International Journal of Fuzzy Logic and Intelligent Systems, vol. 2, no. 1, March 2002 pp. 53-58

# Fuzzy Model Identification Using VmGA

**Jong Il Choi, Jae Heung Oh, and Young Hoon Joo**

**Department of Electronic and Information Engineering, Kunsan National University**

**Abstract**

In the construction of successful fuzzy models for nonlinear systems, the identification of an optimal fuzzy model system is an important and difficult problem. Traditionally, sGA(simple genetic algorithm) has been used to identify structures and parameters of fuzzy model because it has the ability to search the optimal solution somewhat globally. But SGA optimization process may be the reason of the premature local convergence when the appearance of the superior individual at the population evolution. Therefore, in this paper we propose a new method that can yield a successful fuzzy model using VmGA(virus messy genetic algorithms). The proposed method not only can be the countermeasure of premature convergence through the local information changed in population, but also has more effective and adaptive structure with respect to using changeable length string. In order to demonstrate the superiority and generality of the fuzzy modeling using VmGA, we finally applied the proposed fuzzy modeling methodof a complex nonlinear system.

**Key words** : Genetic algorithms, virus messy genetic algorithms, fuzzy modeling, nonlinear system.

## Ⅰ. Introduction

Fuzzy modeling represents more effectively man's mind and natural language than conventional mathematical models, and well suited to deal with complex, nonlinear, ill-defined and uncertain system. There are so much nonlinearity in realistic systems. And, we can hardly obtain the accurate differential equation. Therefore, several fuzzy modeling methods, which can represent nonlinear systems, have been proposed in recent years[1-2].

Recently, fuzzy modeling scheme such as the GA(Genetic Algorithm) by which the parameters and the structures of a fuzzy model are tuned have been applied to various kind of nonlinear system. For example, Joo proposed various hybrid algorithms using GAs with fuzzy c-means clustering methods[2]. Shimojima proposed a new kind of the GA hybrid scheme that has the hierarchical structure[3]. The GA is one of stochastic optimization methods simulating the process of the natural evolution, which is composed of selection, crossover, and mutation.

The GA encodes each point in a parameter space into a binary bit string called a chromosome, and each point is associated with a "fitness" value that, for maximization, is usually equal to the objective function evaluated at the point. Instead of a single point, the GA usually keeps a set of points as a population, which is then evolved repeatedly toward a better overall fitness value. But, the conventional GAs encode the solution space to the fixed position, fixed length strings. Also, it is difficult to acquire properly linkages associated with a given problem because the encoded solution structures are not known. So, this weak linkage means that the building-block is likely to breakdown.

To prevent this problem, Goldberg[4-5] developed mGA(messy genetic algorithm) and Kagupta[6] proposed a gene expression messy genetic algorithm(GEMGA) and applied it to black box optimization, Hoffmann[7] used a mGA to optimize the hierarchical fuzzy inference rules. Chowdhury[8] applied the mGA to the design of a fuzzy neural network based controller for the inverted pendulum.

The virus theory of the evolution is based on the view that the virus transduction is a key mechanism for transporting segments of DNA across species. VEGA generally emulates the coevolution based on the horizontal evolution between the host individuals and the virus individuals[9-10]. And the vertical evolution accomplished generation to generation. The host population and the virus population are defined as a set of candidate solutions and a substring set of the host population respectively.

In this paper, we propose an automatic scheme for identification of the fuzzy model using VmGA. This method identifies optimal rules of he fuzzy inference system and parameters of the membership function simultaneously. To do this, we use two dimensional string representations for the fuzzy system and introduce the multi-objective fitness function. Therefore, the proposed method not only can be the countermeasure of the premature convergence through the local information changed in population by the virus infection, but also has more effective and adaptive structure with respect to using the changeable length string.

Manuscript received November 15, 2001; revised March 5, 2002.
This paper was supported in part by the Brain Korea 21 Project of Kunsan National University.

