A Study of Human Hand Position Control Learning
- Output Feedback Inverse Model -

Eimei Oyama, Taro Maeda
Mechanical Engineering Laboratory
Namiki 1-2, Tsukuba Science City,
Ibaraki 305 Japan

Susumu Tachi
RCAST, The University of Tokyo
.Komaba 4-6-1, Meguro
Tokyo 153 Japan

Abstract

The acquisition of an inverse-kinematic model is required for motor control in humans. With the direct inverse modeling method that is a conventional method, a sufficient inverse model cannot be obtained when the input and output correspondence of the target system is not one-to-one and is non-linear. The problem of seeking the inverse-kinematic model of the human arm, including a wrist, falls into this category. In this report, we propose an inverse model which has an output error feedback path, and determines the input for the target system by means of iterative improvement.

Hand position feedback control of a multi-joint manipulator in working coordinates includes the non-linear gain of the joint angles, for example, the pseudo-inverse of the Jacobian matrix. In this report, we show that learning of the hand position feedback gain is possible with the output feedback inverse model.

1 Introduction

By using neural networks, several models of the human motor control system have been developed. The acquisition of the inverse-kinematic and inverse-dynamic models is the main problem. Kawato et al.[1] pointed out that in motor control in humans, the formation of inverse models is necessary but the inverse model learning is an ill-posed problem.

Kuperstein[2] and others developed human inverse-kinematic models composed of neural networks by using the direct inverse modeling method. With this method, a sufficient inverse model cannot be obtained when the correspondance between the input and the output of the target system is not one-to-one and is non-linear. The learning of an inverse-kinematic model for the human arm including a wrist joint is one such problem.

An inverse model of general systems can be obtained by the forward and inverse modeling method proposed by Jordan et al.[3]. The back-propagation method is necessary for this method. The possiblity that the back-propagation occurs in the human nervous system is very small. Consequently this presents one serious problem as a human model.

The feedback error learning scheme, proposed by Kawato et al.[1], is able to learn an inverse-kinematic model in a motor system with redundancy. However it is considered that the visual feedback controller is acquired by learning. To explain the motion control through eyesight in human, new methods that can acquire the inverse-kinematic model are necessary.

In this report, we propose a kind of inverse model which we call the output feedback inverse model and the learning method of this model. The output feedback inverse model uses the difference between the output of the target system and the desired output as feedback and finds a solution through iterative improvement.

It is shown that the learning of the non-linear gain for the hand position feedback controller is possible through the output feedback inverse model.

2 Inverse Model Learning

Consider the motion of carrying the hand to a target point in space. The generation of a motion which carries the hand to a target position is one type of the inverse problem. In order to solve that problem, an inverse model for the generation of the

CH 3065-0/91/0000-1434 $1.00 © IEEE

proper joint angles from the hand's target position must be formed in the human nervous system.

Denote the m dimension input vector designated as x, and the n dimension output vector as y. When the static system $y = f(x)$ is considered, in the equation $y = f(g(y))$, g represents the inverse model.

When the inputs and the outputs are provided, the non-linear learning element that can approximate any continuous function is designated as Φ. Φ may approximate the target system by minimizing the performance index

$$E = \sum_{i=0}^{N} (f(x_i) - \Phi(x_i))^2 \tag{1}$$

Among the neural network models which are able to approximate any continuous function, there are the multi-layer neural network created through back-propagation learning, Albus' CMAC, Kohonen's topographic mapping, and others. Representative inverse model learning methods are shown in this section.

2.1 Direct Inverse Modeling

The direct inverse modeling (**DIM**) has been applied to the learning of motor control by Albus[4], Kuperstein, and others, and is a method into which much research has been conducted for a long time.

DIM uses the target system's output as input to the inverse model and use the target system's input as the teaching signal of the inverse model. The conceptual diagram of the method is shown in **Fig.1**.

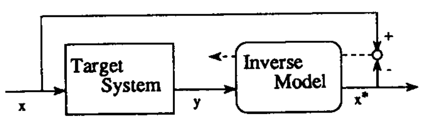

Figure 1 Direct Inverse Modeling

If the learning element Φ is used in the inverse model, **DIM** minimizes the performance index

$$E = \sum_{i=0}^{N} (x_i - \Phi(f(x_i)))^2 \tag{2}$$

When the correspondence between x and y is one-to-one, there are no problems with this method. However, problems arise when the correspondance between input x and output y is not one-to-one.

In cases where there are multiple inputs x which correspond to one output y, the ideal inverse model learned through **DIM** will generate x_m, the average value of x which generates y, when y is inserted. x_m generally does not correspond to y. In cases such as this, the inverse model cannot be realized with sufficient precision. A human arm which includes a wrist joint is one such system.

