

# A Novel Tuning Method for Neural Oscillators with a Ladder-Like Structure Based on Oscillation Analysis

Yuya Hattori<sup>1,2</sup>, Michiyo Suzuki<sup>2</sup>, Zu Soh<sup>1</sup>,  
Yasuhiko Kobayashi<sup>2</sup>, and Toshio Tsuji<sup>1</sup>

<sup>1</sup> Graduate School of Engineering, Hiroshima University,  
1-4-1 Kagamiyama, Higashi-Hiroshima, Hiroshima 739-8527, Japan  
{hattori,sozu,tsuji}@sys.hiroshima-u.ac.jp

<sup>2</sup> Microbeam Radiation Biology Group, Japan Atomic Energy Agency,  
1233 Watanuki, Takasaki, Gunma 370-1292, Japan  
{suzuki.michiyo,kobayashi.yasuhiko}@jaea.go.jp

**Abstract.** Neural oscillators with a ladder-like structure is one of the central pattern generator (CPG) model that is used to simulate rhythmic movements in living organisms. However, it is not easy to realize rhythmical cycles by tuning many parameters of neural oscillators. In this study, we propose an automatic tuning method. We derive the tuning rules for both the time constants and the coefficients of amplitude by linearizing the nonlinear equations of the neural oscillators. Other parameters such as neural connection weights are tuned using a genetic algorithm (GA). Through numerical experiments, we confirmed that the proposed tuning method can successfully tune all parameters.

**Keywords:** Central pattern generator (CPG), neural oscillators, parameter tuning, genetic algorithm, *C. elegans*.

## 1 Introduction

The central pattern generator (CPG) is a network of neuronal cells that control rhythmic reciprocating movements such as walking (humans), meandering (snakes), and swimming (fish) [1]. The CPG generates rhythmic electric signals via signal transduction between neurons. These internally-generated signals are then modified on the basis of sensory information about the external environment, received via sensory neurons. A number of mathematical models, termed ‘neural oscillators’, have been developed to describe how CPG’s function [2], [3].

Recently, several research groups have simulated rhythmic movements in animals using neural oscillators [3], [4]. For example, Ekeberg developed a neural oscillator for the lamprey CPG which was then used to conduct swimming simulations [3]. Suzuki et al. constructed a neural network model of the nematode *C. elegans* [4], [5] using Matsuoka’s neural oscillators and then used this model to simulate rhythmic movements such as sinusoidal locomotion [4], [6].

In all of these simulations, the neural oscillators had a ladder-like structure. This structure incorporates a large number of parameters which makes tuning difficult when attempting to reproduce rhythmic reciprocating movements on a computer. Parameter-tuning rules and the methods for tuning for these parameters to generate the desired output have only been evaluated for neural oscillators that are composed of a small number of oscillators [7], [8]. There is currently no effective tuning method for setting the parameters of a neural oscillator that is composed of a large number of oscillators. Thus, the parameters included in neural oscillators that have a ladder-like structure are currently tuned by trial-and-error.

In this dissertation, we propose an automatic tuning method for all parameters included in neural oscillators with a ladder-like structure. In this method, the tuning rules for both the time constants and the coefficient of amplitude are derived by linearizing the nonlinear equations of neural oscillators. Other parameters, such as the neural connection weights, are tuned using a genetic algorithm (GA) [9]. To avoid stagnation in GA-based tuning ('tuning' denotes 'learning' in a GA) for a large number of parameters, we propose a two-step GA. This consists of a GA for the early stages of tuning and a GA for the mid-late stage of tuning. The former GA reduces large scale learning errors and the latter reduces errors for each oscillator individually. We evaluate the effectiveness of the proposed method through numerical experiments of rhythmic-signal generation in *C. elegans*.

