TR－A－0114

Neural-Network Control for a Closed-Loop System using *Feedback-Error-Learning*

Hiroaki GOMI　　*Mitsuo KAWATO*

1991. 4. 25

ATR視聴覚機構研究所

〒619-02 京都府相楽郡精華町乾谷　☎07749-5-1411

ATR Auditory and Visual Perception Research Laboratories

Inuidani,Sanpeidani, Seika-cho, Soraku-gun, Kyoto 619-02 Japan

Telephone: +81-7749-5-1411
Facsimile: +81-7749-5-1408
Telex: 5452-516 ATR J

Neural-Network Control for a Closed-Loop System using *Feedback-Error-Learning*

Hiroaki GOMI, Mitsuo KAWATO

ATR Auditory and Visual Perception Research Laboratories,
Inui-dani, Sanpei-dani, Seika-cho, Soraku-gun, Kyoto 619-02, Japan
Telephone: +81-7749-5-1453 (Mr. Gomi), +81-7749-5-1452 (Dr. Kawato)
Fax: +81-7749-5-1411 (Each author)

Abstract

New learning schemes using *feedback-error-learning* for a neural network model applied to adaptive nonlinear feedback control are presented here. *Feedback-error-learning* was proposed as a learning method to acquire a feedforward controller which uses the output of a feedback controller as the error for training a neural network model. Using new schemes for nonlinear feedback control, the actual responses after learning correspond to the desired responses which are obtained by an inverse reference model implemented as the conventional feedback controller. In this respect, these methods are similar to Model Reference Adaptive Control (MRAC) applied to linear or linearized systems. It is shown that *learning impedance control* is derived when proposed schemes are used in Cartesian space. Convergence properties of the neural networks employed in these learning schemes are provided by an averaged equation and the Liapunov function method. Some results of simulating these learning schemes by using an inverted pendulum and a 2-link manipulator are also presented.

Key words:
Neural Network, Adaptive Control, Reference Model

Acknowledgment
We would like to thank Drs. E. Yodogawa and K. Nakane of ATR Auditory and Visual Perception Research Laboratories for their continuing encouragement.

Note
A part of this work has been presented at the Conference on Decision and Control 1990, which is entitled "Learning Control for a Closed Loop System using Feedback-Error-Learning".

Contents

1. Introduction

Adaptive control theory has been applied to nonlinear robot control as a method of adjusting linear parameters. Dubowsky [1] applied Model Reference Adaptive Control (MRAC) to a robot manipulator control using the local linearization technique. Craig et al.[2] and Slotine & Li [3] proposed an adaptive control method based on the Liapunov function for a manipulator whose nonlinear characteristics were known in advance.

In previous studies of adaptive learning control using a neural network model, Barto [4], Jordan [5] and Psaltis [6] addressed the problem of how to obtain the error signal for a neural network controller. In supervised learning [7], the difference between the desired response and the actual response cannot directly be used as the error for controller adaptation. The error for controller adaptation should not be trajectory error (plant performance error) but command error. Thus, Jordan proposed *forward-inverse-modeling* [5] and Albus [8], Miller [9], Atkeson [10], Psaltis [6], Kuperstein [11] and Martinets [12] used *direct-inverse-modeling* to obtain command error. In *reinforcement learning* [4], it may be possible to improve plant performance over time by means of on-line learning methods in less structured situations. As an example of *reinforcement learning*, Barto proposed using ASE and ACE techniques [4].

Kawato et al.[13] proposed a learning method to acquire a feedforward controller, which uses the output of a feedback controller as the error for training a neural network model. They called this learning method *feedback-error-learning*. Using this method, the neural network model for feedforward control acquires an inverse dynamics model of a controlled object. They successfully applied the *feedback-error-learning* scheme to learning trajectory and force control of a PUMA robot [14][15], a kinematically redundant manipulator [16] and a manipulator driven by rubber actuators with dynamical redundancy [17][18][19]. This method was originally proposed as a model of voluntary movement learning in the cerebellum [20]. However, the adaptive feedback controller learning method using *feedback-error-learning* had not yet been examined.

In this paper, we propose learning schemes using *feedback-error-learning* for a neural network model applied to an adaptive nonlinear feedback controller. In these learning schemes, a conventional feedback controller (CFC) is provided both as an ordinary feedback controller to guarantee global asymptotic stability and as an inverse reference model of the

response of the controlled object. Convergence properties of the neural networks are shown in Sections 2.1 and 2.2 by using the averaged equation method and Liapunov's second method. If a CFC is prepared as a reference model of the response in Cartesian space, *impedance control* [21] is derived after learning as shown in Section 2.1. In section 3, we introduce simulated results of proposed learning schemes. We also pointed out the relationship of these learning schemes to the posture and locomotion adaptive control mechanism in the cerebellum in Section 4.

2. Adaptive nonlinear feedback controller

In this section, we propose two adaptive learning control schemes using the *feedback-error-learning* technique for neural network feedback controllers.

In the first learning scheme, the neural network feedback controller ultimately acquires an inverse dynamics model of the controlled object. Thus, we call this learning scheme Inverse Dynamics Model Learning (IDML). In the second learning scheme, the neural-network model is trained to become a nonlinear regulator to compensate for the nonlinearity of the controlled object (except for the inertia term) through learning. Accordingly, this is called Nonlinear Regulator Learning (NRL). These two learning schemes are described in detail below. Convergence properties of these two learning schemes are examined by using similar procedures utilized by Kawato [22] for adaptive feedforward control .

2.1 Inverse Dynamics Model Learning (IDML)

We explain the configuration of this learning system, and provide proof of the convergence property of this learning scheme. Subsequently, we show *learning impedance control* as an application of IDML in Cartesian space.

The components, a conventional feedback controller (CFC), a neural network applied to an adaptive nonlinear feedback controller (NNFC) and a controlled object, are connected as shown in Fig.1. The CFC is used both as an ordinary feedback controller to guarantee global asymptotic stability during the learning period, and as a reference model of the responses of the controlled object. The sum of the output of the CFC and the disturbance for a controlled object is fed to the NNFC as the error signal. This corresponds to the parameter modification

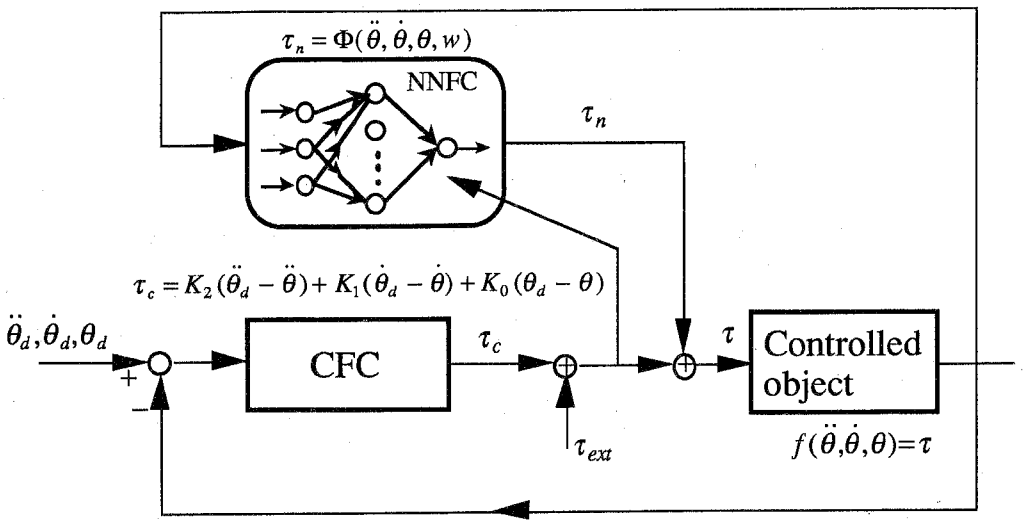


Fig. 1. Inverse Dynamics Model Learning.

inputs. The neural network feedback controller also receives $\theta, \dot{\theta}$ and $\ddot{\theta}$ as ordinary inputs. Here, $\theta, \dot{\theta}, \ddot{\theta}$ are the state vectors of the controlled object, its position, velocity, and acceleration respectively. As the neural network acquires the inverse dynamics model of the controlled object through learning, the responses of the controlled object follow the desired responses defined by the inverse reference model implemented as the CFC.

Following are the equations representing the dynamics of each component and this learning scheme for proof of convergence. The controlled object dynamics f is expressed as:

$$f(\ddot{\theta}, \dot{\theta}, \theta) = \tau \quad (1)$$

A linear controller are used as the CFC in the following explanation.

$$\tau_c = K_2(\ddot{\theta}_d - \ddot{\theta}) + K_1(\dot{\theta}_d - \dot{\theta}) + K_0(\theta_d - \theta) \quad (2)$$

NNFC is expressed as:

$$\tau_n = \Phi(\ddot{\theta}, \dot{\theta}, \theta, w) \quad (3)$$

Here θ is the actual position vector, θ_d is the desired position vector and w is the synaptic weight (adaptive parameter) of NNFC. Any type of neural network model, Multi Layer Perceptron (MLP) [23][24], Cerebellar Model Articulator Controller (CMAC) [8], associative

content addressable memory [10], and Memory Based Reasoning (MBR) [25] , can be used as an NNFC. The assumption about the neural network model here is that f is realized by Φ with an appropriate w. This is guaranteed for several neural network models (for example Irie & Miyake [24]). The equation of the system is

$$\tau = \tau_n + \tau_c + \tau_{ext}. \tag{4}$$

The difference between τ and τ_n is expressed as τ_{imag}.

$$\tau_{imag} = \tau - \tau_n = \tau_c + \tau_{ext} \tag{5}$$

Eq. 5 can be rewritten into the following equation using Eq. 2.

$$K_2(\ddot{\theta} - \ddot{\theta}_d) + K_1(\dot{\theta} - \dot{\theta}_d) + K_0(\theta - \theta_d) = \tau_{ext} - \tau_{imag} \tag{6}$$

The learning rule of the *feedback-error-learning* scheme is represented as:

$$\frac{dw}{dt} = \eta\left(\frac{\partial \Phi}{\partial w}\right)^T (\tau_c + \tau_{ext}) = \eta\left(\frac{\partial \Phi}{\partial w}\right)^T \tau_{imag}, \tag{7}$$

where η is a positive definite matrix which determines the learning rate.

Stochastic space is used for the discussion of the convergence properties of the NNFC below, because the controlled object dynamics is not specified (compare with [3] in which the nonlinearity structure of the controlled object dynamics is perfectly known). Geman's theorem [26] and Liapunov's second method are used under the following three assumptions.

Assumption 1

The learning rate, η, is very small and positive definite.

Assumption 2

The inputs, τ_{ext} and $\ddot{\theta}_d, \dot{\theta}_d, \theta_d$, are strongly mixing and strongly stationary stochastic processes.