2.2 Forward and Inverse Modeling

The forward and inverse modeling (**FIM**) was suggested by Jordan to remove the deficiencies of **DIM**. This method tries to minimize the performance index

$$E = \sum_{i=0}^{N} (y_i - f(\Phi(y_i)))^2 \tag{3}$$

However, the teaching signal cannot be communicated through the target system $f()$. In place of $f()$, the target system's forward model Φ_{fm} can be used, and the performance index is changed to

$$E = \sum_{i=0}^{N} (y_i - \Phi_{fm}(\Phi(y_i)))^2 \tag{4}$$

Φ_{fm} and Φ are regarded as one network, and the learning is carried out. The back propagation method (hereafter referred to as the **BP** method) is essential in this method, because with other learning systems, the error signal cannot be communicated from the forward model to the inverse model. This is shown in the conceptual diagram in **Fig.2**.

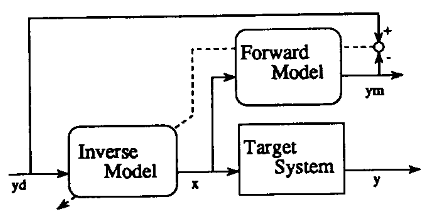

Figure 2 Forward and Inverse Modeling

Since the possibility is only slight that the human nervous system utilizes the BP method, this model is doubtful as a model of human learning.

2.3 Feedback Error Learning Scheme

The feedback error learning scheme (**FEL**) was proposed by Kawato et al. as a human motor learning model. Not only is this reasonable as a human motor learning model, but also it has a high degree of industrial applicability. **FEL** is a method which uses the output of the feedback control circuit as the teaching signal for the inverse model. **FEL** requires feedback control circuits. The conceptual diagram of the method is shown in **Fig.3**.

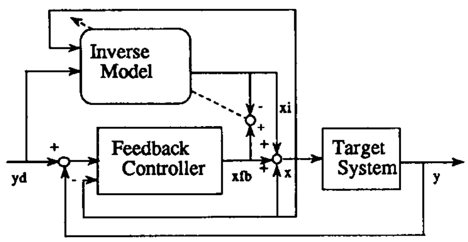

Figure 3 Feedback Error Learning Scheme

It is believed that the correspondence between the hand position measured by vision and the joint angles is acquired by learning. Therefore a human visual feedback controller is acquired by learning, too. **FEL** is not applicable.

3 Output Feedback Inverse Model

The incompleteness of **DIM** is partially avoided by the linearization of the target system and the utilization of a feedback path.

3.1 Direct Inverse Modeling for a Linear System

Consider the inverse model learning for a linear system. Since the target system is linear, the learning element Φ is designated as $\Phi(x) = Wx$, and the learning takes place in W. We provide in succession the sets of input x (m dimension vector) and output y (n dimension vector) as the learning data. Inverse model learning is conducted either by the **BP** method or orthogonal learning. W, which represents the inverse model, is renewed according to the following equation,

$$W^* = W + \eta \sum_{p=0}^{N} (x_p - W y_p) y_p^T \qquad (5)$$

where η is a learning-coefficient. From $W^* = W$, and $y_p = Ax_p$,

$$A^T - WAA^T = 0 \qquad (6)$$

can be realized. When $n < m$, and AA^T is a normal matrix, then the following solution is acquired.

$$W = A^T(AA^T)^{-1} \qquad (7)$$

This is the pseudo-inverse matrix of A. When learning the linear system's inverse model using **DIM**, the inverse model which generates the minimum norm solution can be learned. When $n = m$, and A is a normal matrix, then

$$W = A^{-1} \qquad (8)$$

results. In any case, when the inverse model's output Wy is put into the target system, then the output of the target system becomes y. In the case of $n > m$, W fulfills the conditions of (6). The solutions of equation (6) includes the matrix $(A^TA)^{-1}A^T$ which provides the least squares solution in regard to input y, but the least squares solution is not always obtained. When there is noise with variance rI_n in the teaching signal y, equation (6) becomes

$$A^T - W(AA^T + rI_n) = 0 \qquad (9)$$

and, through the inverse matrix lemma,

$$\begin{aligned} W &= A^T(AA^T + rI_n)^{-1} \\ &= (A^TA + rI_m)^{-1}A^T \end{aligned} \qquad (10)$$

If r is an infinitesimal number, the inverse model approximates the least squares method.

In a linear system, even if there are redundant degrees of freedom, learning of the inverse model is possible through **DIM**. Therefore the incompleteness of **DIM** is avoided by the linearization of the target system.

