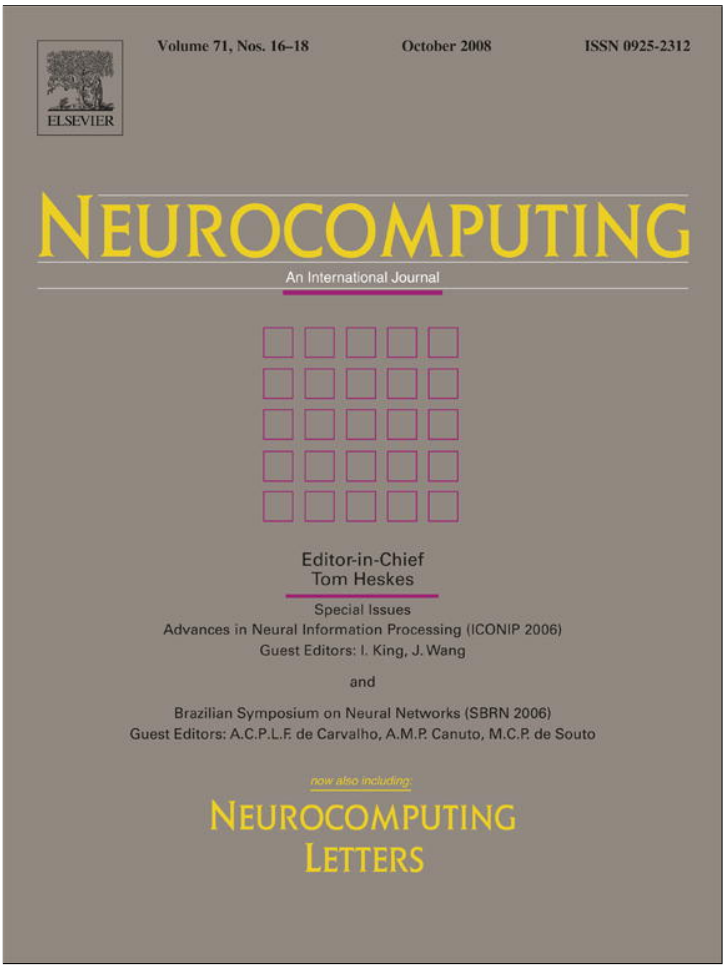

Neurocomputing 71 (2008) 3625–3630

Contents lists available at ScienceDirect

Neurocomputing

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

Letters

Adaptive biomimetic control of robot arm motions

Sungho Jo

Electrical Engineering and Computer Science, KAIST, Daejeon, Republic of Korea

ARTICLE INFO

Article history:
Received 1 December 2006
Received in revised form
22 April 2008
Accepted 29 April 2008
Communicated by M.-J. Er
Available online 20 May 2008

Keywords:
Feedback error learning
Synergy
Arm movement
Adaptation
Biomimetic system

ABSTRACT

By introducing a biologically inspired robotic model that combines a modified feedback error learning, an unsupervised learning, and the viscoelastic actuator system in order to drive adaptive arm motions, this paper discusses the potential usefulness of a biomimetic design of robot skill. The feedback error learning is consistent with the cerebellar adaptation, the unsupervised learning, the synergy network adaptation, and the viscoelastic system of the muscles. The proposed model applies a feedforward adaptive scheme in the low dimensional control space and an adaptive synergy distribution to control redundant actuators effectively. The combination of the two adaptive control schemes is tested by controlling a two-link planar robot arm with six muscular actuators in the gravitational field. The simulation-based study demonstrates that the control scheme adapts the robot arm motions quickly and robustly to generate smooth, human-like motions.

1. Introduction

Humans can routinely attain complex motions by coordinating many degrees-of-freedom skillfully and effortlessly. Current robotic technology still suffers relatively difficulties in performing with such behaviors that are quite natural to humans. Biological inspiration may be able to provide a design framework to achieve smooth motions with less effort. Two major challenges of the biomimetic approach are (1) an efficient and autonomous way of controlling redundant actuators which are potentially a key factor of the efficient and robust generation of motions, and (2) an adaptive control mechanism to acquire skillful motions interacting with the environment or the disturbance.

Computational studies have made progress to attack the problem especially in arm motion control. Reinforcement learning algorithm was used to implement robotic arm movement control [7]. The biological system limited the control subspace where the reinforcement learning is applied in order to handle the redundancy of actuators and demonstrated motion learning without a priori knowledge. However, the model required many trials during learning to attain a simple point-to-point motion. Nakayama and Kimura [14] designed a computational model combining the muscle and muscle spindle system with the cerebellum. The muscle and the muscle spindles are modeled by linear spring-damper systems. The cerebellum does the feedback error learning (FEL) [5,9]. The feedforward torque is calculated through the inverse dynamics, which is computed by the adaptive sliding control [16]. The model performed arm motion tracking tasks quickly and adaptively, but no redundant muscular actuators are considered, and detailed plant inverse dynamics computation is required due to the feature of the adaptive sliding control.

In this paper, we propose a biomimetic adaptive scheme to achieve human-like motions of robot arm in short learning process using redundant actuators, and a simple and less computing adaptive mechanism. The biomimetic approach avoids detailed plant inverse dynamics computation even though it uses the FEL. The effectiveness will be shown by computational experiment.

2. Feedback error learning and viscoelastic muscular actuator

FEL describes the adaptive feedback control as a computational model of the functional role of the cerebellum [5,9,18]. The cerebellum is regarded as a locus of the approximation of the plant inverse dynamics. Initially, a crude feedback controller operates influentially. However, as the system learns the estimation of the plant inverse, the feedforward controller commands the body more dominantly. Fig. 1 illustrates the principal scheme proposed by Gomi and Kawato [5].

