g'). Consequently, for three out of seven movement patterns, our model yields less prediction error in both hand path and velocity profile. By contrast, there is no movement pattern for which the prediction error of the minimum-variance model became smaller in both hand path and velocity profile.

6. Discussion

In this paper, we proposed a computational model for neural motor control of reaching movements. Our model explains not only a mechanism of control but also that of learning. We adopted the feedback-error-learning (FEL) scheme and the actor-critic method so that an inverse statics model and a feedback controller can be trained while executing arm reaching movements. In the FEL scheme, a feedback motor command is used as an error signal to train inverse models of a control object. The original FEL scheme (Kawato et al., 1987), however, did not explain how to acquire a feedback controller itself. In some studies applying the FEL scheme to trajectory control of robotic

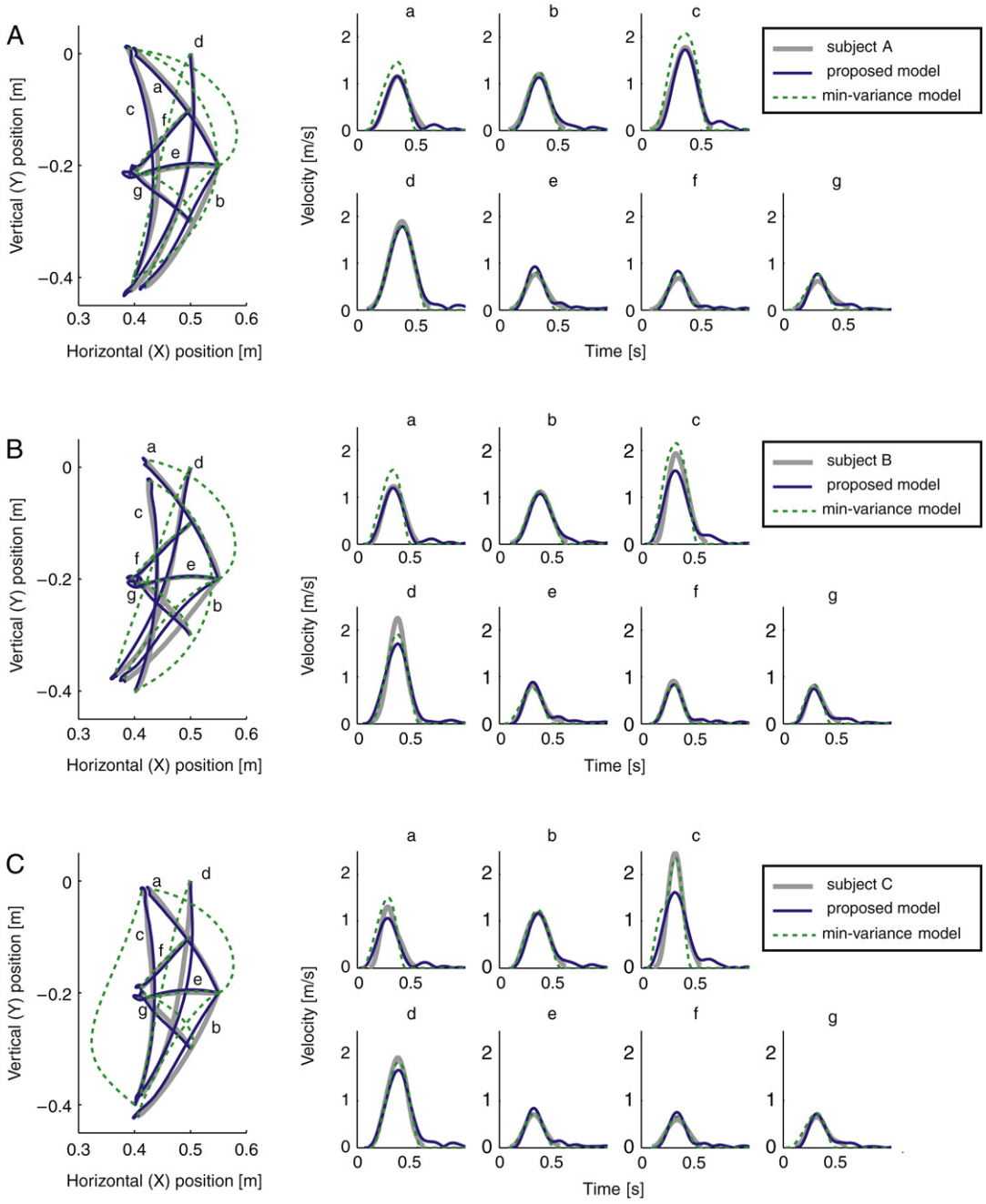


Fig. 11. Reaching movements for seven pairs of initial and target points in the sagittal plane: (A)–(C) Hand trajectory data of subject A–C (gray solid lines), those simulated by our model (blue solid lines), and those simulated by the minimum-variance model (green dashed lines). Left side plots in each of (A)–(C) illustrate hand paths of reaching movements for seven target pairs ('a', …, 'g') shown in Table 1. Right side plots illustrate corresponding hand tangential velocity profiles. The velocity profiles are aligned so that the timing of peak velocity coincides with each other.

manipulators, PID or PD controllers with prefixed gain were used as feedback controllers (Katayama & Kawato, 1991; Miyamoto, Kawato, Setoyoama, & Suzuki, 1988). Although those studies succeeded in acquiring inverse dynamics models for robots, it is hard to assume that those kinds of prefixed gain controllers exist in our brain. The CNS has to learn and adjust the gain in accordance with properties of the body and environment. Although there is a model in which feedback gain is modulated