A Hybrid Robust Identification Using Genetic Algorithm and Gradient Method[†]

Jinglu HU*, Kousuke KUMAMARU* and Katsuhiro INOUE*

This paper deals with the issues related to developing an efficient system identification algorithm which may find "global minimum" of multimodal loss function robustly, on the basis of an *effective combination* of Genetic Algorithm (GA) and gradient method. In order to realize such robust system identification algorithm, a Non-Standard GA (NSGA) is proposed as an effective GA. In the NSGA, a new GA operator named as *development* is introduced to improve its convergent property. We can thus realize a hybrid robust identification, in which parameter estimation is executed by a gradient method based on a good initial value searched by the NSGA. The effectiveness of the proposed algorithm is demonstrated by numerical simulations.

Key Words: multimodality, robust identification, genetic algorithm, gradient method.

1. Introduction

In many model-based approaches, it is an essential problem to build up a good model. System identification has been known as such a technique. When model structure is determined, the main task of the identification is to estimate model parameters, which is usually determined as the global minimum point of the loss function that is not guaranteed to be unimodal. There have been proposed many identification algorithms, most of which are gradient based methods^{1)~3)}. However, it should be noticed that there is a potential risk that a gradient based algorithm is stuck at a local minimum, which may result in a poorly identified model. Such a risk will further increase in the presence of unmodeled dynamics⁴⁾.

On the other hand, recent researches show that nonlinear system identification can be executed in a hierarchical way. In this approach, the characteristics of system in the interesting domain are divided into two parts: linear part and nonlinear part. The linear part is first identified by using a linear AR-MAX model, see 5) for details. There inevitably ex-

ists large unmodeled dynamics in this identification stage. It therefore is crucial to develop a system identification algorithm which may find global minimum robustly.

There is no easy solution to the multimodality problem. It is usually well used effort to spend some time to come up with a good initial value where to start the iterations⁶⁾. This however is not a realistic option in such cases where little prior knowledge is available. A better option is usually to search for a good initial value or to estimate the parameters directly using global searching techniques such as random search and genetic algorithm^{7)~9)}. However, since the searchings based on these techniques are rather time-consuming, it is difficult to implement the techniques in the practical identification algorithms. In this paper, an efficient system identification algorithm, which may find "global minimum"⁽¹⁾ of multimodal loss function robustly, is developed on the basis of an *effective combination* of Genetic Algorithm (GA) and gradient method. The term "robust" mainly means that the algorithm is robust to multimodality problem. However, in the case of multimodality, the presence of unmodeled dynamics will

[†] Presented in the 13th SICE Kyushu Branch Annual Conference (1994.11)

* Faculty of Computer Science and Systems Engineering, Kyushu Institute of Technology, Iizuka, Fukuoka (Received May 29, 1995)
(Revised November 29, 1995)

(1) In this paper, the term "global minimum" is used in the sense of engineering. This means that "global minimum" includes the allowable points around global minimum.

increase the risk of the algorithm being stuck at a local minimum. Therefore, in this sense, the term also means that the algorithm is robust to unmodeled dynamics.

GA is a probabilistic search algorithm based on a simple mechanics which models on genetic processes occurring in nature^{10), 11)}. Recently, GA has attracted much interest in system control and identification community^{9), 12), 13)}, because it has high ability for global optimization. It is well known that GA is rather time-consuming algorithm, some work should therefore be done to improve its efficiency. There have been proposed several hybrid GAs by combining GA with local searching method such as gradient method or simulated annealing^{14), 15)}. However, few of these hybrid GAs can be applied efficiently to system identification. In this paper, we propose a Non-Standard GA (NSGA) which is suitable for system identification. In the NSGA, a new GA operator named as *development* which may be performed by using gradient method is introduced to improve its convergent property. In this way, the NSGA evolves not only in a probabilistic manner (e.g. crossover, mutation) but also in a non-probabilistic manner (e.g. development). Combining this NSGA with gradient method, we can realize a hybrid robust identification, in which first a good initial value is searched using the NSGA and then the identification is continued by gradient method.

