On Nonlinear Dynamics that Generates Rhythmic Motion with Specific Accuracy

Kei SENDA[1] **and Tsuyoshi TANAKA**[2]

[1]Dept. of Aerospace Eng., Osaka Prefecture Univ., Sakai, Osaka 599-8531, Japan, senda@aero.osakafu-u.ac.jp
[2]Graduate School of Eng., Osaka Prefecture Univ.

Abstract

This paper presents a method to generate rhythmic and cyclic motions observed in locomotion of animals or insects by using nonlinear dynamics, e.g., recurrent neural network (RNN). The proposed method enables to specify the approximation accuracy of the generated trajectory to the target trajectory though RNNs cannot easy to specify it. The method is based on a nonlinear oscillator generating cyclic motion and a the Fourier series. The realized dynamics has the desired trajectory as a limit cycle. A realization using neural networks is also shown. Effectiveness of the proposed method is examined by a numerical simulation where a space robot changes its orientation by the cyclic motion of the manipulator.

1. Introduction

Rhythmic and cyclic motions observed in locomotion, fluttering, and swimming of animals and insects are memorized in their brains or nervous networks. Those motion memories would not be stored as time histories but as limit cycles in nonlinear dynamical systems. For the stable generation of the motion, the cyclic trajectory is requested to be a steady attractor, which is called a dynamic associative memory (DAM). This study discusses the methodology to generate the rhythmic and cyclic motions of animals or robots using the DAMs.

Recurrent neural networks (RNNs) have been used generally to realize the DAMs and a back-propagation (BP)[1, 2] was proposed for the RNN. A multi-layered NN, e.g., multi-layered perceptron, can approximates any piecewise continuous function within a specific accuracy if the neural network (NN) has hidden units as many as necessary[3]. On the other hand, RNNs are not sure to express the desired dynamics nor to guarantee their learning convergence.

Consequently, this study proposes a methodology to realize the DAM that generates the desired rhythmic motion. The proposed method realizes a DAM based on a Fourier series and a standard oscillator with nonlinear units generating sinusoidal motion. The achieved DAM can generate any continuous cyclic trajectory, which is steady and multivariable vector function. It also makes the desired rhythmic motion be a limit cycle and attracts the trajectories started from almost all initial states to the desired. Moreover, the design procedure is established that ensures the DAM within the specific approximation accuracy evaluated by the mean squared error. Further, the DAM can be realized by using RNNs and LNNs.

The rest of this paper is organized as follows. Section 2 defines the desired dynamics that should be realized by DAMs and shows problems of the existing RNN, which is designed as in references[1, 2]. Section 3 represents the proposed DAM and illustrates its feasibility through a numerical simulation of a 1-degree-of-freedom (DOF) system. In the same section, a realization of the DAM using NNs is also mentioned. In section 4, effectiveness of the proposed method is examined by a numerical simulation where a space robot changes its orientation by the cyclic motion of the manipulator. Finally, concluding remarks are given by section 5.

2. Desired DAM and Problems in RNN

2.1. Specifications of desired DAM

This study uses DAMs to store the rhythmic and cyclic motions observed in locomotion, fluttering, and swimming of animals and insects. The followings are the specifications of the DAMs.

The desired cyclic trajectory is desired to be an attractor so as to start the trajectory from any initial state. It is also to be steady so as not to change its path. Hence, the desired cyclic trajectory is requested to be a steady limit cycle.

For a desired trajectory $\psi(t)$ with a period T, an allowable trajectory $\theta(t)$ is

$$\theta(t) = \psi(\tau) \tag{1}$$

where $\tau = \alpha(t)t + \Delta t$ must be a monotone increase function of t, $\alpha(t) \simeq 1$, and the phase angle $\phi \stackrel{\triangle}{=} \Delta t/T$ is in $0 \leq \phi < 2\pi$. This requests the generated cyclic trajectory has an almost constant period and any phase angle to the desired trajectory.