simultaneously with improving feed-forward control (Stroeve, 1996), prior knowledge of the dynamics of the arm and that of the cost function are essential for successful learning in that model. In our model, a feedback controller is trained with the actor-critic method in which reward signals dependent on the results of movements are used to improve the performance. The reward signal used in our model is a scalar value and the control system does not know what elements affect the reward value. Therefore, our model requires neither prior knowledge of the dynamics of the arm, nor that of the reward function to execute reaching movements. Furthermore, it has the ability to automatically adapt to new environments. Whenever the properties of the environment change, the feedback controller is re-adjusted so as to improve the performance of the movements under the new environment and, as a consequence, the inverse statics model is also re-adjusted.

In addition to the problem arising in learning feedback controller, the FEL scheme also has a problem arising from desired trajectory formation when it is applied to reaching tasks. When

Fig. 12. Trial by trial hand velocity profiles of subject: each line indicates the hand velocity profile of the reaching movements for the target pair 'b', generated by subject A. The velocity profiles are aligned at the time when the tangential velocities decrease and become 0.15 m/s during the later half of the movements.

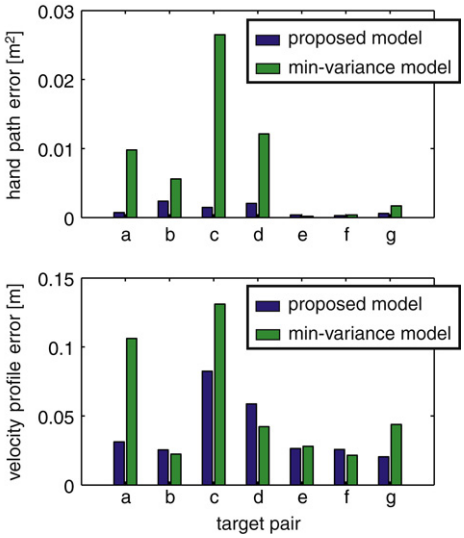


Fig. 13. Trajectory error between subject and the two models: in the upper histogram, areas enclosed with hand paths of subject and each of the two models are shown as a measure of hand path error for seven movement pattern (target pair 'a', . . .,'g'). Likewise, in the lower histogram, areas enclosed with hand velocity profiles of subject and each of the two models are shown as a measure of velocity profile error. Blue bars indicate the trajectory errors of our model, while green bars indicate those of the minimum-variance model. The values in the histograms are the average over three subjects.