## Ⅱ. Fuzzy Modeling

Fuzzy modeling is an approaching method to constructing fuzzy models based on given input-output data or knowledge

of experience human experts by fuzzy set and IF-THEN linguistic rules. Therefore, we must decide to the fuzzy model's form through the modeling process. In this paper, we use a zero-order Takaki-Sugeno fuzzy model whose consequent parts are represented as crisp numbers.

The i-th rule of the TS fuzzy model to be identified is as follow:

Plant Rule $i$ :

IF $x_1(t)$ is $A_1^i$ and ... and $x_n(t)$ is $A_n^i$

THEN $y$ is $\omega_i$ $(i=1,2,\cdots,r)$(1)

where Rule i denotes the i-th fuzzy inference rule, $A_j^i$ (1 $\leq$ j $\leq$ n) are fuzzy sets, $x_j(t)$ (1 $\leq$ j $\leq$ n) are input variables, y is an output variable, $\omega_i$ takes fuzzy singleton(real number), and r is the total number of rules.

Equ, (2) shows the membership function equation form. In Equ. (3), $\mu_i$ is the fitness grade of the $i$th rule for input $x(t)$ . $A_j^i(x_j(t))$ is the grade of membership of $x_j(t)$ in $A_j^i$.

$$A_j^i(x_j; a_{ij}, b_{ij}, c_{ij}) = \begin{cases} \dfrac{x_j - a_{ij}}{b_{ij} - a_{ij}} & a_{ij} \leq x_j \leq b_{ij} \\ \dfrac{b_{ij} - x_j}{b_{ij} - a_{ij}} & b_{ij} \leq x_j \leq c_{ij} \\ 0 & x_j > c_{ij} \text{ or } x_j < a_{ij} \end{cases} \quad (2)$$

$$\mu_i(x(t)) = \prod_{j=1}^{n} A_j^i(x_j(t)) \quad (3)$$

By using the fuzzy inference method with a singleton fuzzifier, product inference, and center of gravity fuzzifier, fuzzy model (1) can be expressed as the following global model:

$$y^* = \frac{\sum_{i=1}^{r} \mu_i(x(t))\omega_i}{\sum_{i=1}^{r} \mu_i(x(t))} \quad (4)$$

## III. Fuzzy Modeling using VmGA

### 3.1 Structure and operators in VmGA

Though the GA is used the optimization method widely, this method has a defect to use fixed length strings. VmGA has changeable length strings in opposition to the conventional GAs. Therefore, the coding scheme of the VmGA is much more flexible than that of the GAs. Also the GA optimization process can be the reason of premature local convergence when the appearance of superior individual at the population evolution. Therefore, population lacks in genetic diversity, and the evolutionary direction of the whole population is decided by much more prominent individual. And improvement of fitness can not be expected. Because the conventional GAs deal with schemata indirectly. Therefore, we introduce virus infection operator based on the virus evolutionary theory. The virus infection operators directly increase effective schemata and ensure genetic diversity of virus individual[9-10]. Figure 1 represents population structure which divided into two phase in the evolutionary process.

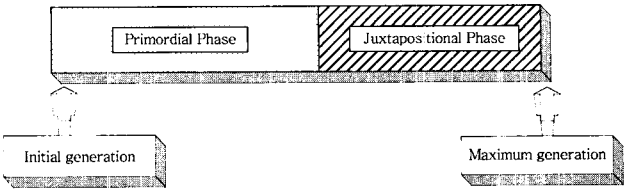


Fig. 1. A typical population reduction schedule

VmGA divides the evolutionary into two phases: a primordial phase and a juxtapositional phase. In the primordial phase, the individual that will just be evolved out of many strings of a population is selected. In the juxtapositional phase, the cut and splice operator is used to evolve individuals. The local and global searching capability of VmGA increases the effective schemata to the horizontal and vertical direction. We can reduce the required time to find the optimal solution through coevolution of host population and virus population. Therefore, we have two populations: a host population and a virus population in Fig. 2. Here a host population and a virus population are defined as a set of candidate solutions and a substring set of the host individuals, respectively.

