

Design of Fuzzy Controller for Car Parking Problem Using Evolutionary Multi-objective Optimization Approach

Joon-Yong Lee

Department of Electrical Engineering
and Computer Science, KAIST
Taejeon, KOREA 305-701
Email: junyoni@odyssey.kaist.ac.kr

Min-Soeng Kim

Department of Electrical Engineering
and Computer Science, KAIST
Taejeon, KOREA 305-701
Email: mibella@odyssey.kaist.ac.kr

Ju-Jang Lee

Department of Electrical Engineering
and Computer Science, KAIST
Taejeon, KOREA 305-701
Email: jjlee@ee.kaist.ac.kr

Abstract— This paper proposes an automatic method to design a fuzzy logic controller for the automated car parking problem within the multi-objective evolutionary optimization framework, which requires three factors: an encoding scheme, design of evaluation criteria, and design of evolutionary operations. Parameters of antecedent membership functions and consequent parameter vectors are encoded into a chromosome so that antecedent/consequent parameters and rule structure of a fuzzy logic controller are optimized simultaneously during evolution. Three optimization criteria are proposed to find a set of good fuzzy logic controllers using Pareto optimality. The proposed algorithm is applied to the well-known car parking problem and produced a Pareto set of fuzzy logic controllers that can control the motion of a vehicle for automated parking. Each fuzzy logic controller in the set has unique characteristics and can be selected according to users' preferences, which is one of the major advantages of using the multi-objective evolutionary optimization.

I. INTRODUCTION

Steering control of a vehicle system is important and indispensable for the safety of drivers and passengers. Parking a car is a typical example of such steering control. Usually, man can perform steering control easily after acquisition of appropriate experience without deep understanding about the vehicle. This fact reveals that an intelligent control scheme such as fuzzy logic controller and neural network controller will be effective to achieve an automated vehicle control system. Thus, automated vehicle control systems were designed using intelligent control schemes such as fuzzy logic control [1] - [3], neural networks (NNs) [4], neuro-fuzzy control [5], and so on. Moreover, some researches stated that a fuzzy controller from human experts' knowledge is more preferable for this kind of problem [6] - [8].

Including control applications, fuzzy logic controllers have been widely applied to the various fields. They can deal with the given problem effectively without any mathematical description about the given problem. A fuzzy logic controller consists of several IF-THEN rules, which describes adequate control strategy for the given

problem in linguistic terms. Human experts' knowledge can build such fuzzy rules directly. In [2] [3], a fuzzy logic controller is manually designed based on human experimental knowledge.

However, such experts' knowledge is not always available. Consequently, 'data-driven' design of a fuzzy logic controller have been studied for decades [2] [3] [9] [10]. Clustering algorithms, linear/nonlinear optimization algorithms are examples of major tools for the data-driven design.

Because a fuzzy logic controller can be specified by a set of parameters, design of a fuzzy logic controller can be formulated as a parameter search problem for which evolutionary algorithms (EA's) such as genetic algorithms (GA's) and evolution strategies (ES's) have shown powerful effectiveness [10]. For example, evolutionary algorithms tunes the parameters of membership functions (MFs) of fuzzy logic controllers with fixed rule structure in [11] - [14]. Because parameters and rule structure of fuzzy logic controllers are co-dependent, simultaneous optimization of rule structure and parameters of a fuzzy logic controller provides better solution [2] [15] [16].

Design a fuzzy logic controller using evolutionary algorithms requires three factors: an encoding scheme which represents a fuzzy logic controller by a chromosome, evaluation criteria which guide evolutionary search direction to find better solution as evolution progresses, and design of relevant evolutionary operators.

Especially, an encoding scheme should be efficient enough so that MFs and rule structure can be evolved simultaneously. Multi-objective evolutionary algorithms can show better performance in finding good solutions when there are more than two possibly conflicting and noncommensurate evaluation criteria.