Assumption 3

The CFC is designed to guarantee the asymptotic convergence of θ during learning when $\tau_{ext}, \theta_d, \dot{\theta}_d$ and $\ddot{\theta}_d$ equal to zero. Especially, $K_2\left(\frac{\partial f}{\partial \ddot{\theta}} - \frac{\partial \Phi}{\partial \ddot{\theta}} + K_2\right)^{-1}$ is positive definite.

Using the above assumptions, the system dynamics are represented in the following random differential equations instead of the deterministic equations, Eq.4 and Eq.7.

$$f(\ddot{\theta}(t,\omega),\dot{\theta}(t,\omega),\theta(t,\omega)) - \Phi(\ddot{\theta}(t,\omega),\dot{\theta}(t,\omega),\theta(t,\omega),w(t,\omega))$$
$$= K_2\left(\ddot{\theta}_d(t,\omega) - \ddot{\theta}(t,\omega)\right) + K_1\left(\dot{\theta}_d(t,\omega) - \dot{\theta}(t,\omega)\right) + K_0\left(\theta_d(t,\omega) - \theta(t,\omega)\right) + \tau_{ext}(t,\omega), \tag{8}$$

$$\frac{dw(t,\omega)}{dt} = \eta\left(\frac{\partial\Phi(\ddot{\theta}(t,\omega),\dot{\theta}(t,\omega),\theta(t,\omega),w(t,\omega))}{\partial w}\right)^T$$
$$\times\left\{\tau_c(\ddot{\theta}(t,\omega),\dot{\theta}(t,\omega),\theta(t,\omega),\ddot{\theta}_d(t,\omega),\dot{\theta}_d(t,\omega),\theta_d(t,\omega)) + \tau_{ext}(t,\omega)\right\}. \tag{9}$$

Here, ω is a sample point in a probability space. The solution of the following averaged equation is a good approximation to the solution of Eq.9 for small η, as proven by Geman's theorem [26].

$$\frac{dM}{dt} = \eta E\left[\left(\frac{\partial\Phi(\ddot{\theta}(t,\omega),\dot{\theta}(t,\omega),\theta(t,\omega),w(t,\omega))}{\partial w}\right)^T\right.$$
$$\left.\times\left\{\tau_c(\ddot{\theta}(t,\omega),\dot{\theta}(t,\omega),\theta(t,\omega),\ddot{\theta}_d(t,\omega),\dot{\theta}_d(t,\omega),\theta_d(t,\omega)) + \tau_{ext}(t,\omega)\right\}\right]_{w=M} \tag{10}$$

Thus, we consider the following function V as a Liapunov function candidate for the averaged equation (10) by using the following function L.

$$L = \frac{1}{2}\left\{\tau_c(\ddot{\theta}(t,\omega),\dot{\theta}(t,\omega),\theta(t,\omega),\ddot{\theta}_d(t,\omega),\dot{\theta}_d(t,\omega),\theta_d(t,\omega)) + \tau_{ext}(t,\omega)\right\}^T$$
$$\times\left\{\tau_c(\ddot{\theta}(t,\omega),\dot{\theta}(t,\omega),\theta(t,\omega),\ddot{\theta}_d(t,\omega),\dot{\theta}_d(t,\omega),\theta_d(t,\omega)) + \tau_{ext}(t,\omega)\right\} \tag{11}$$

$$V(M,t)$$
$$= E\left[L(\ddot{\theta}(t,\omega),\dot{\theta}(t,\omega),\theta(t,\omega),\ddot{\theta}_d(t,\omega),\dot{\theta}_d(t,\omega),\theta_d(t,\omega),\tau_{ext}(t,\omega))\right]_{w=M} \geq 0 \tag{12}$$

The time derivative of V is calculated as follows:

$$\frac{dV(M,t)}{dt}$$
$$= E\left[\left\{\tau_c(\ddot{\theta}(t,\omega),\dot{\theta}(t,\omega),\theta(t,\omega),\ddot{\theta}_d(t,\omega),\dot{\theta}_d(t,\omega),\theta_d(t,\omega)) + \tau_{ext}(t,\omega)\right\}^T\right.$$
$$\left.\times\frac{d}{dt}\left\{\tau_c(\ddot{\theta}(t,\omega),\dot{\theta}(t,\omega),\theta(t,\omega),\ddot{\theta}_d(t,\omega),\dot{\theta}_d(t,\omega),\theta_d(t,\omega)) + \tau_{ext}(t,\omega)\right\}\right]_{w=M} \tag{13}$$

In order to simplify the expression, we shorten each variable by omitting (t,ω).

$$\frac{dV}{dt} = E\left[\left(\tau_c + \tau_{ext}\right)^T \left\{ \frac{\partial \tau_c}{\partial \ddot{\theta}} \frac{d\ddot{\theta}}{dt} + \frac{\partial \tau_c}{\partial \dot{\theta}} \frac{d\dot{\theta}}{dt} + \frac{\partial \tau_c}{\partial \theta} \frac{d\theta}{dt} + \frac{\partial \tau_c}{\partial \ddot{\theta}_d} \frac{d\ddot{\theta}_d}{dt} + \frac{\partial \tau_c}{\partial \dot{\theta}_d} \frac{d\dot{\theta}_d}{dt} + \frac{\partial \tau_c}{\partial \theta_d} \frac{d\theta_d}{dt} + \frac{d\tau_{ext}}{dt} \right\} \right]_{w=M} \quad (14)$$

Eq.8 can be expressed in explicit form with respect to $\ddot{\theta}$ as follows:

$$\ddot{\theta}(t,\omega) = h(\dot{\theta}(t,\omega), \theta(t,\omega), w(t,\omega), \ddot{\theta}_d(t,\omega), \dot{\theta}_d(t,\omega), \theta_d(t,\omega), \tau_{ext}(t,\omega)) \quad (15)$$

By differentiating Eq.15 with respect to t, and then substituting it for $d\ddot{\theta}/dt$ in Eq.14, the following equation is obtained.

$$\frac{dV}{dt} = E\left[\left(\tau_c + \tau_{ext}\right)^T \left\{ -K_2 \left(\frac{\partial \ddot{\theta}}{\partial w} \frac{dw}{dt} + \frac{\partial \ddot{\theta}}{\partial \dot{\theta}} \frac{d\dot{\theta}}{dt} + \frac{\partial \ddot{\theta}}{\partial \theta} \frac{d\theta}{dt} + \frac{\partial \ddot{\theta}}{\partial \ddot{\theta}_d} \frac{d\ddot{\theta}_d}{dt} \right.\right.\right.$$
$$\left.\left.\left. + \frac{\partial \ddot{\theta}}{\partial \dot{\theta}_d} \frac{d\dot{\theta}_d}{dt} + \frac{\partial \ddot{\theta}}{\partial \theta_d} \frac{d\theta_d}{dt} + \frac{\partial \ddot{\theta}}{\partial \tau_{ext}} \frac{d\tau_{ext}}{dt} \right) - K_1 \frac{d\dot{\theta}}{dt} - K_0 \frac{d\theta}{dt} + K_2 \frac{d\ddot{\theta}_d}{dt} + K_1 \frac{d\dot{\theta}_d}{dt} + K_0 \frac{d\theta_d}{dt} + \frac{d\tau_{ext}}{dt} \right\} \right]_{w=M} \quad (16)$$

We sort the above terms into two parts with respect to w and θ in order to eliminate zero terms by averaging.

$$\frac{dV}{dt} = E\left[\left(\tau_c + \tau_{ext}\right)^T \left\{ -K_2 \frac{\partial \ddot{\theta}}{\partial w} \right\} \frac{dw}{dt} \right]_{w=M}$$
$$+ E\left[\left(\tau_c + \tau_{ext}\right)^T \left\{ -K_2 \left(\frac{\partial \ddot{\theta}}{\partial \dot{\theta}} \frac{d\dot{\theta}}{dt} + \frac{\partial \ddot{\theta}}{\partial \theta} \frac{d\theta}{dt} + \frac{\partial \ddot{\theta}}{\partial \ddot{\theta}_d} \frac{d\ddot{\theta}_d}{dt} + \frac{\partial \ddot{\theta}}{\partial \dot{\theta}_d} \frac{d\dot{\theta}_d}{dt} + \frac{\partial \ddot{\theta}}{\partial \theta_d} \frac{d\theta_d}{dt} + \frac{\partial \ddot{\theta}}{\partial \tau_{ext}} \frac{d\tau_{ext}}{dt} \right) \right.\right.$$
$$\left.\left. - K_1 \frac{d\dot{\theta}}{dt} - K_0 \frac{d\theta}{dt} + K_2 \frac{d\ddot{\theta}_d}{dt} + K_1 \frac{d\dot{\theta}_d}{dt} + K_0 \frac{d\theta_d}{dt} + \frac{d\tau_{ext}}{dt} \right\} \right]_{w=M} \quad (17)$$

Then, the second term, $E[\cdot]$, on the right in Eq.17 vanishes, if $\theta, \dot{\theta}, \ddot{\theta}, \theta_d, \dot{\theta}_d, \ddot{\theta}_d, \tau_{ext}$ are strongly stationary stochastic processes. Because:

The second term of Eq.17 is the same with dV/dt when M is constant, as shown in following equation.

$$\frac{dV_{M\,const}}{dt} = E\left[\left(\tau_c + \tau_{ext}\right)^T \left\{ -K_2 \left(\frac{\partial \ddot{\theta}}{\partial \dot{\theta}} \frac{d\dot{\theta}}{dt} + \frac{\partial \ddot{\theta}}{\partial \theta} \frac{d\theta}{dt} + \frac{\partial \ddot{\theta}}{\partial \ddot{\theta}_d} \frac{d\ddot{\theta}_d}{dt} \right.\right.\right.$$
$$\left.\left.\left. + \frac{\partial \ddot{\theta}}{\partial \dot{\theta}_d} \frac{d\dot{\theta}_d}{dt} + \frac{\partial \ddot{\theta}}{\partial \theta_d} \frac{d\theta_d}{dt} + \frac{\partial \ddot{\theta}}{\partial \tau_{ext}} \frac{d\tau_{ext}}{dt} \right) - K_1 \frac{d\dot{\theta}}{dt} - K_0 \frac{d\theta}{dt} + K_2 \frac{d\ddot{\theta}_d}{dt} + K_1 \frac{d\dot{\theta}_d}{dt} + K_0 \frac{d\theta_d}{dt} + \frac{d\tau_{ext}}{dt} \right\} \right] \quad (18)$$

Here, we use the proposition that the expectation of any Baire function of a strongly stationary stochastic process does not depend on time (see for example, Ito [27]). Because of assumption 3, $\ddot{\theta}, \dot{\theta}, \theta$ and L become Baire functions of the strongly stationary processes,

$\ddot{\theta}_d, \dot{\theta}_d, \theta_d$ and τ_{ext}. Consequently, $V_{M\,const}$ does not depend on time, and $dV_{M\,const}/dt$ is equal to zero. Thus, the second term on the right in Eq.17 vanishes.

