ELSEVIER

Fuzzy Sets and Systems 71 (1995) 257–264

FUZZY sets and systems

# Structure optimization of fuzzy neural network by genetic algorithm

Hideyuki Ishigami[a,*], Toshio Fukuda[a], Takanori Shibata[b], Fumihito Arai[a]

[a] *Department of Micro System Engineering, Nagoya University, Furo-cho, Chikusa-ku, Nagoya 464-01, Japan*
[b] *Mechanical Engineering Laboratory, Bio-Robotics Division, Robotics Department, MITI, 1-2 Namiki, Tsukuba 305, Japan*

**Abstract**

This paper presents an auto-tuning method of fuzzy inference using genetic algorithm and delta rule. Fuzzy inference is applied to various problems. However, the determination of membership functions of the fuzzy inference depends on human experts, which is a difficult problem and time-consuming. Therefore, some auto-tuning methods have been proposed to reduce the time-consuming operations. However, the convergence of the tuning by the conventional methods depends on the initial conditions of the fuzzy model. So, we propose an auto-tuning method for the fuzzy neural network by genetic algorithm. The new tuning method realizes to construct minimal and optimal structure of the fuzzy model. This paper shows effectiveness of the tuning system by simulations compared with the conventional method.

*Keywords:* Fuzzy inference; Genetic algorithm; Learning

## 1. Introduction

Recently, fuzzy inference has been applied to the various problems: machine control, manufacturing system, and so on [2, 4–7]. The fuzzy inference has the characteristic that the fuzzy rules can describe the way in which human experts formulate their knowledge. However, the determination of the fuzzy rules, which depends on human experts, is a difficult problem. Therefore, auto-tuning methods are desirable to improve the wasted operations, and much work has been performed about the auto-fuzzy tuning algorithm [4, 5, 7].

The fuzzy neural network, which is one of the auto-tuning methods [4, 5], is actually an excellent method for the adjustment of the fuzzy rules. However, the tuning method has a weak point, because the convergence of tuning depends on the initial conditions.

On the other hand, another method using the genetic algorithm (GA) is proposed for the purpose of auto-tuning and optimization of the structure of the fuzzy model [7]. The genetic algorithm is one of the optimization methods based on the biological evolution process [1, 3, 9, 11, 12]. However, the restriction of the shape of membership

* Corresponding author. Fax: + 81-52-789-3909. Tel.: + 81-52-789-4478.


SSDI 0165-0114(94)00283-5

functions prevents the construction of more compact structure of the fuzzy model.

Considering the background as mentioned above, we propose an auto-tuning method for the fuzzy neural networks based on genetic algorithm. The tuning system has two tuning processes. One is the coarse tuning process, which is based on the multiple point's search with GA. The characteristic of the multiple point's search discriminates this algorithm from the other random search methods [10]. The candidates of the optimal result exist in the search space, and the exchange or development among them enables to search the optimal result without being stuck in the local minimum. The coarse process determines the appropriate structure and coefficient weights of a fuzzy model using the GA.

The other one is the fine tuning process. This process tunes the structure of the changeable membership functions on the antecedent part and coefficient weights on the consequent part of the fuzzy inference, respectively. The fuzzy inference is given by the coarse tuning process and is tuned by using the delta rule [8]. By these tuning processes, we can obtain the minimal and optimal structure of the fuzzy inference. The proposed system can be alternative to the conventional operations by human experts. The relation between inputs and outputs are automatically expressed as the fuzzy rules and the combination of the fuzzy rules describe the linguistic rules. Our method can be applied to the various systems, i.e., robotic motion control, sensing and recognition systems.

The following sections explain the composition of the fuzzy inference and genetic algorithm and the way of the tuning flow of the proposed system. Furthermore, simulation results show effectiveness of our proposed system compared with the conventional method, which is the fuzzy-neural network based on genetic algorithm [7].

## 2. Fine tuning process by the delta rule

This section illustrates the composition of the simplified fuzzy inference and the tuning algorithm of the fuzzy rules, respectively. The simplified fuzzy inference consists of the triangular-shaped membership functions on the antecedant part and the real values on the consequent part. This section corresponds to the fine tuning process as mentioned in Section 1.