Multivariable vector trajectories should be generated by the DAM for locomotion and so on where many joints are moved. For the purpose, independent limit cycle for each joint is not good enough, but they must be synchronized.

It is better to specify the approximation accuracy of the generated trajectory to the desired. The DAM must be able to learn the desired trajectory by using some learning method. In addition, an assurance of the learning convergence is desirable.

One would like to start motion from almost any state because initial joint states are various in locomotion. For the purpose, the desired trajectory is to be a limit cycle with a large convergence region in state space.

The DAM realized by nonlinear differential equations is good for a robot control. However, the neural network realization is desired to understand the brains and the nervous systems, where the NN uses only neuron units observed in living bodies.

2.2. Problems in RNN realization

Recurrent neural networks (RNNs) have been generally studied to realize the DAMs. The RNNs are used to construct dynamical systems and a recurrent back-propagation (RBP)[1, 2] was proposed for learning. A multi-layered NN, e.g., multi-layered perceptron, can approximates any piecewise continuous function within a specific accuracy if it has hidden units as many as necessary[3]. On the other hand, RNNs are not sure to express the desired dynamics nor to guarantee their learning convergence.

Following references [1, 2], Figure 1 shows a numerical simulation, where a RNN has 20 neural units connected by each other, it has studied the desired trajectory 15,000 times by the RBP, and the trajectory is a constant rate circle with (0.5 0.5) in center and 0.4 in radius. The RNN has formed a limit cycle that attracts the generated trajectories to the desired, where no teaching signals for studying are given to the RNN for the first 5 s. But, the learned trajectory is not the right circle. Consequently, one must discuss if the RNN enable to express the desired dynamics before discussing convergence of learning.

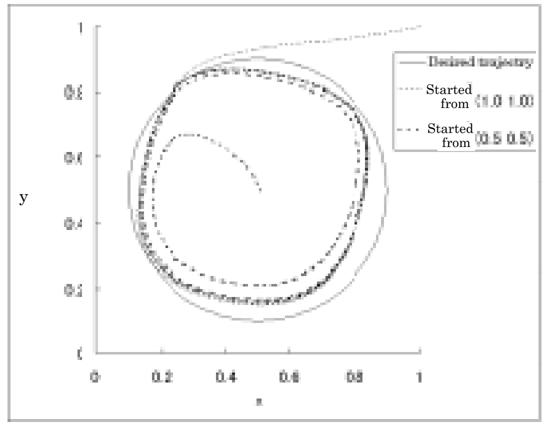

Figure 1: Learned trajectory by RBP following circular trajectory

3. DAM for Any Rhythmic Motion

3.1. Outline of DAM construction

The following outlines the design method of the DAM satisfying the specifications. Firstly, a standard oscillator is equipped to generate a sinusoidal oscillation with the desired period. Higher harmonic oscillators are constructed from the standard oscillator as many as necessary. The desired cyclic trajectory of time is then approximated by a Fourier series and its Fourier coefficients are obtained. The desired cyclic trajectory is generated by multiplying the harmonic oscillations to the Fourier coefficients and summing them up.

3.2. Fourier series approximation

A cyclic function of time t with a period $2L$ is approximated here by using a Fourier series. Assume that the trigonometric function system

$$1,\ \cos\frac{\pi}{L}t,\ \sin\frac{\pi}{L}t,\ \cos\frac{2\pi}{L}t,\ \sin\frac{2\pi}{L}t,\ \ldots \tag{2}$$

of t can be generated by dynamical systems, e.g., oscillators. If the function $f(t)$ with period $2L$ is piecewise smooth on a closed domain $[-L,\ L]$, then a Fourier series is generated from $f(t)$ as:

$$f(t) \sim \frac{a_0}{2} + \sum_{i=1}^{\infty}(a_i \cos\frac{i\pi}{L}t + b_i \sin\frac{i\pi}{L}t) \tag{3}$$

