On-line and Hierarchical Design Methods of Dynamics Based Information Processing System

Masafumi OKADA[*1], Daisuke NAKAMURA[*1] and Yoshihiko NAKAMURA[*1,*2]
[*1]Dept. of Mechano-Informatics, University of Tokyo
7-3-1 Hongo Bunkyo-ku Tokyo, 113-8656 Japan
[*2]CREST Program, Japan Science and Technology Corporation

Abstract

In this paper, we develop the on-line design method and the hierarchical design method of dynamics based information processing system for the robot intelligence. By using the forgetting parameter, dynamics memorizes a new robot motion forgetting an old motion, which means the plasticity of the system. The hierarchical structure enables information processing for complex and continuous environment. We implement the proposed method to a humanoid robot and realize the motion generation and transition.

Keywords: humanoid robots, dynamics, brain-like information processing, on-line design method, hierarchy

1 Introduction

The robot intelligence means a sensory motor mapping system that decides motor commands based on sensor signals. In the conventional approach for robot intelligence, the global goal is divided into sub-goals, which means the modularization of the system. Each module is written by an algorithm. Because algorithm is a calculation that gives an answer, the information processing is represented by the sequence of algorithm and it needs so many modules to handle the information in the real environment. Because it needs more than enough modules to deal with the dynamically changing environment, robots sometimes do not start moving until completing all modules finish caliculation. On the other hand, dynamics is a continous system and always gives output, and it will be an effective tool for the robot intelligence. The biological evidence of the dynamical phenomenon in organismic brain was shown by Freeman[1, 2, 3]. The rabbit olfactory build has dynamical phenomena such as order for known smell and chaos for unknown smell. On the other hand, Tsuda shows the effectiveness of the chaotic dynamics for learning and calls it 'chaotic itinerancy' for the phenomenon that the human brain transits some attractors [4]. These results show the close relationship between the dynamics and intelligence.

Some researches on robot intelligence using nonlinear dynamics have been proposed. These researches realize the human brain function using nonlinear dynamical phenomena. Nakamura proposed the motion control method for a mobile robot using chaotic dynamics[5, 6]. Kotosaka generated rhythmic motion using central pattern generator[7]. Matsuyama used dynamical system for the association of a time sequence data[8]. We proposed the dynamics-based information processing system using nonlinear dynamics that has some attractors with a polynomial configuration and realized a motion generation and transition for a humanoid robot[9].

Human brain cortex totalizes and recognizes the sensory information. This feature is classified into frontal cortex, lobus temporalis, parietal lobe and lobus occipitailis cerebri. These parts go to cyclic stability (attractor) by input signal and transmit the information to each other, which means the hierarchy of the system. This structure enables the information processing in the real world, and adapt to the changing environment with plasticity. In this paper, based on the dynamics based information processing system,

1. We propose on-line design method of the nonlinear dynamics that has an attractor to closed curved lines in N dimensional space. By setting the forgetting parameter, we realize plasticity of the dynamics.
2. We propose the hierarchical design method of the dynamics based information processing for the motion generation based on the input signal.

The proposed method is implemented to humanoid robot and realizes the motion generation and transition based on input sensor signal.

2 Dynamics-based information processing

2.1 Dynamics and robot whole body motion

In this section, we explain the relationship between a dynamics and robot whole body motions. Consider robot motion $\mathcal{M}$. The time sequence data of the joints angle of this motion is assumed to be $\boldsymbol{\xi}[k]$ that composes M as follows.

$$M = [\ \boldsymbol{\xi}[1] \quad \boldsymbol{\xi}[2] \quad \cdots \quad \boldsymbol{\xi}[m]\ ] \tag{1}$$
$$\boldsymbol{\xi}[k] = [\ \xi_1[k] \quad \xi_2[k] \quad \cdots \quad \xi_N[k]\ ]^T \tag{2}$$

m means a number of data and N means the number of degrees of freedom of the robot. Because M composes a carved line in N dimensional space, when $\mathcal{M}$ is a cyclic motion, M shows the closed curved line.