## 2 Characteristic Analysis of Neural Oscillators with a Ladder-Like Structure

### 2.1 Neural Oscillators with a Ladder-Like Structure [2]

Matsuoka's model is representative of neural oscillators that have a ladder-like structure. This model is composed of excitatory oscillators (white circle in Fig. 1) and inhibitory oscillators (gray circle in Fig. 1). An excitatory oscillator connects with adjacent excitatory oscillators that are located in the  $N$ th column of the 2nd row. An inhibitory oscillator connects with the corresponding excitatory oscillator. The strength of signal transduction of the neural connection between

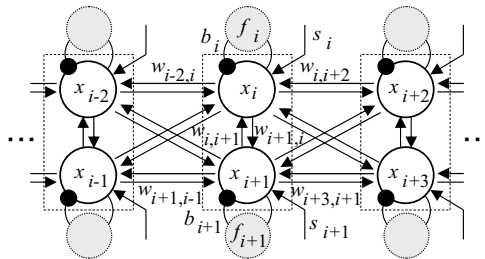


Fig. 1. Schematic of the neural oscillator with a ladder-like structure

two excitatory oscillators is represented by the connection weight  $w_{i,j}$  ( $i \neq j \in \{1, 2, \dots, 2N\}$ ). Three types of signals are input to each excitatory oscillator from adjacent oscillators. These include signals from excitatory oscillators,  $w_{i,j}x_j$ , signals coding for an adaptation effect from an inhibitory oscillator,  $b_i f_i$ , and a sensory feedback signal,  $s_i$ .  $b_i$  is a fatigue coefficient and  $f_i$  is an output of the inhibitory oscillator. The output  $x_i$  of the excitatory oscillator and the output  $f_i$  of the inhibitory oscillator are represented by the following equations [2]:

$$T_{ri} \frac{dx_i}{dt} + x_i = \sum_{j=1, j \neq i}^{2N} w_{i,j} h(x_j) - b_i f_i + s_i \tag{1}$$

$$T_{ai} \frac{df_i}{dt} + f_i = h(x_i) \tag{2}$$

$$h(\epsilon) = \max(\epsilon, 0) \tag{3}$$

$$y_n = \alpha_n (h(x_{2n}) - h(x_{2n-1})) \tag{4}$$

where,  $T_{ri}$  and  $T_{ai}$  are the time constants and  $h(\epsilon)$  is a threshold function of  $x_j$ .  $y_n$  is the total output of a pair of oscillators in the  $n$  ( $n \in \{1, 2, \dots, N\}$ )th column.  $\alpha_n$  is a coefficient of the output amplitude. To generate a desired output for each pair of oscillators,  $w_{i,j}$ ,  $b_i$ ,  $T_{ri}$ ,  $T_{ai}$ , and  $\alpha_n$  must be properly tuned.

### 2.2 Derivation of the Tuning Parameters Based on Oscillation Analysis of the Neural Oscillators

Although proper tuning of the parameters included in a neural oscillator is important to generate the desired output, a detailed analysis of a neural oscillator is difficult because of its nonlinear characteristics. Therefore, we evaluated the relationship between the neural oscillator output and the input parameters. The nonlinear equations that describe the neural oscillator are linearized at the equilibrium position as follows:

First, equation (3) is approximated by the differentiable function  $\tilde{h}(\epsilon) = \epsilon/(1 + e^{-T\epsilon})$ , where,  $T$  is a constant with a large value. Under this condition, equations (1) and (2) are linearized with an equilibrium position  $\bar{x}_i$  of  $dx_i/dt = 0$  and  $\bar{f}_i$  of  $df_i/dt = 0$ , respectively. Thus, equations (1) and (2) are represented by the following:

$$T_{\beta} r_i \frac{dx_i}{dt} + x_i = \sum_{j=1, j \neq i}^{2N} w_{i,j} \tilde{h}'(\bar{x}_j) x_j - b_i f_i \tag{5}$$

$$T_{\beta} a_i \frac{df_i}{dt} + f_i = \tilde{h}'(\bar{x}_i) x_i \tag{6}$$

where,  $\tilde{h}' = d\tilde{h}/d\epsilon$ . The time constants  $T_{ri}$  and  $T_{ai}$  are represented by  $T_{ri} = T_{\beta} r_i$  and  $T_{ai} = T_{\beta} a_i$ .  $T_{\beta}$  is a time constant that is common in all oscillators, and  $r_i$  and  $a_i$  are coefficients of the time constant. In equation (5),  $s_i$  is deleted because of the time-invariant input. To solve the simultaneous linear differential equations (5) and (6), these equations are expressed using the following matrices:

$$\left. \begin{aligned} \begin{bmatrix} \dot{\mathbf{X}} \\ \dot{\mathbf{F}} \end{bmatrix} &= \frac{1}{T_\beta} \mathbf{A} \begin{bmatrix} \mathbf{X} \\ \mathbf{F} \end{bmatrix} \\ \mathbf{A} &= \begin{bmatrix} \mathbf{W}' & \mathbf{B}' \\ \mathbf{H}' & \mathbf{T}' \end{bmatrix} \end{aligned} \right\} \quad (7)$$