The feedback controller is linear as:

$$\bar{\tau}_{\mathrm{fb}} = K_1(\ddot{\bar{\theta}}_{\mathrm{d}} - \ddot{\bar{\theta}}) + K_2(\dot{\bar{\theta}}_{\mathrm{d}} - \dot{\bar{\theta}}) + K_3(\bar{\theta}_{\mathrm{d}} - \bar{\theta}) \quad (1)$$

where K_1, K_2, K_3 are feedback control gains, $\bar{\theta}_{\mathrm{d}}$ denotes the desired position vector, and $\bar{\theta}$ the actual position vector.

E-mail address: shjo@kaist.ac.kr

0925-2312/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.neucom.2008.04.015

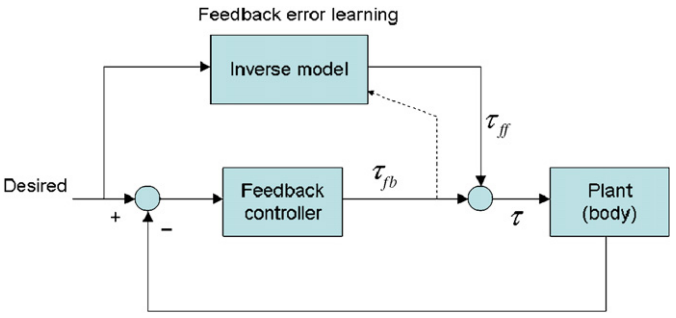

Fig. 1. Feedback error learning scheme.

To acquire the inverse model, different learning schemes could be used. In general, a learning scheme can be expressed by $\bar{\tau}_{\mathrm{ff}} = \phi(\bar{\theta}_{\mathrm{d}}, \bar{\theta}, \dot{\bar{\theta}}_{\mathrm{d}}, \dot{\bar{\theta}}, \ddot{\bar{\theta}}_{\mathrm{d}}, \ddot{\bar{\theta}}, W)$ where W represents the adaptive parameter vector. The adaptive update rule for the FEL is

$$\frac{\mathrm{d}W}{\mathrm{d}t} = \eta \left(\frac{\partial \phi}{\partial W} \right)^{\mathrm{T}} (\bar{\tau}_{\mathrm{fb}} + \bar{\tau}_{\mathrm{ext}}) \tag{2}$$

where $\bar{\tau}_{\mathrm{ext}}$ is the external torque and η is the learning ratio which is small.

The convergence property of the FEL scheme was shown in Gomi and Kawato [5], and Nakanishi and Schaal [12].

Muscle force is generated by the muscular viscoelastic property. The following muscle model has rigorously been studied [4,6,8,11]:

$$\bar{T} = K(\bar{a})(\bar{l}(\bar{a}) - \bar{l}) - B(\bar{a})\dot{\bar{l}} \tag{3}$$

where $K(\bar{a})$ and $B(\bar{a})$ are, respectively, muscle stiffness and viscosity matrices, $\bar{l}$ and $\dot{\bar{l}}$ are, respectively, the vectors of muscle length and its velocity, and $\bar{a}$ is the vector of the descending neural command.

The muscular actuator is functionally a feedback controller for $\bar{l}$ to track $\bar{l}(\bar{a})$, which determines a desired motion. In this case, both control gains ($K(\bar{a})$,$B(\bar{a})$) and shift ($\bar{l}(\bar{a})$) are controllable by neural command. How to adjust the parameters affects energy efficiency, flexibility and rigidness, adaptability and so on in motion performance. For example, increasing stiffness can drive robust motions, but may require high energy consumption. Even with flexed gains, quick shift of $\bar{l}(\bar{a})$ would drive fast motions.

The muscle forces and joint torques, and the muscle length and joint angles, are, respectively, in the geometric relation of

$$\bar{\tau} = A^{\mathrm{T}}\bar{T} = A^{\mathrm{T}}K(\bar{a})(\bar{l}(\bar{a}) - \bar{l}) - A^{\mathrm{T}}B(\bar{a})\dot{\bar{l}}$$

and

$$\bar{l} = \bar{l}_0 + A\bar{\theta}$$

where A is the moment arm matrix and $\bar{l}_0$ is the reference length that indicates the muscle length when joint angles are all zero:

$$M(\bar{\theta})\ddot{\bar{\theta}} + N(\bar{\theta}, \dot{\bar{\theta}}) = \tau = A^{\mathrm{T}}K(\bar{a})(\bar{l}(\bar{a}) - \bar{l}) - A^{\mathrm{T}}B(\bar{a})\dot{\bar{l}} \tag{4}$$

$$A^{\mathrm{T}}K(\bar{a})(\bar{l}(\bar{a}) - \bar{l}_0 - A\bar{\theta}) - A^{\mathrm{T}}B(\bar{a})A\dot{\bar{\theta}} = M(\bar{\theta})\ddot{\bar{\theta}} + N(\bar{\theta}, \dot{\bar{\theta}}) \tag{5}$$

When a desired steady state motion is achieved, it is satisfied that

$$A^{\mathrm{T}}K(\bar{a})(\bar{l}(\bar{a}) - \bar{l}_0) = A^{\mathrm{T}}K(\bar{a})A\bar{\theta}_{\mathrm{d}}$$

The form of $\bar{l}(\bar{a}) = \bar{l}_0 + W_{\mathrm{s}}\bar{a}$ is chosen to satisfy the requirement, and $\bar{a}_{\mathrm{m}} = W_{\mathrm{s}}\bar{a}$ can be interpreted as the motor command to muscles. Therefore, W_{s} determines the distribution of the descending command to muscles and limits the control command space.