Hence, we obtain:

$$\frac{dV}{dt} = E\left[\left(\tau_c + \tau_{ext}\right)^T \left\{ -K_2 \frac{\partial \ddot{\theta}}{\partial w} \right\} \frac{dw}{dt} \right]_{w=M} . \tag{19}$$

We calculate the partial derivative of Eq.15 with respect to w while referring to Eq.8 in order to replace $\partial\ddot{\theta}/\partial w$ in Eq.19.

$$\frac{\partial f}{\partial \ddot{\theta}} \frac{\partial \ddot{\theta}}{\partial w} - \frac{\partial \Phi}{\partial \ddot{\theta}} \frac{\partial \ddot{\theta}}{\partial w} - \frac{\partial \Phi}{\partial w} = -K_2 \frac{\partial \ddot{\theta}}{\partial w}$$

$$\Leftrightarrow \frac{\partial \ddot{\theta}}{\partial w} = \left(\frac{\partial f}{\partial \ddot{\theta}} - \frac{\partial \Phi}{\partial \ddot{\theta}} + K_2 \right)^{-1} \frac{\partial \Phi}{\partial w} \tag{20}$$

Then, Eq.19 is expressed as:

$$\frac{dV}{dt} = E\left[\left(\tau_c + \tau_{ext}\right)^T \left\{ -K_2 \left(\frac{\partial f}{\partial \ddot{\theta}} - \frac{\partial \Phi}{\partial \ddot{\theta}} + K_2 \right)^{-1} \frac{\partial \Phi}{\partial w} \right\} \frac{dw}{dt} \right]_{w=M}$$

$$= E\left[-\left(\tau_c + \tau_{ext}\right)^T K_2 \left(\frac{\partial f}{\partial \ddot{\theta}} - \frac{\partial \Phi}{\partial \ddot{\theta}} + K_2 \right)^{-1} \frac{\partial \Phi}{\partial w} \eta \left(\frac{\partial \Phi}{\partial w} \right)^T \left(\tau_c + \tau_{ext}\right) \right]_{w=M} . \tag{21}$$

Because of the assumption 3, the following matrix is positive definite.

$$K_2 \left(\frac{\partial f}{\partial \ddot{\theta}} - \frac{\partial \Phi}{\partial \ddot{\theta}} + K_2 \right)^{-1} > 0 \tag{22}$$

Therefore, the middle part on the right in Eq.21 is positive semi-definite as follows:

$$K_2 \left(\frac{\partial f}{\partial \ddot{\theta}} - \frac{\partial \Phi}{\partial \ddot{\theta}} + K_2 \right)^{-1} \frac{\partial \Phi}{\partial w} \eta \left(\frac{\partial \Phi}{\partial w} \right)^T \geq 0 \tag{23}$$

Thus, we obtain:

$$\frac{dV}{dt} \leq 0 . \tag{24}$$

The equalities of Eq.12 and Eq.24 hold only when $w = \hat{w}$ for non trivial θ except for the local minimum condition, $\partial\Phi/\partial w = 0$. Consequently, we conclude that w asymptotically converges to the optimal set of synaptic weights, $\hat{w}$, in the sense of quadratic mean convergence if the conditions of local minimum are avoided through learning. By using an annealing method for changing weight value, like [28], the solution of Eq.9 will converge to the global minimum. Then, the response of the controlled object is governed by:

$$K_2\ddot{\xi} + K_1\dot{\xi} + K_0\xi = \tau_{ext} \tag{25}$$

where $\xi = \theta - \theta_d$.

That is to say, trajectory error, $\theta - \theta_d$, converges to zero following the reference response represented by Eq.25 when $\tau_{ext} = 0$. When the disturbance, τ_{ext}, is served, trajectory error is in accordance with Eq.25.

The above describes the learning scheme in joint space. This can also be applied in Cartesian space. By using this learning scheme in Cartesian space, *impedance control* proposed by Hogan [21] can be realized. Thus, we call this learning scheme applied in Cartesian space *learning impedance control*. Next, *learning impedance control* is explained briefly as an application of the IDML. The manipulator dynamics can be expressed as:

$$R(\theta)\ddot{\theta} + N(\theta, \dot{\theta}) = \tau. \tag{26}$$

The CFC is represented in the following equation by using the Cartesian coordinate.

$$F_c = M\ddot{x}_e + B\dot{x}_e + Kx_e \tag{27}$$

where $x_e = x_d - x$.

Eq.27 is the reference model of the manipulator responses. *M, B, K* respectively represent virtual mass, virtual viscosity and virtual stiffness of the manipulator performed by *impedance control*. The function Φ of the neural network is the same equation as Eq.3. The total system is designed as shown in Fig. 2. By using the same procedure as shown in the former proof, we obtain the convergence property of this learning. Consequently, the total feedback compensation, $\tau_n + \tau_c$, becomes the following equation after learning.

$$\tau_n + \tau_c = R(\theta)\ddot{\theta} + N(\theta, \dot{\theta}) - J^T\left(M(\ddot{x} - \ddot{x}_d) + B(\dot{x} - \dot{x}_d) + K(x - x_d)\right). \tag{28}$$

Hence, the response of the end-point of the manipulator is governed by:

$$M(\ddot{x} - \ddot{x}_d) + B(\dot{x} - \dot{x}_d) + K(x - x_d) = F_{ext}. \tag{29}$$

Thus, this learning method derives *impedance control* for the manipulator, which has unknown nonlinear characteristics, when learning is accomplished. Moreover, it is remarkable that we can change the virtual impedance of the manipulator into any value simply by changing the parameters *M,B,K* of the CFC without re-learning, because the NNFC obtains only the inverse dynamics model of the controlled object through this learning.

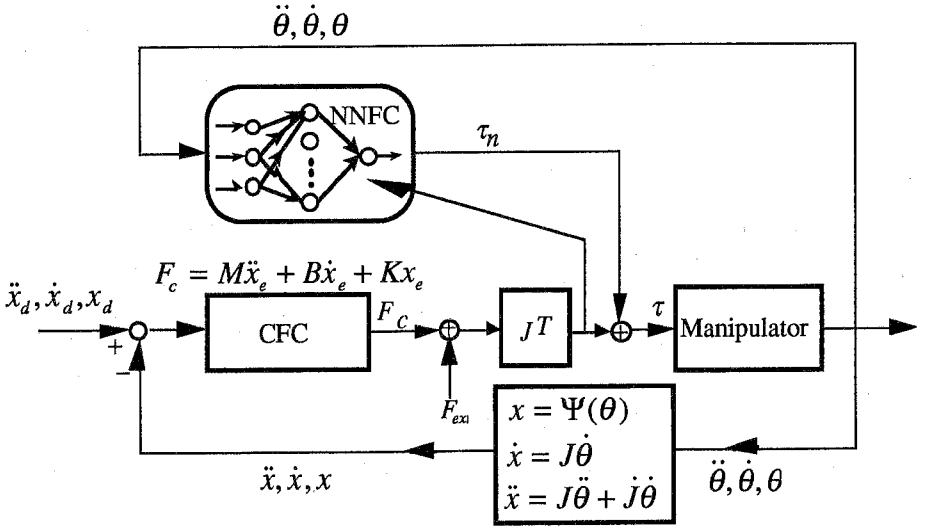


Fig.2. Learning Impedance Control.

2.2 Nonlinear Regulator Learning (NRL)

In this learning scheme, the actual acceleration is not used as the input of the NNFC. Instead, we feed the desired trajectory (position, velocity, and acceleration) to the NNFC in order to obtain reference responses. We explain here the configuration of this learning scheme and the consequence after learning, and then show the convergence property briefly.

The dynamics of the controlled object is represented here as:

$$R(\theta)\ddot{\theta} + N(\theta,\dot{\theta}) = \tau \tag{30}$$

R is an inertia matrix which has nonlinearity. N is another term of the controlled object dynamics. As with the above method, IDML, the CFC serves the same two purposes (see 2.1). We set only the output of the CFC, τ_c, as the error signal (i.e. adaptive modification input) for the NNFC as shown in Fig.3. In this learning scheme, disturbance is considered to be absent in the learning period.

In this case, we feed the desired values, $\ddot{\theta}_d,\dot{\theta}_d,\theta_d$, and the outputs of controlled object , $\theta,\dot{\theta}$, to the NNFC as ordinary inputs. Hence, the function of the NNFC is

$$\tau_n = \Phi\left(\ddot{\theta}_d,\dot{\theta}_d,\theta_d,\dot{\theta},\theta,w\right) \tag{31}$$

The total dynamics is given by the next equation.

$$\tau = \tau_c + \tau_n$$

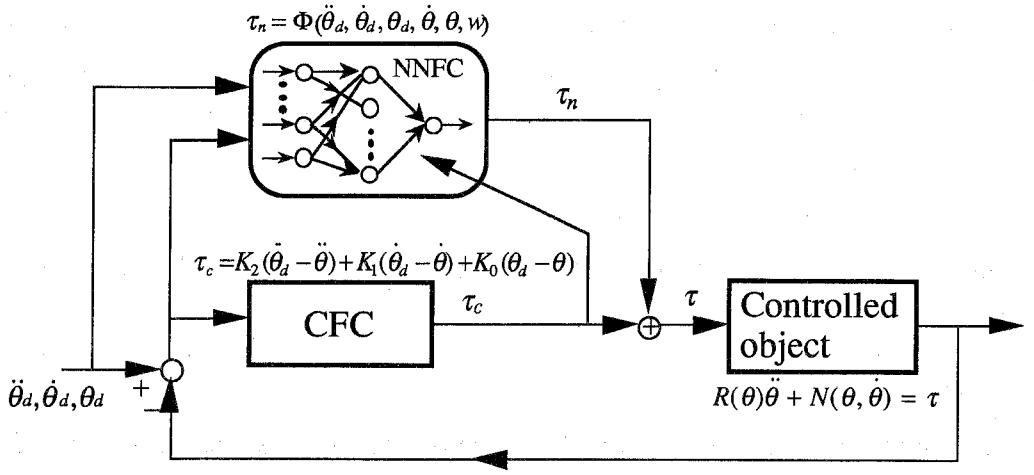


Fig. 3. Nonlinear Regulator Learning.

$$\Leftrightarrow \left(R(\theta)+K_2\right)(\ddot{\theta}_d-\ddot{\theta})+K_1(\dot{\theta}_d-\dot{\theta})+K_0(\theta_d-\theta)+\Phi-N(\theta,\dot{\theta})-R(\theta)\ddot{\theta}_d=0 \tag{32}$$

If Φ is represented as:

$$\Phi_d = N(\theta,\dot{\theta})+R(\theta)\ddot{\theta}_d+R(\theta)K_2^{-1}\left(K_1(\dot{\theta}_d-\dot{\theta})+K_0(\theta_d-\theta)\right), \tag{33}$$

Eq.32 is expressed as:

$$\left(R(\theta)+K_2\right)(\ddot{\theta}_d-\ddot{\theta})+K_1(\dot{\theta}_d-\dot{\theta})+K_0(\theta_d-\theta)+R(\theta)K_2^{-1}\left(K_1(\dot{\theta}_d-\dot{\theta})+K_0(\theta_d-\theta)\right)=0. \tag{34}$$

Then, the next equation is obtained.

$$\left(I+R(\theta)K_2^{-1}\right)\left(K_2(\ddot{\theta}_d-\ddot{\theta})+K_1(\dot{\theta}_d-\dot{\theta})+K_0(\theta_d-\theta)\right)=0 \tag{35}$$

Consequently, this equation yields responses without disturbance (external force) of the controlled objects on the condition, $I+R(\theta)K_2^{-1}\neq 0$, as:

$$K_2\ddot{\xi}+K_1\dot{\xi}+K_0\xi=0. \tag{36}$$

Here $\xi=\theta-\theta_d$.