On the other hand, consider the dynamics represented by the following difference equation.

$$\boldsymbol{x}[k+1] = \boldsymbol{x}[k] + \boldsymbol{f}(\boldsymbol{x}[k]) \tag{3}$$
$$\boldsymbol{x}[k] = [\ x_1[k] \quad x_2[k] \quad \cdots \quad x_N[k]\ ]^T \tag{4}$$

If M is an attractor of this dynamics, the state vector $\boldsymbol{x}[k]$ converges to $\boldsymbol{\xi}[k]$ with initial state $\boldsymbol{x}[0]$, which means that the dynamics memorizes the motion M. And by picking up $\boldsymbol{x}[k]$ $(k = 0, 1, 2, \cdots)$, we can obtain $\boldsymbol{\xi}[k]$ $(k = 0, 1, 2, \cdots)$, which means that the dynamics reproduces time sequence data M of motion $\mathcal{M}$.

2.2 Design algorithm of the dynamics

In equation (3), we consider that $\boldsymbol{f}(\boldsymbol{x}[k])$ defines the vector field in N dimensional space. By using the polynomial functional approximation of the vector field, the dynamics can be calculated [9]. The design algorithm of the dynamics is as follows.

Step 1 Draw the closed curved line M in N dimensional space.

Step 2 Set an area D and define points $\boldsymbol{\eta}_i$ and vector of $\boldsymbol{f}(\boldsymbol{\eta}_i)$ so that the closed curved line M becomes attractor. Figure 1 shows the example of the definition of vector field. Area D contains M and $\boldsymbol{\eta}_i$ are defined near $\boldsymbol{\xi}[k]$

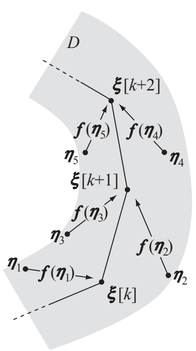


Figure 1: Definition of the vector field

Step 3 The defined vector $\boldsymbol{f}(\boldsymbol{\eta}_i)$ is approximated by the following equation by ℓ-th order polynomial of $\boldsymbol{\eta}_i$.

$$\boldsymbol{f}(\boldsymbol{\eta}) = \sum_{P=0}^{\ell} \sum_{\substack{p_1, \cdots, p_n \\ \sum p_i = P \\ p_i \text{ : non negative integer}}} a_{(p_1 p_2 \cdots p_n)} \prod_{i=1}^{n} \eta_i^{p_i} \tag{5}$$

$$\boldsymbol{\eta} = [\ \eta_1 \quad \eta_2 \quad \cdots \quad \eta_N\ ]^T \tag{6}$$

$a_{(p_1\ p_2\ \cdots p_N)}$ is a constant. Defining $\boldsymbol{f}(\boldsymbol{\eta})$ as

$$\boldsymbol{f}(\boldsymbol{\eta}) = \Phi(a_{(p_1\ p_2\ \cdots p_N)})\boldsymbol{\theta}(\boldsymbol{\eta}) \tag{7}$$
$$\boldsymbol{\theta}(\boldsymbol{\eta}) = [\ \eta_1^{\ell} \quad \cdots \quad \eta_N^{\ell} \quad \eta_1^{\ell-1}\eta_2 \quad \cdots \quad 1\ ]^T \tag{8}$$

Φ is calculated by the least square method as follows.

$$\Phi(a_{(p_1\ p_2\ \cdots p_N)}) = F\Theta^{\#} \tag{9}$$
$$F = [\ \boldsymbol{f}(\boldsymbol{\eta}_1) \quad \boldsymbol{f}(\boldsymbol{\eta}_2) \quad \cdots \quad \boldsymbol{f}(\boldsymbol{\eta}_m)\ ] \tag{10}$$
$$\Theta = [\ \boldsymbol{\theta}(\boldsymbol{\eta}_1) \quad \boldsymbol{\theta}(\boldsymbol{\eta}_2) \quad \cdots \quad \boldsymbol{\theta}(\boldsymbol{\eta}_m)\ ] \tag{11}$$

Φ is a constant parameter matrix that defines the dynamics in equation (3).