A feature of the model structure is to compute the descending commands $\bar{a}$ in a low dimensional space and distribute them via

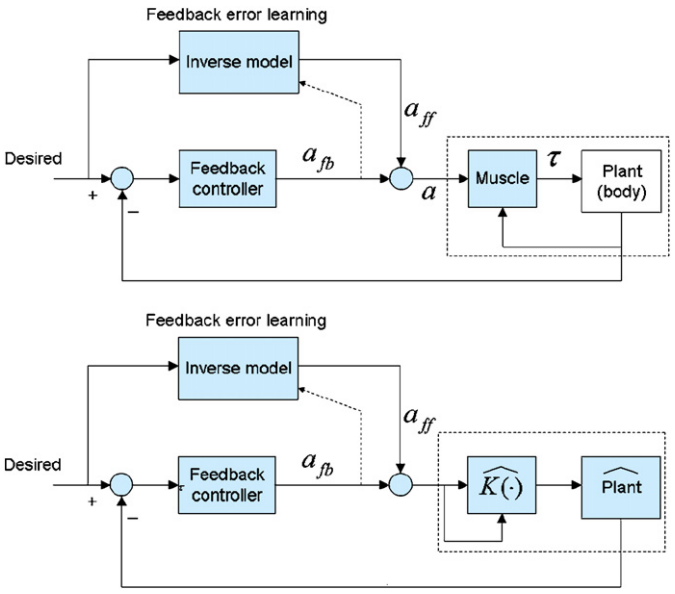

Fig. 2. Modified feedback error learning.

W_{s} rather than to compute the motor command $\bar{a}_{\mathrm{m}}$ directly. It is similar to the muscle synergy concept. The muscle synergy terms a muscle group specified by a principal waveform [2,3,17]. The concept proposes that various muscular motor commands (which correspond to $\bar{a}_{\mathrm{m}}$) can be constructed by a small number of principal waveforms (which correspond to $\bar{a}$). Experimentally it was observed that, using a frog hind limb, synergy underlies a variety of muscular activations that produce different behaviors, and proposed that the synergy is coded within the spinal cord [3,17] and furthermore the spinal motor system participates actively in the behavioral adaptation [2]. W_{s} may be compared with the spinal motor network to effectively and appropriately activate muscles. Eq. (4) can be taken to say that the muscle system includes both characteristics of the alpha and lambda models mentioned in the equilibrium point hypothesis [1]. Stiffness and viscosity can be intensified by the descending command $\bar{a}$ as in the alpha model. The command $\bar{a}$ also shifts as in the lambda model. Therefore, The command $\bar{a}$ is consistent with the virtual equilibrium trajectory in the equilibrium point hypothesis [1]. In this paper, the computation of the descending command $\bar{a}$ is implemented by FEL. The following biologically plausible learning system combines the FEL and the viscoelastic muscle model.

The descending neural command $\bar{a}$ consists of the feedforward component $\bar{a}_{\mathrm{ff}}$ and the feedback component $\bar{a}_{\mathrm{fb}}$.

When the muscle model in Eq. (3) is used, Fig. 2 illustrates that

$$A^{\mathrm{T}}K(\bar{a}_{\mathrm{m}})W_{\mathrm{s}}\bar{a}_{\mathrm{ff}} + A^{\mathrm{T}}K(\bar{a}_{\mathrm{m}})W_{\mathrm{s}}\bar{a}_{\mathrm{fb}} = M(\bar{\theta})\ddot{\bar{\theta}} + N(\bar{\theta}, \dot{\bar{\theta}}) + K_{\theta}(\bar{a}_{\mathrm{m}})\bar{\theta} + B_{\theta}(\bar{a}_{\mathrm{m}})\dot{\bar{\theta}} \tag{6}$$

$$\bar{a}_{\mathrm{ff}} + \bar{a}_{\mathrm{fb}} = (A^{\mathrm{T}}K(\bar{a}_{\mathrm{m}})W_{\mathrm{s}})^{-1}(M(\bar{\theta})\ddot{\bar{\theta}} + N(\bar{\theta}, \dot{\bar{\theta}}) + K_{\theta}(\bar{a}_{\mathrm{m}})\bar{\theta} + B_{\theta}(\bar{a}_{\mathrm{m}})\dot{\bar{\theta}}) = \hat{M}(\bar{a}_{\mathrm{m}}, \bar{\theta})\ddot{\bar{\theta}} + \hat{N}(\bar{a}_{\mathrm{m}}, \bar{\theta}, \dot{\bar{\theta}}) \tag{7}$$

where

$$K_{\theta}(\bar{a}_{\mathrm{m}}) = A^{\mathrm{T}}K(\bar{a}_{\mathrm{m}})A, \quad B_{\theta}(\bar{a}_{\mathrm{m}}) = A^{\mathrm{T}}B(\bar{a}_{\mathrm{m}})A$$

$$\hat{M}(\bar{a}_{\mathrm{m}}, \bar{\theta}) = (A^{\mathrm{T}}K(\bar{a}_{\mathrm{m}})W_{\mathrm{s}})^{-1}M(\bar{\theta})$$

and

$$\hat{N}(\bar{a}_{\mathrm{m}}, \bar{\theta}, \dot{\bar{\theta}}) = (A^{\mathrm{T}}K(\bar{a}_{\mathrm{m}})W_{\mathrm{s}})^{-1}(N(\bar{\theta}, \dot{\bar{\theta}}) + K_{\theta}(\bar{a}_{\mathrm{m}})\bar{\theta} + B_{\theta}(\bar{a}_{\mathrm{m}})\dot{\bar{\theta}})$$

Assuming that the equation on the right hand side represents new body dynamics, $\bar{a}_{\mathrm{ff}}$ and $\bar{a}_{\mathrm{fb}}$ are, respectively, equivalent to $\bar{\tau}_{\mathrm{ff}}$ and $\bar{\tau}_{\mathrm{fb}}$ in the original FEL scheme as long as $A^{\mathrm{T}}K(\bar{a})W_{\mathrm{s}}$ is invertible.

When the position converged to the desired (i.e., $\ddot{\bar{\theta}}_{\mathrm{d}} = \ddot{\bar{\theta}}$, $\dot{\bar{\theta}}_{\mathrm{d}} = \dot{\bar{\theta}}$, $\bar{\theta}_{\mathrm{d}} = \bar{\theta}$), $\bar{a}_{\mathrm{fb}} = 0$ and $\bar{a}_{\mathrm{ff}} = (A^{\mathrm{T}}K(\bar{a}_{\mathrm{ff}})W_{\mathrm{s}})^{-1}(\bar{\tau}_{\mathrm{ff}} + B(\bar{a}_{\mathrm{ff}})\dot{\bar{\theta}}_{\mathrm{d}}) + \bar{\theta}_{\mathrm{d}}$.