3.2 Output Feedback Inverse Model

Another problem of **DIM** is that large-scale connection changes must be carried out before the inverse model is used for control. The desired value must be input into the inverse model instead of the actual value. By adding a feedback path to the inverse model, this problem is partially avoided.

Consider the inverse model Φ which has the output error feedback path.

$$x(t+1) = \Phi(x(t), y_d - y(t)) \quad (11)$$

This system configuration is shown in **Fig.4**.

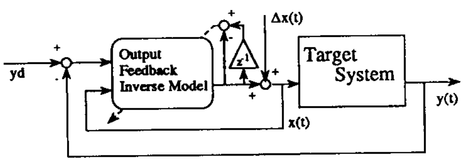

Figure 4 Output Feedback Inverse Model

We assume that the motion generator has two modes, x oriented mode and y oriented mode. The input of the target system is fully controlled in the x oriented mode and the output of the target system is fully controlled in the y oriented mode. The y oriented mode does not work correctly without learning.

In the x oriented mode, x is fully controlled and y_d is equal to $y = f(x)$. The output of $\Phi(x, 0)$ should be x. By learning, the equation

$$\Phi(x, 0) = x \quad (12)$$

is established.

In the main learning mode, learning is carried out as indicated below with the prime leader designated as the changes of the target system's input x.

In the learning mode, assume that the initial value of the input of the target system is $x(0)$ and the output is $y(0)$. The desired output y_d is changed as

$$y_d(t) = y(0) + \Delta y_d(t) \quad (13)$$

The output of the inverse model $\Phi(x(t), y_d(t) - y(t))$ is designated as $x_0(t)$:

$$x_0(t) = \Phi(x(t), y_d(t) - y(t)) \quad (14)$$

and $x(t+1)$ is the sum of $x_0(t)$ and the noise vector $\Delta x(t)$:

$$x(t+1) = x_0(t) + \Delta x(t) \quad (15)$$

The output of the target system becomes

$$y(t+1) = f(x(t+1)) \quad (16)$$

Then the inverse model output becomes

$$x_0(t+1) = \Phi(x(t+1), y_d(t+1) - y(t+1)) \quad (17)$$

$x_0(t)$ is used for the teaching signal for $\Phi(x(t+1), y_d(t+1) - y(t+1))$.

Therefore the performance index

$$\begin{aligned} E &= \sum_{t=0}^{N}(x(t+1) - \Delta x(t) \\ &\quad -\Phi(x(t+1), y_d(t+1) - f(x(t+1)))^2 \end{aligned} \quad (18)$$

is minimized in the learning mode.

First , we analyze the case where the target system is linear and the learning element is also linear.

$$y = Ax \quad (19)$$

$$\Phi(x, \Delta y) = x + W\Delta y \quad (20)$$

The learning takes place in W.

$x(t)$ is renewed according to the following equation,

$$x(t+1) = x(t) + \Delta x(t) + W(y_d(t) - Ax(t)) \quad (21)$$

$e(t)$ is designated as

$$e(t) = x(t) - x(0) \quad (22)$$

and the following equation is obtained.

$$\begin{aligned} e(t+1) &= \Delta x(t) + e(t) + W(\Delta y_d(t) - Ae(t)) \\ &= (I - WA)e(k) + \Delta x(t) + W\Delta y_d(t) \\ &= \sum_{i=0}^{t} B(t-i)(\Delta x(i) + W\Delta y_d(t)) \end{aligned} \quad (23)$$

where $B(j) = (I - WA)^j$.

The performance index become

$$E = \sum_{t=0}^{N} p(t)^2 \quad (24)$$

$$\begin{aligned} p(t) &= x_0(t) - (x(k+1) + W(y_d - Ax(k+1))) \\ &= -(I - WA)\Delta x(t) \\ &\quad -W(I - AW)(\Delta y_d(t) - Ae(k)) \\ &\quad -W(\Delta y_d(t+1) - \Delta y_d(t)) \end{aligned} \quad (25)$$

The value of the error feedback is

$$\begin{aligned} q(t) &= y_d(t+1) - Ax(t+1) \\ &= -A\Delta x(t) + (I - AW)(\Delta y_d(t) - Ae(t)) \\ &\quad +(\Delta y_d(t+1) - \Delta y_d(t)) \end{aligned} \quad (26)$$