where,  $\mathbf{X}$ ,  $\mathbf{F}$ ,  $\dot{\mathbf{X}}$ , and  $\dot{\mathbf{F}}$  are matrices relating to the output of the excitatory oscillator  $x_i$  and the inhibitory oscillator  $f_i$ . These matrices are represented by:

$$\dot{\mathbf{X}} = \begin{bmatrix} \frac{dx_1}{dt} \\ \frac{dx_2}{dt} \\ \vdots \\ \frac{dx_{2N}}{dt} \end{bmatrix}, \dot{\mathbf{F}} = \begin{bmatrix} \frac{df_1}{dt} \\ \frac{df_2}{dt} \\ \vdots \\ \frac{df_{2N}}{dt} \end{bmatrix}, \mathbf{X} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{2N} \end{bmatrix}, \mathbf{F} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_{2N} \end{bmatrix}$$

$\mathbf{W}'$ ,  $\mathbf{B}'$ ,  $\mathbf{H}'$ , and  $\mathbf{T}'$  are matrices describing the neural connection weight  $w_{i,j}$ , the fatigue coefficient  $b_i$ , and the coefficients of the time constants  $r_i$  and  $a_i$ . These matrices are represented by:

$$\mathbf{W}' = \begin{bmatrix} -\frac{1}{r_1} & \frac{w_{1,2}\tilde{h}'(\bar{x}_2)}{r_1} & \dots & \frac{w_{1,2N}\tilde{h}'(\bar{x}_{2N})}{r_1} \\ \frac{w_{2,1}\tilde{h}'(\bar{x}_1)}{r_2} & -\frac{1}{r_2} & \dots & \frac{w_{2,2N}\tilde{h}'(\bar{x}_{2N})}{r_2} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{w_{2N,1}\tilde{h}'(\bar{x}_1)}{r_{2N}} & \frac{w_{2N,2}\tilde{h}'(\bar{x}_2)}{r_{2N}} & \dots & -\frac{1}{r_{2N}} \end{bmatrix}, \mathbf{B}' = \begin{bmatrix} -\frac{b_1}{r_1} & 0 & \dots & 0 \\ 0 & -\frac{b_2}{r_2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & -\frac{b_{2N}}{r_{2N}} \end{bmatrix},$$

$$\mathbf{H}' = \begin{bmatrix} \frac{\tilde{h}'(\bar{x}_1)}{a_1} & 0 & \dots & 0 \\ 0 & \frac{\tilde{h}'(\bar{x}_2)}{a_2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \frac{\tilde{h}'(\bar{x}_{2N})}{a_{2N}} \end{bmatrix}, \mathbf{T}' = \begin{bmatrix} -\frac{1}{a_1} & 0 & \dots & 0 \\ 0 & -\frac{1}{a_2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & -\frac{1}{a_{2N}} \end{bmatrix}$$

Therefore, solving for the output  $x_i$  of the excitatory oscillator in equations (1) and (2) is represented by:

$$x_i = \sum_{k=1}^{4N} C_{qk} Q_{i,k} e^{\frac{\gamma_k}{T_\beta} t} \left[ \cos\left(\frac{\lambda_k}{T_\beta} t\right) + i \sin\left(\frac{\lambda_k}{T_\beta} t\right) \right] \quad (8)$$

where,  $Q_{k,k}$  is an eigenvector of  $\mathbf{A}$ , and  $\gamma_k$  ( $k \in \{1, 2, \dots, 4N\}$ ) and  $\lambda_k$  are the real and imaginary numbers of  $Q_{k,k}$ , respectively.  $C_{qk}$  is calculated using the initial values of  $x_i$  and  $f_i$ . A solution for  $x_i$  can be expressed by an oscillation equation that is a compound trigonometric function. The equation indicates that the angular frequency of  $x_i$  is decided by  $\lambda_k/T_\beta$ . Thus, the relationship between the input parameters and the output of the neural oscillator is partially explained. In the next section, the tuning rules for the time constants and the amplitude of the oscillatory output are derived based on equation (8).