3. Model derivation

3.1. Biological robot arm model

An arm is modeled as in Fig. 4. Shoulder and elbow angles are, respectively, θ_1 and θ_2 ($\bar{\theta} = [\theta_1 \quad \theta_2]^{\mathrm{T}}$). The skeletal system consists of two rigid body segments, upper arm and forearm+hand. The arm is affected by the gravitational force.

The skeletal dynamics are described by:

$$\bar{\tau} = M(\bar{\theta})\ddot{\bar{\theta}} + N(\bar{\theta}, \dot{\bar{\theta}}) + G(\bar{\theta}) + D(\bar{\theta}), \tag{8}$$

where $M(\bar{\theta})$ is the 2×2 inertia matrix, $N(\bar{\theta}, \dot{\bar{\theta}})$ is the Coriolis and centrifugal force vector, $G(\bar{\theta})$ is the vector related to gravitational force, and $D(\bar{\theta})$ is the vector to represent an external disturbance effect if any disturbance is applied to:

$$\begin{aligned} M(\bar{\theta})(1,1) &= m_1 r_1^2 + m_2(l_1^2 + r_2^2 + 2l_1 r_2 \cos\theta_2) + I_1 + I_2 \\ M(\bar{\theta})(1,2) &= M(\bar{\theta})(2,1) = m_2(r_2^2 + l_1 r_2 \cos\theta_2) + I_2 \\ M(\bar{\theta})(1,1) &= m_2 r_2^2 + I_2 \end{aligned} \tag{9}$$

$$N(\bar{\theta}, \dot{\bar{\theta}}) = -m_2 l_1 r_2 \sin\theta_2 \begin{bmatrix} \dot{\theta}_2(2\dot{\theta}_1 + \dot{\theta}_2) \\ -\dot{\theta}_1^2 \end{bmatrix} \tag{10}$$

$$G(\bar{\theta}) = \begin{bmatrix} m_1 g r_1 \sin\theta_1 + m_2 g(l_1 \sin\theta_1 + r_2 \sin(\theta_1 + \theta_2)) \\ m_2 g r_2 \sin(\theta_1 + \theta_2) \end{bmatrix} \tag{11}$$

where m_i, I_i, l_i, and r_i are, respectively, the mass, moment of inertia, length, and distance to the center of mass from the adjacent joint of each segment ($i = 1$: upper arm, $i = 2$: forearm) (Table 1).

Six muscular actuators, two bi-articular and four mono-articular, are around the joints. The actuator's force is determined by the Eq. (3).

The muscle stiffness and viscosity, $K(\bar{a})$ and $B(\bar{a})$, are 6×6 diagonal matrices where each element is expressed by:

$$\begin{aligned} K(\bar{a})(i,i) &= K(\bar{a}_{\mathrm{m}})(i,i) = k_{0,i} + k_{1,i} a_{\mathrm{m},i} \\ B(\bar{a})(i,i) &= B(\bar{a}_{\mathrm{m}})(i,i) = b_{0,i} + b_{1,i} a_{\mathrm{m},i} \\ i &= 1, \ldots, 6 \end{aligned} \tag{12}$$

where $k_{0,i}$, $k_{1,i}$, $b_{0,i}$, and $b_{1,i}$ are coefficients.

For simulation, $k_{0,i} = 1621.6$, $k_{1,i} = 810.8$, $b_{0,i} = 108.1$, $b_{1,i} = 54.1$ for all i, and the moment arm matrix

$$A = \begin{bmatrix} -0.04 & 0.04 & 0 & 0 & -0.028 & 0.028 \\ 0 & 0 & -0.025 & 0.025 & -0.035 & 0.035 \end{bmatrix}^{\mathrm{T}}$$

adapted from Katayama and Kawato [8].

From the FEL, the feedforward controller in this paper is modeled as follows:

$$\bar{a}_{\mathrm{ff}} = \phi(\bar{x}_{\mathrm{d}}, \dot{\bar{x}}_{\mathrm{d}}, \ddot{\bar{x}}_{\mathrm{d}}, W) = W_1 \bar{x}_{\mathrm{d}} + W_2 \dot{\bar{x}}_{\mathrm{d}} + W_3 \ddot{\bar{x}}_{\mathrm{d}} \tag{13}$$

Table 1
Parameter values for simulation

	m_i (kg)	I_i (kg m/s^2)	l_i (m)	r_i (m)
Upper arm ($i = 1$)	1.59	0.0477	0.35	0.18
Forewarm ($i = 2$)	1.44	0.0588	0.35	0.21

where each W_i ($i = 1, 2, 3$) is a 2×2 matrix and $W = [W_1 \quad W_2 \quad W_3]$.

The feedforward command $\bar{a}_{\mathrm{ff}}$ depends on kinematic information in Cartesian coordinates. W is the adaptive weight matrix of the neural network that implicitly approximates the transformation from the Cartesian to joint coordinates and the inverse dynamics as well. The adaptive rule of the FEL is used to find the optimal weight matrices:

$$\frac{\mathrm{d}W_1}{\mathrm{d}t} = \eta \bar{x}_{\mathrm{d}} \bar{a}_{\mathrm{fb}}^{\mathrm{T}}, \quad \frac{\mathrm{d}W_2}{\mathrm{d}t} = \eta \dot{\bar{x}}_{\mathrm{d}} \bar{a}_{\mathrm{fb}}^{\mathrm{T}}, \quad \frac{\mathrm{d}W_3}{\mathrm{d}t} = \eta \ddot{\bar{x}}_{\mathrm{d}} \bar{a}_{\mathrm{fb}}^{\mathrm{T}} \tag{14}$$

where η is a learning coefficient.