Roughly speaking, to obtain the reference response represented by Eq.36 during free movement by compensating for the nonlinearity (except for the inertia term) of the controlled object, the neural network model is trained to become a nonlinear regulator.

Next, the convergence properties of the neural network in this learning scheme will be briefly explained. The difference between the actual output and the desired output of the NNFC, Φ_d, is defined as P. P is represented in the next equation by using Eqs. 30, 32 and 33.

$$P = \Phi - \Phi_d = -\left(I + R(\theta)K_2^{-1}\right)\tau_c \qquad I: \text{ unit matrix} \tag{37}$$

We consider the following function J as a Liapunov function candidate of this learning scheme.

$$J = E\left[\frac{1}{2}P^T P\right]_{w=M} \geq 0 \tag{38}$$

With the same procedure used to prove the IDML convergence properties, the following equation is obtained.

$$\frac{dJ}{dt} \leq 0 \tag{39}$$

Consequently, the function Φ of the neural network acquires nonlinear compensation represented in Eq.32 after learning.

As with the IDML method, we can also apply this learning scheme in Cartesian space. However, perfect *impedance control* cannot be derived by using the NRL method because the NNFC does not have the feedback acceleration signal, $\ddot{\theta}$. Therefore, it is impossible that the inertia term of virtual impedance is made smaller than the inertia term of inherent impedance of the controlled object by using the NRL method in Cartesian space.

3. Simulation results

In this section, we show the simulation results of the learning schemes described in the previous section applied to some control problems. In the first experiment, an inverted pendulum is used as a controlled object for IDML and NRL in joint space. In the second experiment, a two-link manipulator is used as a controlled object for IDML in Cartesian space (i.e., learning impedance control).

3.1 Inverted Pendulum Controlled by IDML

The dynamics of the controlled object, an inverted pendulum, is expressed as:

$$-(J+ml^2)\ddot{\theta}-\left(\frac{2}{1+\exp(-a\dot{\theta})}-1\right)+mgl\sin\theta=ml\tau\cos\theta. \tag{40}$$

here m : pendulum mass l : pendulum length

J : pendulum inertia θ : pendulum angle

τ: input acceleration for the cart g : acceleration of gravity.

The second term on the left side of Eq.40 corresponds to Coulomb's friction. This is a typical regulator problem because the desired values $\ddot{\theta}_d, \dot{\theta}_d, \theta_d$ are always zero.

The following nonlinear functions, p_i, are used as the input of the two-layered neural network in the NNFC. This is because the learning problem could be made considerably easier by using known nonlinearity as shown by Khosla & Kanade[29], Atkeson et al.[30], Craig et al.[2] and Kawato et al.[13][14].

$$\begin{aligned} p_0 &= \tan\theta \\ p_1 &= \left(2/\left(1+\exp(-a\dot{\theta})\right)-1\right)/\cos\theta \\ p_2 &= \ddot{\theta}/\cos\theta \end{aligned} \tag{41}$$

The output of NNFC, τ_n, is

$$\tau_n=\sum_{k=0}^{2} w_k p_k \,. \tag{42}$$

The optimal values of the weights are represented as follows for the controlled object expressed as Eq.40.

$$\hat{w}_0=g,\quad \hat{w}_1=-\frac{1}{ml},\quad \hat{w}_2=-\frac{J+ml^2}{ml}. \tag{43}$$

Using these prepared nonlinearities and a linear parameter adaptation technique, the function, $(\partial\Phi/\partial w)(\partial\Phi/\partial w)^T$, is positive definite except when the condition, $\theta=\dot{\theta}=\ddot{\theta}=0$, is satisfied. Thus, the Liapunov function V is strictly concave because τ_{imag} is not zero except for the condition, $\theta=\dot{\theta}=\ddot{\theta}=0$. Hence, in this case, it is guaranteed that the optimal weights are obtained by the *feedback-error-learning* rule.

For the above conditions, the time courses of weight values during IDML are shown in Fig.4. We used the Ornstein-Uhlenbeck Process as the input disturbance, τ_{ext}, in this learning period. $\hat{w}_2, \hat{w}_1, \hat{w}_0$ are the desired values for the actual values, w_2, w_1, w_0, respectively. Each actual value gradually approached each desired value. Fig.5 shows the moving average of the square of the NNFC output and the error for the NNFC during the IDML period. The NNFC

output increased to compensate for the controlled object dynamics. Such adaptation decreased the error for the NNFC.

The responses of the pendulum to each condition of the input disturbance are compared in Fig.6. The response after learning corresponded perfectly to the desired response:

$$K_2(\ddot{\theta}-\ddot{\theta}_d)+K_1(\dot{\theta}-\dot{\theta}_d)+K_0(\theta-\theta_d)=\tau_{ext}. \tag{44}$$

(In this case, $\ddot{\theta}_d=\dot{\theta}_d=\theta_d=0$, $K_2=1, K_1=3.5, K_0=20$)

The linear controller was designed to obtain the desired response for a linearized controlled object without the friction term:

$$-\left(J+ml^2\right)\ddot{\theta}+mgl\theta=ml\tau. \tag{45}$$

The response driven only by the linear controller did not correspond to the desired response because of nonlinearity of the controlled object.

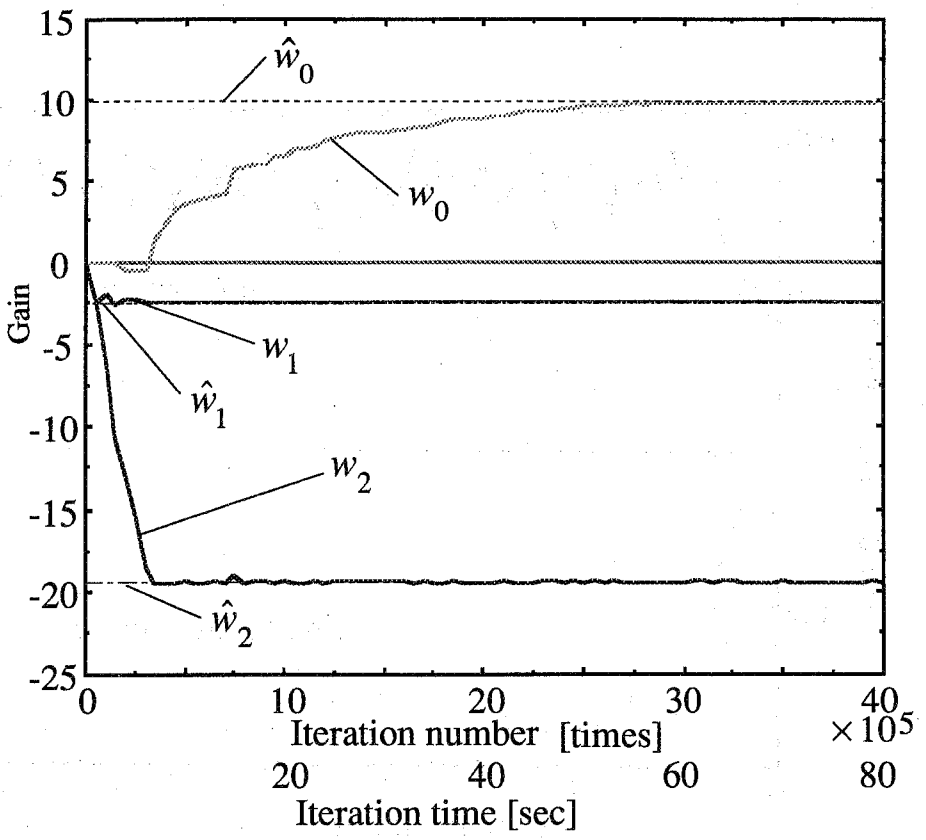


Fig. 4. Time courses of weights during inverse dynamics model learning.

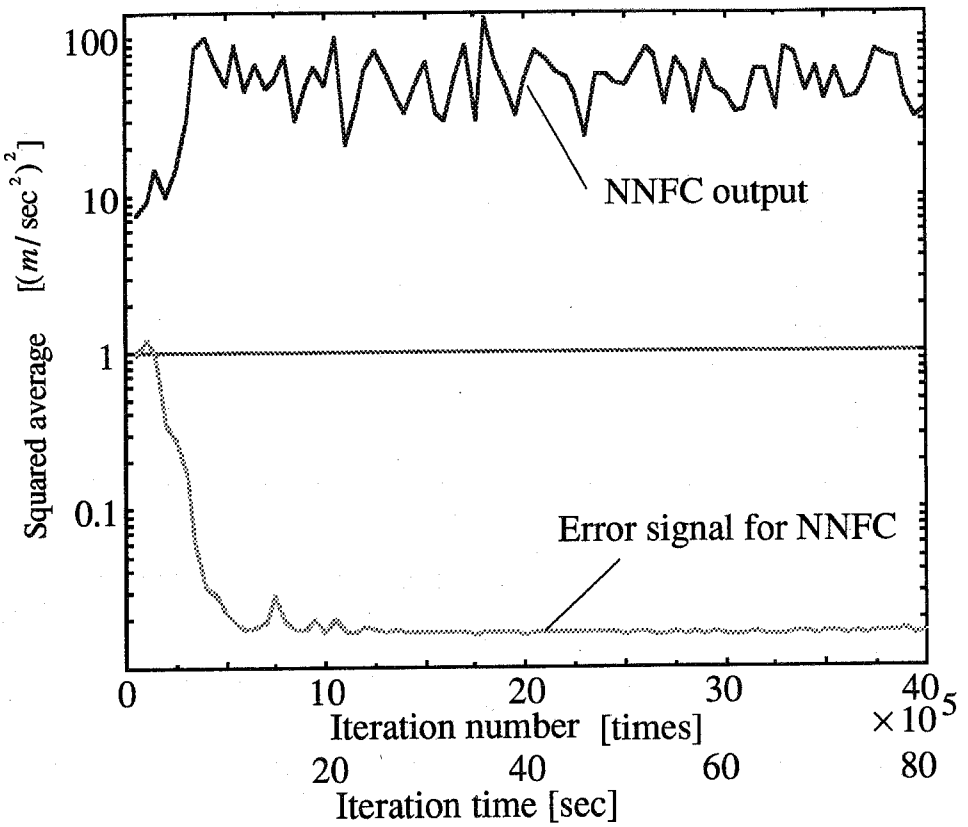


Fig. 5. Moving average of square of the NNFC output and the error for NNFC during inverse dynamics model learning.