The matrix W that minimizes E is given by

$$\sum_{t=0}^{N} p(t)q(t)^T = 0 \tag{27}$$

Because $\Delta x(t)$ and

$$\begin{aligned} r(t) &= (I - AW)(\Delta y_d(t) - Ae(t)) \\ &+ (\Delta y_d(t+1) - \Delta y_d(t)) \end{aligned} \tag{28}$$

are independent,when W converges for very large N,

$$(I - WA)R_1 A^T - WR_2 = 0 \tag{29}$$

$$R_1 = \frac{1}{N}\sum_{t=0}^{N} \Delta x(t)\Delta x(t)^T = r_x I \tag{30}$$

$$R_2 = \frac{1}{N}\sum_{t=0}^{N} r(t)r(t)^T \tag{31}$$

If $r(t)$ converges,

$$\begin{aligned} W &= A^T(AA^T + \frac{1}{r_x}R_2)^{-1} \\ &= (r_x A^T R_2{}^{-1} A + I)^{-1} A^T \end{aligned} \tag{32}$$

is obtained. Because the solution is the product of the transpose of A and and a non-negative symmetrical matrix, this matrix can make $y(t)$ close to $y_d(t)$. But if $|y_d(t+1) - y_d(t)|$ is too large, W dose not converge.

If $y_d(t)$ is kept constant, or $y_d(t)$ is used in place of $y_d(t+1)$ in the equations from (17) to (32), then

$$(I - WA)R_1 A^T - W(I - AW)R_e(I - AW)^T = 0 \tag{33}$$

$$R_e = \frac{1}{N}\sum_{t=0}^{N}(\Delta y_d(t) - Ae(t))(\Delta y_d(t) - Ae(t))^T \tag{34}$$

are obtained. One solution of the above equation is

$$W = A^{\#} = A^T(AA^T)^{-1} \tag{35}$$

W usually converges on this solution by learning.

3.3 OFIM Based on Infinitesimal Changes

To acquire the output feedback inverse model for a non-linear system, $|\Delta y|$, $|e(t)|$ and $|\Delta x(t)|$ must be kept sufficiently small so that Φ and f can be linearized.

$$\Phi(x, \Delta y) = \Phi(x, 0) + W(x)\Delta y \tag{36}$$

$$f(x + \Delta x) = f(x) + A(x)\Delta x \tag{37}$$

Also, unlike the learning of a linear system, the learning of a non-linear system must be carried out for many $x(0)$ and Δy_d. The performance index is

$$E = \sum_{i=0}^{M}\sum_{t=0}^{N}(x_{0i}(t) - \Phi(x_i(t), y_{di} - f(x_i(t+1))))^2 \tag{38}$$

The performance index can be approximated as

$$\begin{aligned} E &\approx \sum_{i=0}^{M}\sum_{t=0}^{N}(\Delta x_i(t) \\ &\quad - W(x_i(t))(y_{di} - f(x_i(t+1)))^2 \\ &= \sum_{i=0}^{M}\sum_{t=0}^{N}((I - W(x_i(t))A(x_i(t)))\Delta x(t) \\ &\quad - W(x_i(t))(\Delta y_{di} - A(x_i(t))e_i(t)))^2 \end{aligned} \tag{39}$$

One solution of W(x(t)) that minimizes the performance index is

$$\begin{aligned} W(x(t)) &= A^{\#}(x(t)) \\ &= A(x(t))^T(A(x(t))A(x(t))^T)^{-1} \end{aligned} \tag{40}$$

Therefore the output feedback inverse model

$$\begin{aligned} \Phi(x, \Delta y) &= x + W(x)\Delta y \\ &= x \\ &+ A(x)^T(A(x)A(x)^T)^{-1}(y_d - f(x)) \end{aligned} \tag{41}$$

is obtained.

When the difference between y_d and y is sufficiently small, the feedback circuit conducts the replacement operation for the purpose of bringing y close to the target y_d. Even in the case where x is separated from the true solution, x ultimately can be made to converge upon the approximate value of the true solution through iterative improvement.

In the event that the initial value of x is far from the solution, $|\Delta y| = |y_d - f(x)|$ becomes larger, and when $(x, \Delta y)$ enters the domain where no learning is being carried out, the guarantee that the output y approaches y_d disappears, when the inverse model's output $\Phi(x, \Delta y)$ is put into the target system. Instead of y_d, the virtual desired value y'_d can be used. y'_d guarantees the stability of convergence and is defined as

$$y'_d = y + \Delta y = y + K(y_d - y) \tag{42}$$

using a sufficiently small gain K or, using the saturation element *sat* concerning each component of y,

$$y_d' = y + \Delta y = y + sat(y_d - y) \quad (43)$$

Changing $y_d(t)$ by

$$y_d(t) = y(t-1), \quad (44)$$

the learning becomes faster and the precision is improved. In this case, the performance index

$$E = \sum (x - \Phi(x + \Delta x, f(x) - f(x + \Delta x)))^2 \quad (45)$$

should be minimized. This performance index is equivalent to

$$\begin{aligned} E &= \sum (x + \Delta x - \Phi(x, f(x + \Delta x) - f(x)))^2 \\ &= \sum (\Delta x - W(x)\Delta y)^2 \end{aligned} \quad (46)$$