The hand end position is also geometrically a function of joint angles: $\bar{x} = p(\bar{\theta})$.

Furthermore, differentiating both sides of $\bar{x} = p(\bar{\theta})$ also results in $\dot{\bar{x}} = J(\bar{\theta})\dot{\bar{\theta}}$. Using equations, the hand end position, and its velocity are obtained from joint kinematic information:

$$\begin{bmatrix} \bar{x} \\ \dot{\bar{x}} \end{bmatrix} = P(\bar{\theta}, \dot{\bar{\theta}}) = \begin{bmatrix} p(\bar{\theta}) \\ J(\bar{\theta})\dot{\bar{\theta}} \end{bmatrix} \tag{15}$$

Human sensory information such as visual, proprioceptive, and cutaneous information may be used to detect the hand end location and movement in the spatial configuration and human brain may estimate somehow spatial information such as $P(\bar{\theta}, \dot{\bar{\theta}}, \ddot{\bar{\theta}})$.

Comparison of the estimated hand end position and its velocity with their desired values drives error signals: $\bar{e} = \bar{x}_{\mathrm{d}} - \bar{x}$, $\dot{\bar{e}} = \dot{\bar{x}}_{\mathrm{d}} - \dot{\bar{x}}$.

The error signals are with respect to Cartesian coordinate.

The feedback controller is designed in the form of:

$$\bar{a}_{\mathrm{fb}} = \eta(\bar{e}, \dot{\bar{e}}) = J^{-1}(\bar{\theta})(K_1 \bar{e} + K_2 \dot{\bar{e}}) \tag{16}$$

where K_i is a diagonal matrix.

$J^{-1}(\bar{\theta})$ is multiplied to describe the feedback command in terms of the joint coordinate. From the geometric relation,

$$\bar{x} = \bar{p}(\bar{\theta}_{\mathrm{d}} - \delta\bar{\theta}) \approx \bar{p}(\bar{\theta}_{\mathrm{d}}) - \frac{\partial \bar{p}(\bar{\theta}_{\mathrm{d}})}{\partial \bar{\theta}} \delta\bar{\theta}$$

where $\delta\bar{\theta}$ is small.

The error in Cartesian coordinates,

$$\bar{e} = \bar{x}_{\mathrm{d}} - \bar{x} \cong \frac{\partial \bar{p}(\bar{\theta}_{\mathrm{d}})}{\partial \bar{\theta}} \delta\bar{\theta} = J(\bar{\theta}_{\mathrm{d}})\delta\bar{\theta}$$

and

$$\delta\bar{\theta} = J^{-1}(\bar{\theta}_{\mathrm{d}})\bar{e}$$

Because the desired joint angles are not directly available, $J^{-1}(\bar{\theta}_{\mathrm{d}})$ is replaced by $J^{-1}(\bar{\theta})$. As long as $\bar{\theta}$ is close to its desired value, it will hold good.

An unsupervised learning scheme updates the synergy network weight W_{s}. The W_{s} is selected to minimize the descending command with respect to the joint coordinate. The minimization has an effect in two aspects, energy efficiency, and feedback error minimization. Therefore,

$$\frac{\mathrm{d}W_{\mathrm{s}}}{\mathrm{d}t} = \eta_{\mathrm{s}} W_{\mathrm{s}} \bar{a}\, \bar{a}^{\mathrm{T}} = \eta_{\mathrm{s}} \bar{a}_{\mathrm{m}} \bar{a}^{\mathrm{T}} \tag{17}$$

where η_{s} is a learning coefficient.

In another viewpoint, the learning scheme is equivalent to the Hebbian rule [10]. The input to the synergy network is $\bar{a}$, and the output from the network is $W_{\mathrm{s}}\bar{a}$.

Fig. 3 illustrates the whole system. A feature of the model is that the descending command ($\bar{a}$) is generated in the lower dimensional space than actuators. The command is distributed to muscles by a network (represented by the matrix W_{s}) as mentioned previously. In addition, the feedforward controller adapts the command in

Cartesian coordinate different from the body joint configuration. Therefore, the adaptive weight matrix (W) of the feedforward controller includes implicitly the space transformation or the dimensional reduction of the controller space.

Plugging Eqs. (13) and (16) into Eq. (7) results

$$W_3 J(\bar{\theta}_d)\ddot{\bar{\theta}}_d + (W_2 J(\bar{\theta}_d) + W_3 \dot{J}(\bar{\theta}_d) + K_2 J^{-1}(\bar{\theta}) J(\bar{\theta}_d))\dot{\bar{\theta}}_d + (W_1 + K_1 J^{-1}(\bar{\theta}))P(\bar{\theta}_d) = \hat{M}(\bar{a}_m, \bar{\theta})\ddot{\bar{\theta}} + \hat{N}(\bar{a}_m, \bar{\theta}, \dot{\bar{\theta}})$$

When a task is stably achieved, i.e., $\bar{\theta} \to \bar{\theta}_d$, it is expected that

$$A^T K(\bar{a}_m) W_s W_3 J(\bar{\theta}_d) \to M(\bar{\theta}_d)$$

and

$$A^T K(\bar{a}_m) W_s ((W_2 J(\bar{\theta}_d) + W_3 \dot{J}(\bar{\theta}_d) + K_2 J^{-1}(\bar{\theta}) J(\bar{\theta}_d))\dot{\bar{\theta}}_d + (W_1 + K_1 J^{-1}(\bar{\theta}))P(\bar{\theta}_d)) \to N(\bar{\theta}_d, \dot{\bar{\theta}}_d)$$

and

$$\bar{a}_m \to \bar{a}_{ff}$$

Eqs. (14) and (17) adapt parameters to closely satisfy the above convergence. Each learning rule is simple, however, biological inspiration implicates that combination of such learning rules with viscoelastic actuator and coordinate translation may be able to imitate nonlinear system dynamics without computing detailed inverse dynamics directly. This does not guarantee stable task achievement explicitly. In case that the simple feedforward form of Eq. (13) may not be sufficient to generate quite complicated motions, biological inspiration suggests hierarchical or cascade control structure rather than a specific highly nonlinear control structure. This is relevant to gain scheduling or switching control scheme. This issue is further mentioned in discussion.