inverse dynamics models are trained based on the FEL scheme, the desired trajectories must be specified. The original FEL scheme, however, does not take into account the mechanism of trajectory planning in the CNS. Note that the term 'trajectory' includes not only a sequence of positions, but also sequences of velocities and accelerations. The musculoskeletal system is extremely complex, including non-linear dynamics and many redundancies. Therefore, complex calculation is often necessary to acquire the human-like trajectories minimizing some cost function. In addition, movement duration must be specified to determine the desired trajectory. Although there is a neural network model for trajectory formation (Wada & Kawato, 1993), it requires a pre-trained inverse dynamics model. Thus, the trajectory formation problem in the FEL scheme remains to be solved. An important advantage of combining reinforcement learning with the inverse statics model is that the desired trajectories are no longer needed. In our motor control-learning model, the feedback controller is trained so as to maximize expected cumulative "reward" (which is the same as minimizing cost), and it drives the arm directly toward the target. The concept of using a goal-directed control strategy, instead of following a desired trajectory, has produced successful results in predicting the features of several human movements (Todorov & Jordan, 2002). The inverse statics model also serves as a goal-directed feed-forward controller in our model. It transforms the target position into a set of motor commands that shift the arm's equilibrium position to the target. Combining two goal-directed controllers, the inverse statics model and the feedback controller, our model starts driving the arm directly toward a target as soon as the target is specified. Consequently, the desired trajectories are not needed in our model and movement durations are determined as results of movement executions.

In our model, a forward dynamics model was introduced in addition to the feedback controller and the inverse statics model. Since muscle force develops gradually in time, there exists a time delay between the motor command and muscle force (Mannard & Stein, 1973). Therefore, the arm becomes unstable if a feedback motor command is determined from current joint angle and its time derivative. To compensate for the time delay of force development, we designed the forward dynamics model to predict future state of the angles and angular velocities of arm joints. The predicted future state is then used to calculate the feedback motor command. Through the simulations, we demonstrated that our model can learn stable control of an arm with muscles modeled as a second order system. Granted, the prediction of future state seems unnecessary for stable control if the feedback motor command is properly determined from, in addition to current joint angles and angular velocities, variables representing muscle tension and its time derivative. However, it is still unknown whether the CNS can estimate such quantities as time derivative of muscle tension. Furthermore, the number of muscles in the body is quite large compared with that of joints, and so the number of state variables that the control system has to take into account greatly increases. From a technical point of view, the increase in dimensionality causes the so-called "curse of dimensionality" and makes the control problem harder to solve by reinforcement learning. We avoided the increase in dimensionality of state space by designing the forward dynamics model to predict future angles and angular velocities of the arm and utilize them to determine the current motor command. The existence of the forward dynamics model in the CNS is supported by psychophysical experiments (Bard et al., 1999; Flanagan, Vetter, Johansson, & Wolpert, 2003; Wolpert et al., 1995). Fine predictive adjustments of body movement are also found in some types of voluntary arm movements (De Wolf, Slijper, & Latash, 1998; Flanagan & Wing, 1997; Gribble & Ostry, 1999). We have not analyzed in detail the effects of using the forward dynamics model, and in particular how far in the future it should attempt to predict. However, there is a possibility that the CNS is predicting the future state and utilizing it intelligently so as to control our complex body.

The hand paths of reaching movements simulated by our model were qualitatively in strong agreement with those of subjects' movements. Moreover, our model succeeded in reproducing smooth bell-shaped velocity profiles that is a common invariant feature in arm reaching movements. The main reason that our model produced the smooth bell-shaped velocity profiles seems to come from the properties of the arm model, especially the gradual development of muscle force. One might think that smooth basis functions used for the description of the artificial networks of the FBC and the ISM induced smooth changes in motor commands, and hence, smooth movements were realized. However, this is not the case for our simulation. If we do not assume low-pass filtering of motor commands such as that used in Eq. (5), velocity profiles show steep initial rises at the movement onset even though the networks of the FBC and the ISM are described using