Thus, this paper proposes a method to design a fuzzy logic controller design within the multi-objective evolutionary algorithm framework. The proposed algorithm mainly consists of a new encoding scheme and new evaluation criteria, i.e., fitness functions. The proposed encoding scheme helps to achieve simultaneous optimization

of rule structures and relevant parameters. Parameters of antecedent membership functions and consequent parameter vectors are encoded into a chromosome so that antecedent/consequent parameters and rule structure of a fuzzy logic controller are optimized simultaneously during evolution. The proposed fitness functions consist of three objectives which finds a more compact and transparent fuzzy logic controller based on Pareto optimality.

The remainder of the paper is as follows: Section II gives a description about the automated car parking problem. Section III explains design considerations when designing a fuzzy logic controller with grid partitioning of input space. Section IV proposes a new encoding scheme and new fitness functions. Section V presents simulation results to show the effectiveness of the proposed algorithm. Finally, Section VI concludes the paper.

II. AUTOMATED CAR PARKING PROBLEM

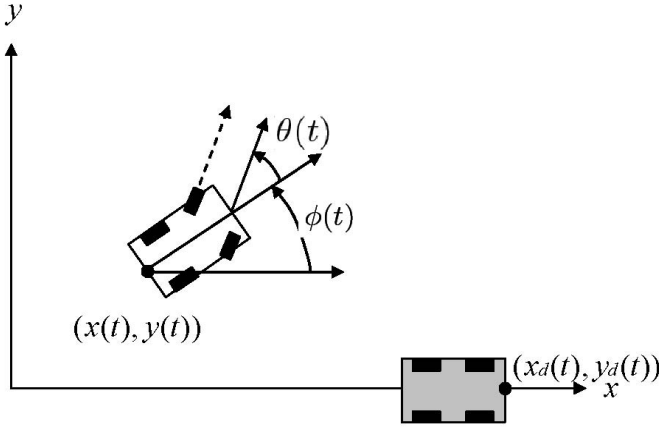


Fig. 1. Car parking problem.

Figure 1 depicts the model of the automated car parking problem. Motion of the car is described by the following equations [1] [2].

$$\phi'(t) = \phi(t) + vt/l \tan(\theta(t)), \quad (1)$$

$$x'(t) = x(t) + vt_s \cos(\phi(t)), \quad (2)$$

$$y'(t) = y(t) + vt_s \sin(\phi(t)), \quad (3)$$

where $\phi(t)$ is the angle of the car with respect to the x axis and $\theta(t)$ is the steering angle. $x(t)$, $y(t)$ represent horizontal and vertical position of the rear end of the car, respectively. l is the length of the car and t_s is the sampling time. v is linear velocity of the car which is assumed to be constant value. In this paper, $l = 3.0(m)$, $v = 1.0(m/s)$, and $t_s = 1.0(s)$.

In the automated car parking problem, initial posture of a car is given as $P(0) = (x(0), y(0), \phi(0))$. The goal of the problem is to take the car in minimum time from the initial posture to the desired location and orientation, $P_d = (x_d, y_d, \phi_d)$. $P(t) = (x(t), y(t), \phi(t))$ represents the

current posture of the car at time t , and $P_{err}(t) = (e_x(t), e_y(t), e_\phi(t)) = P_d - P(t)$ denotes errors in position and orientation in time t . Thus, the goal becomes to design a fuzzy logic controller that can reduce the errors efficiently. In this paper, $e_x(t)$ and $e_\phi(t)$ are inputs of a fuzzy controller and $\theta(t)$ is an output of the fuzzy controller. Vertical position error is not considered as in [1] - [5] with the assumption that the car is far from the parking lot along the y direction.