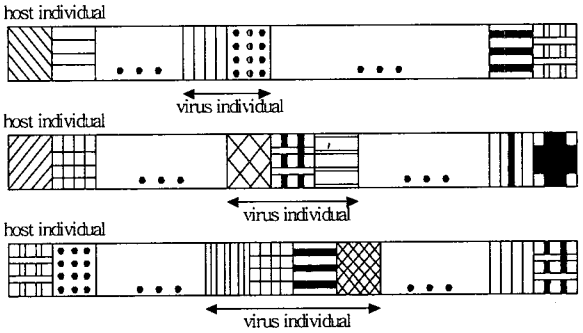


Fig. 2. Infected host individuals

In a crossover operation of the conventional GA, the position of the crossover point for two parent individuals has a common locus. But the crossover operator is no longer suitable to handle strings of variable length. Therefore VmGA use a cut and splice operator instead. Figure 3 represents the cut and splice operation process. The cut operator cuts a string at an arbitrary point with probability $p_0$. The splice operator splices two arbitrarily selected strings with probability $p_s$. A proportional selection in the GAs often causes a premature local convergence because of selecting an individual with the high fitness value many times.

And we consider both vertical inheritance of genetic information and horizontal propagation of effective schemata. Therefore, we consider both local and global search space by the virus infection operator. Fig. 3.5 represent virus infection operator, reverse transcription and transduction are introduced into the mGA as new searching operators. Reverse

transcription operator enables virus to overwrite its substring on the string of a host individual for generating new host individuals. And a virus performs the reverse transcription to a host individual randomly selected out of the host population.

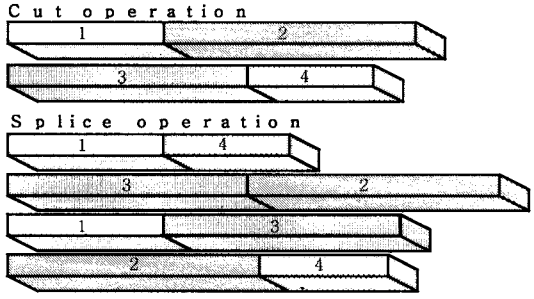


Fig. 4. Cut and splice operations

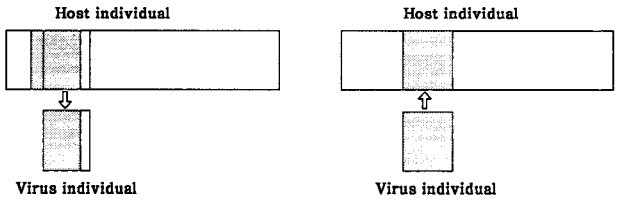


Fig. 5. Reverse transcription and transduction operators

VmGA operate in accordance with the fitness of virus individual. Therefore VmGA have the parameters of infection rate, virus fitness value and life force.

The fitness of the virus is shown by Equ. (5).

$$fitvirus_{ij} = fithost'_j - fithost_j$$
$$fitvirus_i = \sum_{j \in S} fitvirus_{ij} \tag{5}$$

where $fithost_j$ and $fithost'_j$ are the fitness value of a host individual j before and after the infection, respectively. The $fitvirus_{ij}$ denotes the difference between $fithost_j$ and $fithost'_j$. And the $fitvirus_i$ is $i$-th virus fitness. Equ. (6) represent the infection rate and each virus has $fitvirus_i$ for the virus infection. where $\beta$ is coefficient. After the infection rate is initialized, if $fitvirus_i$ has the positive value, the virus performs the transduction by taking out partially new substring from one of the infected host individuals. Otherwise, a virus shortens the genotype by removing some genes.

$$infrate_{i,t+1} = \begin{cases} (1+\beta) infrate_{i,t} & \text{if } fitvirus_i \geq 0 \\ (1-\beta) infrate_{i,t} & fitvirus_i < 0 \end{cases}, \beta < 0 \tag{6}$$

Each virus has a life force and represented by as follow:

$$life_{i,t+1} = r \times life_{i,t} + \alpha \times fitvirus_i \tag{7}$$

where, $life_{i,t+1}$ and $life_{i,t}$ is the Life force at the generation of t+1 and t, r is the life reduction rate, $\alpha$ is the life coefficient. If $life_{i,t+1}$ takes a negative value, the virus individual takes out a new substring with the transduction operator from the randomly selected host individual.