$$\begin{cases} a_0 = \frac{1}{L}\int_{-L}^{L} f(t)dt \\ a_i = \frac{1}{L}\int_{-L}^{L} f(t)\cos\frac{i\pi}{L}tdt \\ b_i = \frac{1}{L}\int_{-L}^{L} f(t)\sin\frac{i\pi}{L}tdt \end{cases} \quad (4)$$

where $a_0, \ a_1, \ a_2, \cdots$ and $b_1, \ b_2, \ \cdots$ are called Fourier coefficients of $f(t)$. One can consider that the function $f(t)$ with period $2L$ is piecewise smooth and continuous because it describes a motion. The Fourier series then converges to $f(t)$ absolutely and uniformly.

The trigonometrical polynomial using finite number of trigonometric functions

$$\begin{aligned} S_n(t) = \ & \frac{a_0}{2} + a_1\cos\frac{\pi}{L}t + \cdots + a_n\cos\frac{n\pi}{L}t \\ & + b_1\sin\frac{\pi}{L}t + \cdots + b_n\sin\frac{n\pi}{L}t \end{aligned} \quad (5)$$

approximates $f(t)$ with the minimum mean squared error

$$E(f - S_n) = \frac{1}{2L}\int_{-L}^{L}(f(t) - S_n(t))^2 dt \quad (6)$$

when the Fourier coefficients are used as $a_0, \ a_1, \ a_2, \ \cdots, \ a_n$ and $b_1, \ b_2, \ \cdots, \ b_n$. Therefore, one can specify the approximation accuracy of S_n by selecting the number n that makes the mean squared error $E(f - S_n)$ of Eq. (6) less than the specific value.

3.3. Standard oscillator and higher harmonic oscillations

In the previous subsection, one assumes that the trigonometric function system, Eq. (2), is generated by dynamical systems. This subsection presents a method to generate it. A standard oscillator is firstly equipped by von der Pol's equation. Higher harmonic oscillators are then constructed from the standard oscillator as many as necessary.

3.3.1. von der Pol's oscillator

A nonlinear dynamics is realized by von der Pol's oscillator, where the dynamics has a limit cycle with a constant rate trajectory on an unit circle:

$$\frac{d\boldsymbol{x}}{dt} = \begin{bmatrix} \dot{x} \\ \epsilon(\omega^2 - \omega^2 x^2 - \dot{x}^2)\dot{x} - \omega^2 x \end{bmatrix} \quad (7)$$

$$\boldsymbol{y} = \boldsymbol{\Omega}\boldsymbol{x} \quad (8)$$

where $\boldsymbol{x} = [x \ \ \dot{x}]^T$, $\boldsymbol{x} = [y_1 \ \ y_2]^T$ and $\boldsymbol{\Omega} = \text{diag}[1, \ 1/\omega]$, $\epsilon > 0$. To discuss the stability of the oscillator, let a Lyapunov function be

$$V(x, \dot{x}) = \frac{1}{2}(\omega^2 x^2 + \dot{x}^2) \quad (9)$$

then $V(0,0) = 0$ and $V(x, \dot{x}) > 0$ is satisfied when $\boldsymbol{x} \neq \boldsymbol{0}$ or $\dot{\boldsymbol{x}} \neq \boldsymbol{0}$. Since the time derivative is

$$\begin{aligned} \dot{V} &= \omega^2 x\dot{x} + \dot{x}\ddot{x} \\ &= \omega^2 x\dot{x} + \dot{x}(\epsilon(\omega^2 - \omega^2 x^2 - \dot{x}^2)\dot{x} - \omega^2 x) \\ &= \epsilon(\omega^2 - \omega^2 x^2 - \dot{x}^2)\dot{x}^2 \\ &= \epsilon(\omega^2 - 2V)\dot{x}^2 \end{aligned} \quad (10)$$

sign of $\dot{V}$ changes beyond $V = \frac{\omega^2}{2}$. Hence V increases in $0 < V < \frac{\omega^2}{2}$ with $\dot{V} > 0$ and V decreases in $\frac{\omega^2}{2} < V$ with $\dot{V} < 0$. Therefore $V \to \frac{\omega^2}{2}$ as $t \to \infty$. It means that the solution would be constrained on the ellipse $\omega^2 x^2 + \dot{x}^2 = \omega^2$. The ellipse is a limit cycle because no singular point exists on the ellipse and the state vector travels with nonzero rate. The parameter ϵ gives the speed of convergence to the ellipse and the convergence becomes quicker the larger ϵ is.