### 3 Parameter-Tuning Method of Neural Oscillators with a Ladder-Like Structure

The parameters included in the neural oscillators are divided into the following three groups: (1) time constants,  $T_{r_i}$  and  $T_{a_i}$ , (2) the amplitude coefficient,  $\alpha_n$ , and (3) the connection weight,  $w_{i,j}$ , and fatigue coefficient,  $b_i$ . The parameters in groups (1) and (2), but not (3), were derived using tuning rules that were based on the oscillation analysis in 2.2. The parameters in group (3) were tuned using a genetic algorithm (GA) [9].

In the proposed tuning algorithm, the tuning methods for (1) and (2) are combined with a GA for (3). The parameters are tuned by both a local GA and a global GA. The former is used for all parameters during the early stages of tuning whereas the latter is used for a limited pair of oscillators at the mid-late stage of tuning. This two-step tuning method is designed to avoid learning stagnation during GA-based tuning. A procedure of the proposed parameter-tuning algorithm is outlined below.

#### Step 0: Initialization

The parameters for each pair of oscillators are arranged in order (shown in Fig. 2). All the parameters included in the neural oscillators are represented as individual genes. A string including all the parameters (genes) of the neural oscillator is treated as an individual in the GA. During the initialization step,  $P$  individuals are produced and the initial value for each gene is given as a uniform random number.

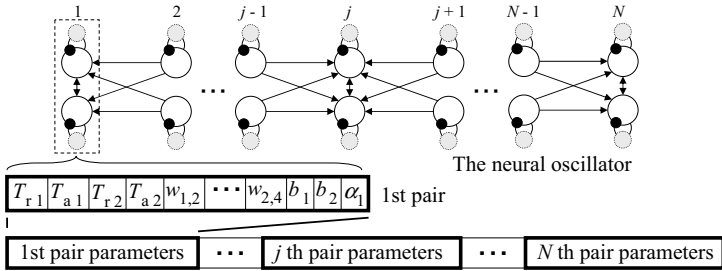


Fig. 2. A string of a GA

#### Step 1: Global Parameter-Tuning Using a GA (GA1)

##### (1) The tuning rule of the time constants

Based on the oscillation analysis in 2.2, the output-cycle length for neural oscillators with a ladder-like structure is proportional to the time constants. Thus, when a desired output-cycle length  $C_s^D$  is given, the time constants after tuning,  $T_{r_i}^{new}$  and  $T_{a_i}^{new}$ , are represented by:

$$T_{ri}^{\text{new}} = \frac{C_s^{\text{D}}}{C_s} T_{ri}^{\text{old}} \tag{9}$$

$$T_{ai}^{\text{new}} = \frac{C_s^{\text{D}}}{C_s} T_{ai}^{\text{old}} \tag{10}$$

where  $C_s$  is the output-cycle length of the neural oscillators and  $T_{ri}^{\text{old}}$  and  $T_{ai}^{\text{old}}$  are time constants prior to tuning. Thus, the time constants  $T_{ri}$  and  $T_{ai}$  at the  $g$  ( $g \in \{1, 2, \dots, G_1\}$ ) th generation may be set to a unique value as  $T_{ri}^{\text{new}}$  and  $T_{ai}^{\text{new}}$  that are calculated from equations (9) and (10) using  $T_{ri}^{\text{old}}$ ,  $T_{ai}^{\text{old}}$  and  $C_s$  at the  $g - 1$ th generation.

**(2) The tuning rule of the amplitude coefficient**

In equation (4), the output,  $y_n$ , of the neural oscillator in the  $n$  ( $n \in \{1, 2, \dots, N\}$ )th column is calculated by multiplying the difference in output between a pair of excitatory oscillators by  $\alpha_n$ . Thus, the error between the total output  $y_n$  of a pair of oscillators and its desired output  $y_n^{\text{D}}$  is decreased by tuning  $\alpha_n$ . An amplitude error,  $e^{\alpha_n}(l)$  ( $l \in \{1, 2, \dots, L\}$ ) denotes an output error between  $y_n(l)$  and its desired output  $y_n^{\text{D}}(l)$ , where  $y_n$  is discretized with  $L$  and the output from the  $l$ th sampling is  $y_n(l)$ . A square summation of  $e^{\alpha_n}(l)$  is expressed by:

$$E^{\alpha_n} = \sum_{l=1}^L (y_n(l) - y_n^{\text{D}}(l))^2 \tag{11}$$

The amplitude coefficient  $\alpha_n$  that minimizes  $E^{\alpha_n}$  is given by the following equation:

$$\alpha_n^{\text{new}} = \frac{\sum_{l=1}^L (y_n(l)y_n^{\text{D}}(l))}{\sum_{l=1}^L (y_n(l))^2} \alpha_n^{\text{old}} \tag{12}$$

where,  $\alpha_n^{\text{new}}$  is the amplitude coefficient after tuning and  $\alpha_n^{\text{old}}$  is the coefficient before tuning.  $y_n$  is expressed by  $y_n = \alpha_n^{\text{old}}(h(x_{2n}) - h(x_{2n-1}))$ . Thus, the amplitude coefficient  $\alpha_n$  at the  $g$ th generation is set to a unique value of  $\alpha_n^{\text{new}}$  that is calculated from equation (12) using  $\alpha_n^{\text{old}}$  at the  $(g - 1)$ th generation.

**(3) Genetic evolution-inspired operation for parameter optimization**

An individual of a GA consists of a string arraying a set of parameters included in the neural oscillators. For each GA generation, the adequacy of each individual is evaluated to determine which individuals will be included in the next generation. The function for evaluating error values during tuning is defined by the following equation.

$$J = \frac{1}{N} \frac{1}{L} \sum_{n=1}^N \sum_{l=1}^L |y_n(l) - y_n^{\text{D}}(l)| \tag{13}$$

where the desired output  $y_n^D$  is given by an arbitrary function. The value of  $J$  decreases accordingly with a decrease in the error between the total output  $y_n(l)$  of the  $n$ th pair of neural oscillators and its desired output  $y_n^D(l)$ . At each generation, individuals are sorted into ascending order based on  $J$ .  $J_{\text{elite}}$  denotes the smallest value of  $J$  that is obtained by an elite individual. Based on this evaluation, individuals with greater diversity are produced by a series of GA-operation with (a) selection, (b) crossover, and (c) mutation.

The above procedures from (1) to (3) are repeated until  $g_1$  reaches  $G_1$ .

### Step 2: Local Parameter-Tuning with a GA (GA2)

During local parameter-tuning using a GA (GA2), the parameters ( $T_{ri}$ ,  $T_{ai}$ ,  $b_i$ ,  $\alpha_n$ , and  $w_{i,j}$ ) that are arranged for each pair of oscillators from the first column to the  $N$ th column are tuned individually. All parameters are tuned simultaneously using GA1, whereas the tuning is limited to the parameters in one column of the neural oscillator using GA2. During this process, the following procedures: (1) tuning of the time constants, (2) tuning of the amplitude coefficient, and (3) genetic evolution-inspired operation for parameter optimization, are conducted for each column of the neural oscillator until  $g_2$  reaches  $G_2$ . The parameters of the first column are tuned first. The process is then repeated up to the  $N$ th column. Once the parameter tuning of  $N$ th column is completed, the tuning is repeated for individual columns using a GA2, beginning with first column. During this phase of individual tuning, the parameters, except those of the column targeted for tuning, are set to their previously tuned values. The tuning process is then repeated until the total generation,  $g_{\text{total}}$ , for GA1 and GA2 reaches  $G_{\text{total}}$ .

## 4 Numerical Experiments

We evaluated the effectiveness of the proposed tuning method of the ladder-like neural oscillators. As an example to reproduce the complex rhythmic signals like living animals, we applied the proposed tuning method to a rigid link model of *C. elegans*, which includes 12 pairs of neural oscillators [4]. We conducted the parameter tuning by using the conventional method (the ‘simple tuning method’) as well as the proposed method, and compared these results.

### 4.1 Acquisition of *C. elegans* Rhythmic Signal

The rigid link model for *C. elegans* [4] used in this experiment represents the body of *C. elegans* and is based on the actual neuromuscular structure, innervated by 12 pairs of motor neurons (see Fig. 3). The angle  $q_n$  of the  $n$ th link is controlled by the output  $y_n$  of the  $n$ th pair of ladder-like neural oscillators. To acquire observations of rhythmic signals in *C. elegans*, five or more animals were placed on a plate containing nematode growth medium [5]. Their sinusoidal locomotion was recorded using a video camera mounted on a stereomicroscope for approximately 1 min at 24 frames per second. Each frame of the video was

