Formation of Voluntary Movements in Recurrent RBF Networks with Feedback-error-learning

Tetsuya MIYOSHI, Hidetomo ICHIHASHI, Takuya NISHIKADO and Kazunori NAGASAKA
Department of Industrial Engineering, College of Engineering, Osaka Prefecture University
1-1 Gakuencho, Sakai, Osaka 593, JAPAN
e-mail: miyoshi@center.osakafu-u.ac.jp
Tel:+81-722-52-1161 ext.2374 Fax: +81-722-59-3340

Key Word: fuzzy modeling, recurrent network, RBF network, voluntary movement, feedback-error-learning

Abstract

In this paper we propose the learning model of voluntary movement with feedback-error-learning. Our learning model consists of the hierarchical recurrent RBF network which represents the dynamics of voluntary movement. Once the dynamics of voluntary movement is formed in the network, it generates the torque commands which realize the desired trajectory.

1. Introduction

It has been recognized that the synaptic plasticity must play the most important roles in information processing for voluntary movements. In order to control voluntary movements, the central nervous system(CNS) must perform complex information processing. A computational model of voluntary movement consists of the following three components as: (1) determination of a desired trajectory, (2) transformation of visual coordinates, (3) generation of motor commands to realize the desired trajectory[1].

In the study of control in voluntary human arm movements, M.Kawato *et al.* proposed the learning scheme called as "feedback-error-learning", in which the feedback torque is used as the error signal of the heterosynaptic learning[2]. The neural network model represents the inverse dynamics of upper limb, and requires the desired trajectory for generating the torque command which realizes the desired trajectory.

Recurrent neural network (RNN) with feedback and self-connection seems suited for temporal dynamics which expresses the input-output relation depending on time. The RNN learns to identify the system's dynamic characteristic. We have proposed the learning method for the recurrent neural networks, whose basic building blocks are represented by Radial Basis Functions(RBF). By the excellent function approximation capability of the RBF networks, some nonlinear dynamics were recovered [3]. As the neurons of RNN increase, the unknown parameters increase exponentially. Y.Kuroe proposed the architecture of recurrent neural network in which dynamic and static neurons are connected[4].

In this paper, we propose a learning scheme of voluntary movement with "feedback-error-learning" for hierarchical RNN. In order to decrease the unknown parameters a new architecture of recurrent neural networks, which we call hierarchical recurrent RBF networks, is introduced and their training algorithm is derived. This hierarchical recurrent RBF networks contains both the feedforward and feedback network.

2. Feedback-error-learning

Let us consider RNN as a feedforward controller of the voluntary movement of human upper limb. For simplicity we consider a manipulator as human upper limb model, which is shown in Fig.1. In general the dynamics equation of the manipulator is written as

$$\boldsymbol{H}(\ddot{\boldsymbol{\theta}}) + \boldsymbol{C}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \boldsymbol{B}(\dot{\boldsymbol{\theta}}) + \boldsymbol{G}(\boldsymbol{\theta}) = \boldsymbol{T} \tag{1}$$

where $\boldsymbol{\theta}$ is the vector of joint angle, $\boldsymbol{H}(\ddot{\boldsymbol{\theta}})$ is inertia term, $\boldsymbol{C}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})$ is centripetal and Coriolis force, $\boldsymbol{B}(\dot{\boldsymbol{\theta}})$ is the frictional force, $\boldsymbol{G}(\boldsymbol{\theta})$ is the gravitational force and $\boldsymbol{T}$ represents a vector of joint torque. The dynamics of manipulator can be represented by a system of equations consists of 43 nonlinear terms (see e.g.[5]).

The problem is to generate a motor command, which realizes the desired movement pattern. We assume that in training the movement the desired trajectories are given, and after learning the movement, the voluntary movement is realized without the desired trajectories.

Fig.2 shows the feedback-error-learning scheme, in which K is a feedback controller and RNN is the recurrent neural network. Since RBF networks are potentially faster than the sigmoidal basis function networks and learning algorithms are relatively simple[6, 7, 8],

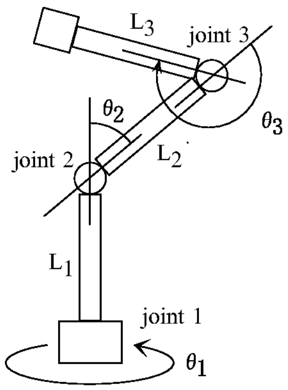

Figure 1: A vertical type 3 link manipulator

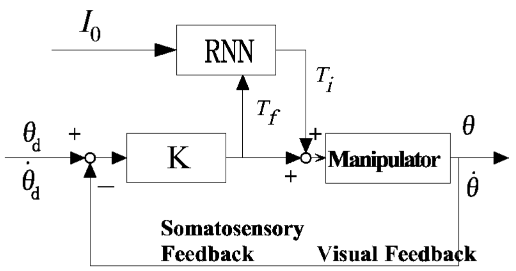

Figure 2: Feedback-Error-Learning

then in our study we use the hierarchical recurrent RBF network as RNN, which will be described later in section 3. RNN has the trigger input I_0 which raises the voluntary movement. Once the dynamics of voluntary movement is formed in the RNN, only by the trigger input I_0 the torque T_i which realizes the desired trajectory is generated.