smooth gaussian basis functions (Kambara, Kim, Sato, & Koike, 2004). There might be also the suggestion that signal-dependent biological noise constrains the trajectories to become smooth so as to minimize movement variance (Harris & Wolpert, 1998). However, this is not the case either. We designed the muscle model to have properties similar to biological muscles. The reason we introduced signal-dependent noise into the arm model was that it is an inherent property of the muscles. We did not expect it to make trajectories smoother. In fact, we made the same learning simulation as described in Section 4 under the condition that the signal-dependent noise was excluded from the arm model. From the results of the simulation, we verified that the hand velocity profiles become smooth and bell-shaped such as those shown in Fig. 12. In addition, the velocity profiles in Harris and Wolpert (1998) became smooth and bell-shaped only if the second-order muscle model was used. If there were no time delay between motor command and force generation, a steep initial rise appeared in the velocity profile even though signal-dependent noise was introduced (see Figure 2b in Harris and Wolpert (1998)). It has been widely thought that smooth bell-shaped velocity profiles result from the execution of smooth desired movement planned before movement onset (Flash & Hogan, 1985; Harris & Wolpert, 1998; Nakano et al., 1999; Uno et al., 1989). However, it has been shown in recent study that smooth movement trajectories can be acquired by filtering sparse and discontinuous command signals (Sakaguchi & Ikeda, 2007). In the simulations we made, the desired position of the arm was instantaneously shifted to the target position. Therefore, it can be said that the desired trajectory given to the control system changed discontinuously. However, the velocity profiles became smooth and bell-shaped. This is attributed to the fact that muscles in the simulation were modeled as a second-order system just like biological muscles.

We also considered the biological plausibility of our model. The model consists of three main components, a forward dynamics model, inverse statics model, and feedback controller. The forward dynamics model predicts the future state of the arm given the current state and ongoing motor commands. It has been suggested that the forward dynamics model is acquired in the cerebellum (Miall & Wolpert, 1993; Wolpert et al., 1998). It can be trained in a supervised learning manner using actually sensed state as the teaching signal. The circuit structure of the cerebellum has been shown to be capable of implementing the supervised learning paradigm (Albus, 1971; Doya, 1999; Ito, 1989; Marr, 1969). The mossy fibers, input pathway to the cerebellum, carry both afferent sensory and cerebral efferent signals. Thus, information about current state of the arm and efferent copies of ongoing motor commands can be fed to the cerebellum through the mossy fiber input pathway. The activity of some of the Purkinje cells, whose axons provide cerebellar output through deep cerebellar nuclei, correlates with kinematical state of movement (Coltz, Johnson, & Ebner, 2000; Fu, Flament, Coltz, & Ebner, 1997). The climbing fibers, another input pathway to the cerebellum, are suggested to carry information about error signals of cerebellar outputs. Although there is no direct evidence indicating that the climbing fibers encode prediction error of the body's state, it is reported that the climbing fibers have sensitivity to unexpected sensory events (Andersson & Armstrong, 1987; Gellman, Gibson, & Houk, 1985). Taking this evidence into account, there is a possibility that the forward dynamics model of the arm is acquired in the cerebellum, and its output, that is the prediction of future state, is relayed to the cerebral cortex through the thalamus and utilized to generate a feedback command signal.

The inverse statics model in our model handles the static component of the inverse dynamics of the arm and controls the equilibrium of the arm. It has been suggested that the inverse dynamics models of the body are acquired by feedback-error-learning in the cerebellum (Kawato & Gomi, 1992; Schwighofer, Spoelstra, Arbib, & Kawato, 1998). Although there is direct neurophysiological support for the cerebellar feedback-error-learning model of eye movements (Kawato, 1999), it is unknown whether the inverse dynamics model of the arm exists in the cerebellum. However, taking into account the uniformity of neural circuitry in the cerebellum, the possibility of cerebellar feedback-error-learning of arm movements cannot be eliminated. Since complex spikes of the Purkinje cells, driven by the climbing fibers' activities, occur during steady posture (Miall, Keating, Malkmus, & Thach, 1998), we believe that the inverse statics model is acquired in the cerebellum through feedback-error-learning. It has been shown that the activities of arm muscles correlate with the activities of some of the Purkinje cells in the intermediate part of lobules IV–VI (Yamamoto, Kawato, Kotosaka, & Kitazawa, 2007). In addition, the arm area of the primary motor cortex receives input from the Purkinje cells in that part (Kelly & Strick, 2003). Therefore, it seems possible that the motor command generated from the cerebellum is integrated with the feedback motor command at the primary motor cortex and is sent to the muscles through the spinal motor neurons.