2. Problem Description

Let us consider the problem of estimating a model for a dynamic system on the basis of the observation of an N point input-output data sequence $Z^N = [\{u_k\}, \{y_k\}]$. For simplicity we assume that the observed data Z^N is generated by a SISO system. The system is considered to be a general one, which may be linear or nonlinear, time-invariant or time-variant and even with time-delay.

When the system is modeled, we assume that the parametric model is a linear time-invariant ARMAX model

$$A(q^{-1}, \theta)y(k) = B(q^{-1}, \theta)u(k) + C(q^{-1}, \theta)e(k) \quad (1)$$

with

$$\begin{aligned} A(q^{-1}, \theta) &= 1 + a_1 q^{-1} + \dots + a_n q^{-n} \\ B(q^{-1}, \theta) &= b_1 q^{-1} + \dots + b_m q^{-m} \\ C(q^{-1}, \theta) &= 1 + c_1 q^{-1} + \dots + c_l q^{-l} \end{aligned} \quad (2)$$

where $\theta = [a_1 \dots a_n \ b_1 \dots b_m \ c_1 \dots c_l]^T$ is parameter vector, $y(k)$ is the output at time k ($k = 1, 2, \dots$), $u(k)$ the input and $e(k)$ the white noise with zero mean.

3. Estimating Model Parameter θ

A gradient based numerical search is the most commonly-used method to estimate the model parameters θ . Without loss of generality, we will use Prediction Error Method (PEM) in our discussion. Introducing a regression vector

$$\varphi(k, \theta) = [-y(k-1) \dots -y(k-n) \ u(k-1) \dots u(k-m) \ \varepsilon(k-1, \theta) \dots \varepsilon(k-l, \theta)]^T,$$

we can rewrite (1) into the form

$$y(k) = \varphi(k, \theta)^T \theta + e(k) \quad (3)$$

$\varepsilon(k, \theta)$ in $\varphi(k, \theta)$ denotes prediction error defined by

$$\begin{aligned} \varepsilon(k, \theta) &= y(k) - \hat{y}(k, \theta) \\ &= y(k) - \varphi(k, \theta)^T \theta \end{aligned} \quad (4)$$

For this identification problem, most numerical schemes select $\theta = \hat{\theta}_N$ so that

$$V_N(\theta) = \frac{1}{N} \sum_{k=1}^N \|y(k) - \varphi(k, \theta)^T \theta\| \quad (5)$$

is minimized for some norm $\|\cdot\|$.

A typical choice is a quadratic norm

$$V_N(\theta) = \frac{1}{N} \sum_{k=1}^N |y(k) - \varphi(k, \theta)^T \theta|^2 \quad (6)$$

Note that the loss function (6) is multimodal because there exists a noise model $C(q^{-1})e(k)$ in (1). Minimizing (6) with respect to θ can be carried out by a recursive algorithm³⁾.

$$\hat{\theta}(k) = \hat{\theta}(k-1) + L(k)\varepsilon(k, \hat{\theta}(k-1)) \quad (7)$$

$$L(k) = \frac{P(k-1)\psi(k, \hat{\theta}(k-1))}{1 + \Gamma(k)\psi(k, \hat{\theta}(k-1))} \quad (8)$$

$$P(k) = P(k-1) - L(k)\Gamma(k) \quad (9)$$

where $\Gamma(k) = \psi^T(k, \hat{\theta}(k-1))P(k-1)$ and

$$\begin{aligned} \psi(k, \hat{\theta}(k-1)) &= \left(\frac{\partial \varepsilon(k, \theta)}{\partial \theta} \right)^T \Bigg|_{\theta = \hat{\theta}(k-1)} \\ &= \frac{\varphi(k, \hat{\theta}(k-1))}{C(q^{-1}, \hat{\theta}(k-1))} \end{aligned}$$

Because the loss function is not unimodal, the algorithm (7)-(9) has a potential risk to get caught at a local minimum. The risk has been found to be increased when the modeling contains unmodeled dynamics. In order to find the global minimum robustly, we shall develop a hybrid identification algorithm by combining GA with the PEM, in which an NSGA is introduced to search for a good initial value.