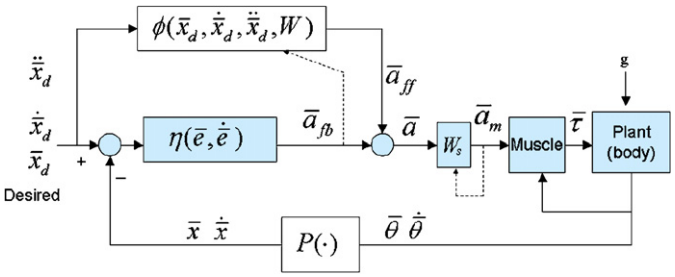

Fig. 3. The model of biological robot arm movement control.

The proposed learning structure is simpler with respect to inverse dynamics computation than using neural nets or reinforcement learning. On the other hand, the proposed control structure is more effective to implement quick motions variously in comparison with servo-type control, or equilibrium-point hypothesis [11].

3.2. Simulation problems

To test the proposed learning mechanism computationally, five arm motions in the gravitational field are selected by defining the desired motions of the hand end point. In this model, each desired motion is described in Cartesian coordinates. Five desired hand end trajectories are as follows in $0 \leqslant t \leqslant T_f$:

- *Case* (1):

$$x_d(t) = 0.35\frac{t}{T_f} + 0.2, \quad y_d(t) = 0.35\frac{t}{T_f}; \quad T_f = 1.$$

- *Case* (2):

$$x_d(t) = 0.35, \quad y_d(t) = 0.6062\frac{t}{T_f}; \quad T_f = 1.$$

- *Case* (3):

$$x_d(t) = 0.1\ \sin\left(\frac{\pi}{2}t - \frac{\pi}{2}\right) + 0.45,$$
$$y_d(t) = 0.1\ \cos\left(\frac{\pi}{2}t - \frac{\pi}{2}\right),$$
$$T_f = 4.$$

- *Case* (4):

$$x_d(t) = \begin{cases} 0.6\frac{t}{T_f} + 0.2, & 0 \leqslant t < \frac{T_f}{2}, \\ 0.5, & \frac{T_f}{2} \leqslant t \leqslant T_f, \end{cases}$$
$$x_d(t) = \begin{cases} 0.3 - 0.6\frac{t}{T_f}, & 0 \leqslant t < \frac{T_f}{2}, \\ 0.6\left(\frac{t}{T_f} - 0.5\right), & \frac{T_f}{2} \leqslant t \leqslant T_f, \end{cases}$$
$$T_f = 2.$$

- *Case* (5):

$$x_d(t) = 0.17\ \sin(\pi t) + 0.35,$$
$$y_d(t) = 0.17\ \cos\left(\frac{\pi}{2}t - \frac{\pi}{2}\right),$$
$$T_f = 4.$$

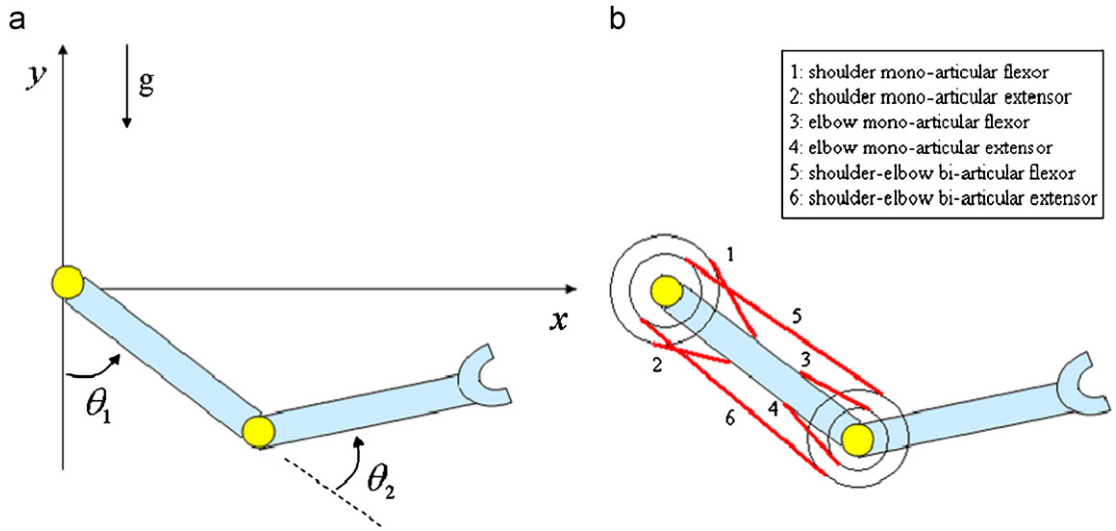

Fig. 4. The biological robot arm musculoskeletal model: (a) body configuration and (b) muscle actuators.

Initially, W is a zero matrix, and W_s is set to A. The learning coefficients η and η_s are, respectively, 0.5 and 0.1:

$$K_1 = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}, \quad K_2 = \begin{bmatrix} 0.6 & 0 \\ 0 & 0.6 \end{bmatrix}$$

The initial position of hand end is equal to $(x_d(0), y_d(0))$, however, the hand remains still. The iterative learning continues until the condition of $\sqrt{(e_k - e_{k-1})^2} < 0.0001$ is satisfied where k is the iteration number and $e_k = \int_0^{T_f} \sqrt{\bar{e}^T \bar{e}}\, dt$.