Finally, the feedback controller in our model is trained by reinforcement learning. There are several suggestions that reinforcement learning is implemented in the neural circuits involving cortico-striatal and striato-nigral loops (Barto, 1995; Doya, 1999; Haruno & Kawato, 2006; Joel, Niv, & Ruppin, 2002). In addition, the dopamine neurons in the basal ganglia (substantia nigra pars compacta) are known to encode the signals acting like the TD error in reinforcement learning paradigm (Schultz, Dayan, & Montague, 1997; Suri, 2002). We have not found, so far, direct evidence suggesting the existence of feedback controller for the arm movements in the cerebral cortex. However, taking into account the fact that Huntington's disease patients showed a dysfunction in error feedback control during reaching movements (Smith, Brandt, & Shadmehr, 2000), there seems a possibility that the feedback controller is trained by reinforcement learning and acquired somewhere in cortico-striatal loop, especially involving the pre- and primary motor cortices.

Table 2
Link parameters.

	Upper arm ($j = 1$)	Lower arm ($j = 2$)
m_j (kg)	1.59	1.44
l_j (m)	0.3	0.35
l_{gj} (m)	0.18	0.21
I_j (kg m^2)	6.78×10^{-2}	7.99×10^{-2}

Table 3
Moment arms:

	a_{11}	a_{21}	a_{32}	a_{42}	a_{51}	a_{52}	a_{61}	a_{62}
Moment arm (cm)	4.0	4.0	2.5	2.5	2.8	2.8	3.5	3.5

7. Conclusion

In this paper, we proposed a computational model for arm reaching control and learning. Our model consists of neural networks implementing feedback controller, inverse statics model, and forward dynamics model of the arm. Each of them is trained with reinforcement learning, feedback-error-learning, and

Table 4
Muscle parameters:

	k_{0i} (N/m)	k_{1i} (N/m)	b_{0i} (N s/m)	b_{1i} (N s/m)	l_{1i}^{rest} (cm)	$l_{0i} - l_{0i}^{rest}$ (cm)
Shoulder flexor ($i = 1$)	1000	3000	50	100	15.0	7.7
Shoulder extensor ($i = 2$)	1000	2000	50	100	15.0	12.8
Elbow flexor ($i = 3$)	600	1400	50	100	15.0	10.0
Elbow extensor ($i = 4$)	600	1200	50	100	15.0	4.0
Double-joint flexor ($i = 5$)	300	600	50	100	15.0	2.0
Double-joint extensor ($i = 6$)	300	600	50	100	15.0	1.9

supervised learning, respectively. The error signals given to our model are the state vector of the arm and scalar reward indicating the task performance. Prior knowledge about the dynamics of the arm is not necessary for our model to accomplish a reaching task. Instead, the performance of reaching movement is improved through training trials in which the arm is controlled in a trial-and-error manner. We applied our model to a multi-joint reaching task in the sagittal plane. Simulation results demonstrated the ability of our model to generate quite accurate reaching movements toward various target points.

The other important feature of our model is that it drives the arm directly toward the target points, and hence, it does not require desired trajectories to generate reaching movements. We compared the movement trajectories of the simulation with those of human subjects. The hand paths and velocity profiles in the simulation were qualitatively in good agreement with experimental data. It is quite reasonable based on the present result that the CNS can realize smooth movements without desired trajectory planning if it controls the arm in the way our model adopted.

Acknowledgements

We thank Professor Emanuel Todorov at University of California San Diego for his helpful discussions and comments. A part of this research was supported by Japan Science and Technology Agency, CREST. Also this research was supported by Japan Society for the Promotion of Science, Grant-in-Aid for Young Scientists (Start-up 19860031).