Because the steady state solution is $\omega^2 x^2 + \dot{x}^2 = \omega^2$, substituting it into Eq. (7) yields

$$\dot{\boldsymbol{x}} = \begin{bmatrix} 0 & 1 \\ -\omega^2 & 0 \end{bmatrix} \boldsymbol{x} \quad (11)$$

The solution becomes $x(t) = \sin\omega t$ if the initial state is $\boldsymbol{x}(0) = [0 \ \omega]^T$ at $t = 0$ and one obtains $\dot{x}(t) = \omega\cos\omega t$. Hence the output becomes

$$\boldsymbol{y} = \begin{bmatrix} \sin\omega t \\ \cos\omega t \end{bmatrix} \quad (12)$$

Because Eq. (7) generates the limit cycle, a solution started from another initial state converges to

$$\boldsymbol{y} = \begin{bmatrix} \sin(\omega t + \phi) \\ \cos(\omega t + \phi) \end{bmatrix} \quad (13)$$

where a phase ϕ in $0 \leq \phi < 2\pi$ depends on the initial state. Eq. (12) can be considered as the steady state output when $t + \phi/\omega$ is redefined as t.

3.3.2. Higher harmonic oscillators

Higher harmonic oscillators are needed for the Fourier series approximation. For the purpose, a standard oscillator $g(x, \dot{x})$ is firstly equipped by the von der Pol's equation with a frequency $\omega = \pi/L$ and the output $\boldsymbol{y} = [y_1 \ y_2]^T = [\sin\omega t \ \ \cos\omega t]^T$ is obtained. A higher harmonic oscillation with a frequency $j\omega$ is generated by

$$\sin j\omega t = \sum_{i=0}^{(j-1)/2} \begin{pmatrix} j \\ 2i+1 \end{pmatrix} (-1)^i y_1^{2i+1} y_2^{j-(2i+1)}$$
$$\cos j\omega t = \sum_{i=0}^{j/2} \begin{pmatrix} j \\ 2i \end{pmatrix} (-1)^i y_1^{2i} y_2^{j-2i} \quad (14)$$

As a result, all the higher harmonic oscillations can be generated by the standard oscillator $g(x, \dot{x})$.

3.4. Generation of rhythmic motion

The cyclic function $f(t)$ with a period $2L$ is approximated by the following procedure.

(i) Evaluate the approximation accuracy of Eq. (5) to $f(t)$ by the mean squared error of Eq. (6) and select the number n. Calculate Fourier coefficients by Eq. (4) and define $\boldsymbol{w}$ as:

$$\boldsymbol{w}^T = \begin{bmatrix} \frac{1}{2}a_0 & b_1 & a_1 & \cdots & b_n & a_n \end{bmatrix} \quad (15)$$

(ii) Equip the standard oscillator with the frequency $\omega = \pi/L$ and the amplitude 1 by Eqs. (7) and (8). Any initial state of $\boldsymbol{x} \neq \boldsymbol{0}$ can be selected because the output $\boldsymbol{y}_1 = [y_{11} \ y_{12}]^T$ converges to the standard oscillation from any initial state other than $\boldsymbol{x} = \boldsymbol{0}$. Select $\boldsymbol{x}(0) = [0 \ \omega]^T$ as the initial condition when the standard oscillation should follow just after the initial time. Initial condition setting will be discussed later.