### 3.2 Coding Method

In the VmGA, genes are composed of index of the gene and the value corresponds to it. For example, a gene (1, 3) corresponds to the first gene in the string whose allele value is 3. Furthermore, unlike the conventional GAs, the order of genes in the string is not important in VmGA, i.e., the strings {(1, 3) (3, 1) (2, 1)} and {(2, 1) (1, 3) (3, 1)} are considered to be identical. Notice that we have not required all genes to be presented, nor have we precluded the possibility of multiple, possible contradictory, genes. For example, string {(1, 3) (2, 1)} and string {(1, 3) (2, 1) (3, 2) (1, 1)} are both valid. The former is said to be under-specified since there is no gene that corresponds to the third gene (3, x), and the latter is said to be over-specified. Fig. 6 can be shown as under specification of fuzzy rules in VmGA. In most problems, the string needs to be fully complemented with genes. In VmGA, this under-specification problem is easily tackled by using templates. Templates are used to fill unspecified genes in the string with locally optimal solutions. One way of solving the over-specified problem is to select a gene among conflicting genes based on the first-come-first-serve rule. Fig. 7 can be shown as over specification of fuzzy rules in VmGA. The next step is to design a proper structure of strings that best represent a given fuzzy inference system. In this paper, we use the fuzzy model (1). The parameters and the structure of the fuzzy model (1) are encoded into one or more substrings in a string. Parameters and the structure of the fuzzy inference system can be represented in a two-dimensional matrix form, as shown in Fig. 7 and Fig. 8 can be shown as the parameter matrix of the fuzzy inference system and the raw structure of the string not represented by VmGA coding. In this paper, we slightly modify the coding of the standard mGA string to effectively represent the fuzzy inference system. In the proposed method, one gene is composed of three elements, i.e., the gene {(i, j, p)} corresponds to the (i, j)the element in the parameter matrix with value p. Figure 9 illustrates an example of the parameter matrix of the fuzzy inference system and Fig. 10 shows an example of decoding the string. The resulting inference system is shown in Fig. 3.11.