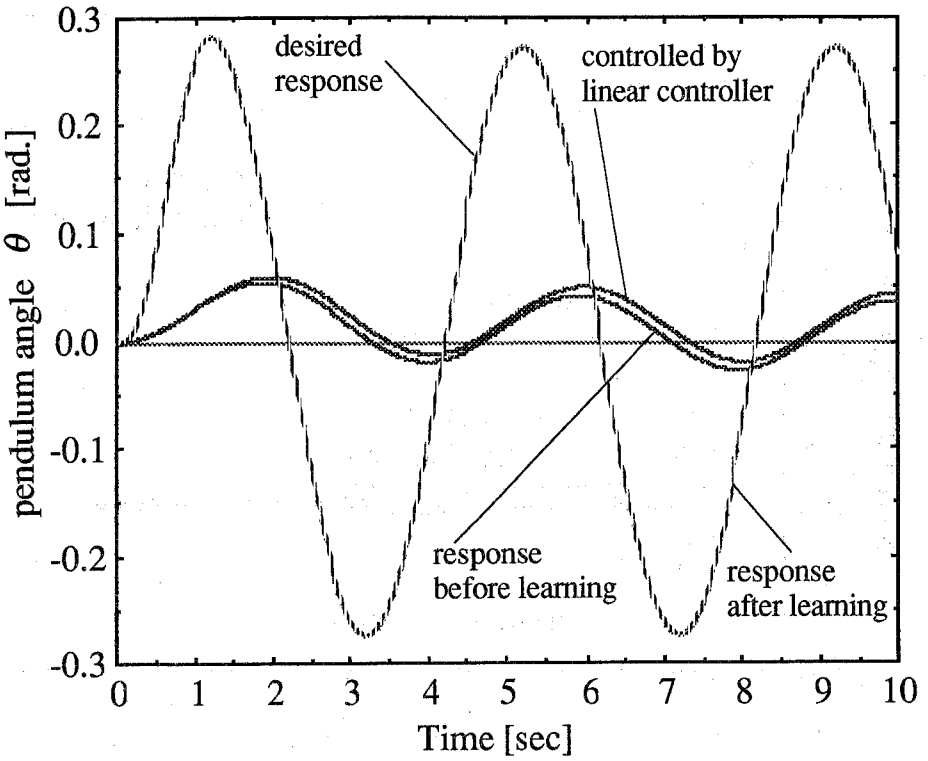


Fig. 6. Controlled object response for disturbance.

$$\tau_{ext} = -5\sin(\pi t / 2)[m / \sec^2]$$

3.2 Inverted Pendulum Controlled by NRL

We show here a result of the second learning scheme, NRL, using the same controlled object, an inverted pendulum. By using prepared nonlinearities, we have already confirmed that the actual weight values in this learning scheme approached the desired weight values. Accordingly, the following result was derived by using a 3-layered neural network to confirm the ability of this learning scheme.

Fig.7 shows the desired response, the response using the linear controller, the response before learning, and the response after learning. The linear controller was designed to perform the desired response for the controlled object expressed in Eq.45. However, we did not obtain the desired response by using a linear controller because of the nonlinearity and friction of the actual controlled object. The actual response after learning approached the

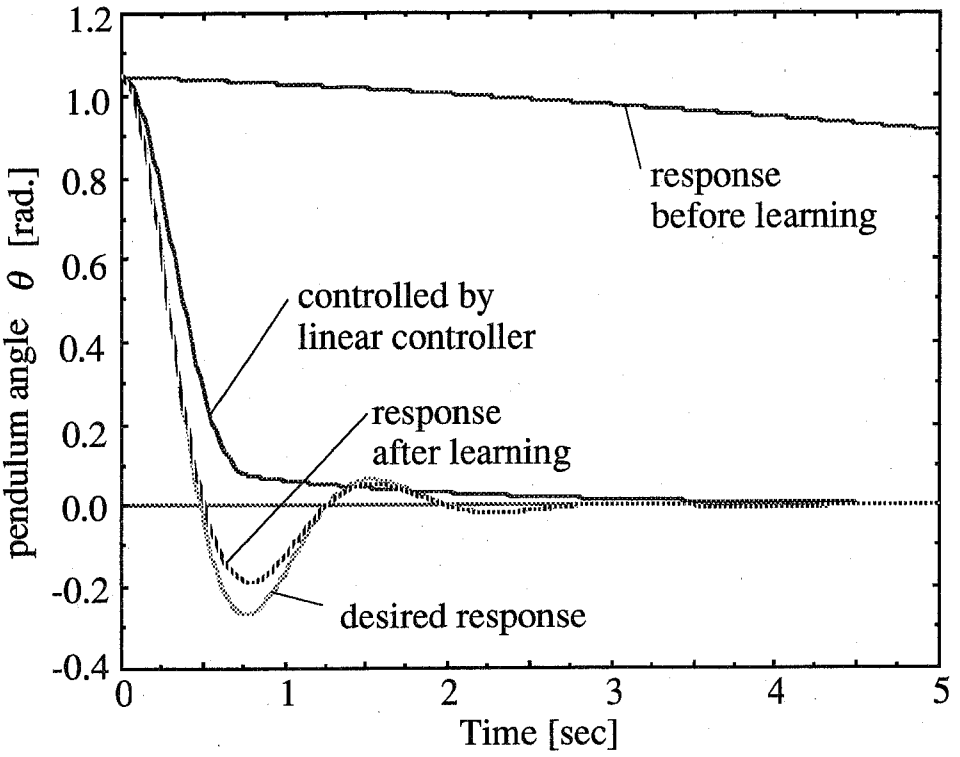


Fig. 7. Controlled object response from initial position of 60[deg].

$$\tau_c = -\left[1.0(\ddot{\theta} - \ddot{\theta}_d) + 3.5(\dot{\theta} - \dot{\theta}_d) + 20.0(\theta - \theta_d)\right]$$

desired response because the nonlinear adaptive controller compensated for the nonlinearity of the controlled object through learning.

3.3 Learning Impedance Control of a 2-link Manipulator

In this section, we show some results of *learning impedance control* using a 2-link manipulator on the horizontal plane. A 3-layered neural network model with 9 input units, 13 hidden units and 2 output units was used in this simulation.

To demonstrate the efficiency of *learning impedance control*, we compared the responses of the end-point of the manipulator to the external force input in each condition (before learning, after learning, and ideal response) in Fig.8. The virtual impedance which is decoupled along the x and y axis, is required for the manipulator. Then, the ideal response for

the step force input along the x or y axis, traces the straight path along each axis as shown in Fig.8. As shown in Fig.8, the responses after learning were closer to the ideal response than before learning. If an ideal inverse model is supplied as the NNFC, ideal responses are obtained. Thus, we can conclude that the NNFC acquired an approximate model of the inverse dynamics model in a work space which was explored during learning.

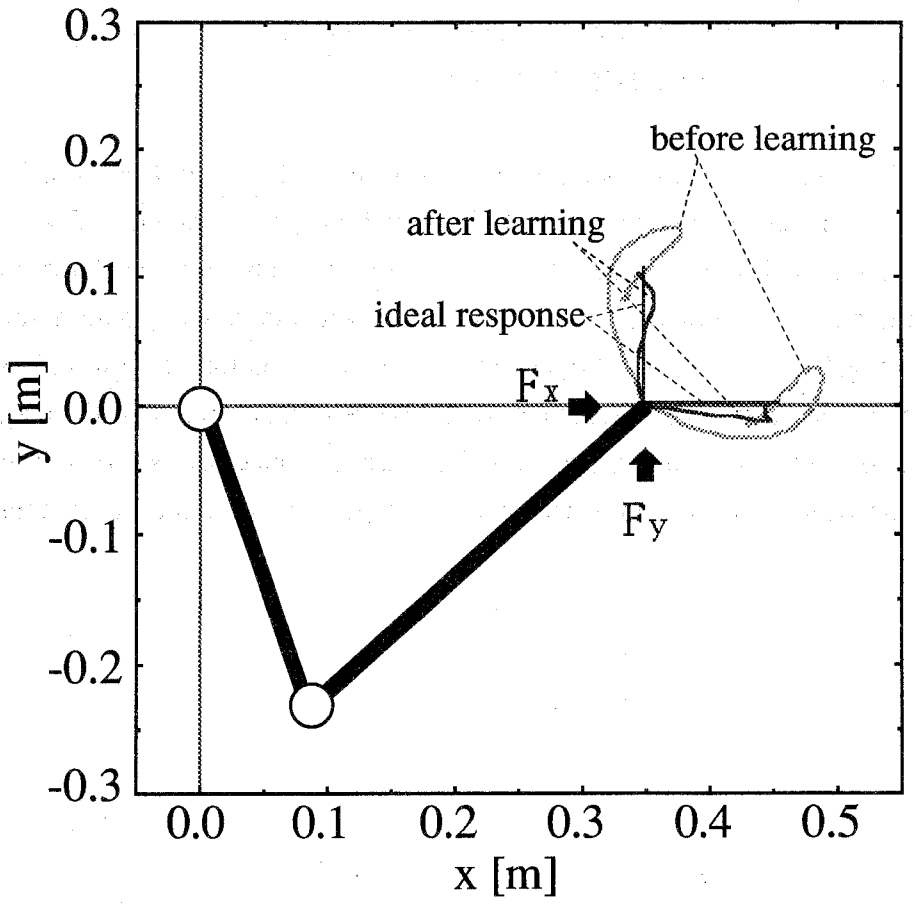


Fig. 8. Improved response for the force inputs at the end effector.

$$F_c = -\left[0.1(\ddot{x} - \ddot{x}_d) + 2.0(\dot{x} - \dot{x}_d) + 20.0(x - x_d)\right]$$

Next, we examined the responses of the contact task for the wall (which is modeled as a strong spring). The parameters of the CFC are shown below in each condition.