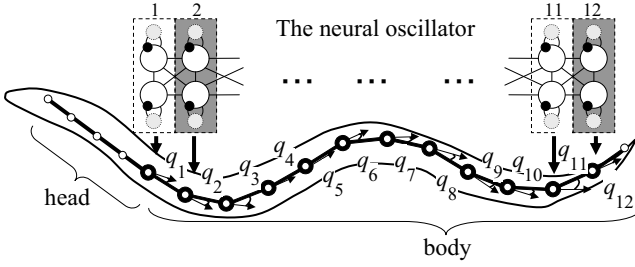


Fig. 3. A rigid link model of *C. elegans*

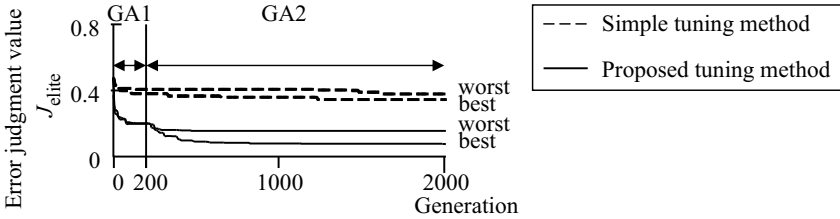


Fig. 4. Evolution of the error judgment value,  $J_{elite}$

processed using the following procedures: (a) binarization, (b) denoising, (c) skeletonizing, and (d) division of body line into the 16 parts, using image processing software. Because the rhythmic signals acquired from *C. elegans* had high levels of background noise, the signal was approximated using the following equation:

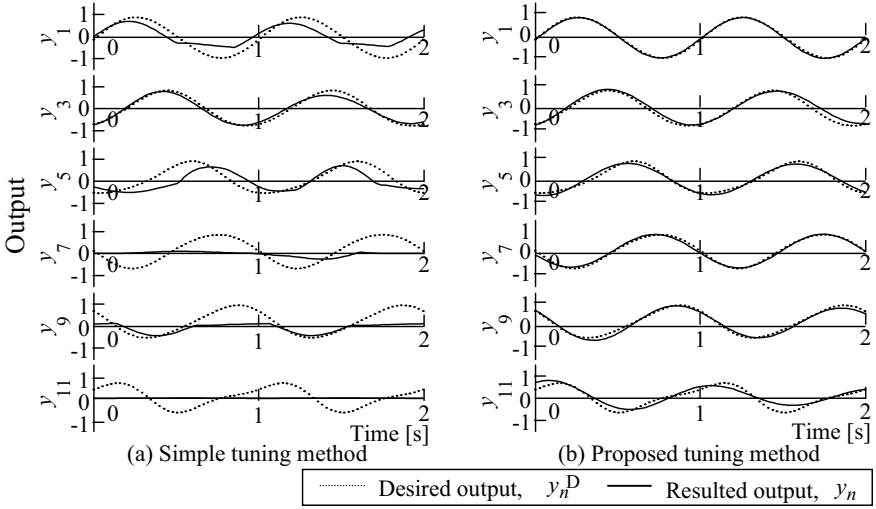
$$q_n^D = A_{0n}^D + \sum_{k=1}^K A_{n,k}^D \sin \left( \frac{2k\pi}{C_s^D} t - d_{n,k}^D \right) \quad (14)$$

$A_{0n}^D$  is a bias,  $A_{n,k}^D$  is the amplitude of oscillatory output,  $C_s^D$  is the output-cycle length, and  $d_{n,k}^D$  is the phase. The link angle was approximated by  $K = 2$ . The data were discretized by sampling  $L$ , and were set to the desired link angle  $q_n^D(l)$ . In addition,  $q_n^D(l)$  was normalized to the maximum value of the desired link angle and denoted as the desired output  $y_n^D(l)$  of the corresponding pair of oscillators.

### 4.2 *C. elegans* 's Rhythmic Signal Generation

To generate the desired outputs from the 12 columns of ladder-like neural oscillators, the parameters were tuned using the proposed tuning method. The number of neural oscillator columns was set to  $N = 12$ , and the desired output of the  $n$ th pair of oscillators was calculated using data acquired from *C. elegans*. The range of parameters used in this study were  $T_{ri} = [0, \infty)$  [sec],  $T_{ai} = [0, \infty)$  [sec],  $w_{i,j} = (-\infty, \infty)$ ,  $b_i = [0, \infty)$ ,  $s_i = 1.0$ . The generation number of tuning method





**Fig. 5.** Results of reproduction of the rhythmic signals in *C. elegans*)

were  $G_1 = 200$ ,  $G_2 = 20$ , and  $G_{total} = 2,000$ . To reduce the computational load, the time constants  $T_r$ ,  $T_a$  and the fatigue coefficient  $b_i$  were represented by equivalence at all oscillators.