4. Robust Identification Algorithm

A robust identification algorithm based on GA and gradient method can usually be described as a two-stage process, in which first a good initial value is searched using GA, and then the identification is continued by gradient method. However, since the convergence of a standard GA is rather slow, some improvements are needed to make it easily implemented in identification algorithm. Usually, individual developmental process is disallowed in a standard GA because of premature problem. However, considering the peculiarity of system identification, a limited development may be allowed to improve the convergent property of GA. By introducing a new GA operator named as *development*, we propose an NSGA, in which the *development* may be performed by gradient method. A hybrid robust identification algorithm based on these consideration is shown in figure 1.

4.1 Description of the NSGA

Genetic Algorithms, which model on the genetic processes occurring in nature, are adaptive methods that may be used to solve search and optimization problem¹⁶⁾. They work with a population of individuals (also known as chromosomes), which represent possible solutions to a given problem. Each individual is assigned a fitness score according to how good a solution to the problem it is. The highly fit individuals are given opportunities to reproduce, by cross breeding with individuals in the population. The least fit of the members of the population are less likely to get selected for reproduction, and so die out. Over several generations, individuals tend to be identity and the algorithm is converged.

Figure 2 shows the principle of the random search via crossover and mutation in a standard GA. The search based on this kind of "probabilistic" searching rules has been demonstrated to be robust to the

Fig. 1 Basic idea of hybrid robust identification

multimodality problem, but its convergent speed is low. In order to improve the convergence, the optimal searching rule that appears in gradient method is incorporated into GA, which is named as *individual development* (or *development*). That is, before the fitness is evaluated, each individual is allowed to develop itself via optimal rule with a limited degree. In this way, we have introduced a Non-Standard GA (NSGA). The NSGA evolves in a manner of "probabilistic $\Rightarrow$ optimal $\Rightarrow$ probabilistic $\Rightarrow$ optimal $\Rightarrow \dots$ ", which combines the advantages of both GA and gradient method, and makes tradeoff between the robustness and the convergence.

The structure of the NSGA discussed above is shown in figure 3. We discuss it further in the following subsections.

4.1.1 Representation of the Solutions

To solve a problem with GA, potential solutions have first to be encoded as chromosomes. Generally, chromosomes have been string of bits, but also other alphabets can be used¹⁷⁾. For an ARMAX model described by (1), it would seem particularly natural to represent the genes directly as real parameters. Then a chromosome vector is a vector of floating point numbers. This is known as Real Coded GA.

$$C^t = (a_1, \dots, a_n, b_1, \dots, b_m, c_1, \dots, c_l) \quad (10)$$

Fig. 2 Crossover and Mutation in a standard GA

4.1.2 Crossover and Mutation

In the case of Real Coded GA, each of the randomly chosen positions is naturally located between two genes. Usually, two strategies, Simple Crossover and Max-Min-Arithmetical Crossover can be used¹⁷⁾.

Suppose two chromosomes of $C_a^t = (c_1, \dots, c_k, \dots, c_H)$ and $C_b^t = (\tilde{c}_1, \dots, \tilde{c}_k, \dots, \tilde{c}_H)$ to be crossed. For Simple Crossover, two offsprings will be obtained:

$$\begin{aligned} C_a^{t+1} &= (c_1, \dots, c_{k-1}, \tilde{c}_k, \dots, \tilde{c}_H) \\ C_b^{t+1} &= (\tilde{c}_1, \dots, \tilde{c}_{k-1}, c_k, \dots, c_H) \end{aligned} \quad (11)$$

where the crossover point (between $k-1$ and k) is determined randomly. And for Max-Min-Arithmetical Crossover, four offsprings will be generated:

$$\begin{aligned} C_1^{t+1} &= \alpha C_a^t + (1 - \alpha) C_b^t \\ C_2^{t+1} &= \alpha C_b^t + (1 - \alpha) C_a^t \\ C_3^{t+1} &= (\min\{c_1, \tilde{c}_1\}, \dots, \min\{c_H, \tilde{c}_H\}) \\ C_4^{t+1} &= (\max\{c_1, \tilde{c}_1\}, \dots, \max\{c_H, \tilde{c}_H\}) \end{aligned} \quad (12)$$

where $\alpha \in [0, 1]$ is a random value.