3 On-line design of the dynamics

3.1 On-line design method

In equation (9), Φ is designed based on the least square method. By using on-line least square method algorithm, the dynamics is on-line designed. The parameter matrix Φ^m in time step m is calculated by the iteration of the following on-line least square algorithm using a non-singular matrix P_m.

$$P_{m+1} = P_m - \frac{P_m \boldsymbol{\theta}(\boldsymbol{\eta}_{m+1})\boldsymbol{\theta}^T(\boldsymbol{\eta}_{m+1})P_m}{1 + \boldsymbol{\theta}^T(\boldsymbol{\eta}_{m+1})P_m\boldsymbol{\theta}(\boldsymbol{\eta}_{m+1})} \tag{12}$$
$$\Phi^{m+1} = \Phi^m + (\boldsymbol{f}(\boldsymbol{\eta}_{m+1}) - \Phi^m\boldsymbol{\theta}(\boldsymbol{\eta}_{m+1}))\boldsymbol{\theta}^T(\boldsymbol{\eta}_{m+1})P_{m+1} \tag{13}$$

P_m corresponds to

$$P_m = (\Theta\Theta^T)^{-1} \tag{14}$$

Figure 2 shows an example of the successive change of vector field $\boldsymbol{f}(\boldsymbol{x}[k])$ with $N = 2$. "+" means the arrow of the vector field. The length of vectors is normalized for easy view.

3.2 Set of the forgetting parameter

Set of the forgetting parameter to the on-line least square method causes the plasticity of the dynamics. As the alternation of equation (9), we consider the following weighted equation.

$$\Phi = \widehat{F}\widehat{\Theta}^{\#} \tag{15}$$
$$\widehat{F} = [\ \alpha F \quad \boldsymbol{f}(\boldsymbol{\eta}_{m+1})\ ] \tag{16}$$
$$\widehat{\Theta} = [\ \alpha\Theta \quad \boldsymbol{\theta}(\boldsymbol{\eta}_{m+1})\ ] \tag{17}$$

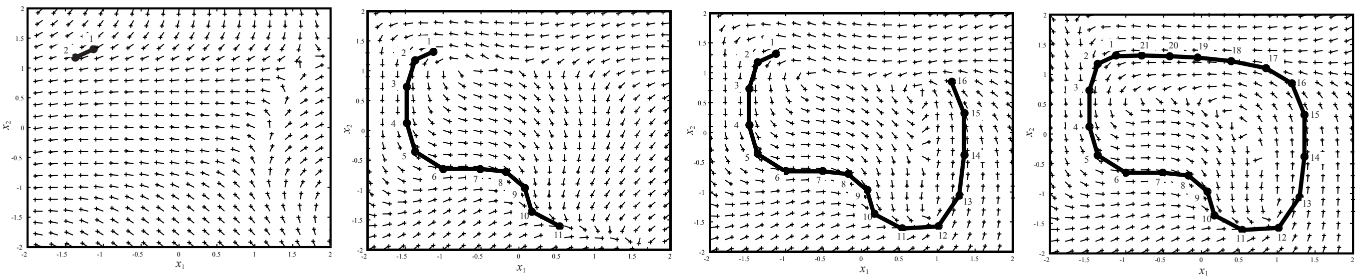


Figure 2: On-line design of dynamics

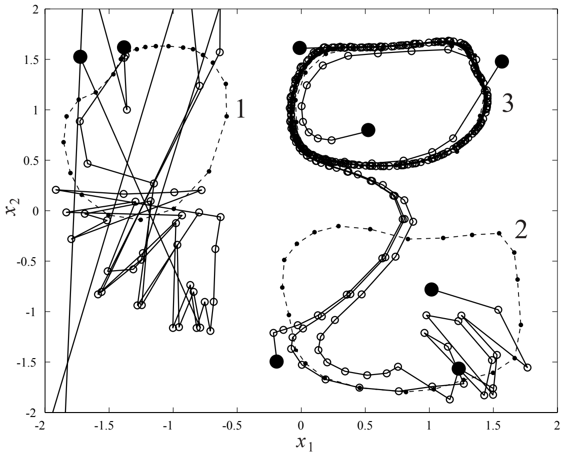


Figure 3: Motions of dynamics with weighted on-line algorithm

α $(0 < \alpha < 1)$ is a constant parameter. Φ is calculated by the same way as equation (12) and (13) after P_m is replaced by