We conducted five trials using different initial values for both tuning methods, the proposed tuning method and simple tuning method. The best trial  $T^{best}$  was the one that yielded the lowest error judgment value,  $J$ , at  $G_{total} = 2,000$ th generation of a GA among five trials. The worst trial  $T^{worst}$  yielded the highest value of  $J$ . The values for  $J$  in the elite individual at each GA generation in the two trials,  $T^{best}$  and  $T^{worst}$ , are shown in Fig. 4. The solid line represents  $J_{elite}$  the proposed tuning method and the dotted line represents the simple tuning method. The evolution curves,  $J_{elite}$  produced by the proposed tuning method decreased smoothly prior to the 200th generation in comparison with  $J_{elite}$  of the simple tuning method. In addition,  $J_{elite}$  was lower for the proposed tuning method than for the simple tuning method at the 2,000th generation. The total output,  $y_n$ , of the  $n$ th pair of oscillators which used the parameters tuned by  $T^{best}$  in Fig. 4 is shown in each panel of Fig. 5 (solid line; data not shown for  $n = 2, 4, 6, 8, 10, 12$ ). Panels (a) and (b) illustrate the results of the proposed and simple tuning methods, respectively. The dotted line represents the desired outputs,  $y_n^D$ , of the  $n$ th pair of the oscillator. From the results, we confirmed that the outputs of all pairs of oscillators correlate with the desired output for the parameters tuned using the proposed method. In contrast, the output of a few pairs of oscillators did not reproduce the desired output using the simple tuning method. Although, it has been difficult to tune the parameters for 12 pairs of neural oscillators to generate complex signals of animals [4], the proposed tuning method appears to generate the desired oscillatory outputs (shown in Fig. 5).

## 5 Conclusions

There was no effective tuning method for setting the parameters of a nonlinear neural oscillators that have a ladder-like structure. Therefore, we proposed a novel method for automatically tuning the parameters included in neural oscillators with a ladder-like structure. To clarify the relationship between parameters and each output of oscillators, we linearized the equations of the neural oscillators and derived the tuning rules for both the time constants and amplitude of the oscillatory outputs. Undrivable other parameters were tuned by using a genetic algorithm (GA). To avoid the stagnation of GA-learning, we formulated a two-step GA. In this algorithm after tuned all parameters of neural oscillators generally by a Global GA, parameters of each pair of oscillators were individually tuned by a Local GA. Based on numerical simulation of mimicked rhythmic-signal generation in *C. elegans*, the proposed method successfully tuned all the parameters included in the neural oscillators, and generated sinusoidal complex signals. Given the success of the proposed method, future studies may evaluate tuning methods for neural oscillators that have a non-ladder-like structure.

## References

1. Fred, D.: Foundation of Neurobiology. W. H. Feeman and Company, New York (1999)
2. Matsuoka, M.: Sustained oscillations generated by mutually inhibiting neurons with adaptation. *Bio. Cybern.* 52, 367–376 (1985)
3. Ekeberg, Ö.: A combined neuronal and mechanical model of fish swimming. *Bio. Cybern.* 69, 363–374 (1993)
4. Suzuki, M., Tsuji, T., Ohtake, H.: A model of motor control of the nematode *C.elegans* with neuronal circuits. *Artif. Intell. Med.* 35, 75–86 (2005)
5. Brenner, S.: The genetics of *Caenorhabditis elegans*. *Genetics* 77, 71–94 (1974)
6. Suzuki, M., Goto, T., Tsuji, T., Ohtake, H.: A dynamic body model of the nematode *C. elegans* with neuronal oscillators. *J. Robotics Mechatronics* 17(3), 318–326 (2005)
7. Artur, M.A.: On stability and tuning of neural oscillators: Application to rhythmic control of a humanoid robot. *Neural Networks* 1, 99–104 (2004)
8. Dinggou, Z., Kuanyi, Z.: Computer simulation study on central pattern generator: From biology to engineering. *Int. J. of Neural Sys.* 16(6), 6–16 (2006)
9. Goldberg, D.E.: Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley Publishing, Reading (1989)