### 2.1. Composition of the fuzzy inference

This section shows the composition of the fuzzy inference. As mentioned above, the simplified fuzzy inference consists of the membership functions and real values. Fig. 1 shows the structure of an input layer that consists of some membership functions. As shown in Fig. 1, the membership function is triangular shaped and variable, i.e., the horizontal coordinates of the summit of a membership function and the bottom vertex take arbitrary position. On the other hand, the conventional method [7] also has the triangular-shaped membership functions. However, the horizontal coordinates of the summit of a membership function corresponds to the coordinate of the bottom vertex of the next positioned membership function. This restriction in the membership function prevents the construction of more compact structure of the fuzzy model. The effectiveness of proposed method compared with the conventional method is shown in Section 5.

First, we describe formulas between input and output values of the fuzzy inference. The output value $y_k$, with respect to the $k$th input pattern is expressed by

$$\mu_i = \prod_{j=1}^{m} A_{ij}(x_j), \tag{1}$$

$$y_k = \frac{\sum_{i=1}^{n} \mu_i \cdot W_i}{\sum_{i=1}^{n} \mu_i}, \tag{2}$$

where $x_j$ is the input value in the $j$th fuzzy rule, $m$ the dimension of input space, $n$ the number of fuzzy rule, $\mu_i$ the fitness in the $i$th fuzzy rule, $A_{ij}$ the membership function, and $w_i$ the interconnection weight.

### 2.2. Tuning for antecedent part and consequent part

This section shows the tuning algorithm for the variable membership function and real value by delta rule [8]. The delta rule tunes the parameters,

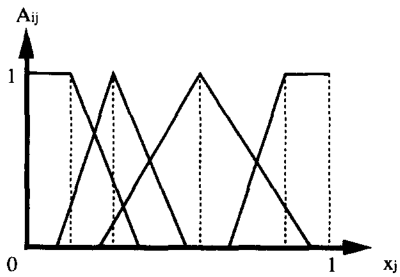


Fig. 1. Structure of an input layer (this example consists of four membership functions).

which construct fuzzy inference, by using the error between teach signal and output value of the fuzzy inference. Based on formulas (1) and (2), the tuning formula for the antecedent part is given by

$$E = \sum_{k=1}^{p} \tfrac{1}{2}(y_k - y_{tk})^2, \tag{3}$$

$$\begin{aligned} Aw(t+1) &= Aw(t) - \alpha \frac{\partial E}{\partial Aw(t)} \\ &= Aw(t) - \alpha \frac{\prod_{j \neq k}^{m} A_{ij}(x_j)}{\sum_{i=1}^{n} \mu_i} \\ &\qquad \times (y_k - y_{tk})(w_i - y_k), \end{aligned} \tag{4}$$

where $E$ is the summation of squared error between output value of the fuzzy model and desirable one, $Aw(t)$ the width between vertex and base of a membership function in $t$th tuning, $y_{tk}$ the teach signal with respect to $k$th pattern, and $\alpha$ the coefficient constant.

The membership functions are tuned by the delta rule as shown in Fig. 2.

In the same way as expressed before, the tuning formula of the consequent part is expressed by

$$\begin{aligned} w_i(t+1) &= w_i(t) - \beta \frac{\partial E}{\partial w_i} \\ &= w_i(t) - \beta \frac{\mu_i}{\sum_{i=1}^{n} \mu_i}(y_k - y_{tk}), \end{aligned} \tag{5}$$

where $\beta$ is the coefficient constant.

The tuning for the antecedent and consequent part by the delta rule corresponds to the fine tuning process.

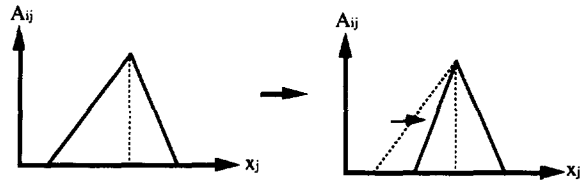


Fig. 2. Tuning of a membership function.

## 3. Coarse tuning process by genetic algorithm

This section describes the genetic algorithm [3] used in this paper. Next, we propose the coding method, the fitness function, and the genetic operations (cross-over, mutation, and selection) in the proposed method with GA. This section corresponds to the coarse tuning process.

### 3.1. *What is genetic algorithm?*

Genetic algorithm is an optimization method in which the stochastic search algorithm is based on biological principles (selection, cross-over and mutation) [9, 11, 12]. One of the GA's characteristics is the multiple points' search, which discriminates the GA from the other random search methods [10]. That is, each string corresponds to a candidate of the optimal result in the search space and many such results exist in the GA's search. In the proposed method, the string, which is a model of chromosome, represents the parameters to construct the fuzzy model. A population consists of the several strings.