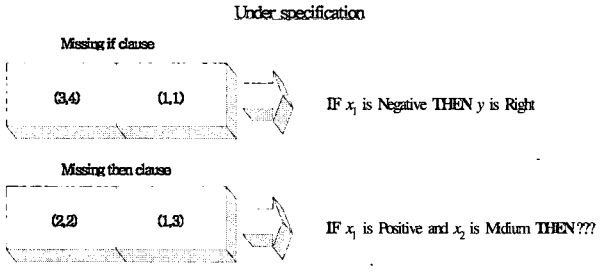


Fig. 6. Under specification of fuzzy rules in VmGA

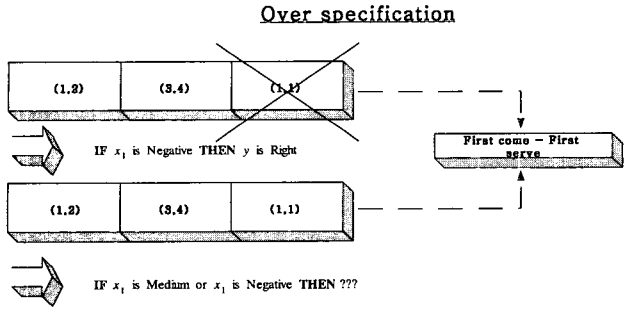


Fig. 7. Over specification of fuzzy rules in VmGA

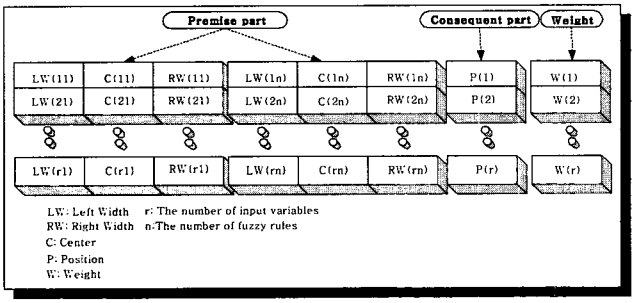


Fig. 8. The structure of a string in fuzzy modeling

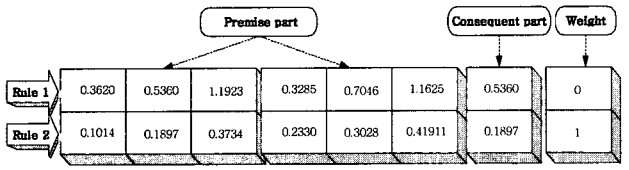


Fig. 9. An example of the parameter matrix of the fuzzy inference system

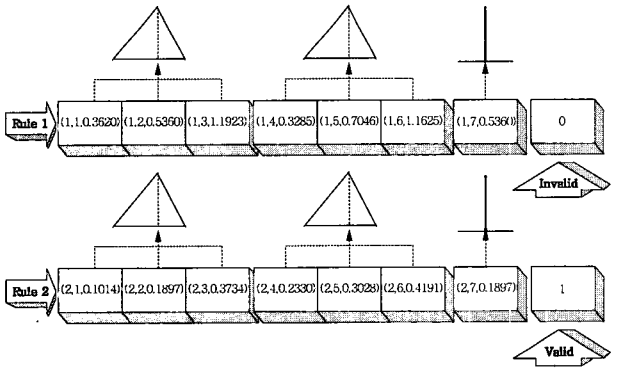


Fig. 10. An example of the proposed VmGA decoding process

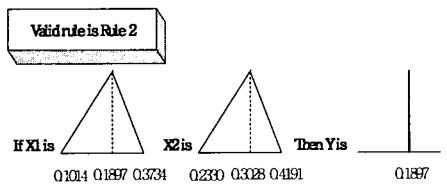


Fig. 11. The resulting fuzzy inference system

### 3.3 Multi-Objective Fitness Function

Since the GA is guided only by the fitness function, we have to consider more delicate fitness functions to better determine the accuracy and the size of the fuzzy inference system. The performance measures of the accuracy and size of the fuzzy inference system are as follows:

$$PI_{accuracy} = \frac{1}{N}\sum_{i=1}^{N}(y_i - y_i^d)^2 \tag{8}$$

$$PI_{size} = r \tag{9}$$

where r is the number of rules in the fuzzy inference system and $y_i^d$ are the desired outputs. As stated above, the purpose of identification is to reduce $PI_{accuracy}$ and $PI_{size}$. However, VmGA uses the fitness value, which has to be maximized in general, so we have to determine the transformation form performance index to fitness function:

$$f(PI_{accuracy}, PI_{size}) = \lambda\frac{1}{1+PI_{accuracy}} + (1-\lambda)\frac{1}{1+PI_{size}} \tag{10}$$

where $\lambda$ $(0 \le \lambda \le 1)$ is the weighting factor.

Fuzzy model with null sets cannot infer suitable outputs from the given input data set. In order to solve this problem, we introduce the following type of penalty function for the fitness function:

$$P = \sum_{i=1}^{N} p_i \tag{11}$$

where $p_i$ is a constant value if there exists null sets in the fuzzy inference system for the given $i$th input data set, otherwise $p_i$ is zero.