Appendix A. Dynamic equations of the 2-link arm

The dynamic equations of the 2-link arm in the sagittal plane shown in Fig. 2 are given by

$$\tau_1 = M_{11}\ddot{\theta}_1 + M_{12}\ddot{\theta}_2 + h_{122}\dot{\theta}_2^2 + 2h_{112}\dot{\theta}_1\dot{\theta}_2 + g_1$$
$$\tau_2 = M_{21}\ddot{\theta}_1 + M_{22}\ddot{\theta}_2 + h_{211}\dot{\theta}_1^2 + g_2 \quad (28)$$

where

$$M_{11} = I_1 + I_2 + m_2(l_1^2 + 2l_1 l_{g2}\cos\theta_2)$$
$$M_{12} = M_{21} = I_2 + m_2 l_1 l_{g2}\cos\theta_2$$
$$M_{22} = I_2$$
$$h_{122} = h_{112} = -h_{211} = -m_2 l_1 l_{g2}\sin\theta_2$$
$$g_1 = m_1\hat{g}l_{g1}\cos\theta_1 + m_2\hat{g}(l_1\cos\theta_1 + l_{g2}\cos(\theta_1 + \theta_2))$$
$$g_2 = m_2\hat{g}l_{g2}\cos(\theta_1 + \theta_2). \quad (29)$$

Here, m_j, l_j, l_{gj}, I_j are the mass, the length, the distance from the center of mass to the joint, and the rotary inertia of the jth link around the joint, respectively. θ_1 and θ_2 are angles of shoulder and elbow joints, respectively. $\hat{g}$ is the gravitational acceleration. The parameters of the 2-link arm are shown in Table 2.

Appendix B. Values of muscle parameters

We set the values of moment arms and muscle parameters as shown in Tables 3 and 4, respectively. The meaning of muscle parameters are explained in Section 3.1. We determined the value of $(l_{0i} - l_{0i}^{rest})$ so that the muscle tension does not become negative for all possible situations.

References

Abbeel, P., Coates, A., Quigley, M., & Ng, A. Y. (2007). An application of reinforcement learning to aerobatic helicopter flight. In B. Schölkopf, J. Platt, & T. Hofmann (Eds.), *Advaces in neural information processing system*: *Vol. 19* (pp. 1–8). Cambridge, MA: MIT Press.

Abend, W., Bizzi, E., & Morasso, P. (1982). Human arm trajectory formation. *Brain*, *105*(2), 331–348.

Albus, J. (1971). A theory of cerebellar function. *Mathematical Bioscience*, *10*(1–2), 25–61.

Andersson, G., & Armstrong, D. M. (1987). Complex spikes in purkinje cells in the lateral vermis (b zone) of the cat cerebellum during locomotion. *The Journal of Physiology*, *385*(1), 107–134.

Atkeson, C. G., & Hollerback, M. (1985). Kinematic features of unrestrained vertical arm movements. *The Journal of Neuroscience*, *5*(9), 2318–2330.

Bard, C., Turrell, Y., Fleury, M., Teasdale, N., Lamarre, L., & Martin, O. (1999). Deaf-ferentation and pointing with visual double-step perturbations. *Experimental Brain Research*, *125*(4), 410–416.

Barto, A. G. (1995). Adaptive critics and the basal ganglia. In C. J. Houk, J. L. Davis, & B. D. G. (Eds.), *Models of information processing in the basal ganglia* (pp. 215–232). Cambridge, MA: MIT Press (Chapter 11).

Bugmann, G. (1998). Normalized gaussian radial basis function networks. *Neurocomputing*, *20*(1–3), 97–110.

Coltz, J. D., Johnson, M. T. V., & Ebner, T. J. (2000). Population code for tracking velocity based on cerebellar purkinje cell simple spike firing in monkeys. *Neuroscience Letters*, *296*(1), 1–4.

De Wolf, S., Slijper, H., & Latash, L. M. (1998). Anticipatory postural adjustments during self-paced and reaction-time movements. *Experimental Brain Research*, *121*(1), 7–19.

Doya, K. (1999). What are the computations of the cerebellum, the basal ganglia and the cerebral cortex. *Neural Networks*, *12*(7–8), 961–974.

Doya, K. (2000). Reinforcement learning in continuous time and space. *Neural Computation*, *12*(1), 219–245.

Feldman, A. G. (1966). Functional tuning of the nervous system during control of movement or maintenace of a steady posture–3. Mechanographic analysis of the execution by man of the simplest motor tasks. *Biophysics*, *11*, 766–775.

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

Flanagan, J. R., & Wing, A. M. (1997). The role of internal models in motion planning and control: Evidence from grip force adjustments during movements of hand-held loads. *The Journal of Neuroscience*, *17*(4), 1519–1528.