The GA typically starts by randomly generating initial population of strings. Each string is transformed into the fitness value to obtain a quantitative measure. On the basis of the fitness value, the strings undergo genetic operations. The goal of genetic operations is to find a set of parameters that search an optimal solution to the problem or to reach the limited generations.

In the following sections, we describe the coding method, the fitness function, and the genetic operations (cross-over, mutation, and selection).

### 3.2. Coding

Coding operation means to transform a fuzzy model into a column of numerical parameters in one dimension (string) [1]. In other words, the string contains the parameters to construct a fuzzy model. As mentioned in Section 1, the type of fuzzy model is a simplified fuzzy inference that the antecedent part consists of triangular-shaped membership functions and the consequent part consists of real values.

So, a string has the following genetic information: (1) ratio that constructs the position for the summit of a membership function, (2) width between the cortex and the base of a membership function and (3) real value on the consequent part. Fig. 3 shows the relationship between numerical parameters in the string and the structure of an input space of the membership function. As shown in the Fig. 3, coordinates of the summit of each membership function corresponds to the ratio divided by the sum of values (1) to construct an input unit. By using three species of parameters in a string, the fuzzy model can be constructed.

A generation consists of the several strings. The strings are generated at random in the initial generation. Each string undergoes the genetic operations.

### 3.3. Cross-over

Cross-over operation means changing the position for the summit of the membership function coarsely. The way to carry out the operation is shown in the Fig. 4. Two strings (parents) in a population are arbitrary selected. Using the selected strings, cross-over point is also selected among the values that constructs the position for the summit of a membership function. The structure is exchanged between each string based on the cross-over point. The offsprings, which undergo the cross-over operation, generate four species.

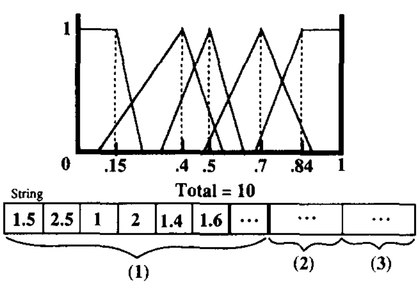


Fig. 3. Coding operation

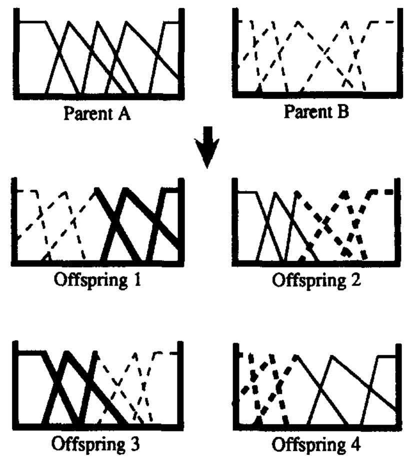


Fig. 4. Cross-over operation.

(1) The values of the right side of the string are inherited from one string (parent A) and the values of the other side are inherited from the other string (parent B), where each value is similitude by the summation of the value on the other string.
(2) The values of the right side of the string are inherited from the other string (B) and the values of the other side are inherited from one string (A), where each value is similitude by the summation of the values on one string.
(3) In the same way, the values of the left side are inherited from one string (A).
(4) The values of the left side are inherited from the other string (B).

As shown in the Fig. 4, the summit of membership function is changed with respect to each offspring. Offspring can inherit superior genetic information from two parents by cross-over operation. The second and third values in the coding operation are inherited without changing.

### 3.4. Mutation

Mutation operation occurs with respect to the strings, which undergo the cross-over operation, with probability $P_m$. Mutation operation in this paper means that the selected membership function is pruned, which is illustrated in Fig. 5. We can expect to reduce the number of fuzzy rules and obtain the minimal structure of the fuzzy model by the operation.

In this way, new strings are generated by the cross-over and the mutation operations. These operations, which give the appropriate structures of the fuzzy model, correspond to the coarse tuning process.

### 3.5. Fitness function

We set the fitness function to evaluate each string whether it is desirable or not. The fitness function consists of the number of the membership functions and the summation of squared error between output value of the fuzzy model and desirable one. We define the function $F$ as

$$F(s_i) = E + A \times N, \tag{6}$$

where $s_i$ is the $i$th string, $E$ the summation of squared error between output value of the fuzzy model and desirable one, $A$ is the coefficient, and $N$ the number of the membership function. We define that the smaller the fitness value is, the more the string has high fitness.