In Fig.2 the torque fed to the manipulator is the sum of the feedback torque generated by the feedback controller and the feedforward torque generated by the recurrent neural networks.

$$T(t+1) = T_f(t) + T_i(t) \tag{2}$$

where $T_f(t)$ is the feedback torque and $T_i(t)$ is the feedforward torque. The feedback torque T_f is calculated from the trajectory error $\theta_d - \theta$ multiplied by the feedback gain K_p and K_d as

$$T_f(t) = K_p(\theta_d - \theta) + K_d(\dot{\theta}_d - \dot{\theta}) \tag{3}$$

where θ_d and θ are the desired trajectory and the realized trajectory respectively. In feedback-error-learning scheme the feedback torque is chosen as the error signal for RNN, the error between the desired trajectory

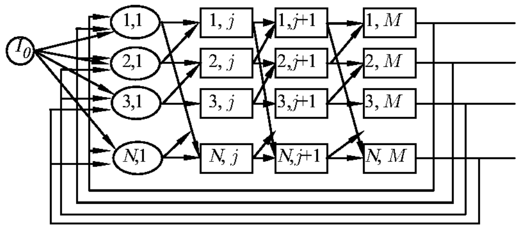

Figure 3: Hierarchical Recurrent RBF Network

and the realized trajectory tends to zero as learning proceeds. That is, the dynamics to generate the voluntary movement is learned in RNN as learning proceeds.

3. Hierarchical recurrent RBF network

We have proposed the learning method for the recurrent neural networks(RNN), whose basic building blocks are represented by Radial Basis Function(RBF)[3]. In this section a new architecture of recurrent neural networks, which we call hierarchical recurrent RBF networks, is introduced and their training algorithm is derived.

The hierarchical recurrent RBF network model as RNN is shown in Fig.3. The network has M layers, each of which consist of N submodels and the outputs of the Mth layer are fed to the 1st layer. In the network the input of the i-th model in the j-th layer is the output variables of the i-th and the $(i+1)$th submodel in the $(j-1)$th layer.

Let the outputs of i-th submodel in j-th layer be y_{ij} which is represented by

$$y_{ij} = \sum_k w^i_{kj} \cdot \mu^i_{kj} \tag{4}$$

where μ^i_{kj} is the k-th basis function of the i-th submodel in the j-th layer. μ^i_{kj} is defined as:

$$\mu^i_{kj}(x_1, x_2) = \exp\left(-\frac{(x_1 - a^{i1}_{kj})^2}{b^{i1}_{kj}} - \frac{(x_2 - a^{i2}_{kj})^2}{b^{i2}_{kj}} \right) \tag{5}$$

where x_1,x_2 are the input of the i-th submodel in the j-th layer, that is, $x_1 = y_{i,j-1}$, $x_2 = y_{i+1,j-1}$. The parameters w^i_{kj}, a^{il}_{kj} and $b^{il}_{kj}(l = 1,2)$ are given for each i, j, k and are changed in the training procedure.

Let $\boldsymbol{y}_j$ and $\boldsymbol{g}_j$ $(j = 1, \cdots, M)$ be the output vector and the nonlinear vector function of submodel at j-th layer respectively. RBF networks are used as the nonlinear function $\boldsymbol{g}_j$, which consist of the Gaussian functions. The 1st layer has the trigger input I_0.

Since the 1st layer has the feedback loop from the Mth layer, it is governed by the system of ordinary differential equations as

$$\frac{\boldsymbol{y}_1}{dt} = \boldsymbol{g}_1(\boldsymbol{y}_M, I_0, \boldsymbol{w}) \tag{6}$$

and since the inputs of j-th ($j \neq 1$) layer are y_{j-1}, then the outputs of the j-th layer are represented by

$$\boldsymbol{y}_j = \boldsymbol{g}_j(\boldsymbol{y}_{j-1}, \boldsymbol{w}) \qquad (j = 2, \cdots, M) \tag{7}$$

where $\boldsymbol{w}$ is the unknown parameter. The objective is to minimize the deviation of $\boldsymbol{y}$ from $\boldsymbol{y}^*$ which is the desired torque to generate the voluntary movement, then the cost function is defined as

$$\begin{aligned} E &= \int_{t_0}^{t_f} L(\boldsymbol{y}_M)\, dt \\ &= \int_{t_0}^{t_f} \frac{1}{2}(\boldsymbol{y}_M - \boldsymbol{y}^*)^T \cdot (\boldsymbol{y}_M - \boldsymbol{y}^*) dt \end{aligned} \tag{8}$$

The learning rule is written as

$$w_{ij}^{NEW} = w_{ij}^{OLD} - \eta \frac{\partial E}{\partial w_{ij}}$$
$$(i = 1, \cdots, N, \quad j = 1, \cdots, M) \tag{9}$$

We can calculate the gradient vector with respect to $\boldsymbol{w}_1$ of the 1st layer by solving the adjoint equations. The gradient with respect to $\boldsymbol{w}_j$ in other layers can be calculated by the backpropagation method[9].