III. FUZZY LOGIC CONTROLLER

A typical fuzzy logic controller consists of several IF-THEN rules where rule consequent values represents control efforts. Since there are two input variables for the automated car parking problem, there are at most $N_R = m_1 \times m_2$ fuzzy rules where m_1 and m_2 are the number of MFs for horizontal positional error and car angle error, respectively. Then, a fuzzy rule is described as follows:

$$R_k : \text{IF } e_x \text{ is } A_i^{e_x} \text{ and } e_\phi \text{ is } A_j^{e_\phi} \text{ then } \theta_k = w_k \quad \text{for } k = 1, 2, \dots, N_R \quad (4)$$

where θ_k is the steering angle of the car for the k th rule, and w_k is real-valued consequent parameters of the k th fuzzy rule. N_R is the total number of available rule. $A_i^{e_x}$ ($i = 1, 2, \dots, m_1$) and $A_j^{e_\phi}$ ($j = 1, 2, \dots, m_2$) denote the i th and j th antecedent MFs for the both input variable, respectively. Antecedent MFs, $\mu_{A_i^{e_x}}(e_x)$, is described by a Gaussian function as follows:

$$\mu_{A_i^{e_x}}(e_x) = \exp\left(\frac{-(e_x - c_i^{e_x})^2}{(\sigma_i^{e_x})^2}\right) \quad (5)$$

where $c_i^{e_x}$ and $\sigma_i^{e_x}$ are the center value and the width value of the i th MF of the horizontal positional error, respectively. Similar definition is made for the antecedent MFs for the car angle error. As horizontal positional error and car angle error change each fuzzy rule produces a rule strength for the current input values. Rule strength of the k th fuzzy rule, $\mu_k(\mathbf{x})$, is calculated as follows:

$$\mu_k(e_x, e_\phi) = \mu_{A_i^{e_x}}(e_x) \times \mu_{A_j^{e_\phi}}(e_\phi) \quad (6)$$

The fuzzy logic controller produces final control $\hat{y}$ by means of weighted mean defuzzification method as follows:

$$\hat{y} = \frac{\sum_{k=1}^{N_R} \mu_k(e_x, e_\phi) \cdot w_k}{\sum_{k=1}^{N_R} \mu_k(e_x, e_\phi)} \quad (7)$$

Design of a fuzzy logic controller is to find a set of parameters for the fuzzy controller shown above. The number of antecedent MFs for each input variable should be determined, parameter values for each MF should be adjusted, fuzzy rule consequent parameters should be optimized, and a set of useful fuzzy rules should be selected from the entire fuzzy rules for proper motion control of the car. To resolve these issues, a new encoding scheme and evaluation criteria are proposed in the next section.

IV. PROPOSED ALGORITHM

A. Encoding a fuzzy logic controller

Whenever solving a certain problem under the multi-objective optimization framework, one of the most important steps is to provide an efficient encoding method. In this paper, this step corresponds to encoding a fuzzy logic controller into a chromosome. The number of antecedent MFs for both input variables are defined as m_1 and m_2 *a priori*. Two real values are necessary for each antecedent MF while optimization of consequent parameters and rule selection is performed using $m_1 \times m_2$ real values. A p th individual, i.e., an encoded fuzzy model, is represented in a vector form as follows:

$$s_p(t) = [v_1^{e_x} \ v_2^{e_x} \ \dots \ v_{m_1}^{e_x} \ v_1^{e_\phi} \ v_2^{e_\phi} \ \dots \ v_{m_2}^{e_\phi} \ w_1 \ w_2 \ \dots \ w_{N_R}] \quad (8)$$

where $v_i^{e_x} = [c_i^{e_x}, \omega_i^{e_x}] (i = 1, \dots, m_1)$ and $v_j^{e_\phi} = [c_j^{e_\phi}, \omega_j^{e_\phi}] (j = 1, \dots, m_2)$ represent the parameters (i.e., center and width) of the i th and j th antecedent MFs of the horizontal positional error and the car angle error, respectively. $w_k (k = 1, \dots, N_R)$ is used for calculating control output and rule selection and ranged $[0.0 \ 1.0]$. If a rule of which w_k is less than predefined design parameter ϵ then the rule is excluded from the rule base. Otherwise, w_k is normalized into control efforts as shown in the following equation.

$$\theta_k = \theta_{max} - \left(\frac{1 - w_k}{1 - \epsilon} \right) (\theta_{max} - \theta_{min}) \quad (9)$$

where θ_{max} and θ_{min} are the maximum and the minimum value for the steering angle.