$$P_m = \alpha^{-2} P_m \tag{18}$$

This method means the on-line forgetting least square method. By using this method, the dynamics memorizes the newer time sequence data forgetting the older one. Figure 3 shows the motion of the designed dynamics. The attractor is embedded by the order of $1 \to 2 \to 3$. Some $\bullet$ mean some initial states $\boldsymbol{x}[0]$. The first embedded closed curved line is forgot, a part of the attractor of the second closed curved line is remained. This means that the dynamics has plasticity.

4 Hierarchical design of the dynamics based information processing system

4.1 Hierarchical design

We successfully designed nonlinear dynamics for the information processing system. However, this system has only outputs. For robot intelligence, the dynamics is necessary to have input and output signals for sensory motor mapping system. In this section, by setting the inputs signal for the dynamics, we design the hierarchical model so that the dynamics transits the attractors because of the external inputs.

The hierarchical structure means the increase of the dimension of the dynamics, which causes the increase of design parameters. Design of the large dimensional dynamics is not only difficult because of the small computer power but also causes unclearness of the structure of the system, which prevent the heuristic design. On the other hand, the hierarchical structure enables the large dimensional dynamics with has clear structure.

4.2 Hierarchical structure

We set two spaces shown in Figure 4. One is the sensor space and the other is the motor space.

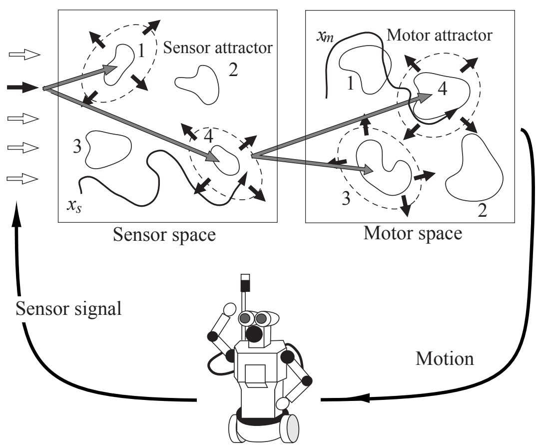


Figure 4: Hierarchical design of sensor space and motor space

Sensor space : The sensor space is a virtual torus space. There are some sensor attractors in the sensor space. The state vector $\boldsymbol{x}_s[k]$ moves based on the vector field and the basins of each attractor are decided by the external sensor signal. When the vector field on $\boldsymbol{x}_s[k]$ is zero (undefined), it takes chaotic motion so that $\boldsymbol{x}_s[k]$ goes around and searches another basin. By the change of the sensor signal, $\boldsymbol{x}_s[k]$ transits to the other attractor.

Motor space : In the motor space, there are some motor attractors that define the humanoid motions directory. The entrainment of the dynamics in the sensor space decides the basin of the motor space. The state vector $\boldsymbol{x}_m[k]$ in motor space moves according to the vector field and produces the humanoid motion. The change of motion causes the change of the sensor signal that means the feedback of the signal through the environment.

4.3 Local entrainment area and global basin

For the hierarchical design of the dynamics based information processing system, it needs the design strategy of the dynamics that has some attractors whose basin is changeable. Because the dynamics is designed using polynomial function with locally defined vector field $\boldsymbol{f}(\boldsymbol{\eta}_i)$ in equation (10), $\boldsymbol{f}(\boldsymbol{x}[k])$ should be effective only in the neighborhood of the attractor.