$F_c = -\left[M\ddot{x} + B\dot{x} + K\left(x - x_d\right)\right]$ $\quad x = [\mathrm{x}, \mathrm{y}]^T$

Before learning M=*diag*[0.1, 0.1][kg]
B=*diag*[7.0, 7.0][N/(m/sec)]
K=*diag*[100, 100][N/m]

After learning M=*diag*[0.1, 0.1][kg]
B=*diag*[7.0, 7.0][N/(m/sec)] (noncontact phase)
B=*diag*[700, 7.0][N/(m/sec)](contact phase)
K=*diag*[100, 100][N/m]

Before learning, the actual trajectory differed from the desired trajectory in free movement, and the performance of the contact task was much poorer as shown in Fig.9 and 10. After learning, the actual trajectory in free movement was much better as shown in Fig.11. The impact force at the time when collision occurred, was much lower than before learning as shown in Fig.12, because virtual impedance of the manipulator was lowered. Moreover, the contact task was so smooth that the external force was proportional to the magnitude of the difference between the actual trajectory and the desired trajectory as shown in Fig.12. This is because the virtual impedance was changed to perform a stable contact task in this phase.

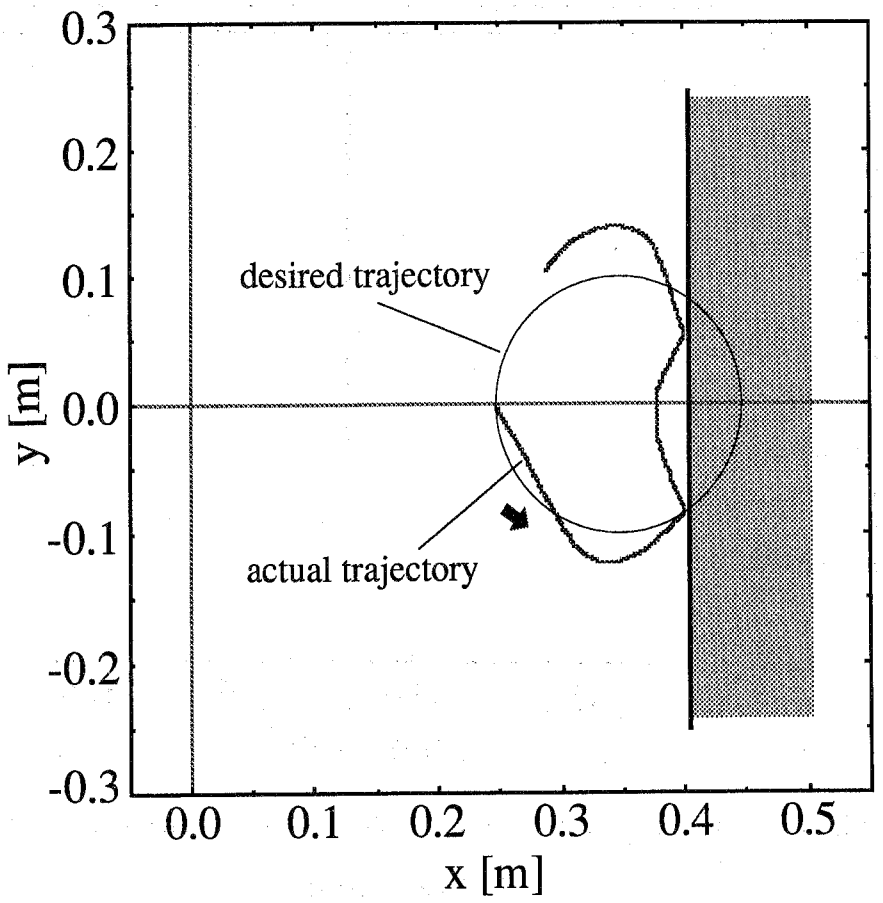


Fig. 9. Trajectory of free movement & contact task. **before learning**

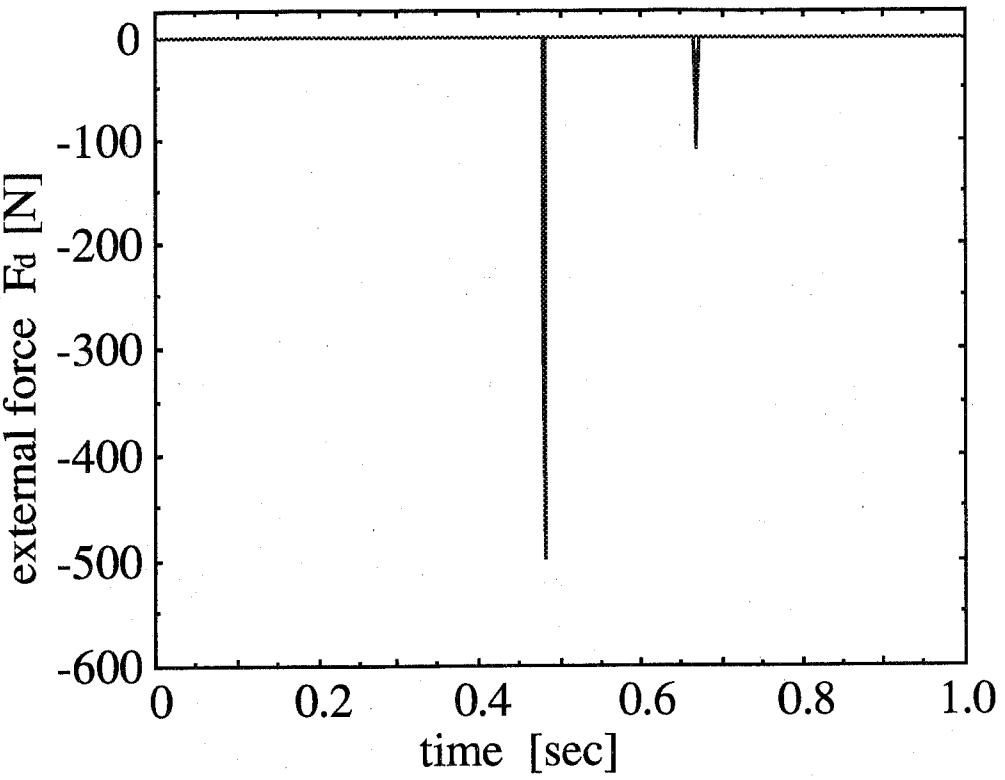


Fig. 10. External force at the end effector of free movement & contact task.

before learning

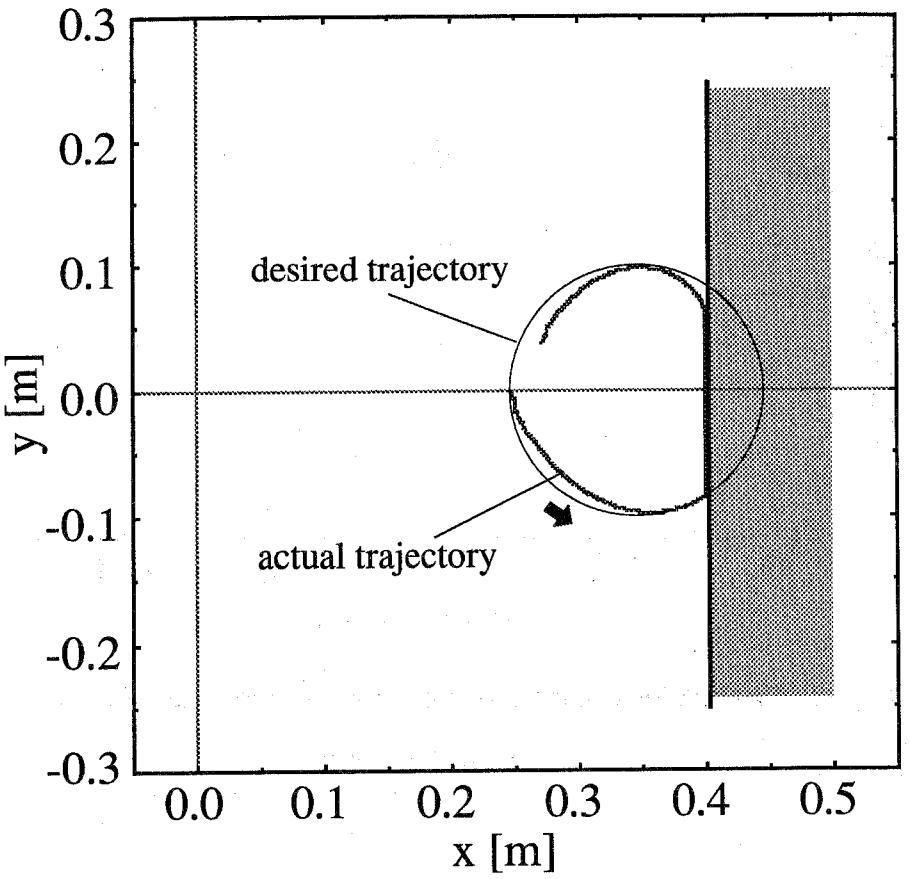


Fig. 11. Trajectory of free movement & contact task. **after learning**

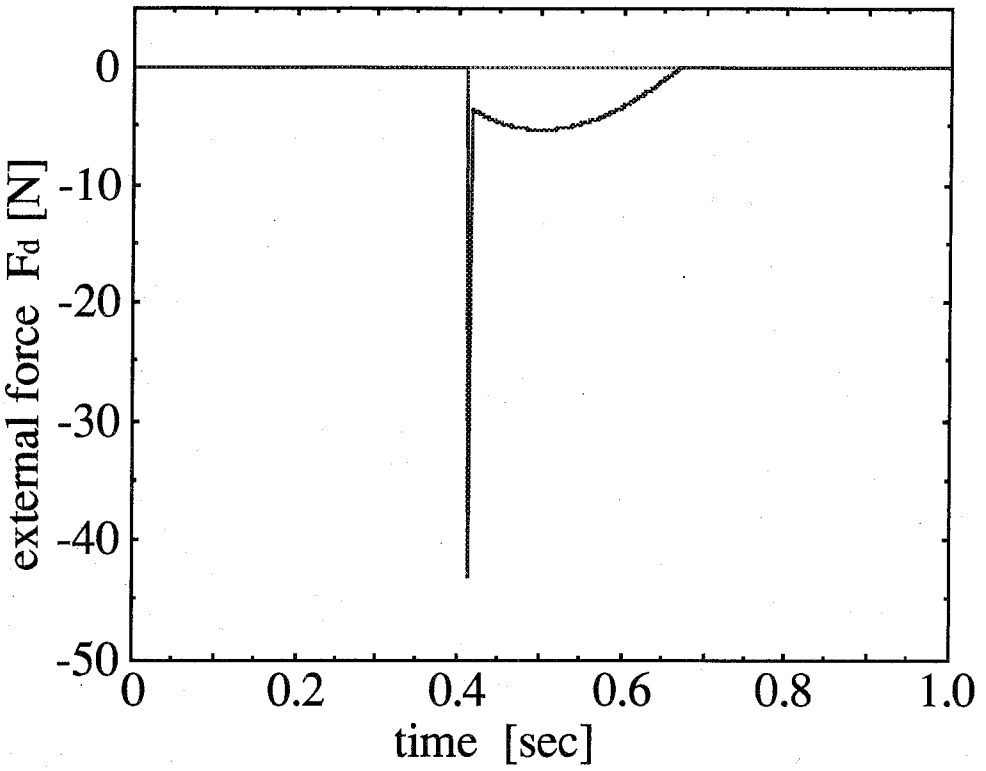


Fig. 12. External force at the end effector of free movement & contact task.

after learning

4 Discussion

4.1 Difference between two learning schemes, IDML and NRL

Comparing the results of IDML and NRL, the same linear responses during free movement ($\tau_{ext} = 0$) can be obtained after learning as shown in Eq.25 and Eq.36. However, different responses are obtained for same disturbance (i.e., constrained movement) from each learning scheme. In other words, the NRL method cannot provide a reference response for the disturbance, τ_{ext}. This is because the dynamics of the controlled object cannot be fully compensated for by using feedback signals in the NRL scheme. However, because the

disturbance is not used as an error signal for training the NNFC in NRL, we need not measure or estimate the disturbance in the learning period. Therefore, the choice of learning schemes depends on the controlled object and practical usage.