4. Results

In all cases, the hand end follows its desired position reasonably under the gravitational force after learning in several trials (Fig. 5). The hand position is static initially, but its desired velocity at time zero is nonzero. Therefore, the hand end tends to lag the desired position initially. The third column in Fig. 5 shows that integrated least square errors (e_k) during a task decrease as learning is processed, and finally converge.

Table 2 shows the adapted control weights. In cases (1), (2), and (4), desired accelerations are all zeros, so that there is no adaptation in corresponding locations and zero elements appear. In W_s, the principal activations of extensor and flexor by commands are conserved over all motions (signs of elements are consistent). However, relative contributions are differently adapted depending on motion tasks.

5. Discussion

The learning algorithm combined with viscoelastic muscular property drives the biological robot arm motion skillfully and autonomously. The model proposes the feedforward control scheme in Fig. 6.

The FEL in the feedforward control scheme does not compute the exact inverse dynamics but only approximates it, and uses the

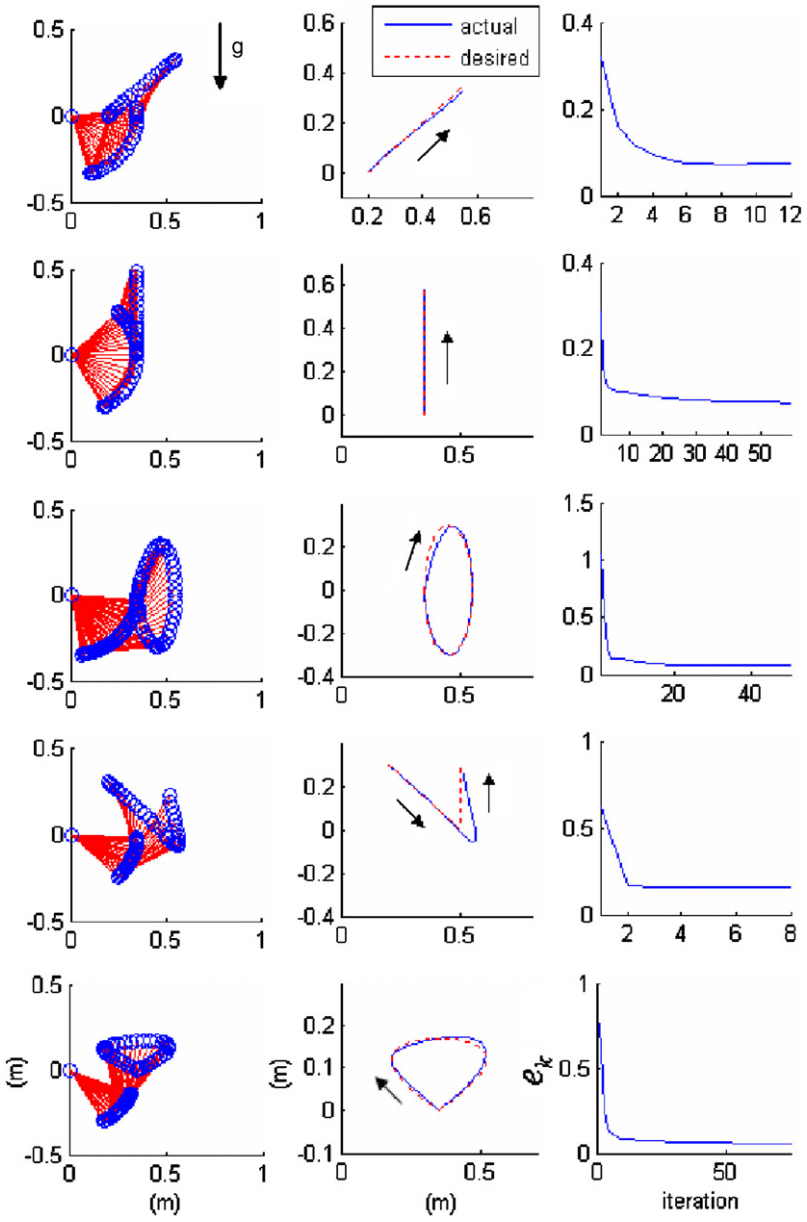

Fig. 5. Simulated motions: four different motions are tested. The first column: stick figures, the second column: motion trajectories, the third column: error in each iteration.

Table 2
The values of W and W_s after learning

	W	W_s
Case (1)	$\begin{bmatrix} -0.4812 & -0.2868 & -0.3402 & -0.3402 & 0 & 0 \\ -0.4077 & -0.0807 & -0.5723 & -0.5723 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0.1592 & -0.1280 & 0.0037 & -0.0027 & 0.0964 & -0.0812 \\ 0.0219 & -0.0176 & 0.1467 & -0.1087 & 0.2461 & -0.1357 \end{bmatrix}^T$
Case (2)	$\begin{bmatrix} 0.0214 & -0.8844 & 0 & 0.0371 & 0 & 0 \\ -0.3317 & 0.9314 & 0 & -0.5744 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0.3599 & -0.2893 & 0.0865 & -0.0641 & 0.3424 & -0.2501 \\ 0.1178 & -0.0947 & 0.2183 & -0.1617 & 0.4147 & -0.2438 \end{bmatrix}^T$
Case (3)	$\begin{bmatrix} -0.3368 & -0.0120 & -0.0377 & 0.0463 & 0.1455 & 0.1186 \\ -0.5083 & 0.0071 & 0.0222 & -0.0558 & -0.1752 & -0.0698 \end{bmatrix}$	$\begin{bmatrix} 0.1906 & -0.1532 & 0.1185 & -0.0877 & 0.2970 & -0.1943 \\ 0.1360 & -0.1093 & 0.2881 & -0.2134 & 0.5361 & -0.3121 \end{bmatrix}^T$
Case (4)	$\begin{bmatrix} -0.5938 & -0.2426 & -0.1846 & -0.0570 & 0 & 0 \\ -0.7195 & -0.2393 & -0.4418 & 0.3144 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0.2372 & -0.1907 & 0.0555 & -0.0411 & 0.2233 & -0.1636 \\ 0.0757 & -0.0609 & 0.1255 & -0.0929 & 0.2428 & -0.1440 \end{bmatrix}^T$
Case (5)	$\begin{bmatrix} -0.1491 & -0.1568 & 0.2938 & -0.0274 & 0.1499 & 0.0967 \\ -0.2858 & -0.0238 & -0.2362 & 0.0167 & -0.2910 & 0.0147 \end{bmatrix}$	$\begin{bmatrix} 0.4533 & -0.3643 & 0.2791 & -0.2066 & 0.7016 & -0.4595 \\ 0.3368 & -0.2703 & 0.5988 & -0.4435 & 1.1442 & -0.6746 \end{bmatrix}^T$