Also by combining Eqt. (8) and (9), we have following fitness function:

$$f(PI_{accuracy}, PI_{size}) = \left\{\lambda\frac{1}{1+PI_{accuracy}} + (1-\lambda)\frac{1}{1+PI_{size}}\right\}\frac{1}{1+P} \tag{12}$$

### 3.4 Fine Tuning

In this paper, we use the solution that is identified by VmGA by initial value of gradient descent method. In this hybrid scheme, the objective function to be minimized is:

$$E = \frac{1}{2}(y_i - y_i^d)^2 \tag{13}$$

The parameter update rules for the fuzzy inference system can be easily derived by using the chain rule as follows:

$$\begin{aligned} a_{ij}(k+1) &= a_{ij}(k) - K_a\frac{\partial E}{\partial a_{ij}} \\ b_{ij}(k+1) &= b_{ij}(k) - K_b\frac{\partial E}{\partial b_{ij}} \\ c_{ij}(k+1) &= c_{ij}(k) - K_c\frac{\partial E}{\partial c_{ij}} \\ w_{ij}(k+1) &= w_{ij}(k) - K_w\frac{\partial E}{\partial w_{ij}} \end{aligned} \tag{14}$$

where $K_a$, $K_b$, $K_c$, and $K_w$ are learning rates and the partial derivatives can be further derived as:

## IV. Simulation Results

To demonstrate the superiority of the proposed fuzzy modeling method, we consider the nonlinear system shown in Equ. (15).

$$y = (1 + x_1^{-2} + x_2^{-1.5})^2, \quad 1 \le x_1, x_2 \le 5 \qquad (15)$$

This nonlinear system has two inputs and one output. We use 50 input-output data pairs extracted from nonlinear equation as same as Sugeno's. Table 1 shows the parameters for modeling of nonlinear system

Table 1. Initial parameters for modeling of nonlinear system

| Parameter's name | Parameter's value |
|---|---|
| Maximum generation number | 200 |
| Population number | 100 |
| Cut, Splice, and Mutation rate | 0.2, 1.0, 0.2 |
| Iterated generation number | 1000 |
| Reduction rate, Life coefficient | 0.1, 0.9 |
| Ka, Kb, Kc, Kw | 10-5, 10-5, 10-5, 10-4 |
| $\beta$, $\lambda$ | 0.1, 0.9 |
| Initial length of virus | 3 |

After the proposed identifying method, the rules of the resulting fuzzy model are 4 and the MSE is 0.037359. The parameters of the fuzzy model identified by the proposed method are listed in Table 2. In Table 3, we compare the performance of our constructed fuzzy model with Sugeno's, in which our model has more excellent performance with the smaller number of rules.

Table 2. The membership parameters of the fuzzy rules

| | $X_1$ | | | $X_2$ | | | $Y$ |
|---|---|---|---|---|---|---|---|
| | $a_{i1}$ | $b_{i1}$ | $c_{i1}$ | $a_{i2}$ | $b_{i2}$ | $c_{i2}$ | $w_i$ |
| **rule1** | 0.9501 | 0.8913 | 0.8214 | 0.9218 | 0.9355 | 0.0579 | 0.1389 |
| **rule2** | 0.8965 | 0.0210 | 0.2492 | 0.8432 | 0.0635 | 0.8690 | 0.9348 |
| **rule3** | 0.3118 | 0.2591 | 0.8069 | 0.4346 | 0.3446 | 0.2802 | 0.4463 |
| **rule4** | 0.4267 | 0.4228 | 0.9717 | 0.3126 | 0.4114 | 0.8909 | 0.0274 |

Table 3. Comparisons of our model with other ones

| Model | The Number of rules | MSE |
|---|---|---|
| Sugeno's | 6 | 0.079 |
| Our model | 4 | 0.037 |

## V. Conclusions

In this paper, we propose an automatic algorithm for identification of fuzzy inference systems based on VmGA. In order to settle premature convergence in sGA, we introduce virus infection operator. Therefore we consider both vertical inheritance of genetic information and horizontal propagation of effective schemata. As the mGA use a cut and splice operator instead of crossover operator in conventional sGA, we can increase flexibility of identification. Conclusionally, the proposed method not only can be the countermeasure of premature convergence through the local information changed in population, but also has more effective and adaptive structure with respect to using changeable length string. In order to demonstrate the superiority and efficiency of the proposed scheme, we applied this method to the approximation of a complex nonlinear system.