Local entrainment area : The difference equation in equation (3) is changed as follows.

$$\boldsymbol{x}[k+1] = \boldsymbol{x}[k] + w_1 \boldsymbol{f}(\boldsymbol{x}[k]) \quad (19)$$

where w_1 is defined using a_1 as follows.

$$w_1 = 1 - \frac{1}{1+\exp\{a_1(\omega_1(\boldsymbol{x}[k])-1)\}} \quad (20)$$

$$\omega_1(\boldsymbol{x}[k]) = (\boldsymbol{x}^T[k] - \boldsymbol{X}_0^T) Q (\boldsymbol{x}[k] - \boldsymbol{X}_0) \quad (21)$$

Q defines an ellipsoid that contains closed curved line M with the center $\boldsymbol{X}_0$. When $\boldsymbol{x}[k]$ is inside the ellipsoid, w_1 is 1 and otherwise 0.

Global basin : Global basin is defined by w_2 as the following equation,

$$\boldsymbol{x}[k+1] = \boldsymbol{x}[k] + w_2 (w_1 \boldsymbol{f}(\boldsymbol{x}[k]) + (1-w_1)\delta(\boldsymbol{X}_0 - \boldsymbol{x}[k])) \quad (22)$$

where δ $(0 < \delta < 1)$ and w_2 are defined as follows with a_2 and $\boldsymbol{X}_0$.

$$w_2 = 1 - \frac{1}{1+\exp\{a_2(\omega_2(\boldsymbol{x}[k])-1)\}} \quad (23)$$

$$\omega_2(\boldsymbol{x}[k]) = K(\boldsymbol{x}^T[k] - \boldsymbol{X}_0^T) Q (\boldsymbol{x}[k] - \boldsymbol{X}_0) \quad (24)$$

This means that K defines the range of global basin. Figure 5 shows the pattern diagram of the dynamics. w_1 is 1 when $\boldsymbol{x}[k]$ exists inside E_1 otherwise $w_1 = 0$. $\boldsymbol{f}(\boldsymbol{x}[k])$ is effective inside E_1 and inside E_2. Basin is defined outside E_1 and inside E_2. The attractor will disappear at $K \to \infty$.

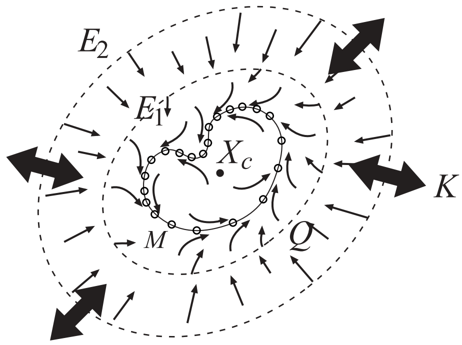


Figure 5: Change of basin of attractor

4.4 Design of nonlinear dynamics with multi attractors

The dynamics with multi attractors is designed by sum of vector fields as the following equation.

$$\boldsymbol{x}[k+1] = \boldsymbol{x}[k] + \sum_i w_{2i} \{w_{1i} \boldsymbol{f}_i(\boldsymbol{x}[k]) + (1-w_{1i})\delta_i(\boldsymbol{X}_{0i} - \boldsymbol{x}[k])\} \quad (25)$$

5 Motion generation of the humanoid robot

5.1 Upper body humanoid robot

In this section, we implement the proposed system to the humanoid robot. Figure 6 shows the upper body humanoid robot ' Robovie '. This robot has 3 DOF on