4.2 Relation to previous works

(1) Adaptive control methods

As we noted in Section 1, previous adaptive control methods mainly use the local linearization technique [1] or a priori knowledge of the controlled object nonlinearity [2] [3] to regulate linear parameters. However, some nonlinear adaptive controllers, such as artificial neural networks, can be utilized for nonlinear problems directly. Having a sufficiently powerful training method in a work-space and preparing a sufficient capacity of a neural network make it unnecessary to know the exact structure of the controlled object nonlinearity in advance. In practice, however, it is difficult to realize these ideal conditions. Thus, it is better to only use neural network controllers for controlled objects that include unknown characteristics.

(2) Neural network controllers

Some robot control research using neural networks which employed the *direct inverse modeling* [5][7] method has been done. This research can be categorized into 2 areas: creating an inverse kinematics model [11][12] and creating an inverse dynamics model [6][9].

As we pointed out in the introduction, the fundamental problem of training a nonlinear adaptive controller such as a neural network is how to obtain the error signal for that controller. The *forward-inverse-modeling* method proposed by Jordan [5] uses the back-propagation method [23] through a forward dynamics model (direct dynamics model) to obtain the input-error (command error) from the output-error (trajectory error). Mathematically, this corresponds to a steepest descent method. On the other hand, Kawato [22] pointed out that the *feedback-error-learning* corresponds to a Newton-like method. These two methods, a steepest descent method and a Newton-like method, are well known numerical solutions. Thus, we can employ not only the *feedback-error-learning* method but also the *forward-inverse-modeling* method in the IDML and the NRL schemes as shown in Fig.13. In this case, the reference model of trajectory-error convergence is prepared explicitly as shown. Narendra and Parthasarathy [31] examined in detail neural network controllers that

use an input-output reference model and error-back-propagation through a forward dynamics model. Any appropriate reference model, such as the trajectory-error reference model or the input -output reference model, can be utilized to accomplish each purpose.

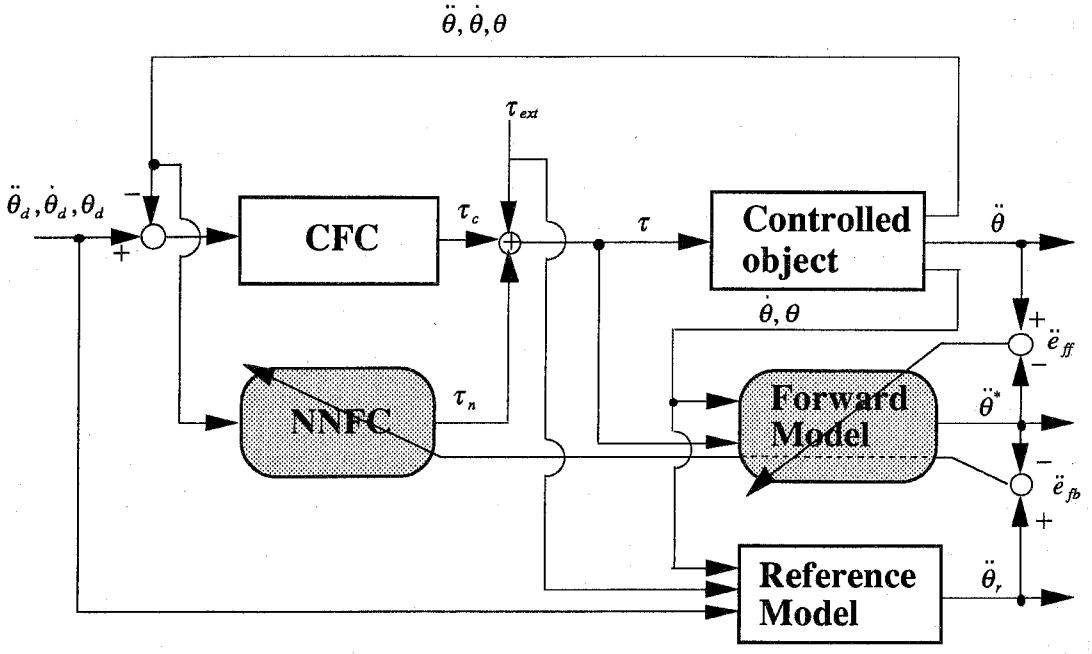


Fig. 13. IDML by Forward-modeling.

The error for NNFC is produced from the error, e_{fb}, by back-propagation through the forward model of a controlled object. Each component is expressed as follows.

- Controlled object dynamics is

$$R(\theta)\ddot{\theta} + N(\dot{\theta}, \theta) = \tau .$$

- Estimated forward model by neural network is

$$\ddot{\theta}^* = R^{*-1}\left(\tau - N^*(\dot{\theta}, \theta)\right) .$$

- Reference response is

$$\ddot{\theta}_r = K_2^{-1}\left(\tau_{ext} + K_1\left(\dot{\theta}_d - \dot{\theta}\right) + K_0\left(\theta_d - \theta\right)\right) + \ddot{\theta}_d .$$

- Conventional feedback controller output is

$$\tau_c = K_2\left(\ddot{\theta}_d - \ddot{\theta}\right) + K_1\left(\dot{\theta}_d - \dot{\theta}\right) + K_0\left(\theta_d - \theta\right) .$$

4.3 Adaptive motor control in the central nervous system

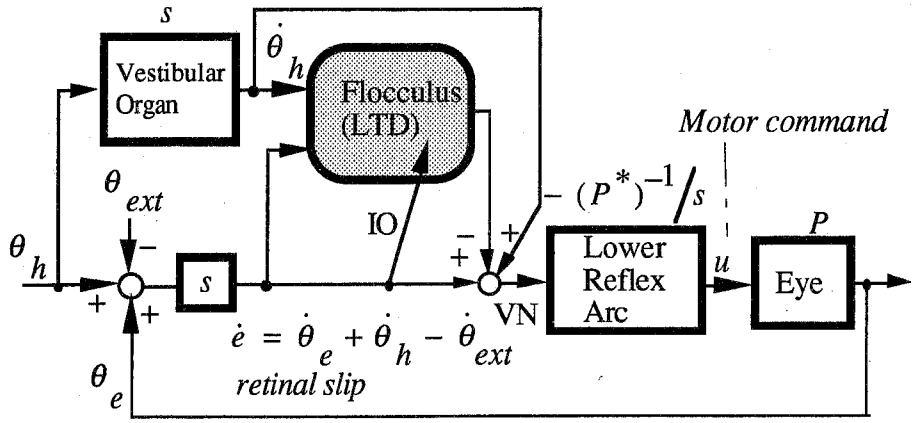
Originally, this research was launched to reveal the computational schemes of motor learning in the central nervous system. Hence, we attempt to discuss here the relationships between the learning schemes proposed above and actual animal motor learning ability.

Human beings can stand stably upright and walk smoothly because various sensors observe the present state and the central nervous system controls posture with a real-time feedback loop. This ability is gradually achieved in a growth process. It has been pointed out that proprioceptors such as the muscle spindles and the tendon organs of Golgi, the vestibulo organ and vision all play a major part as sensors for observing the present state. The proprioceptors observe the position, velocity, and force of the limbs and the body trunk. The vestibulo organs observe the head velocity and acceleration, and the visual sensor also detects the head velocity. Even if one sensor is seriously damaged, other sensors can compensate in order to maintain upright posture control. It is known that there are some tracts descending in the spinal cord from the brain-stem and the cerebellum to limb and body trunk muscles which serve to harmoniously control posture and locomotion [32].

Ito [33] has revealed physiologically that the cerebellum plays an important role in adaptation for the vestibulo-ocular reflex. There are four main parts of the cerebellum: the flocculus, the vermis, the hemisphere intermediate part and the hemisphere lateral part. Fujita [34] proposed the hypothesis of the adaptive mechanism regulating vestibulo-ocular reflex in the flocculus, and Kawato [13] proposed the hypothesis of the adaptive mechanism regulating voluntary movement in the hemisphere lateral part. Nashner found that the adaptation of posture control is severely impaired in patients with cerebellar disease [32]. Thus, it is supposed that the cerebellum plays an important role also in the adaptation of posture control.