## References


[1] H. Takagi and Sugeno, "Fuzzy Identification of Systems and Its Application to Modeling and Control", *IEEE Trans. on Sys.* Man and Cybern., vol. 15, pp. 116-132, 1985.

[2] Y. H. Joo, H. S. Hwang, K. B. Kim and K. B. Woo, "Linguistic Model Identification for Fuzzy System", *Electronics Letters*, vol. 31, no. 4, pp. 330-331, 1995.

[3] K. Shimojima, T. Fukuda and Y. Hasegawa, Self-Tuning Fuzzy Modeling with Adaptive Membership Function, Rules, and Hierarchical Structure Based on Genetic Algorithm, Fuzzy Sets and Systems, vol. 71, pp. 295-309, 1995.

[4] D. E. Goldberg, Genetic Algorithms in Search, Optimization & Machine Learning, Addison- Wesley, 1990.

[5] D. E. Goldberg, B. Korb, and K. Deb, "Messy Genetic Algorithms: Motivation, Analysis, and First Results", *Complex Systems*, vol. 3, no. 5, pp. 493-530, 1989.

[6] H. Kargupta, "The Gene Expression Messy Genetic Algorithm", *Proc. of IEEE Int.* Conf. on Evolutionary Computation, Nagoya, Japan, 1996

[7] F. Hoffmann and G. Pfister, "A New Learning Method for the Design of Hierarchical Fuzzy Controllers Using Messy Genetic Algorithms", *Proc. IFSA'95*, July 1995.

[8] M. Chowdhury and Y. Li, "Messy Genetic Algorithm Based New Learning Method for Structurally Optimized Neurofuzzy Controllers", *Proc. IEEE Int.* Conf. on Industrial Tech., Dec. 1996.

[9] N. Kubota, T. Fukuda and K. shimojima, "Virus-Evolutionary Genetic Algorithm for a Self- Organizing Manufacturing System", *Computers & Industrial Engineering*, vol. 30, Issue 4, pp. 1015-1026, Sep. 1996.

[10] N. Kubota, K. shimojima, T. Fukuda, "The Role of Virus Infection in Virus-Evolutionary Genetic Algorithm", *Proc. of IEEE Int.* Conf. on, pp. 182-187, 1996

**Jong-Il Choi**
He received the B. S. and M. S. degrees in the school of the electronic and information engineering from Kunsan National University, Korea in 2000 and 2002, respectively. His current research interests are in intelligent control, fuzzy modeling, and genetic algorithms.

Phone: +82-63-469-4706 , Fax: +82-63-469-4706
E-mail: jongiree@hanmir.com

**Jae-Heung Oh**
He received the B. S. degree in electrical engineering from Kunsan National University, Korea in 2002, He is currently working towards the MS degree in the department of Electronic and information engineering, Kunsan National University, Korea. His current research interests are in vision, pattern recognition, neural network, and genetic algorithms.

Phone: +82-63-469-4706 , Fax: +82-63-469-4706
E-mail: jaeheung@dreamwis.com

**Young-Hoon Joo**
He received the B.S., M.S., and Ph.D. degrees in electrical engineering from the Yonsei University, Korea, in 1982, 1984, and 1995, respectively. He worked with Samsung Electronics Company, Korea, from 1986 to 1995, as a Project Manager. He was with University of Houston, TX, from 1998 to 1999, as a Visiting Professor in the Department of Electrical and Computer Engineering. He is currently Associate Professor in the School of Electronic and Information Engineering, Kunsan National University, Korea. His major is mainly in the field of mobile robots, fuzzy modeling and control, genetic algorithms, intelligent control, and nonlinear systems control. Prof. Joo is serving as the Associate Editor for the Transactions of the KIEE and Journal of Fuzzy Logic and Intelligent Systems.

Phone: +82-63-469-4706, Fax: +82-63-469-4706
E-mail:yhjoo@kunsan.ac.kr