(iii) Define new output $\boldsymbol{\eta}$ including the higher harmonic oscillations as:

$$\boldsymbol{\eta} \triangleq \begin{bmatrix} 1 & \boldsymbol{y}_1^T & \boldsymbol{y}_2^T & \cdots & \boldsymbol{y}_n^T \end{bmatrix}^T \quad (16)$$

Generate $\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots$, and $\boldsymbol{y}_n$ by Eq. (14).

(iv) Finally, generate the motion $z \simeq f(t)$ as:

$$z = \boldsymbol{w}^T \boldsymbol{\eta} \quad (17)$$

One often would like to start the trajectory from any initial condition $(z(0), \ \dot{z}(0))$ because the generated $z(t)$ is the motion of animals or robots. In this case, solve $\boldsymbol{y}_1$ from Eq. (17) as an independent variable and determine $\boldsymbol{y}_1(0)$. Calculate $\boldsymbol{x}(0)$ satisfying Eq. (12) with $\boldsymbol{y}_1(0)$ and give it to Eq. (7) as the initial condition. One can then start the motion from the specific initial condition $(z(0), \ \dot{z}(0))$ and attract it to the desired cyclic trajectory.

3.5. Extension to multivariable vector function

Multivariable vector function $\boldsymbol{f}(t)$ would be approximated by the following equations for the motion of animals and robots though the scalar function $f(t)$ has been discussed:

$$\begin{aligned} \boldsymbol{f}(t) &\triangleq [f_1(t) \ \ f_2(t) \ \cdots \ f_m(t)]^T \\ &\simeq [z_1(t) \ \ z_2(t) \ \cdots \ z_m(t)]^T \triangleq \boldsymbol{z}(t) \end{aligned} \quad (18)$$

where each z_i is obtained as

$$z_i = \boldsymbol{w}_i^T \boldsymbol{\eta}_i \quad (19)$$

by using Eq. (17). All $\boldsymbol{\eta}_i$s, i.e., $\boldsymbol{x}_i$s, must become equal since z_is are to be synchronized. In other words, phase angle ϕ_is must be equivalent for all is. An approach is to use only one common standard oscillator for all z_is but one cannot set the independent initial condition for each z_i. Accordingly, a standard oscillator of Eqs. (7) and (8) is equipped to each i and their phase angles are synchronized by the following equation. In order to catch up the oscillator j with the maximum phase angle ϕ_j, frequency ω_i $(i \neq j)$ of other oscillators are modulated as:

$$\omega_i = \omega + \varepsilon \left(1 - \frac{\boldsymbol{x}_i \cdot \boldsymbol{x}_j}{|\boldsymbol{x}_i| \cdot |\boldsymbol{x}_j|} \right) \quad (20)$$

where parameter ε gives speed of the synchronization.

3.6. NN realization

Discussed here is the realization of the DAM with NNs. Figure 2 shows a schematic diagram of the DAM, where the oscillator part is a RNN and the Fourier series part is a layered NN (LNN). It is known that the oscillators are realized by the RNNs composed of nonlinear neuron units [4, 5]. The Fourier series part is obviously achieved by a LNN if the Fourier coefficients are considered as the LNN's connecting weights. Therefore, the proposed DAM can be realized by the NN composed of the RNN and the LNN.

3.7. Numerical example

Figure 3 is a phase portrait of the desired trajectory and the trajectory generated by the proposed DAM. The generated trajectory asymptotically converges to the desired trajectory from the initial state that is not on the desired trajectory. The generated trajectory tracks the desired with almost no error after the convergence. The DAM successfully makes the desired trajectory be a limit cycle by using the von der Pol's oscillators.