4. Simulation of training

In this section we show the simulation results of training upper limb voluntary movement. Hierarchical recurrent RBF network has 2 layers (M=2) and 2 submodels (N=2) in each layer. The submodels at 1st layer have feedback loop from the last layer(2nd layer) and the last layers (2nd layer) have inputs from the 1st layer only. Feedback gains were set appropriately as K_p=60.0, K_d=1.2 and the number of Gaussian basis in the 1st layer and in the 2nd layer were set as 18 and 9 respectively. The fourth order Runge-Kutta-Gill method with a time step of 10ms was used, and sampling period was 10ms. We deal with two voluntary movements shown in Fig.5, whose desired trajectories for 1 second are drawn by dashed line (a) in Fig.6. Assuming that θ_{d1} and θ_{d2} are given at every 10ms in the interval [0,1.00], 100 data are obtained as the teaching signals.

We first compare the learning performance of proposed hierarchical recurrent RBF network($M = 2$) against RNN with single layer($M = 1$) by the a voluntary movement shown in Fig.5(a). Fig.4 shows how

Figure 4: Learning history of square sum of errors (E) during 1500 iteration (1500sec).

sum of square errors is minimized as iteration number , it is recognized that sum of square error by using the hierarchical recurrent RBF network reduces faster than the other.

The learning result using two movement patterns are shown in Fig.5(a) and (b). During learning process the parameter w, a and b, which consist of the hierarchical recurrent RBF network, are adjusted to minimize feedback torque T_f. In each learning iteration they are updated for a movement and another movement by turns. In this learning simulation we set the iteration number and learning rate to 500 and 0.003 respectively. Learning results are shown in Fig. 5. Fig. 5 shows the desired trajectories and the trajectories generated by the hierarchical recurrent RBF network. Fig.6 shows the comparison between feedback torque T_f and feedforward torque T_i before learning and ones after learning.

As the leaning proceeds, the feedback torque decreases and the feedforward torque increases. It is recognized that the hierarchical recurrent RBF network can learns to identify the dynamic characteristics of upper limb voluntary movement.

5. Conclusion

We proposed a learning model of voluntary movement formation in brain, which consist of the recurrent and feedforward RBF networks, and derived the learning algorithm for the model. Once the dynamics of voluntary movement is formed in the hierarchical recurrent RBF network, by the trigger input it can provide the torque which realize the desired trajectory.

We have applied the proposed learning model and algorithm to adaptive control of a manipulator which simulate the upper limb movement, and have shown the simulation results of training upper limb voluntary movement.

a) pattern 1 b) pattern 2
desired trajectories
trajectories generated by HRNN

Figure 5: Desired trajectories and trajectories generated by HRNN

(1) before learning

(2) after learning

Figure 6: Comparison feedback torque T_f and feedforward torque T_i before learning (1) with ones after learning (2). (a) is the torque which generates the voluntary movements. (b) is the feedback torque which is generated by feedback controller. (c) is the feedforward torque which is generated by the RNN.

References

[1] Y.Uno, M.Kawato and R.Suzuki, "Formation and Control of Optimal Trajectory in Human Multijoint Arm Movement", Biological Cybernetics, Vol.61, pp.89-101, 1989.

[2] M.Kawato, K.Furukawa and R.Suzuki: A Hierarchical Neural-Network Model for Control and Voluntary Movement, Biological Cybernetics, Vol.57, pp.169–185(1987)

[3] T. Miyoshi, H. Ichihashi, S. Okamoto and T. Hayakawa: Learning Chaotic Dynamics in Recurrent RBF Network, Proc. of ICNN'95, pp.588–593(1995)

[4] Y.Kuroe, I.Kimura: Modeling of Unsteady Heat Conduction Field by Using Composite Recurrent Neural Networks, Proc. of ICNN'95, pp.232–237(1995)

[5] H.Miyamoto, M.Kawato, T.Setoyama and R.Suzuki: Feedback-Error-Learning Neural Network for Trajectory Control of a Robotic Manipulator, Neural Networks, Vol.1, pp.251–265(1988)

[6] J.Moody and C.Darken: Fast Learning Networks of Locally-Tuned Processing Units, Neural Computation, Vol.1, No.2, pp.281-294 (1989)

[7] H.Ichihashi, Y.Satoh, T.Miyoshi and K.Nagasaka, Neuro-Fuzzy Finite Element Method and a Posteriori Error Estimation, Proc. of FUZZ-IEEE/IFES'95, pp.2335-2342(1995).

[8] H.Ichihashi, T.Miyoshi, K.Nagasaka, M.Tokunaga and T.Wakamatsu, A Neuro-Fuzzy Approach to Variational Problems by Using Gaussian Membership Functions, International Journal of Approximate Reasoning, Vol.13, No.4, pp.287-302(1995).

[9] D.E.Rumelhart, G.E.Hinton and R.J.Williams, Learning Representations by Back-Propagation Errors, Nature, Vol.323, No.9, pp.533-536 (1986).