Crossover is not usually applied to all pairs of individuals selected for mating. A random choice is made, where the likelihood of crossover being applied is typically between 0.6 and 1.0.

Unlike a standard GA, in the Real Coded GA, mutation is applied to each child individually. It randomly alters each gene with a small probability. A non-uniform stage can be used¹⁷⁾.

Suppose chromosome $C^t = (c_1, \dots, c_k, \dots, c_H)$ is selected to be mutated, the result chromosome will be

```
t:=0;
initpopulation P(t);
develop P(t);
evaluate P(t);
while not done do
  t:=t+1
  P':=selectparents P(t);
  cross P'(t);
  mutate P'(t);
  develop P'(t);
  evaluate P'(t);
  P:=survive P, P'(t);
od
```

Fig. 3 Structure of the NSGA

$C^{t+1} = (c_1, \dots, \tilde{c}_k, \dots, c_H)$, with

$$\tilde{c}_k = \begin{cases} c_k + \Delta(t, c_{max} - c_k) & \text{if } r = 0 \\ c_k - \Delta(t, c_k - c_{min}) & \text{if } r = 1 \end{cases} \quad (13)$$

where r is a randomly generated binary digit, $c_k \in [c_{min}, c_{max}]$, and the function $\Delta(t, y)$ returns a value in the range $[0, y]$ such that the probability of $\Delta(t, y)$ being close to 0 increases as t increases. This property causes that the operator makes a uniform search into the initial space when t is small, and very locally at later stages.

4.1.3 Development and Fitness Evaluation

In a genetic algorithm, selection is done based on the fitness of individual. A fitness function should be chosen that the fitness of an individual indicates the "goodness" of the individual which represents a solution to the problem. In the case of the PEM, the fitness function could simply be the inversion of the

variance of prediction error or the loss function (6).

Generally, there is no individual developmental process in a standard GA. In our NSGA, we introduced a limited development via gradient to each individual. This can improve the convergence of the GA. The *degree* of the development should be chosen carefully according to each problem to be solved in order to keep the robustness.

In the hybrid robust identification algorithm, *fitness evaluation* and *development* are done in one function described as (14). By given the old individual (*old_indi*), the degree of development (*m*) and input-output data Z^m , the function will return developed individual (*new_indi*) and the fitness (*fit*).

$$[fit, new\_indi] = RPEM(Z^m, old\_indi, m) \quad (14)$$

where *RPEM* is the recursive PEM described by (7)-(9), and the fitness is calculated by inverting the variance of the variance of prediction error.

$$fit = \frac{1}{\beta + \frac{1}{m} \sum_{k=1}^m \varepsilon^2(k, \hat{\theta}(k-1))} \quad (15)$$

where β is an appropriate constant, which is introduced to change the shape of fitness function.

4.1.4 Reproduction

During the reproductive phase of the GA, the individuals of the next generation should be selected based on the rule of natural selection: The highly fit individuals are given opportunities to reproduce, by cross breeding with individuals in the population, while the least fit of the individuals are less likely to get selected for reproduction, and so die out.

Usually reproduction is carried out as a two-stage process. It starts with the *current population*. Selection is applied to the current population to create an *intermediate population*. Then the *next population* is created from the *intermediate population*. Following two strategies can be chosen.

In one strategy, the individuals of *intermediate population* are generated from selected parents which give their offsprings proportional to their fitness. Then pairs of individuals are chosen randomly as parents for mating by crossover and give their offsprings to form the *next population*.

In the other strategy, *intermediate population* is generated by including the individuals of *current population* and their offsprings generated by applying

Max-Min-Arithmetical crossover. The fitness of each individual of the *intermediate population* is evaluated. Then good individuals of the *intermediate population* are chosen to form the *next population*. This method is known as truncation(or extinctive) selection.

4.2 Hybrid Identification Algorithm

We can now summarize the hybrid robust identification algorithm into the following steps.

Step 1: Initialize parameters of the NSGA.

Specify population size, crossover probability, mutation probability, development step and maximum generation with appropriate values.

Step 2: Generate the first generation.

If there is no *a priori* information, individuals in the first generation are usually given to be random values.

Step 3: Search with the probabilistic rules.