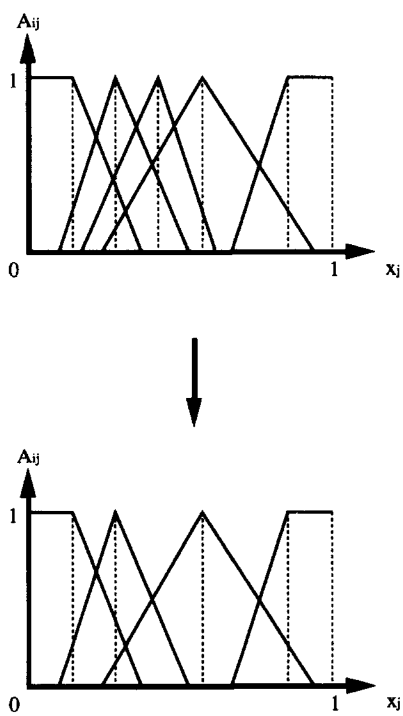

Fig. 5. Mutation operation.

### 3.6. Selection

Based on the fitness value, each string is selected or not to the next generation. In this paper, we adopt elite preservation strategy that the strings having high fitness remain to the next generation and sampling strategy that the strings are selected randomly remain to the next generation. The more a string that undergoes the genetic operation has high fitness value, the more it will survive in the next generation by selection. This selection, which changes the population of strings, performs the "evolution".

## 4. Auto-fuzzy tuning strategy

This section illustrates the tuning algorithm to obtain the optimal fuzzy model. This procedure is shown in Fig. 6.

(1) The initial generation that has several strings, in which each string includes the parameter to construct a fuzzy model, are generated at random. (2) Cross-over and mutation operations perform to create new strings. GA determines the structure of the fuzzy model coarsely and minimizes the number of the fuzzy rule. These operations correspond to the coarse tuning process, as mentioned before. (3) Using the strings that undergo the genetic operations, the membership functions and the real values in each string are tuned with delta rule [8]. The delta rule determines the structure of the fuzzy model finely. This operation corresponds to the fine tuning process. (4) Each string is evaluated based on a fitness function to obtain the quantitative measure. (5) Based on the fitness value, each string

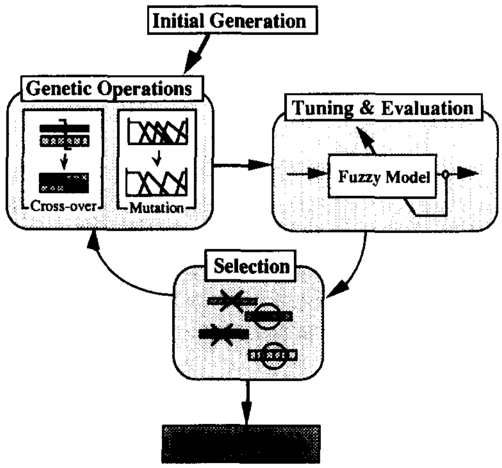


Fig. 6. Auto-fuzzy tuning strategy.

undergoes the selection operation, which changes the population of string and performs the "evolution". (6) If we can obtain the target string or the searching reaches the limited generations, the searching is over; otherwise the flow goes back to operation (2) (one generation). We define the target string as the one whose fitness value converges within the target error.

As shown in Fig. 6, we obtain the optimal model by integrating the coarse tuning with GA and the fine tuning with delta rule.

## 5. Simulation results

This section shows the effectiveness of the proposed system by simulations. The system is applied to the EX-OR problem, which is famous for a simple non-linear problem. A function approximation problem is compared with the conventional method [7], which allows us to obtain the minimal and optimal structure of the fuzzy model.

Each string is expressed as a fuzzy model by the coding operation. In both problems, the structure of the fuzzy model consists of two inputs and one output. Each input consists of five membership functions in the greatest number and the both sides of membership function are constant functions. In the simulation, the population size in a generation consists of 50 strings.

### 5.1. EX-OR problem

In this section, we verify the convergence of the tuning and optimization of the structure concerning the fuzzy model. We carry out searching with four teach signals expressed by

$$(\text{Input 1, Input 2, Output}) = (0,0,0), (1,0,1), (0,1,1), (1,1,0). \tag{7}$$

Probability of the mutation operation, which reduces the number of fuzzy rules, is 1%.