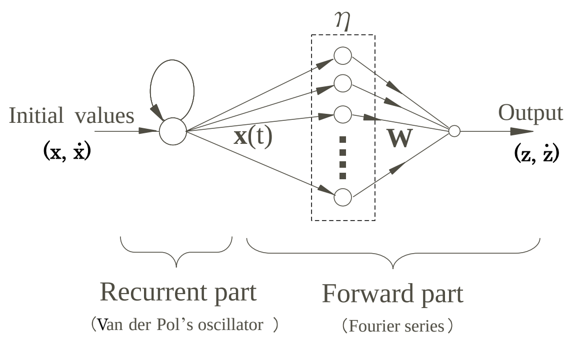

Figure 2: Construction of NN for proposed dynamic associative memory

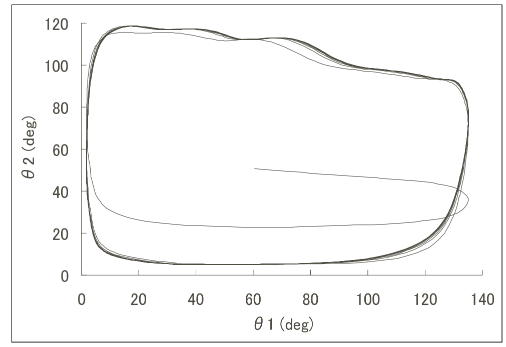

Figure 4: Generated trajectory in configuration space

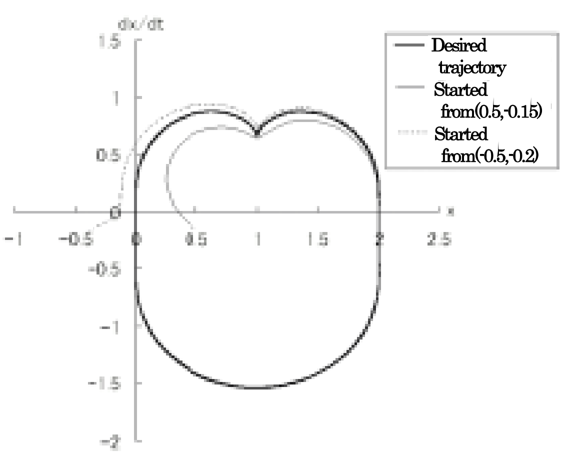

Figure 3: Phase portrait of generated and desired trajectories

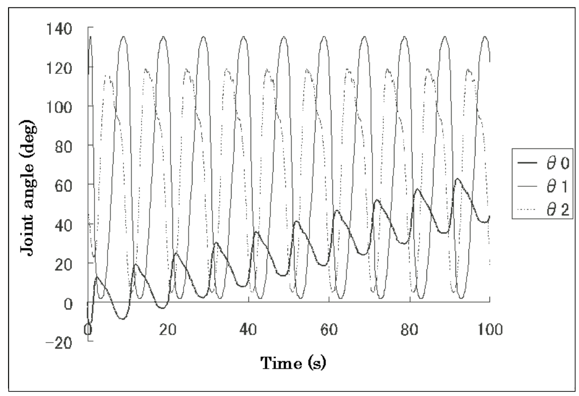

Figure 5: Time history of joint and attitude angles

4. Application to Space Robot Reorientation

A free-floating space robot is subjected to the non-holonomic constraint due to the angular momentum conservation. The attitude of the satellite vehicle may be changed during the manipulator operation. The system's orientation in the final state is not determined uniquely by the specific configuration of the manipulator since the final attitude of the satellite vehicle is dependent on the trajectory of the manipulator. Using the characteristics, the satellite attitude can be controlled. Shown here is a numerical simulation, where a space robot changes its orientation by the cyclic motion of the manipulator that is generated by the proposed DAM.

The mathematical model corresponds to the experimental system[6] simulating a space robot where the robot model composed of two SCARA type manipulators and a satellite vehicle can move freely on a two-dimensional planar table without friction by using air-bearings. For the reorientation, the robot drives only the shoulder and the elbow joints of one arm. The desired trajectory is based on the trajectory planned in references[7, 8]. Figure 4 illustrates the generated trajectory and the desired trajectory in the configuration space of the the shoulder angle θ_1 and the elbow θ_2. The generated trajectory converges to the desired as time passes. Figure 5 is the time history of the joint angles and satellite attitude angle θ_0. Figure 6 shows the motion of the space robot. The satellite attitude changes gradually.