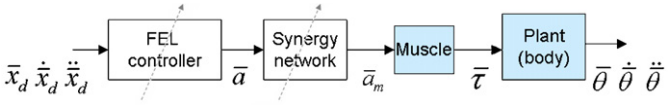

Fig. 6. The proposed biomimetic feedforward control scheme.

Cartesian coordinate space. The space transformation from the Cartesian to the joint configuration is implicitly implemented by the learning algorithm. In addition, the control scheme generates simply the control command in the lower dimensional space than the redundant actuator. The synergy network is self-adaptively updated to map the control command appropriately to the actuators according to each task goal. Therefore, two adaptive control schemes, FEL and adaptive synergy network, operate synchronously. Several example tests illustrate the effectiveness of the proposed learning scheme. However, for future research, it is necessary to investigate gainscheduling inverse dynamics approximations or synergy networks. To implement more complicated or more rapidly converging or highly accurate robot arm motions, a linear approximation of the inverse dynamics and a constant synergy distribution may not be sufficient. In fact, composite adaptive control schemes [13] have been proposed for piecewise linear approximations of the inverse dynamics. However, the biomimetic approach in this study proposes that piecewise adapted synergy distributions or a mixture of the gainscheduling of both linear approximation and synergy distribution may be a more efficient strategy for motion tasks. The enhancement would be further effective to generate complicated multi-joint motions. In addition, a more sophisticated feedback controller would be helpful to implement high-degree-of-freedom motions dexterously. An example of such feedback controller was studied in Ref. [15]. The adaptive schemes in this study could be integrated with a more intelligent feedback controller to improve performance.

References

[1] E. Bizzi, N. Hogan, F.A. Mussa-Ivaldi, S.F. Giszter, Does the nervous system use equilibrium-point control to guide single and multiple joint movements?, Behav. Brain Sci. 15 (1992) 603–613.
[2] E. Bizzi, M.C. Tresch, P. Saltiel, A. d'Avella, New perspectives on spinal motor systems, Nat. Rev. Neurosci. 1 (2000) 101–108.
[3] A. d'Avella, P. Satiel, E. Bizzi, Combinations of muscle synergies in the construction of a natural motor behavior, Nat. Neurosci. 6 (2003) 300–308.
[4] T. Flash, The control of hand equilibrium trajectories in multi-joint arm movements, Biol. Cybern. 57 (1987) 257–274.
[5] H. Gomi, M. Kawato, Learning control for a closed loop system using feedback-error-learning, in: Proceedings of the 29th Conference on Decision and Control, 1990, pp. 3289–3294.
[6] N. Hogan, The mechanics of multi-joint posture and movement control, Biol. Cybern. 52 (1985) 315–331.
[7] J. Izawa, T. Kondo, K. Ito, Biological robot arm motion through reinforcement learning, in: Proceedings of the IEEE International Conference on Robotics and Automation, 2002, pp. 3398–3403.
[8] M. Katayama, M. Kawato, Virtual trajectories and stiffness ellipse during multijoint arm movement predicted by neural inverse models, Biol. Cybern. 69 (1993) 353–362.
[9] M. Kawato, H. Gomi, A computational model of four regions of the cerebellum based on feedback-error learning, Biol. Cybern. 68 (1992) 95–103.
[10] D. Mackay, Information Theory, Inference, and Learning Algorithms, Cambridge University Press, 2003.
[11] J. McIntyre, E. Bizzi, Servo hypotheses for the biological control of movement, J. Motor Behav. 25 (3) (1993) 193–202.
[12] J. Nakanishi, S. Schaal, Feedback error learning and nonlinear adaptive control, Neural Networks 17 (2004) 1453–1465.
[13] J. Nakanishi, J.A. Farrell, S. Schaal, Composite adaptive control with locally weighted statistical learning, Neural Networks 18 (2005) 71–90.
[14] T. Nakayama, H. Kimura, Trajectory tracking control of robot arm by using computational models of spinal cord and cerebellum, Syst. Comput. Jpn. 35 (11) (2004) 1–13.
[15] A.M. Sekimoto, H. Hashiguchi, R. Ozawa, Natural resolution of ill-posedness of inverse kinematics for redundant robots: a challenge to Bernstein's degrees-of-freedom problem, Adv. Robot. 19 (4) (2005) 401–434.
[16] J.E. Slotine, W. Li, Applied Nonlinear Control, Prentice-Hall, 1991.
[17] M.C. Tresch, P. Saltiel, E. Bizzi, The construction of movement by the spinal cord, Nat. Neurosci. 2 (2) (1999) 162–167.
[18] D.M. Wolpert, M. Kawato, Multiple paired forward and inverse models for motor control, Neural Networks 11, 1317–1329.

Sungho Jo received his B.S. degree (1999) from School of Mechanical Engineering, Seoul National University, Seoul, Korea, his M.S. (2001) in mechanical engineering and Ph.D. (2006) in electrical engineering and computer science from Massachusetts Institute of Technology, MA, USA. He was a postdoctoral associate in Biomechatronics group, MIT Media Laboratory. Since December in 2007, he has been with electrical engineering and computer science at KAIST as an assistant professor. His research interests include sensorimotor neural engineering, neurobiological system modeling, human postural balance and biped locomotion, biomimetic systems, etc.