Apply crossover operator (11) or (12) and mutation operator (13) to individuals of population and then reproduce offsprings.

Step 4: Search with the optimal rule (*development*).

Develop each individual with limited degree based on optimal rule (7)-(9).

Step 5: Evaluate fitness of each individual via (15) and select new generation based on the fitness.

Step 6: Determine whether the algorithm is converged or maximum generation is reached. If the answer is 'no', go to step 3 otherwise go to next step.

Step 7: Identify model parameters.

Average individuals in the final generation to obtain a good initial value and then continue the identification using (7)-(9)

5. Numerical Simulations

In this section, some numerical simulations are carried out to demonstrate the effectiveness of the proposed robust identification algorithm. The results are compared with those using the PEM (variance of prediction error method).

All the simulations are carried out as Monte Carlo tests. In each of these cases, first, 2000 input-output data sets are sampled from the system; Then Monte Carlo tests with 300 trials are conducted by using both the proposed method and the PEM, in which the

initial values of each trial are given randomly. Considering the fact that the value of $\hat{\sigma}^2 = \sum_{t=1}^N \varepsilon(t)^2 / N$ becomes larger when an algorithm is stuck at a local minimum, we show the results in histogram of $\hat{\sigma}^2$. It should be noticed that the robustness of the proposed approach is demonstrated using the results of its Monte Carlo test, while the results of Monte Carlo test of PEM with random initial value are mainly used to show the multimodality of loss function.

In these cases, when PEM is used, the identification begins with a randomly given initial value and continues for 2000 steps; When the proposed robust method is used, a good initial value is searched using the NSGA, where the parameters of the NSGA used are *population*=15, *crossover probability*=0.8, *mutation probability*=0.1, *development step*=30, *maximum generation*=10. Then the identification is continued using the PEM for 500 steps.

Example 1: Consider a second order SISO system described by

$$G(s) = \frac{k_0}{s(T_0 s + 1) + k_0} \quad (16)$$

where $T_0 = 0.74$, $k_0 = 3$. After being sampled with a period of 0.25sec, the discrete-time system is corrupted by a non-white noise sequence $\{\nu(k)\}$:

$$\nu(k) = e(k) + c_1 e(k-1) + c_2 e(k-2) \quad (17)$$

where $c_1 = -1.2$, $c_2 = 0.5$, and $e(k) \in N(0, 0.1)$ is white gaussian noise. The input of the test is pseudo-random binary sequence (PRBS).

When the system is identified, a second order AR-MAX models described by (1) is used as model, where $n = m = l = 2$. Since there exists a noise model in (1), the loss function is multimodal hence there is a risk that a gradient method such as the PEM is stuck at a local minimum. Figure 4 shows the results of Monte Carlo tests in histogram, where x-axis denotes $\hat{\sigma}^2$ and y-axis denotes numbers(probabilities), (a) is the result using proposed method and (b) is the result using the PEM.

Example 2: In this example, we consider a discrete-time SISO system governed by

$$y(k) = (1.5 - 0.01y(k-1))y(k-1) - 0.7y(k-2) + 0.1u(k-1) + 0.1u(k-2) + \nu(k) \quad (18)$$

where $\nu(k)$ is a non-white noise described by (17) and $u(k)$ is a PRBS.

Fig. 4 Histograms of $\hat{\sigma}^2$ for example 1 (a) by using the proposed method; (b) by using PEM

Fig. 5 Histograms of $\hat{\sigma}^2$ for example 2 (a) by using the proposed method; (b) by using PEM

The model used to identified this system is the same as that of example 1. The results of Monte Carlo tests are shown in figure 5 where (a) is the result using proposed method and (b) is the result using the PEM. Since in this example, the nonlinearity of system caused unmodeled dynamics, it seems that the probability of a gradient based method being stuck at local minima was increased. However, our proposed method did not get caught in all the trials.

In the *Example 1* and *Example 2*, we have shown that the proposed algorithm is very *efficient*, which can find the "global minimum" *robustly* (see concentrated histograms Fig.4(a) and Fig.5(a)) using short data sequence (500) and reasonable computing time (about 3200 steps of PEM) in multimodal problems (see distributed histograms Fig.4(b) and

Fig. 6 Histograms of σ^2 for example 3 (a) by using the proposed method; (b) by using PEM with random initial value; (c) by using PEM with LS estimations as initial value.