5. Concluding Remarks

This study has proposed the methodology to realize the dynamic associative memory (DAM) that generates rhythmic and cyclic motions of animals and insects. The proposed DAM is based on the nonlinear oscillator and a Fourier series. It has the following characteristics.

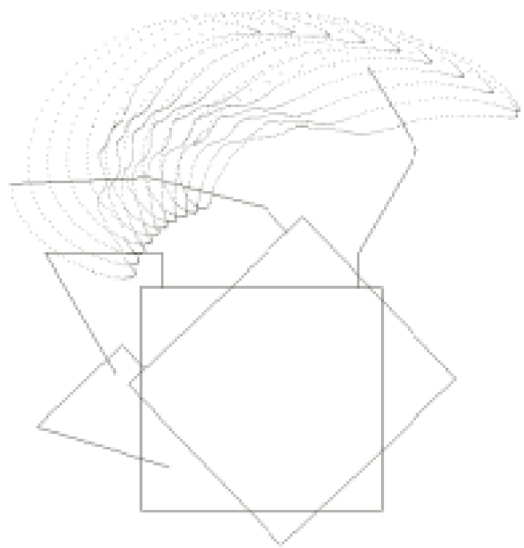

Figure 6: Motion of space robot through cyclic motion

(a) The desired cyclic trajectory can be a steady attractor, which does not change as time passes. Hence the generated trajectory is attracted to the desired trajectory.

(b) The DAM can generate multivariable vector functions.

(c) The DAM can generate the trajectory with the specific approximation accuracy evaluated by the mean squared error to the desired. The Fourier coefficients give the best approximation to the desired trajectory.

(d) The motion can start from almost any initial state because the von der Pol's oscillator is a limit cycle that attracts all trajectories started form points other than the origin.

(e) The proposed DAM can be realized by the NN composed of the RNN and the LNN.

Effectiveness of the proposed method has been examined by the numerical simulation of the space robot reorientation by the cyclic motion of the manipulator.

References

[1] Pearlmutter, B. A., "Learning State Space Trajectories in Recurrent Neural Networks," *Neural Computation*, Vol. 1, No. 2, 1989, pp. 263–269.

[2] Pineda, F. J., "Generalization of Backpropagation to Recurrent Neural Networks," *Physical Review Letters*, Vol. 59, No. 19, 1987, pp. 2229–2232.

[3] Funahashi, K. I., "On the Approximate Realization of Continuous Mapping by Neural Networks," *Neural Networks*, Vol. 2, 1989, pp. 183–192.

[4] Nakano, K., *An Introduction to Neurocomputing*, Corona Publishing, 1990. (in Japanese)

[5] Sawada, M. and Okabe, Y., "Generation of Rhythmic Motions of Multi-Joint Robots with Autonomic Distributed Neural Network Which Have Learning Systems," Proc. of 12th SICE Symposium on Decentralized Autonomous Systems, 2000, pp. 471–474

[6] Senda, K. et al., "A Hardware Experiment of Space Truss Assembly by Using Space Robot Simulator," *Proc. of 9th Workshop on Astrodynamics and Flight Mechanics*, Sagamihara, ISAS, July 22–23, 1999, B-16.

[7] Senda, K., Murotsu, Y., and Ozaki, M., "A Method of Attitude Control for Space Robots," *Trans. Japan Society of Mechanical Engineers*, Ser. C, Vol. 57, No. 539, 1991, pp. 2356–2362. (in Japanese)

[8] Senda, K., "Dynamics and Control of Rigid/ Flexible Space Manipulators", *Ph. D. Thesis, Osaka Prefecture University*, Osaka, Japan, 1993.