The problem with OFIM is that there are two modes, a learning mode and a control mode, and that the modes cannot be conducted simultaneously. $|\Delta x(t)|$ must be zero during the control mode. However, it is difficult to believe that the eyesight-based control is properly established in infancy, during which time it is believed that the correspondance between the hand position and the eyesight and the joint angles is acquired in humans. We think that this period corresponds to the learning mode, and that the inverse model formation is carried out.

The second problem in the case of the non-linear system which has a redundant degree of freedom in the input is that, strictly speaking, y does not conform to y_d. The distribution of $\Delta x(t)$ is limited to a small domain ($|\Delta x(t)| < r_x$). An error which is proportionate to the square of r_x remains. When r_x becomes greater, the correspondence between input and output can no longer be approximated by a linear relationship, and there is a decrease in accuracy. But if Δx is too small, then many iterative operations become necessary, and it takes too much time to calculate the solution.

The third problem is that since learning is conducted with regard to x only in a small domain which can be linearized, when the function of the target system contains multiple crests and valleys, x do not always converge on a correct solution through the initial value of x. x sometimes converges upon local minimum solutions.

There is also a formation in which the inverse model's output is added onto x's current value, as a variation of the output feedback inverse model. Its conceptual diagram is shown in **Fig.5**.

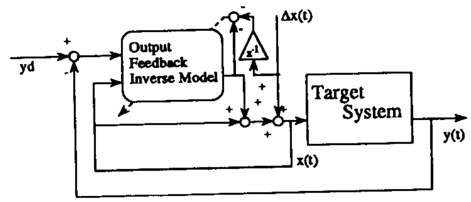

Figure 5 Output Feedback Inverse Model

The output of **OFIM** approximates the minimum norm solution locally. **OFIM** fulfills the characteristics of human hand positioning, because large joint angle changes seldom occur in human hand positioning.

3.4 OFIM Based on Velocity

OFIM based on infinitesimal changes converges on approximate solutions. Paying attention to the input and output time differential relation

$$\dot{y} = A(x)\dot{x} \quad (47)$$

the correspondance between the time differential of the input and that of the output of the system is linear. As well as **OFIM** based on infinitesimal changes, a kind of inverse model

$$\begin{aligned} \dot{x_m} &= \dot{x} + W(x)(\dot{y_d} - A(x)\dot{x}) \\ &\approx \dot{x} + A^{\#}(x)(\dot{y_d} - A(x)\dot{x}) \end{aligned} \quad (48)$$

can be obtained. If $\dot{y_d}$ is kept constant,

$$W(x) = A^{\#}(x) \quad (49)$$

is obtained.

Assuming

$$\dot{y_d} = K(y_d - y) \quad (50)$$

and that a controller that realizes velocity commands exists in humans, a velocity integration type inverse model

$$x = x(0) + \int A^{\#}(x)K(y_d - y)dt \quad (51)$$

can be obtained. The configuration of the model is shown in **Fig.6**.

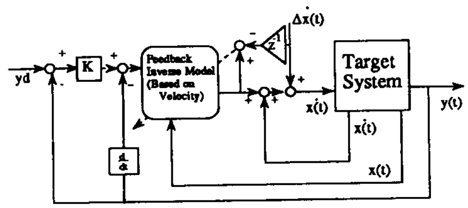

Figure 6 Output Feedback Inverse Model Based on Velocity

If Euler's method is used as the method of integration, an equation, which is the same as (4)

$$x(t+1) = x(t) + A^*(x(t))K(y_d - y(t))\Delta t \quad (52)$$

may be obtained. **OFIM** based on velocity converges precisely upon the true solution, but as with **OFIM** based on infinitesimal changes, learning and control cannot be carried out at the same time and the deficiency of lapsing into local optimum solutions still remains.

In this method, the velocity components of the system are necessary as teaching signals and a sensor for the detection of velocity is required. And a controller that realizes velocity commands is required. It is believed, however, that the motor control system of humans has such sensors and eye movement control is carried out in terms of velocities of eye, head and body.

4 Hand Position Feedback Control of Manipulator

In this report, manipulator control through eyesight is regarded as manipulator control using hand position.

Designating the hand position vector x and the joint angle vector θ, the manipulator kinematics are expressed as

$$x = f(\theta) \quad (53)$$

where f is a non-linear function. Ordinary manipulator dynamics can be expressed as

$$R(\theta)\,\ddot{\theta} + S(\theta,\dot{\theta})\,\dot{\theta} + g(\theta) = u \quad (54)$$

The joint angle torque is represented by u. $g(\theta)$ is a function of θ and represents gravitational forces.

4.1 Joint Angle Feedback Control

For the feedback control of the path defined by the manipulator's joint angles, **PD** feedback control with compensation for the gravitational effect is used as follows.

$$u = g(\theta) - K_p(\theta - \theta_d) - K_D(\dot{\theta}) \quad (55)$$

PID feedback control can also be used. Sufficient control is not possible with only feedback control, but the effects of parameters such as acceleration, velocity, gravity, etc., can be learned by using the feedback error learning scheme of Kawato et al.

4.2 Hand Position Feedback Control

The actual movement path of a robot is usually defined in the working coordinate system. With trajectory control defined in the working coordinate system of the robot arm with a redundant degree of freedom, the transpose of the Jacobian Matrix

$$J(\theta) = \partial x/\partial\theta \quad (56)$$

is used, and **PD** feedback control is carried out by

$$u = g(\theta) - J(\theta)^T K_P(x - x_d) - K_D\dot{\theta} \quad (57)$$

or with the pseudo-inverse matrix $J^*(\theta)$, by

$$u = g(\theta) - J^*(\theta)K_P(x - x_d) - K_D\dot{\theta} \quad (58)$$

The **PID** feedback control can be applied in a similar manner.

The position feedback control system usually contains the non-linear gains $J(\theta)^T or J^*(\theta)$. With a constant gain, control on some states of the arm is not possible. It is unlikely that the non-linear gain is innate in humans. It is believed that this is acquired through learning.

4.3 Application of the Output Feedback Inverse Model to Hand Position Control

When applying **OFIM** to the hand positioning system, one type of inverse model which renews the joint angles may be obtained as

$$\theta_d = \theta + KJ^*(\theta)(x_d - x) \quad (59)$$

When this is provided to the joint angle feedback controller,

$$\begin{aligned} u &= K_P(\theta_d - \theta) + K_D(-\dot{\theta}) \\ &= K_P\{\theta + J^*(\theta)K(x_d - x) - \theta\} + K_D(-\theta) \\ &= K_P J^*(\theta)K(x_d - x) + K_D(-\dot{\theta}) \end{aligned} \quad (60)$$

a position feedback control circuit may be obtained. The composition of a control circuit is shown in **Fig.7**.

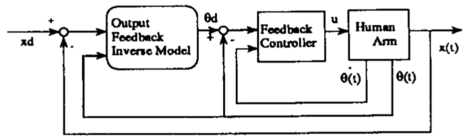

Figure 7 Hand Position Feedback Controller

5 Simulation

Numerical experiments were performed in order to evaluate the performance of **OFIM**. The approximation problem of $y = \cos(x)$ and the problem of calculating the joint angles for the purpose of bringing a hand to a target position were considered. In those experiments, the learning elements were the multi-layer neural network using the BP method.

5.1 Output Feedback Inverse Model

Hereafter, **OFIM1** refers to **OFIM** on infinitesimal changes, and **OFIM2** refer to **OFIM** based on velocity. In these experiments, no technique for the stabilization of convergence was used for **OFIM1**. Hereafter, in the tables, σ will refer to the root square sum of output errors of the data that converges on the true solution ($|y_d - y| < r_y$), PS will refer to the percentage of success of the convergence in simulation, **EM** will refer to the maximum value of output error norm, and **NI** refers to the average number of repeated loops until convergence.

OFIM2 used equation (52) and $K\Delta t$ was 1.0. The output of the neural network has been scaled at $(-0.45, 0.45)$. The variance of x for **OFIM** is fixed at 0.01.

(1) Approximation of $\cos(x)$

The approximation of $\cos(x)$ was carried out with respect to the domain $(-\pi,\ \pi)$. The composition of the forward model was four layer network (1-4-4-1), that of the inverse model was (1-8-8-1), and that of **OFIM** was (2-8-8-1). The results are shown on **Tab.1**. The initial value of x in **OFIM** was fixed at $(x = 30°)$. In all cases, x converged on the true solution with both **OFIM1** and **OFIM2**. By **FIM**, **OFIM1**and **OFIM2**, inverse models were successfully obtained, but this was not possible by **DIM**.

Table 1 Simulation Results

	DIM	*FIM*	*OFIM1*	*OFIM2*
σ	1.2	0.041	0.0015	0.0067
EM	1.9	0.087	0.0056	0.011
NI	-	-	25.6	23.7

(2)Arms With Two Degrees of Freedom on a Plane

Consider the control of a two degree of freedom manipulator moving on a two-dimensional plane. The relationship between the joint angles (θ_1, θ_2) and the hand position (x, y) is expressed as

$$x = x_0 + L_1\cos(\theta_1) + L_2\cos(\theta_1 + \theta_2) \quad (61)$$

$$y = y_0 + L_1\sin(\theta_1) + L_2\sin(\theta_1 + \theta_2) \quad (62)$$

We sssume a range for θ_2 which corresponds roughly to the human wrist joint $(-60°, 60°)$. Without considering the attitude of the hand, consider the problem of finding the joint angles (θ_1, θ_2) which realize the hand position (x, y).

The composition of the inverse model was (2-15-15-2) and the composition of the feedback circuit was (4-15-15-2). Learning was carried out over 100,000 examples. The initial value of the joint angle (θ_1, θ_2) was $(45°, 15°)$.

The error for each inverse model is shown in **Tab.2**. The units are in cm.

Table 2 Simulation Results

	DIM	*FIM*	*OFIM1*	*OFIM2*
σ	4.0	1.2	0.3	0.5
EM	5.8	3.8	1.5	2.0
NI	-	-	33.2	83.2

OFIM1 and **OFIM2** did not converge on any local minimum solusion. By **FIM**, **OFIM1**and **OFIM2**, inverse models were successfully obtained, but this was not possible by **DIM**.

(3) Arms With Three Degrees of Freedom on a Plane

Let's consider the control of a three degree of freedom manipulator moving on a two-dimensional plane. The relationship between the joint angles $(\theta_1, \theta_2, \theta_3)$ and the hand position (x, y) is expressed as

$$x = x_0 + L_1\cos(\theta_1) + L_2\cos(\theta_1 + \theta_2) + L_3\cos(\theta_1 + \theta_2 + \theta_3) \quad (63)$$

$$y = y_0 + L_1 \sin(\theta_1) + L_2 \sin(\theta_1 + \theta_2) + L_3 \sin(\theta_1 + \theta_2 + \theta_3) \quad (64)$$

This is a simplified model of the human arm moving on either a horizontal or vertical plane. Assuming movement over a vertical surface, the possible range for θ_3 which corresponds to the wrist joint is designated as $(-60°, 60°)$. Without considering the attitude of the hand, consider the problem of finding the joint angles $(\theta_1, \theta_2, \theta_3)$ which realize the hand position (x, y). The composition of the inverse model for **DIM** and **FIM** was (3-15-15-3) and the composition for **OFIM1** and **OFIM2** was (5-15-15-3). The composition of the forward model for **FIM** was (3-15-15-2). Learning was carried out over 1,000,000 examples. The initial values of the joint angles and the desired position were given as random numbers. The simulation was carried out for 1000 examples. r_y,the threshold that distinguishes the success of the convergence, is 10 cm. The accuracy is expressed in **Tab.3**.

Table 3 Simulation Result

	DIM	*FIM*	*OFIM1*	*OFIM2*
σ	5.6	2.4	0.6	1.2
PS	46.6	98.0	99.3	96.0
EM	11.2	9.3	3.3	4.5
NI	-	-	30.2	102.0

By **FIM**, **OFIM1** and **OFIM2**, inverse models were successfully obtained, but this was not possible by **DIM**. **OFIM1** and **OFIM2** sometimes converged on local minimum solutions. This problem can be avoided by changing the initial value. This composition of the circuit cannot produce accuracy on the level of a human, but it is considered possible if there are sufficient elements.

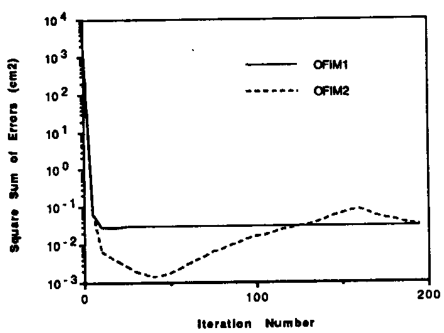

Figure 8 Rate of Convergence

The rate of the convergence for **OFIM** is shown in **Fig.8**. **OFIM2** quickly minimize the output errors but takes some time until the convergence. The path of hand using **OFIM1** after various iteration number is shown in **Fig.9** and the path of hand using **OFIM2** is shown in **Fig.10**.

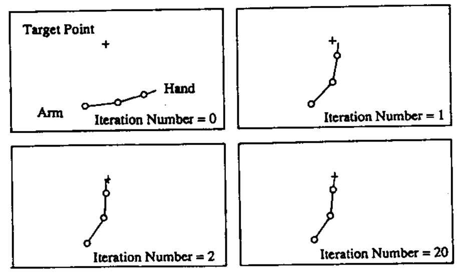

Figure 9 Path of Hand (OFIM1)

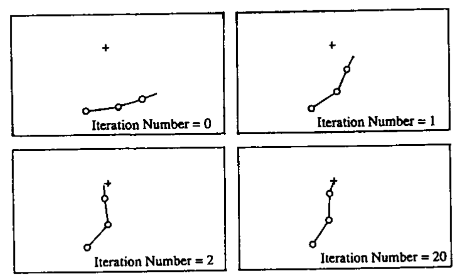

Figure 10 Path of Hand (OFIM2)

(4) Feedback Effects

In order to see the feedback effects, after the learning of the inverse model, the parameters of the target system were changed, and the experiment was repeated.

We tried extending the length of the wrist to 5 cm, and the results are shown in **Tab.4**. It is understood that model error compensation is possible through feedback circuit.

Table 4 Simulation Results

	DIM	*FIM*	*OFIM1*	*OFIM2*
σ	8.7	7.4	1.1	1.4
PS	1.0	20.0	94.5	95.7
EM	9.9	9.8	7.57	9.4
NI	-	-	34.4	105.1

5.2 Hand Position Feedback Control

In order to investigate the possibility of a hand position feedback controller which combines a joint angle feedback controller and **OFIM**, a numerical experiment of control of a manipulator with three degree of freedom over a two-dimensional plane was conducted. The composition of **OFIM** is (5-20-20-3), and learning was carried out with one million pieces of data. The results of the feedback control conducted using a pseudo-inverse matrix and the feedback control using **OFIM1** obtained through learning are shown in **Fig.11**. And the feedback control using **OFIM2** is showm in **Fig.12**. Renewal of the value of the joint angle in **OFIM** was carried out every 50 msec.

Sufficient control is not possible because the feedback gain is relatively small and the feedforward effect is not considered, but finally there is convergence on the target position (position error is about 2mm). In order to improve the ordinary feedback controller's performance, it is advisable to use Kawato et al.'s feedback error learning.

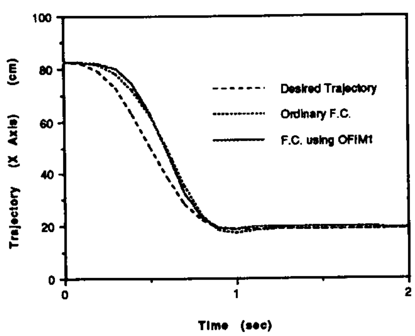

Figure 11 Hand Position Feedback Control using OFIM1

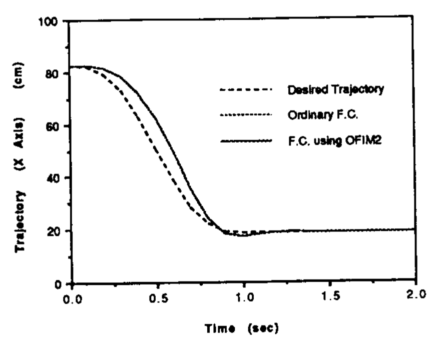

Figure 12 Hand Position Feedback Control using OFIM2

6 Conclusion

We proposed a new method which obtains an inverse model of a system whose inverse model cannot be obtained using the direct modeling method. We corroborated its performance through numerical experiments. The output feedback inverse model (**OFIM**) had a satisfactory performance over all experiments.

When modeling the human arm, only an inadequate kinematics-inverse model can be obtained with customary direct inverse modeling, but satisfactory accuracy could be obtained with the output feedback inverse model.

In addition, by using the output feedback inverse model, it is shown that learning the non-linear gain of the hand position feedback controller is possible.

In infancy, humans cannot control their hands or feet well. It is believed that learning the inverse model of changes of hand position through eyesight is done and that may be used as a basis for carrying out control. This leaning is possible by using the output feedback inverse model.

OFIM based on infinitesimal changes and **OFIM** based on velocity still have some problems. However human hand position control learning is possible with them.

References

[1] M. Kawato, K. Furukawa and R. Suzuki : "A Hierarchical neural-network model for control and learning of voluntary movement", Biol. Cybern. 57, pp.169-185, 1987

[2] M. Kuperstein : "Neural Model of Adaptive Hand-Eye Coordination for Single Postures", Nature, 239, pp.1308-1311, 1988

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

[4] Albus, "A New Approach to Manipulator control : the Cerebellar Model Articulation Controller(CMAC)", vol.97, pp.220-227, ASME, J. Dynamic Syst. Meas. Contr., vol.97, Sept. 1975