As the result of the tuning, we can obtain the string for which the summation of squared error between the desired value and the output value converges to 0 and that is obtained independent of the initial conditions and within 9.6 generations on the average. Moreover, we can also obtain the optimal structure, which is pruned of all the six redundant middle-positioned membership functions. We can obtain the same optimal string that consists of the minimum fuzzy rule and produces the same output as the teach signal.

### 5.2. Function approximation problem

In this section, we deal with another function approximation problem. The function to approximate is expressed by

$$y = \frac{\sin \pi \sqrt{2(x_1^2 + x_2^2)} + 1}{2}, \tag{8}$$

where $x_1$, $x_2$ are input variables ($0 \leqslant x_1, x_2 \leqslant 1$), and $y$ is the output variable ($0 \leqslant y \leqslant 1$).

We verify the optimization of structure concerning the fuzzy model. In addition to the conditions as mentioned above, the condition of the search is as follows.

(1) Search is carried out for 50 generations.
(2) The number of teach signals for search is nine. Each teach signal consists of two input data and one output data.
(3) Tuning parameters for antecedant part and consequent part are $\alpha = 0.05$ and $\beta = 0.1$.
(4) Probability of the mutation operation is 1%.

Fig. 7 illustrates fitness curve in the process of search. The target string, in case the summation of

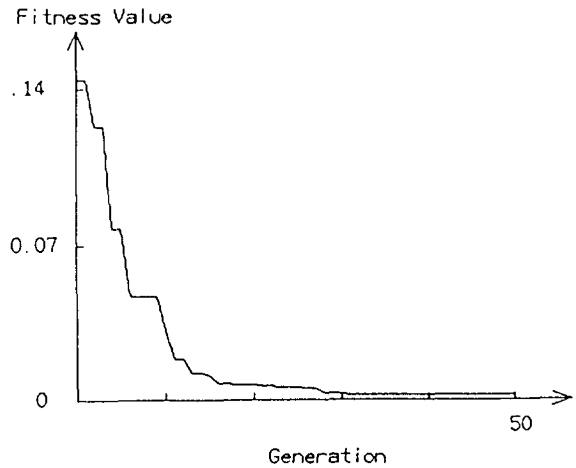


Fig. 7. Fitness value.

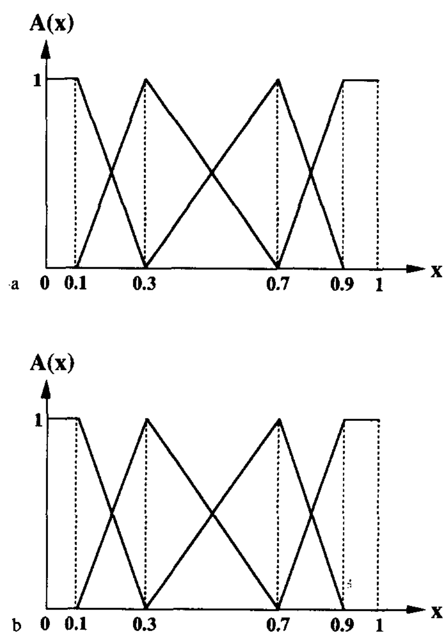


Fig. 8. Optimized membership function by using the conventional method. (a) Membership functions in the first input. (b) Membership functions in the second input.

squared error between desirable value and output value converges in $1.27 \times 10^{-6}$, is obtained without depending on the initial conditions as shown in Fig. 7. The obtained fuzzy model has the following structure. One input consists of three membership functions and the other input consists of four membership functions. In other words, the fuzzy model is pruned three redundant membership functions.

To show more effectiveness, we compare the proposed method with the conventional method [7]. By using the conventional method, we obtain the fuzzy model based on the same condition in the case of the proposed method. The conventional method indicates almost the same convergence in $3.37 \times 10^{-5}$ as the proposed method. However, we obtain the string that is pruned two redundant membership functions. In other words, the number of fuzzy rules obtained by the proposed method reduces by 25% compared with the conventional methods.