Flash, T. (1987). The control of hand equilibrium trajectories in multi-joint arm movements. *Biological Cybernetics*, *57*(4–5), 257–274.

Flash, T., & Hogan, N. (1985). The coordination of arm movements: An experimentally confirmed mathmatical model. *The Journal of Neuroscience*, *5*(7), 1688–1703.

Fu, Q.-G., Flament, D., Coltz, J. D., & Ebner, T. J. (1997). Relationship of cerebellar purkinje cell simple spike discharge to movement kinematics in the monkey. *Journal of Neurophysiology*, *78*(1), 478–491.

Gellman, R., Gibson, A. R., & Houk, J. C. (1985). Inferior olivary neurons in the awake cat: Detection of contact and passive body displacement. *Journal of Neurophysiology*, *54*(1), 40–60.

Gribble, P. L., & Ostry, D. J. (1999). Compensation for interaction torques during single- and multijoints limb movement. *Journal of Neurophysiology*, *82*(5), 2310–2326.

Gribble, P. L., Ostry, D. J., Sanguineti, V., & Laboissiere, R. (1998). Are complex control signals required for human arm movement? *Journal of Neurophysiology*, *79*(3), 1409–1424.

Hamilton, A. F. C., & Wolpert, D. M. (2002). Controlling the statistics of action: Obstacle avoidance. *Journal of Neurophysiology*, *87*(5), 2434–2440.

Harris, C. M., & Wolpert, D. M. (1998). Signal-dependent noise determines motor planning. *Nature*, *394*(6695), 780–794.