Fig.5(b)). On the other hand, the proposed method is basically a gradient method (PEM), while only the initial value is searched by the NSGA. In next example, we will compare it with those using initial values given in different ways.

Example 3: a third order SISO system governed by (19) is considered as true system, in which a time delay of 0.25sec is added to increase the unmodeled dynamics.

$$G(s) = \frac{ke^{-\tau s}}{s(T_0s + 1)(T_1s + 1) + k} \quad (19)$$

where $k = 2.5$, $T_0 = 0.1$, $T_1 = 0.5$, $\tau = 0.25\text{sec}$.

The sampling period and the noise sequence for the discrete-time system and the identification model used are the same as those of example 1.

In the Monte Carlo tests, each of trials is carried out by PEM with 2000 steps using an initial value (a) searched by using the NSGA; (b) given randomly; (c) a_i and b_i searched by LS while c_i given randomly.

Figure 6 shows the results of Monte Carlo tests. We can see that the proposed method can achieve "global minimum" robustly, while other two methods can not. This example again confirms the effectiveness of the proposed method.

Table 1 Mean values and Standard variations

	PEM with random initial value		PEM with LS estimation as initial value		the proposed method	
	Mean Value	Standard Variation	Mean Value	Standard Variation	Mean Value	Standard Variation
a_1	-1.3163	0.4444	-1.4972	0.1451	-1.5544	0.00003
a_2	0.6123	0.3360	0.7585	0.1086	0.7707	0.00001
b_1	-0.0919	0.0864	-0.1008	0.0444	-0.0777	0.00004
b_2	0.2126	0.1072	0.2325	0.0585	0.2638	0.00002
c_1	-0.6442	0.7078	-0.9871	0.3365	-1.2410	0.00026
c_2	0.3438	0.1669	0.4082	0.1512	0.5082	0.00027

Table 1 shows mean values of the identified parameters and their standard variations in the Monte Carlo tests of example 3. We can see that using the proposed method, the standard variations of identified model parameters are very small.

6. Discussions and Conclusions

It has been shown that when the loss function is not unimodal, there is a potential risk that a gradient based identification algorithm is stuck at a local minimum. The risk has been found to be increased in the case of presence of unmodeled dynamics. This usually results in a poorly identified model, it therefore is crucial to develop an identification algorithm which may find global minimum robustly.

Such a robust identification algorithm has been developed on the basis of the effective combination of GA and gradient method. First, an NSGA has been proposed, in which a new GA operator named as *development* was introduced. This new GA operator may improve convergent property of the GA. Then the NSGA is used to search for a good initial value and the identification is continued by gradient method. In this way, we can realize a robust system identification which may find global minimum of multimodal loss function with a probability close to one. Simulation results show that the proposed algorithm is robust in finding the global minimum while the computing time required for searching is not increased significantly.

In our current research, we have applied the robust identification algorithm to fault detection and nonlinear system identification^{4), 5)}, and found it very effective and useful. We thus believe that the robust identification algorithm will find wide applications in

the practical cases.