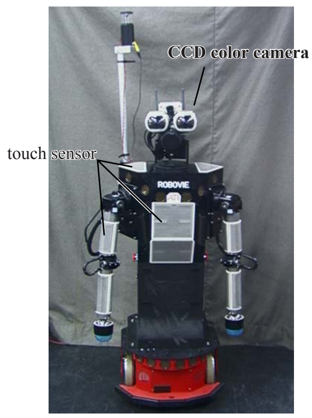
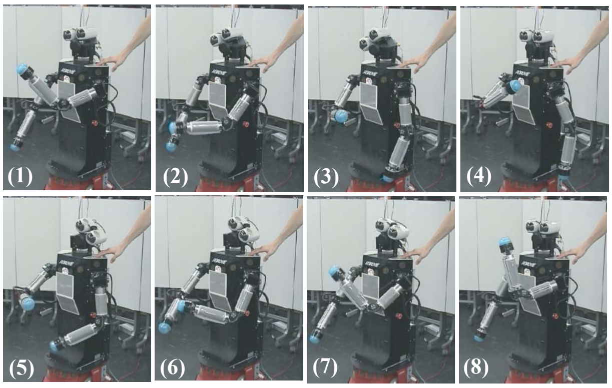
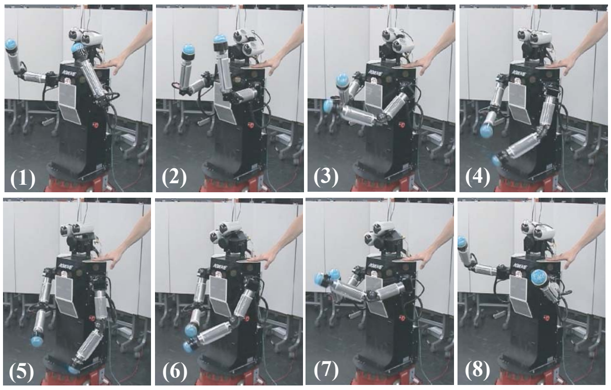


Figure 6: Humanoid robot Robovie

neck, 3 DOF on each shoulders and 1 DOF on each elbow. The total DOF is 11. This robot has 16 tactile sensors on head, shoulder, breast, upper arm, lower arm and wrist. Each sensors yield on-off signal. The color CCD camera obtains color information of the environment (percentage of red, green and blue). Total number of sensor signals is 19. We design 10 motions to this robot. Figure 7 shows the sample of the motion (motion1 and motion 2).



Motion 1



Motion 2

Figure 7: Motions of the humanoid robot

5.2 Design of the dynamics and network

Because this robot has 11 DOF, motor space (joint angle space) has 11th order dimension. However, to design the high dimensional dynamics is unrealistic because of the small computer power and numerical instability. In this paper, motor space is reduced to 3rd order dimension using principal component analysis reduction method [9] and the dynamics is designed as follows.

$$\begin{aligned} \boldsymbol{x}^m[k+1] = \boldsymbol{x}^m[k] + \sum_{i=1}^{10} w_{2i}^m \{ w_{1i}^m \boldsymbol{f}_i^m(\boldsymbol{x}^m[k]) \\ + (1 - w_{1i}^m)\delta_i^m(\boldsymbol{X}_{ci}^m - \boldsymbol{x}^m[k])\} \end{aligned} \quad (26)$$

$\boldsymbol{x}^m[k] \in \boldsymbol{R}^3$ means the state vector in motor space. Using $F_i(\in \boldsymbol{R}^{11\times 3})$ that decompresses $\boldsymbol{x}^m[k]$ to 11th order vector, the robot joint angle $\boldsymbol{y}[k](\in \boldsymbol{R}^{11})$ is calculated by

$$\boldsymbol{y}[k] = \sum_{i=1}^{10} w_{2i}^m w_{1i}^m F_i \boldsymbol{x}^m[k] \quad (27)$$

where $w_{2i}^m w_{1i}^m$ is used for the index of entrainment to i-th attractor. The parameter K_i^m in equation (24) that defines the basin of i-th attractor, is decided by the entrainment of $\boldsymbol{x}_i^s[k]$ that is the state vector in sensor space (subscript s means the sensor space). 10 attractors are embedded in the sensor space. The input vector from tactile sensor space (0 or 1) u_{ti} $(i = 1,\ 2,\ \cdots, 16)$ is defined as

$$\boldsymbol{u}_t = \left[\ u_{t1} \quad u_{t2} \quad \cdots \quad u_{t16}\ \right]^T \quad (28)$$

and the input vector from visual sensor space (percentage of red, green and blue) is defined as

$$\boldsymbol{u}_i = \left[\ u_{iR} \quad u_{iG} \quad u_{iB}\ \right]^T \quad (29)$$