We can indicate how to obtain the minimal and optimal structure by using the proposed system. Fig. 8 and Table 1 show the obtained structures and fuzzy rules by the conventional method, while Fig. 9 and Table 2 show the obtained structures and fuzzy rules by the proposed method. Reduction

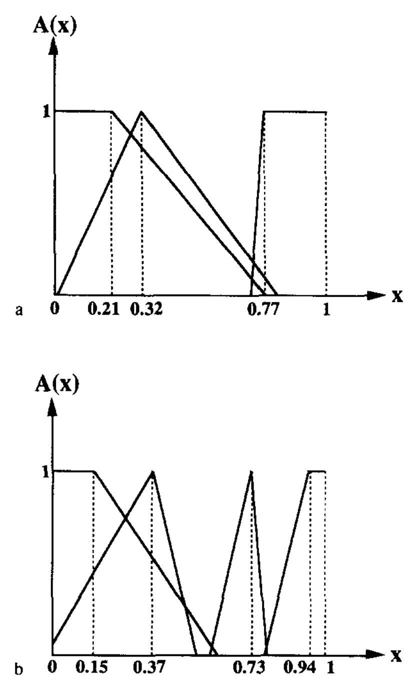


Fig. 9. Optimized membership function by using the proposed method. (a) Membership functions in the first input. (b) Membership functions in the second input.

Table 1
Fuzzy rules obtained by the conventional method

| | Input 1 | | | |
|---|---|---|---|---|
| Input 2 | 0.5000 | 1.1075 | 0.3062 | 0.0180 |
| | 1.1077 | 0.9834 | 0.1316 | 0.9946 |
| | 0.3061 | 0.1186 | 0.6061 | − 0.9638 |
| | 0.0180 | 1.0299 | − 0.9981 | 0.4998 |

Table 2
Fuzzy rules obtained by the proposed method

| | Input 1 | | |
|---|---|---|---|
| Input 2 | 0.4519 | 1.4342 | 0.0227 |
| | 1.3192 | 0.8583 | − 0.0646 |
| | 0.0594 | 0.7401 | − 0.1685 |
| | 0.0180 | − 0.0159 | 0.4999 |

of the fuzzy rules contributes the easiness to convert the fuzzy rule into the linguistic rule.

## 6. Conclusions

This paper has proposed a synthesis method of fuzzy neural network and genetic algorithm for auto-fuzzy tuning. The characteristic of the proposed system is to obtain the minimal and the optimal structure of a fuzzy model. The antecedent and the consequent part of the fuzzy model are tuned by the delta rule and the number of membership functions is optimized by the genetic algorithm.

Simulations show the effectiveness of the proposed method compared with the conventional method. The proposed method is superior to the conventional method with respect to the number of the fuzzy rules and convergence.

We expect to substitute the proposed system for the human experts' operations to construct the fuzzy model. The system can be applied to the various systems such as the robotic motion control, sensing and recognition problems.

## References

[1] L. Davis, Ed., *Handbook of Genetic Algorithms* (Van Norstrand Reinhold, New York, 1991) 61–70.

[2] T. Fukuda, S. Shiotani et al., Recognition and counting of cells by image processing, Preprint for the 4th Toyota Conference (1990) 5-1/5-18.

[3] D.E. Goldberg, *Genetic Algorithm in Search Optimization and Machine Learning* (Addison-Wesley, Reading, MA, 1989) 1–23.

[4] C.M. Higgins and R.M. Goodman, Learning fuzzy rule-based neural networks for function approximation, *Proc. IEEE IJCNN*, Vol. 1 (1992) 251–256.

[5] S. Horikawa, T. Furuhashi et al., A study on fuzzy modeling using fuzzy neural networks, *Proc. IFES* (1991) 562–573.

[6] Y.F. Li and C.C. Lau, Application of fuzzy control for servo system, *Proc. IEEE Robotics and Automation*, Vol. 3 (1988) 1511–1519.

[7] H. Nomura et al., A self-tuning method of fuzzy reasoning by genetic algorithm, *Proc. Internat. Fuzzy Systems and Intelligent Control Conf.* (1992) 236–245.

[8] D.E. Rumelhart, J.L. McClelland and The PDP Research Group, *Parallel Distributed Processing* (MIT Press, Cambridge, MA, 1986).

[9] T. Shibata, T. Fukuda et al., Path-planning for multiple mobile robots by using genetic algorithm, *Proc. IMACS '92*, Vol. 2 (1992) 1657–1662.

[10] J. Sun, W.I. Grosky and M.H. Hassoun, A fast algorithm for finding global minima of error functions in layered neural networks, *Proc. IEEE IJCNN*, Vol. 1 (1990) 715–720.

[11] T. Ueyama, T. Fukuda and F. Arai, Structure configuration using genetic algorithm for cellular robotic system, *Proc. IEEE IROS*, Vol. 3 (1992) 1542–1549.

[12] D. Whitley, T. Starkweather and C. Bogart, Genetic algorithms and neural networks: optimizing connection and connectivity, *Parallel Comput.* **14** (1990) 347–361.