The inputs and outputs of these four parts have already been investigated in anatomical studies [35]. The principal afferent inputs of the vermis in the cerebellum are from the vestibular labyrinth, the proximal body parts and the visual organs. The principal outputs of the vermis are to the medial brain stem and the axial regions of the motor cortex. The function of the vermis is comprehended as axial and proximal motor control. The hemisphere intermediate part receives the afferent information from the spinal cord, and sends the output to the red nucleus and the distal region of the motor cortex. The function of the hemisphere intermediate part is comprehended as distal motor control.

a. Adaptive modification of VOR (vestibulo-ocular reflex) and OKR (optokinetic response)



b. Adaptive Control for Posture

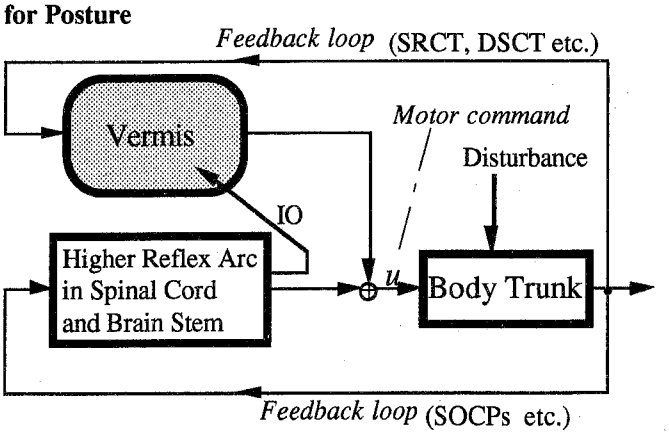
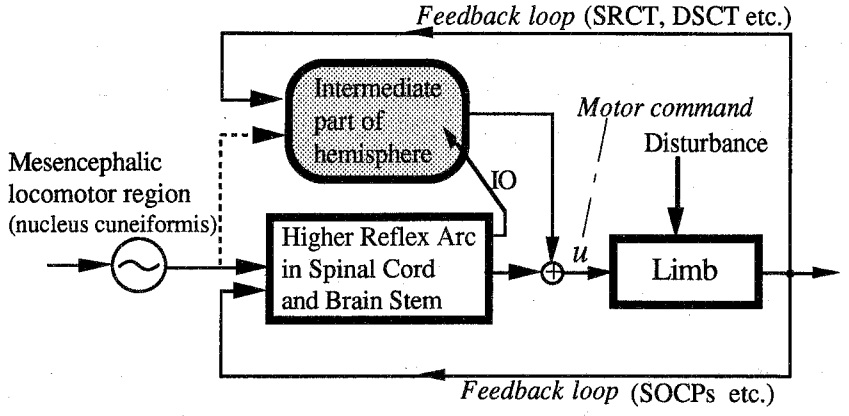


Fig. 14. Adaptive motor control in cerebellum.(a), (b)

IO: inferior olivary nucleus, VN: vestibular nuclei, SRCT: spino-reticulo-cerebellar tracts, DSCT: dorsal spino-cerebellar tracts, SOCPs: spino-olibo-cerebellar paths.

c. Adaptive Control for Locomotion



d. Learning Control for Voluntary Movement

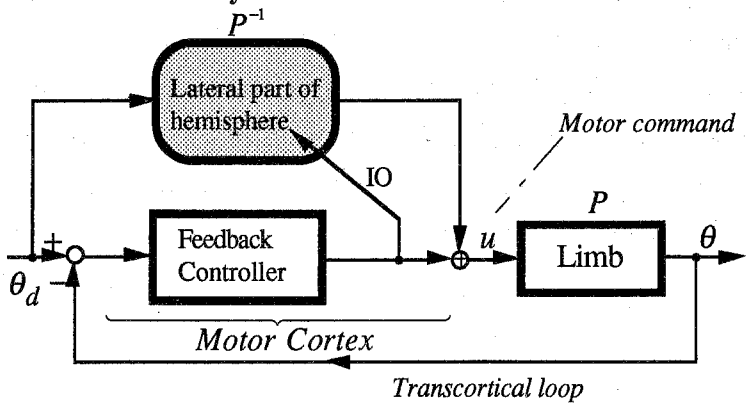


Fig. 14. Adaptive motor control in cerebellum. (c), (d)

These physiological and anatomical observations support the hypothesis that the vermis and the hemisphere intermediate part have an adaptive mechanism of posture control and locomotion control. In addition to the previous two models by Fujita and Kawato, we propose two other models of adaptive control in the cerebellum for posture and locomotion control, as shown in Fig.14. These adaptive mechanisms are based on the *feedback-error-learning* mechanism. Two adaptive models of the vermis and the hemisphere intermediate part correspond to the learning scheme that we proposed above in this paper. The physiological evidence for this hypothesis is not yet adequate, but it might play an important role in revealing the computational schemes of motor learning by the central nervous system.

5.Conclusion

We have proposed a new learning scheme using *feedback-error-learning* for a neural network model applied to adaptive nonlinear feedback control. After the neural network compensated perfectly or partially for the nonlinearity of the controlled object through learning, the responses of the controlled object finally follow the desired responses which are obtained by inverse reference model implemented as the conventional feedback controller.
This learning scheme does not require knowledge of the nonlinearity of a controlled object in advance. The learning schemes proposed here, can also be used together with previous adaptive control methods to use a priori knowledge. Therefore, these learning schemes can be employed for controlling many kinds of objects, such as chemical plants, machines, robots, etc..

References

[1] Dubowsky, S., DesForges,D.T.: The Application of Model Referenced Adaptive Control to Robotic Manipulators; Journal of Dynamic Sys. *Meas. and Cont*, Vol.101, pp.193-200 (1979)

[2] Craig, J.J., Hsu, P., Sastry, S.S.: Adaptive control of mechanical manipulators, *Int. Conf. Robotics and Automation,* San Francisco (1986)

[3] Slotine, J.E., Li, W.: On the adaptive control of robot manipulators, *Int. J. Robotics Research,* 6(3) (1987)

[4] Barto, A.G., Sutton, R.S., Anderson, C.W.: Neuronlike adaptive elements that can solve difficult learning control problems; *IEEE Trans. on Sys. Man and Cyb.* SMC-13, pp.834-846 (1983)

[5] Jordan, M.I.: Supervised learning and systems with excess degrees of freedom, COINS Technical Report 88-27, pp.1-41 (1988)

[6] Psaltis, D., Sideris, A., Yamamura, A.: Neural Controllers, IEEE Int. Conf. Neural Networks, Vol.4, pp.551-557 (1987)

[7] Barto, A. G.: Connectionist Learning for Control. An Overview, COINS Technical Report 89-89, pp.1-38 (1989)

[8] Albus, J.S.: A new approach to manipulator control: The cerebellar model articulation controller, Trans. ASME, vol.97 pp.220-227 (1975)

[9] Miller, III, W.T., Hewes, R.P., Glanz, F.H., Kraft, III, L.G.: Real-Time Dynamic Control of an Industrial Manipulator Using a Neural-Network-Based Learning Controller; IEEE Trans. Robotics & Autom., vol.6, No.1, pp.1-9 (1990)

[10] Atkeson, C.G., Reinkensmeyer, D.J.: Using associative content-addressable memories to control robots, Proc. IEEE CDC, pp.792-797 (1988)

[11] Kuperstein, M., Rubinstein, J.: Implementation of an Adaptive Neural Controller for Sensory-Motor Coordination; IEEE Control Systems Magazine, vol.8, No.2, pp.25-30 (1989)

[12] Martinets, T.M., Ritter, H.J., Schulten, K.J.: Three-Dimensional Neural Net for Learning Visuomotor Coordination of a Robot Arm; IEEE Trans. Neural Net., vol.1, No.1, pp.131-136 (1990)

[13] Kawato, M., Furukawa, K., Suzuki, R.: A Hierarchical Neural-Network Model for Control and Learning of Voluntary Movement; Biol. Cybern. 57, pp.169-185 (1987)

[14] Miyamoto, H.,Kawato, M., Setoyana, T., Suzuki, R.:Feedback-error-learning neural network for trajectory control of a robotic manipulator, Neural Networks, Vol.1, pp.251-265 (1988)

[15] Kawato, M.: Computational schemes and neural network models for formation and control of multijoint arm trajectory. In: Miller, T., Sutton, R.S., Werbos, P.J.(Eds.) Neural networks for control, The MIT Press, Cambridge, Massachusetts (1990)

[16] Kano, M., Kawato, M., Uno, Y., Suzuki, R.: Learning trajectory control of a redundant arm by feedback-error-learning, IEICE technical report, Vol.89, No.463, pp.1-6 (1990) (in Japanese)

[17] Katayama, M., Kawato, M.: A parallel-hierarchical neural network model for motor control of a musculo-skeletal system, System & Computers in Japan (1991) (in Press)

[18] Katayama, M., Kawato, M.: Learning trajectory and force control of an artificial muscle arm by parallel-hierarchical neural network model, Advances in Neural Information Processing Systems **3** (1991) (in Press)

[19] Katayama, M., Kawato, M.: Virtual trajectory and stiffness ellipse during force-trajectory control using a parallel-hierarchical neural network model, Proceeding of fifth international conference on advanced robotics, June, Pisa Italy (1991)

[20] Tsukahara, N., Kawato, M.: Dynamic and plastic properties of the brain stem neuronal networks as the possible neuronal basis of learning and memory; In Amari, S., Arbib, M.A. (Eds.) Competition and cooperation in neural nets, chap. 25, pp.430-441, Springer-Verlag (1982)

[21] Hogan, N.: Impedance Control: An Approach to Manipulation: Part I - Theory, Part II - Implementation, Part III - Applications, ASME Journal of Dynamic Systems, Measurement, and Control, Vol.107, pp.1-24 (1985)

[22] Kawato, M.: The feedback-error-learning neural network for supervised motor learning. In: Eckmiller, R.(ed) *Advanced neural computers*. Elsevier, Amsterdam, pp.365-372 (1990)

[23] Rumelhart, D. E., McClelland, J.L. and PDP Research Group: Parallel Distributed Processing Vol.1, The MIT Press (1986)

[24] Irie, B.,Miyake, S.: Capabilities of three-layered perceptrons, Proc. ICNN 88, vol.1, pp.641-648 (1988)

[25] Stanfill, C., Waltz, D.: Toward memory-based reasoning, Commun. ACM, vol.29, pp.1213-1228 (1986)

[26] Geman, S.: Some averaging and stability results for random differential equations; SIAM J. Appl. Math. vol.36 No.1, pp.86-105 (1979)

[27] Ito, K.: Probability Theory, Iwanami, Tokyo (1953)

[28] Geman, S., Hwang, C.: Diffusions for global optimization, SIAM J. control and optimization, vol.24, No.5, pp.1031-1043 (1986)

[29] Khosla, P., Kanade, T.: Parameter Identification of Robot Dynamics, IEEE Conf. Decision and Control (1985)

[30] Atkeson, C.G., An, C.G., Hollerbach, J.M.: Estimation of Inertial Parameters of Manipulator Loads and Links, Int. Symp. Robotics Research (1985)

[31] Narendra, K.S., Parthasarathy, K.: Identification and Control of Dynamical Systems Using Neural Networks; IEEE Trans. Neural Networks, vol.1,No.1, pp.4-27 (1990)

[32] Carew, T.J.: Posture and Locomotion. In : Kandel, E.R.(ed) Principles of Neural Science, Section 37, Elsevier Inc., 1985.

[33] Ito, M.: *The Cerebellum and Neural Control,* Raven Press, Section 25, pp.354-373 (1984)

[34] Fujita, M. : Simulation of adaptive modification of the vestibulo-ocular reflex with an adaptive filter model of the cerebellum, *Biol. Cybern.*, vol.45, pp.207-214 (1982)

[35] Ghez, C., Fahn, S. : The Cerebellum. In : Kandel, E.R.(ed) *Principles of Neural Science,* Section 39, Elsevier Inc.(1985).