The parameter $\boldsymbol{K}$ that decides the basin of the sensor space

$$\boldsymbol{K}^s = \left[\ K_1^s \quad K_2^s \quad \cdots \quad K_{10}^s\ \right]^T \quad (30)$$

is represented as follows using the weighting matrix ${}^sW_s$

$$\boldsymbol{K}^s = {}^sW_s \begin{bmatrix} \boldsymbol{u}_t \\ \boldsymbol{u}_i \end{bmatrix} \quad (31)$$

The parameter $\boldsymbol{K}^m$

$$\boldsymbol{K}^m = \left[\ K_1^m \quad K_2^m \quad \cdots \quad K_{10}^m\ \right] \quad (32)$$

that decides the basin of motor attractor is defined as follows.

$$\boldsymbol{K}^m = {}^mW_s \boldsymbol{w}_1^s \quad (33)$$

$$\boldsymbol{w}_1^s = \left[\ w_{21}^s w_{11}^s \quad w_{22}^s w_{12}^s \quad \cdots \quad w_{2\cdot 10}^s w_{1\cdot 10}^s\ \right]^T \quad (34)$$

$w_{2i}^s w_{1i}^s$ is used as the index of the entrainment of attractor that is same as equation (27).

5.3 Motion generation

Using the designed dynamics, we realize the humanoid motion generation and transition based on the sensor signal. The input of tactile sensor is changed as right shoulder $(t_1 < t < t_2) \rightarrow$ right head $(t_3 < t < t_4) \rightarrow$ right arm $(t_5 < t)$ for time t. We made two experiments with different timing of touches. Figure 8 shows the motion of the state vector in motor space. By the difference of the timing of the tactile sensor inputs, the different motion is generated, that means the humanoid motion depends on the internal space of the humanoid robot. Figure 9 shows the index of the entrainment of the attractor (fired attractor) in sensor space and generated motion. Density means the index value of the entrainment. Because the sequence of the tactile sensor input is same, same attractors are fired in case 1 and 2, however, because of the difference of timing of input signal, different motions are generated.

This results show that we realize the sensory motor mapping system using nonlinear dynamics with a hierarchical configuration. This system obtains the motor command successively. The network weighting function ${}^sW_s$ in equation (31) is designed so that the robot generates appropriate motions.

6 Conclusions

In this paper, we proposed the hierarchical design method using the dynamics based information processing system