Haruno, M., & Kawato, M. (2006). Heterarchical reinforcement-learning model for integration of multiple cortico-striatal loops: fmri examination in stimulus-action-reward association learning. *Neural Networks*, *19*(8), 1242–1254.
Hogan, N. (1984). An organizing principle for a class of voluntary movements. *The Journal of Neuroscience*, *4*(11), 2745–2754.
Ito, M. (1989). Long-term depression. *Annual Review of Neuroscience*, *12*, 85–102.
Izawa, J., Kondo, T., & Ito, K. (2004). Biological arm motion through reinforcement learning. *Biological Cybernetics*, *91*(1), 10–22.
Joel, D., Niv, Y., & Ruppin, E. (2002). Actor-critic models of the basal gaglia: New anatomical and computational perspectives. *Neural Networks*, *15*(4–6), 535–547.
Julier, S., Uhlmann, J., & Durrant-Whyte, H. F. (2004). A new method for the nonlinear transformation of means and covariances in filtering and estimators. *IEEE Transaction on Automatic Control*, *45*(3), 477–482.
Kambara, H., Kim, J., Sato, M., & Koike, Y. (2004). Learning arm's posture control using reinforcement learning and feedback-error-learning. In *Proceedings of the 26th annual international conferenece of the IEEE engineering in medicine and biology society* (pp. 486–489).
Katayama, M., & Kawato, M. (1991). Learning trajectory and force control of an aritificial muscle arm by parallel-hierachical neural network model. In R. P. Lippmann, J. E. Moody, & D. S. Touretzky (Eds.), *Advaces in neural information processing system: Vol. 3* (pp. 436–442). San Mateo, CA: Morgan Kauffmann (Chapter 8).
Kawato, M. (1999). Internal models for motor control and trajectory planning. *Current Opinion in Neurobiology*, *9*(6), 718–727.
Kawato, M., Furukawa, K., & Suzuki, R. (1987). A hierachical neural-network model for control and learning of voluntary movement. *Biological Cybernetics*, *53*(2), 56–66.
Kawato, M., & Gomi, H. (1992). A computational model of four regions of the cerebellum based on feedback-error learning. *Biological Cybernetics*, *68*(2), 95–103.
Kelly, R. M., & Strick, P. L. (2003). Cerebellar loops with motor cortex and prefrontal cortex of a nonhuman primate. *The Journal of Neuroscience*, *23*(23), 8432–8444.
Koike, Y., & Kawato, M. (1995). Estimation of dynamic joint torques and trajectory formation from surface electromyography signals using a neural network model. *Biological Cybernectics*, *73*(4), 291–300.
Konczak, J., & Dichgans, J. (1997). The development toward stereotypic arm kinematics during reaching in the first 3 years of life. *Experimental Brain Research*, *117*(2), 346–354.
Mannard, A., & Stein, R. B. (1973). Determination of the frequency response of isometric solues muscle in the cat using random nerve stimulation. *The Journal of Physiology*, *229*(2), 275–296.
Marr, D. (1969). A theory of cerebellar cortex. *The Journal of Physiology*, *202*(2), 437–470.
Miall, R. C., Keating, J. G., Malkmus, M., & Thach, W. T. (1998). Simple spike activity predicts occurrence of complex spikes in cerebellar purkinje cells. *Nature Neuroscience*, *1*(1), 13–15.
Miall, R. C., & Wolpert, D. M. (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.
Miyamoto, H., Kawato, M., Setoyoama, T., & Suzuki, R. (1988). Feedback-error-learning neural network for trajectory control of robotic manipulator. *Neural Networks*, *1*(3), 251–265.
Miyamoto, H., Nakano, E., Wolpert, D. M., & Kawato, M. (2004). Tops (task optimization in the presence of signal-dependent noise) model. *Systems and Computers in Japan*, *35*(11), 48–58.
Nakano, E., Imamizu, H., Osu, R., Uno, Y., Gomi, H., Yoshioka, T., et al. (1999). Quantative examinations of internal representations for arm trajectory planning: Minimum commanded torque change model. *Journal of Neurophysiology*, *81*(5), 2140–2155.
Ozkaya, N., & Nordin, M. (1999). *Fundamentals of biomechanics: Equilibrium, motion, and deformation* (2nd ed.). New York, NY: Springer.
Sakaguchi, Y., & Ikeda, S. (2007). Motor planning and sparse motor command representation. *Neurocomputing*, *70*(10–12), 1748–1752.
Schultz, W., Dayan, P., & Montague, P. R. (1997). A neural substrate of prediction and reward. *Science*, *273*(5306), 1593–1599.
Schwighofer, N., Spoelstra, J., Arbib, M. A., & Kawato, M. (1998). Role of the cerebellum in reaching movements in humans. ii. A neural model of the intermediate cerebellum. *European Journal of Neuroscience*, *10*(1), 95–105.
Smith, M. A., Brandt, J., & Shadmehr, R. (2000). Motor disorder in huntington's disease begins as a dysfunction in error feedback control. *Nature*, *403*(6769), 544–549.
Stroeve, S. (1996). Learning combined feedback and feedforward control of a musculoskeletal system. *Biological Cybernetics*, *75*(1), 73–83.
Suri, R. E. (2002). Td models of reward predictive responses in dopamine neurons. *Neural Networks*, *15*(4–6), 523–533.
Sutton, R., & Barto, A. (1998). *Reinforcement learning: An introduction*. Cambridge, MA: MIT Press.
Todorov, E., & Jordan, M. (2002). Optimal feedback control as a theory of motor coordination. *Nature Neuroscience*, *5*(11), 1226–1235.
Uno, Y., Kawato, M., & Suzuki, R. (1989). Formation and control of optimal trajectory in human multijoint arm movement. *Biological Cybernetics*, *61*(2), 89–101.
Wada, Y., & Kawato, M. (1993). A neural network model for arm trajectory formation using forward and inverse dynamics models. *Neural Networks*, *6*(7), 919–932.
Wolpert, D. M., Ghahramani, Z., & Jordan, M. I. (1995). An internal model for sensorimotor integration. *Science*, *269*(5232), 1880–1882.
Wolpert, D. M., Miall, R. C., & Kawato, M. (1998). Internal models in the cerebellum. *Trends in Cognitive Sciences*, *2*(9), 338–347.
Yamamoto, K., Kawato, M., Kotosaka, S., & Kitazawa, S. (2007). Encoding of movement dynamics by purkinje cell simple spike activity during fast arm movements under resistive and assistive force fields. *Journal of Neurophysiology*, *97*(2), 1588–1599.
Zaal, F. T. J. M., Daigle, K., Gottlieb, L., & Thelen, E. (1999). An unlearned principle for controlling natural movement. *Journal of Neurophysiology*, *82*(1), 255–259.