References

- 1) T. Söderström and P. Stoica, *System Identification*, Prentice-Hall, Englewood Cliffs, NJ, 1989.
- 2) G.C. Goodwin and K.S. Sin, *Adaptive Filtering Prediction and Control*, Prentice-Hall, Inc., 1984.
- 3) L. Ljung and T. Söderström, *Theory and Practice of Recursive Identification*, MIT press, Cambridge, Mass, 1983.
- 4) K. Kumamaru, J. Hu, K. Inoue, and H. Ono, "Fault detection via KDI in presence of unmodelled uncertainty", in *Proc. of the 26th ISIE International Symposium on Stochastic Systems Theory and Its Applications (Osaka, Japan)*, 1994, pp. 173-178.
- 5) J. Hu and K. Kumamaru, "Identification of nonlinear systems based on adaptive fuzzy systems embedding quasi-ARMAX model", in *Proc. of the 34th SICE Annual Conference (international session)*, Sapporo, 1995, pp. 1211-1216.
- 6) J. Sjöberg, Q. Zhang, L. Ljung, A. Benveniste, B. Deglon, P.Y. Glorennec, H. Hjalmarsson, and A. Juditsky, "Nonlinear black-box modelling in system identification: a unified overview", *Automatica*, vol. 31, no. 12, pp. 1691-1724, 1995.
- 7) K.C. Sharman and T.S. Durrani, "Annealing algorithms for adaptive array processing", in *Proc. of the 8th IFAC Symposium on Identification and System Parameter Estimation (Beijing, China)*, 1988.
- 8) J. Sjöberg, H. Hjalmarsson, and L. Ljung, "Neural networks in system identification", in *Proc. of the 10th IFAC Symposium on Identification (Copenhagen)*, 1994, vol. 3, pp. 49-72.
- 9) K. Kristinsson and G.A. Dumont, "System identification and control using genetic algorithms", *IEEE Trans. Syst., Man & Cybern.*, vol. 22, no. 5, pp. 1033-1046, 1992.
- 10) D.E. Goldberg, *Genetic Algorithms in Search, Optimization & Machine Learning*, Addison-Wesley Publishing Company, Inc., 1989.
- 11) L. Davis, *Genetic Algorithms and Simulated Annealing*, London, Pitman Publishing, 1987.
- 12) Z.J. Yang, T. Hachino, T. Tsugi, and S. Sagara, "Identification on parameters and time delay of continuous systems using the genetic algorithm", in *Proc. of the 10th IFAC Symposium on Identification (Copenhagen)*, 1994, vol. 3, pp. 657-662.
- 13) M. Tanaka, T. Hattonri, and T. Tanino, "Jump detection and identification of linear systems by the genetic algorithm (in Japanese)", *Trans. of the Society of Instrument and control Engineers*, vol. 28, no. 11, pp. 1383-1385, 1992.
- 14) H. Kitano, *Genetic Algorithm (in Japanese)*, SanGyou Tosyo, Japan, 1993.
- 15) T. Murada, H. Ishibuchi, and H. Tanaka, "Flowshop scheduling by genetic algorithm and its application to multi-objective problem (in Japanese)", *Trans. of the Society of Instrument and control Engineers*, vol. 31, no. 5, pp. 583-590, 1995.
- 16) D. Beasley, D.R. Bull, and R.R. Martin, "An overview of genetic algorithms: Part 1, fundamentals", *University Computing*, vol. 15, no. 2, pp. 58-69, 1993.
- 17) F. Herrera, M. Lozano, and J.L. Verdegay, "Genetic algorithms applications to fuzzy logic based

systems", Tech. Rep. DECSAI-93116, Facultad de Ciencias, Universidad de Granada, 1993.

Jinglu Hu (Student Member)

He received the M. Sci. degree in 1986 from Zhongshan University, Guangzhou, China. From 1988, he was a Lecturer in Zhongshan University. Since 1993, he has been an overseas student under the Japanese Government Scholarship at the Faculty of Computer Science and Systems Engineering, Kyushu Institute of Technology, Japan. Now, he is an Ph.D candidate at the same university. His current research interests are system identification and its applications.

Kousuke KUMAMARU (Member)

He received the Ph.D degree in 1972 from Kyushu University. From 1973, he was an Associate Professor, and from 1987 a Professor in Kyushu University. Since 1989, he has been a Professor in Kyushu Institute of Technology. His current research interests are in the field of estimation and identification of dynamics systems, and their applications. Dr. Kumamaru is a member of the Institute of Electrical Engineers, the Institute of Systems, Control and Information Engineers, and the Japan Society of Medical Electronics Society and Biological Engineering.

Katsuhiko INOUE (Member)

He received the Ph.D degree in 1983 from Kyushu University. From 1981 to 1983, he was a Teaching Assistant in Kyushu University, from 1983, a Lecturer, and then an Associate Professor in Oita University. Since 1990, he has been an Associate Professor in Kyushu Institute of Technology. His current research interests are system identification, pattern recognition theory and their applications. Dr. Inoue is a member of the Institute of Electrical Engineers, the Institute of Systems, Control and Information Engineers, and the Japan Society of Medical Electronics Society and Biological Engineering.