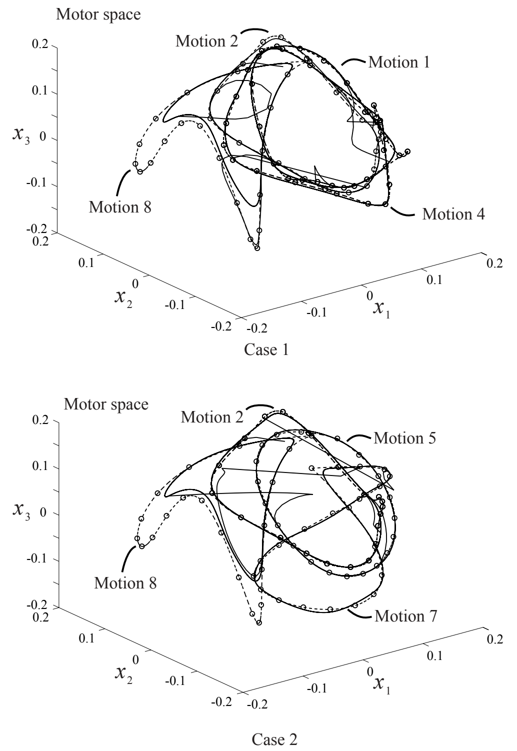


Figure 8: Motion of the dynamics in motor space

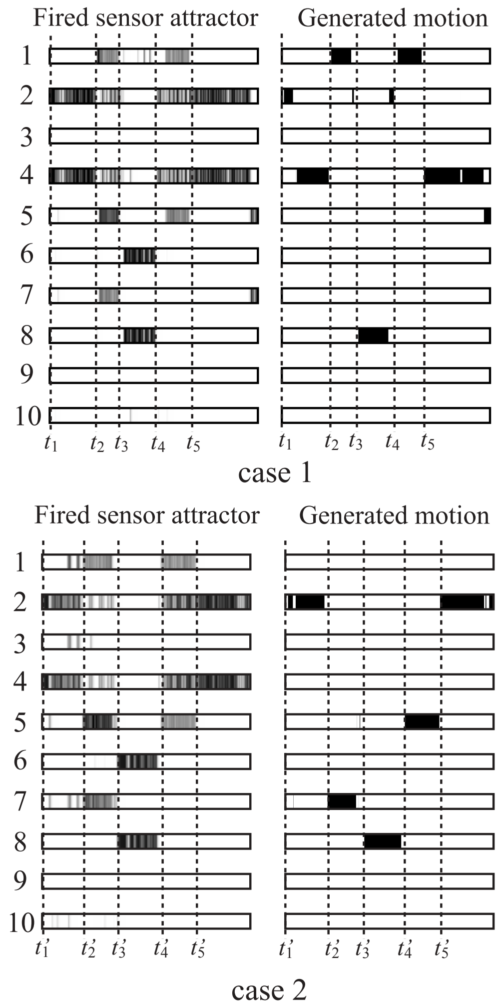


Figure 9: Fired attractors and generated motions

with polynomial function. The results of this paper are as follows.

- By setting the input and output to nonlinear dynamics, we design the hierarchical structure of dynamics based information processing system.
- By the change of basins of attractors, the state vector of the dynamics is entrained and transits to attractors.
- By the implementation of the designed dynamical system to the humanoid robot, we realize the motion generation and transition of the humanoid robot based on the sensor signal and the internal state.

Acknowledgments: This research is supported by the "Robot Brain Project" under the Core Research for Evolutional Science and Technology (CREST program) of the Japan Science and Technology Corporation.

References

[1] W.J.Freeman and W.Schneider : Changes in Spatial Patterns of Rabbit Olfactory EEG with Conditioning to Odors, Psychophysiology, Vol.19, pp.44–56, 1982.

[2] W.J.Freeman: Simulation of Chaotic EEG Patterns, Nonlinear Dynamic model of the Olfactory Systems, Biological Cybernetics, Vol.56, pp.139–150, 1987.

[3] Y.Yao and W.J.Freeman: Model of Biological Pattern Recognition with Spatially Chaotic Dynamics, Neural Networks, Vol.3 pp.153–160, 1990.

[4] I.Tsuda: Chaotic itinerancy as a dynamical basis of Hermeneutics in brain and mind, World Futures, 31 pp.105–122, 1991

[5] A.Sekiguchi and Y.Nakamura : The Chaotic Mobile Robot, Proc. of Systemics, Cybernetics and Informatics 2000 Vol.9, pp.463–468, 2000

[6] A.Sekiguchi and Y.Nakamura : Behavior Control of Robot Using Orbits of Nonlinear Dynamics, Proc. of IEEE International Conference on Robotics and Automation, pp.1647–1652, 2001

[7] S.Kotosaka and S.Schaal: Synchronized robot drumming by neural oscillators, Proc. of the International Symposium on Adaptive Motion of Animals and Machines, pp.8–12, 2000.

[8] H.Kawashima and T.Matsuyama: Integrated Event Recognition from Multiple Sources, Proc. of International Conference on Pattern Recognition, Vol.2, pp.785-789, 2002.

[9] M.Okada, K.Tatani and Y.Nakamura: Polynomial Design of the Nonlinear Dynamics for theBrain-Like Information Processing of Whole Body Motion, Proc. of IEEE International Conference on Robotics and Automation, pp1410-1415, 2002: