EFFECT OF FORCE LOAD IN HAND REACHING MOVEMENT ACQUIRED BY REINFORCEMENT LEARNING

*Katsunari Shibata**

Dept. of Electr. & Electronic Eng.,
Oita Univ., 870-1192, Japan
shibata@cc.oita-u.ac.jp

Koji ITO

Dept. of Comp. Intelli. & Sys. Sci.,
Tokyo Inst. of Tech., 226-8502, Japan
ito@dis.titech.ac.jp

ABSTRACT

It has been known that when a human moves its hand to a target, the trajectories becomes almost a straight line from the start point to the target. When a viscosity force field is loaded to the hand unexpectedly, it is pulled toward the force direction once and then goes back to the target. However, after the learning in the force field, the trajectory becomes a straight line again, and when the force field is removed, it is pulled toward the opposite direction of the force that was loaded to the hand[6]. This is called after-effect.

In this paper, a neural network, whose inputs are visual sensory signals and the state of a manipulator, and whose outputs are joint torques, was trained by reinforcement learning. The effect of the first force field exposure and after-effect could be observed. This means that the system obtains inverse dynamics of its hand and environment in the neural network through reinforcement learning. Further, when the neural network learned in a random force at every trial, it became to control its hand based on the feedback control rather than feedforward control.

1. INTRODUCTION

Hand reaching to some object is a primitive movement for humans. It has been well investigated for the analysis of our motion learning. It is known that the shape of the human hand reaching path is almost a straight line and the associated speed profile is single-peaked and bell-shaped in the case of short unconstrained horizontal movements. There are also some exceptions in the case of long-distance reaching[1]. In order to explain such hand trajectories as optimization problem, "minimum jerk","minimum torque-change", "minimum motion-command-change" criteria and so forth have been proposed[2][1][3]. In order to control the arm to follow the computed trajectory based on feedforward control, "feedback error learning" has been also proposed[4].

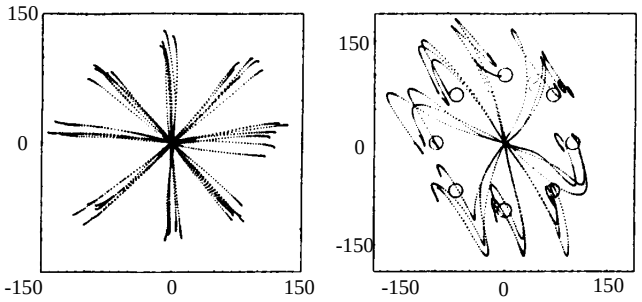
On the other hand, the authors showed that a neural network whose inputs are visual sensory signals and state of a manipulator, and whose outputs are joint torques, can learn the hand reaching movement by reinforcement learning[5]. The hand dynamics is considered and no preprocessing of the visual sensory signals are executed. In this model, it is not necessary to compute the trajectory explicitly on any coordinates, and so the iterative computation to generate the trajectories is not also needed. The obtained hand path is almost a straight line and speed profile is bell-shaped, but not as similar to the human's as that derived from the path planning model mentioned above. However, it can be considered that the path is obtained by the learning of whole the process from sensors to motors without any knowledge about the task and arm dynamics under insufficient simulation settings such as low resolution of the visual sensor.

It is known that in human hand reaching movement, the hand is pulled by the force, and the hand path is curved. However, after some learning in the force field, the path becomes close to a straight line. If the force field is removed, the path is curved to the opposite direction[6]. That is called after-effect.

The purpose of this paper is to examine the effect of the force field loaded to the hand when the hand reaching task is trained by reinforcement learning. Through this, the possibility that reinforcement learning is executed in our living things is shown. It is also shown that no explicit trajectories on any coordinates are necessary to be computed for controlling its arm. Moreover, depending on the situation in the learning, appropriate control can be selected between feedforward control and feedback control only by executing reinforcement learning in the hand reaching task.

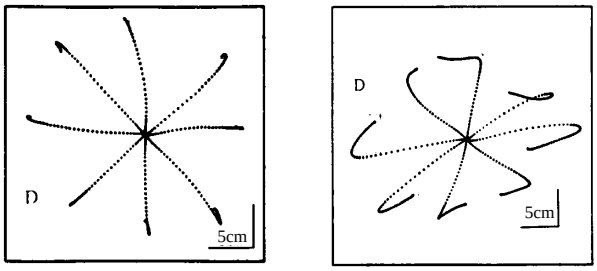
Conventionally reinforcement learning has been thought of a learning for motion planning, in other words, control in wide meaning. However, the authors believe that it can be

*We would like to say thank to Prof. Uno, Mr. Kubotera and Mr. Izawa to give us much knowledge about the human hand reaching movement and the conventional models, and to discuss with us. A part of this research was supported by the Sci. Res. Foundation of the Ministry of Edu., Sci., Sports and Culture of Japan (#13780295,#14350227) and by "The Japan Society for the Promotion of Science" as "Biologically Inspired Adaptive Systems" (JSPS-RFTF96I00105) in "Research for the Future Program".



(a) after learning without force field (ff)

(b) ff is loaded to the hand after learning without ff



(c) ff is loaded to the hand after learning with ff

(d) aftereffect ff is removed after learning with ff

Figure 1: Effect of the force field in human hand reaching task. [6]. [[[[[the copy has not been permitted yet. under application]]]]]

the learning for whole the process from sensors to motors, including recognition, attention, memory and so forth. By combining it to the neural networks, whole the process can be obtained purposively, adaptively and in harmony without being divided into some functional modules. This is expected to result in the real intelligence that fills up the gap between humans and the present robots. This research has an aspect of one of the process to show this ability and the possibility that reinforcement learning is utilized for the learning of many functions in our living things.

2. EFFECT OF FORCE FIELD IN HUMAN HAND REACHING MOVEMENT

In the environment with unknown dynamics, the trajectory of human reaching movement is curved at first. After the learning of a hand reaching task, the hand path becomes almost a straight line as shown in Fig. 1 (a). If a viscosity force field, in which loaded force is varied according to the hand speed vector, as shown in Fig. 2 is loaded to the hand, the hand is pulled to the direction of the force field, and the path is curved as shown in Fig. 1(b). However, after some learning in the force field, the path becomes close to a straight line again as shown in Fig. 1(c). When the

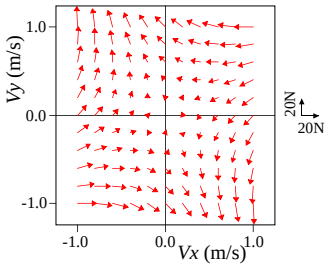


Figure 2: The viscosity force field loaded to the hand.

environment is restored to the normal dynamics, the trajectory is curved inversely[6] as shown in Fig. 1(d). It is also known that when the trajectory curves, the stiffness of the hand becomes large by the simultaneous activation of the paired muscles[7]. In other words, the trajectory error becomes less by making the feedback gain large.

3. ACTOR-CRITIC AND TS LEARNING

Here, actor-critic architecture[8] is employed, and implemented in one layered neural network. The output neurons are divided into one critic output and the other actor outputs. For the learning of the critic, TS (temporal smoothing) based learning is employed[9]. It is very similar to TD (temporal difference) learning[8] and the details can be seen in [9]. The training signal for the critic p_s and for the actor $\mathbf{m_s}$ are

$$p_s(t-1) = p(t) - P_{range}/N_{max}[i], \quad (1)$$

$$\mathbf{m}_s(t-1) = \mathbf{m}(t-1) + \mathbf{rnd}(t-1)\{p(t) - p(t-1)\}, \quad (2)$$

where p: the critic output, P_{range}: the value range of the ideal critic output, here, $0.4-(-0.4)=0.8$, $\mathbf{m}$: the actor output vector, and $\mathbf{rnd}$: random vector added to $\mathbf{m}$ as a trial and error factor before a robot actually moves. N_{max} is computed as

$$N_{max}[i] = \begin{cases} N[i] & if N[i] > \lambda N_{max}[i-1] \\ \lambda N_{max}[i-1] & otherwise. \end{cases} \quad (3)$$

N_{max} tries to represent the maximum time steps until its goal state, but if the present time steps is less than N_{max}, N_{max} decades gradually for adaptation. The slope of the critic is trained to be P_{range}/N_{max}. Accordingly, the ideal critic output changes as a straight line in TS learning, while it changes as an exponential curve in TD learning during the time steps with no reward.

4. SIMULATION

Here, two-link arm as shown in Fig. 3 is supposed, and the task is to learn the hand reaching movement to the object on the visual sensor.

4.1. Arm Dynamics

The arm dynamics is as follows that is the same as [1].

$$\begin{aligned}\tau_1 &= (I_1 + I_2 + 2M_2 l_1 s_2 cos\theta_2 + M_2 (l_1)^2 \ddot{\theta}_1 \\ &\quad +(I_2 + M_2 l_1 s_2 cos\theta_2)\ddot{\theta}_2 \\ &\quad -M_2 l_1 s_2 (2\dot{\theta}_1 + \dot{\theta}_2)\dot{\theta}_2 sin\theta_2 + B_1 \dot{\theta}_1 \qquad (4) \\ \tau_2 &= (I_2 + M_2 l_1 s_2 cos\theta_2)\ddot{\theta}_1 + I_2 \ddot{\theta}_2 \\ &\quad +M_2 l_1 s_2 (\dot{\theta}_1)^2 sin\theta_2 + B_2 \dot{\theta}_2 \qquad (5)\end{aligned}$$

where $\tau_i, M_i, l_i, s_i, I_i$: torque for the joint i, mass, length, distance between joint and center of gravity and inertia of the link i respectively. If the joint angle 1 becomes larger than 180 degree, the angle is fixed at 180 degree, and computed the dynamics as one link. Each parameter is set as shown in Table 1. The maximum torque is 5.0[Nm] for the joint 1 and 3.0[Nm] for the joint 2. The differential equation is solved numerically by Runge-Kutta method with the sampling time of 0.02 second.

Table 1: Parameters used in the dynamical arm model.

Parameter	link1	link2
M_i (kg)	2.0	
l_i (m)	0.3	
s_i (m)	$l_i/2$	
I_i ($kg\ m^2$)	$M_i * l_i^2 / 3.0$	
B_i ($kg\ m^2/s$)	0.4	0.2
$\tau_{max\ i}$ ($N\ m$)	5.0	3.0

4.2. Task Setting

The visual sensor consists of $5 \times 5 = 25$ sensory cells, and the receptive field of each cell is a square without overlapping. The output of the cell is the area ratio occupied by a projected object. The size of the hand and the target is supposed to be just the same as one sensory cell, and they cannot be distinguished on the visual sensor. Each of the joint angles and joint angular velocities is a continuous signal, but is localized into some signals as shown in the bottom of Fig. 3 for making the learning of non-linear mappings easy. Here, the first joint angle that is from 0 to $\pi/2$ is divided into 8 signals, and the second one from 0 to π is divided into 12. Each of joint angular velocity from $-\pi$ to π is divided into 10. In order to examine whether the system learns feedback control or not, the joint angles, angular velocities and torques at one time step before are appended as inputs. Here feedback controlled is defined to control based on the past states and motion outputs. Totally 125 signals are the inputs of a layered neural network.

Actor-Critic architecture[8] is implemented in a four-layered neural network. The number of the output is three

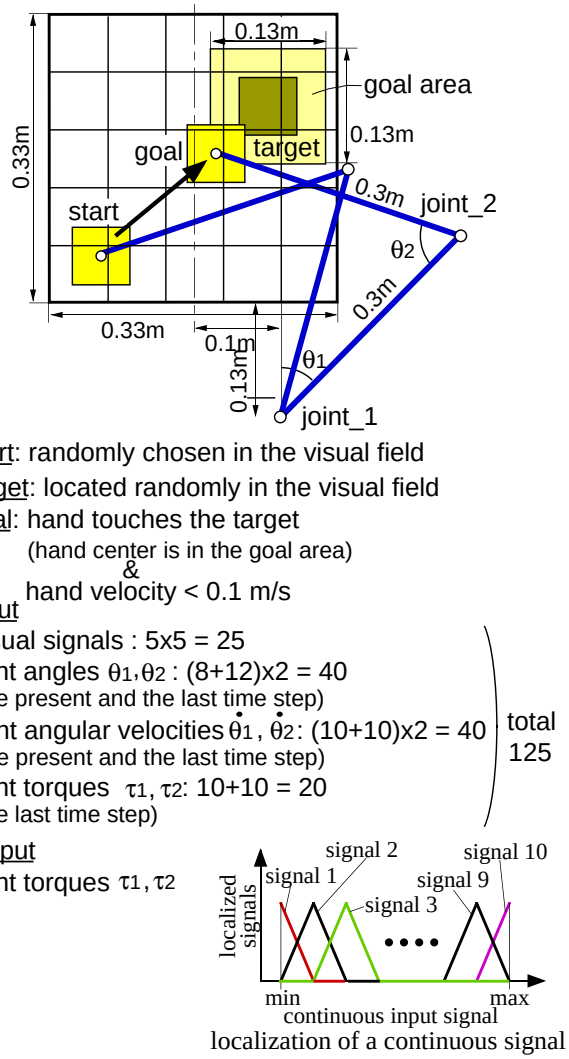


Figure 3: The robot hand-reaching task with a force field.

those are one for the critic and two for the actor. The output function of each neuron is sigmoid function with value range from -0.5 to 0.5. The two actor outputs are used as the torques τ for joint 1 and joint 2 respectively after linear transformation to the range in Table 1. The critic output is used without any transformations. The network has 2 hidden layers, and the numbers of hidden neurons are 30 for the lower layer and 10 in the upper respectively. The initial hand and target locations are decided randomly in the visual field at every trial. When the hand is overlapped with the target and the hand tangential velocity is less than 0.1 [m/s], reward 0.4 is given. When one joint angle becomes less than 0 degree or joint 1 angle becomes more than 90 degree, the trial is stopped and the penalty -0.4 is given. Otherwise, reinforcement signal is 0.0. At the early phase of the learning, The target is located close to the hand, and then the initial

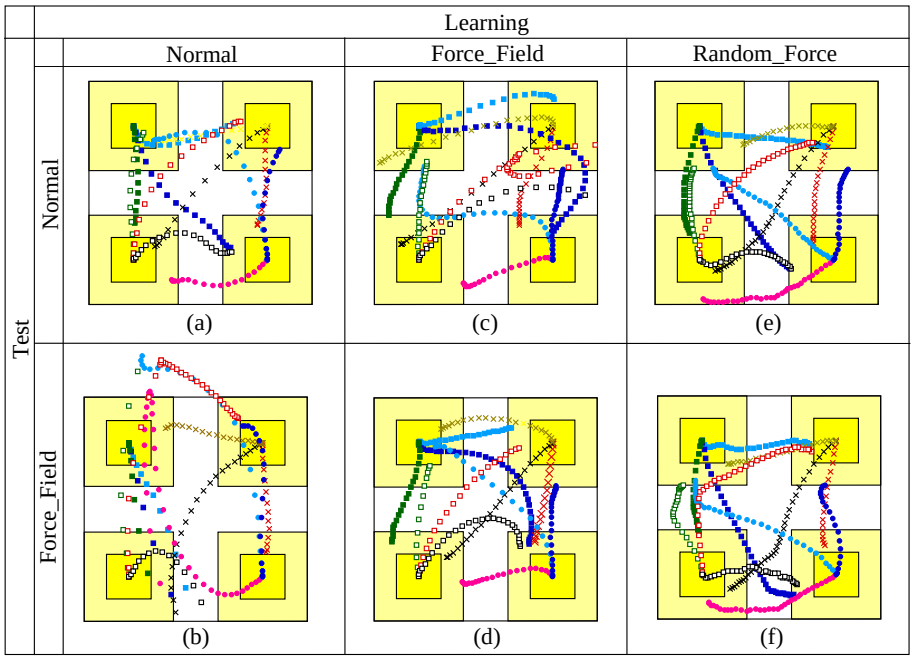


Figure 4: The difference of trajectories depending on the exposure of the force field. The start positions are the center of the squares at four corners of visual field. The goal state is that the hand center exists in the larger square that is filled with light gray with less than 0.1 m/s tangential velocity. The path from the upper left, lower left, upper right, lower right corner is drawn by small filled squares, empty squares, x, and filled circle respectively.

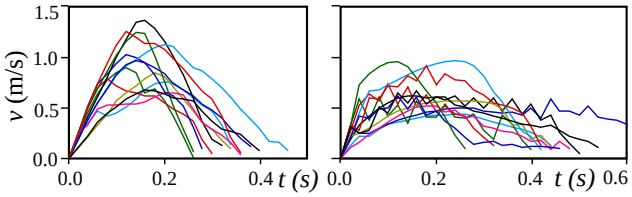
distance to the target is made larger gradually according to the learning performance. If the hand often fails to reach the target, the target is moved to the hand gradually in one trial.

4.3. External Force Load

In addition to the setting in the last section, external force was loaded to the hand. The learning are done under following three conditions. (1)viscosity force field as in [6] was loaded, (2)random force within 8[N] for each of x, y directions that was decided at every trial was loaded, and (3)no external force was loaded. Then in the test phase, the learning results are compared in the cases of no force field and viscosity force field as in [6]. The reason why the random force is loaded in the learning phase is to examine that by experience of various force load, the system becomes to control mainly by feedback control rather than feedforward control based on learning of inverse dynamics of the arm and the environment.

4.4. Results

Some example trajectories after 2,000,000 trials of learning are shown in Fig. 4 for two cases in which external force field is loaded or not in the test phase. The trajectories are drawn for the cases when the initial hand location and target location are chosen among upper right, upper left, lower



(a) in the case of in Fig. 4 (a) (b) in the case of Fig. 4 (d)

Figure 5: Profiles of the hand tangential velocity.

right and lower left. When the environment is the same between learning phase and test phase ((a) and (d)), the trajectories are expected to be almost a straight line, but are not so close to a straight line. However, comparing with the case that the force field is loaded in one of the learning phase and the test phase ((b) and (c)), it can be said that the paths are closer to a straight line. When the force field was loaded after the learning without the force field (b), the hand sometimes did not reach the target, but the paths are curved to the direction of the force field. When the force field was removed after the learning in the force field (c), the paths are curved to the other direction. The angle of the direction that the hand was pulled in Fig. (b) and the direction in Fig. (c), is almost 90 degree. That is the same as the result of human's case as shown in section 2. It can be said that the

after-effect was observed.

In the case of random force in the learning phase ((e) and (f)), the curve of the paths is not so strong in both force field and no force filed in the test phase. The paths are similar to each other.

The change of the hand tangential velocities is shown in Fig. 5 for the case of (a) and (d) in Fig. 4. In the case that the force field was loaded in both learning and test phases, velocity profile is not similar to bell shape.

Next, the reaching time is observed. The target and object were located on one of the $8 \times 8 = 64$ lattice on the visual sensor. Excluding the case in which the hand is reaching the target before it moves, the number of successful trials and failures were counted for the total 1040 combinations. The failure is classified into two cases, that the joint angle is out of the limit, and that the hand does not reach in 4 seconds. The numbers for two simulations for the different sets of initial weight values in the network are added and shown in Table 2.

Table 2: Learning results. The three numbers indicate the number of success-out-fail respectively. "Out" means that one of the joint angle is over the limitation. "Fail" means the the hand cannot reach in 4 sec. The number in parenthesis indicates the average reaching time to the target.

learning \ test	no force field	force field
no force_field	2071- 5- 4 (0.324sec)	1139-926- 15 (2.03sec)
force_field	1869-193- 18 (0.793sec)	2035- 38- 7 (0.467sec)
random_force	2038- 5- 37 (0.546sec)	1833-227- 20 (0.909sec)
random_force (no past state inputs)	1887- 0-193 (0.896sec)	1165-853- 62 (2.12sec)

The average time steps are also shown in parenthesis. When it failed to reach, the time was set to be 4.0 seconds. In the both cases in which the force field is loaded and not, the inverse dynamics is acquired only through reinforcement learning, and when the environment changes, it cannot reach the target in many cases.

In the case of random force, the number of success is not relatively small not depending on the force field loaded to the hand in the test phase. However, if the past information is removed from the inputs as the last row of Table 2, it often happens that the hand cannot reach when the force field is loaded. This indicates that the feedback control is employed. That may be because since the random force cannot be predicted, the feedforward control does not work, and only feedback control is effective. Only by applying reinforcement learning, the system changed to control selectively based on the feedback control through experiences.

5. CONCLUSION

By the combination of reinforcement learning and neural network, the system can reach its arm to the target on the visual sensor even if the force field is loaded to the hand. The neural network obtains the inverse dynamics of the arm and environment, and can control based on feedforward control. When the force field was loaded after the learning without force field, the hand path curved to the direction of the force. When the force field was removed after the learning in the force field, after-effect was observed. When the random force was loaded at every trial, feedback control is performed through learning. Since the hand path and velocity profile is not so similar to human's, there are still many things to be improved, but the authors think that the possibility could be shown that the human utilizes reinforcement learning to obtain the hand movement.

6. REFERENCES

[1] Uno, Y., et al. (1989) "Formation and control of optimal trajectory in human multijoint arm movement", *Biol. Cybern.*, **61**, pp. 89–101

[2] Flash, T. & Hogan, N. (1985) The coordination of arm movements: An experimentally confirmed mathematical model, *J. Neurosci.*, **5**: 1688-1703

[3] Kawato, M. (1992) "Optimization schemes and neural network models for formation and control of coordinated movement", *Attention and Performance XIV*, MIT Press, pp. 821–849

[4] Kawato, M., et al. (1987) "A hierarchical neural-network model for control and learning of voluntary movement", *Biol. Cybern.*, **57**, pp. 169–185

[5] Shibata, K., Sugisaka, M. and Ito, K. (2000) "Hand reaching movement acquired through reinforcement learning", *Proc. of The 2000 KACC (Korea Automatic Cont. Conf.)*, 90rd (CD-ROM)

[6] Shadmher, R. (1994) "Adaptive representation of dynamics during learning of a motor task", *J. NeuroSci.*, **15** (5), pp. 3208–3224, 1994.

[7] Thoroughman, K.A. and Shadmher, R. (1999) "Electromyographic correlates of learning an internal model of reaching movement", *J. Neurosci.*, **19** (19), pp.8573-8588

[8] A. G. Barto, R. S. Sutton & C. W. Anderson (1983) Neuronlike Adaptive Elements That Can Solve Difficult Learning Control Problems, *IEEE Trans. SMC*, **13**: 835-846

[9] Shibata, K., Ito, K. & Okabe, Y. (1998) "Direct-Vision-Based Reinforcement Learning in "Going to an Target"Task with an Obstacle and with a Variety of Target Sizes," *Proc. of Int'l. Conf. on Neu. Net. & Their Appli.(NEURAP) '98*, pp. 95–102.