B. Designing evaluation criteria

Given an encoding scheme, the next step is to evaluate each fuzzy logic controller's performance according to appropriate criteria. Thus, three objective functions are proposed in this paper considering control error, distribution of antecedent MFs, and compactness of fuzzy logic controllers.

Because one of the most important objectives of automated car parking problem is to park the vehicle at the desired position with desired pose, in several previous approaches [1] [2], this objective is usually defined considering errors as follows:

$$f_{err}^1(s_p) = \sum_k (|e_x^k| + |e_\phi^k|), \quad (10)$$

where k represents the time index. Also, there was different objective function which gives more penalty for the steady-state errors as shown in the following equation [5].

$$f_{err}^2(s_p) = \sum_k (((e_x^k)^2 + 1)((e_\phi^k)^2 + 1)k). \quad (11)$$

As shown in the last term of the Eq. (11, k operates as a weighting error values as time increases.

A new objective function is proposed for evaluating the error between current and desired position and orientation by compromising f_{err}^1 and f_{err}^2 as in the following equation.

$$f_{err}(s_p) = \sum_k \sqrt{k(w_x(e_x^k)^2 + w_\phi(e_\phi^k)^2)}, \quad (12)$$

where w_x and w_ϕ are weight for scaling the error of position and posture. In this paper, $w_x = 0.5$ and $w_\phi = 1$.

The second objective is related to the distribution of antecedent MFs of a fuzzy logic controller for better interpretability [16].

$$f_{intr}(s_p) = 0.5 \cdot (1.0 - \frac{1}{N_I} \sum_{i=1}^{N_I} \frac{1}{N_{ov}^i} \sum_{h=1}^{N_{ov}^i} \frac{\lambda_{ih}}{|\chi_i|}) + 0.5 \cdot (1.0 - \frac{1}{N_I} \sum_{i=1}^{N_I} \frac{1}{N_{co}^i} \sum_{h=1}^{N_{co}^i} \frac{\gamma_{ih}}{|\chi_i|}) \quad (13)$$

where λ_{ih} represents the length of overlap at the predefined level ξ_D between two MFs of the i th input while γ_{ih} the length of gap at the predefined level ξ_C between two MFs of the i th input. χ_i is the length of the i th input universe of discourse. N_I represents the number of input and is 2. N_{ov}^i and N_{co}^i represent the number of calculating the length of overlap and the length of gap for the i th input, respectively. This objective function is useful to preserve the distinguishability of antecedent MFs.

The third objective function considers the compactness of a fuzzy logic controller. The compactness of a fuzzy logic controller is measured by the ratio between the actually-being-used number of fuzzy rules by a specific individual and the maximum number of rules pre-determined. The measure for the compactness of a fuzzy model is defined as follows [16]:

$$f_{comp}(s_p) = \frac{N_R^p}{N_R}, \quad (14)$$

where N_R^p is the actual number of fuzzy rules that the p th individual uses.

In addition to these three fitness function, a penalty function is used heuristically to prevent a solution of which error is far bigger than predefined error e_{max} and defined as follows:

$$f_{pen}(s_p) = \begin{cases} P_{max}, & \text{if } f_{err}(p_i) > e_{max} \\ 0, & \text{otherwise} \end{cases}, \quad (15)$$

where P_{max} is the penalty value given for the violated p th individual.

C. Assigning fitness value

The multi-objective optimization problem is to find the point $\mathbf{x} = (x_1, \dots, x_n)$ which optimizes the values of a set of objective functions $\mathbf{f} = (f_1, \dots, f_m)$ within the feasible

region of $\mathbf{x}$ [17]. For two arbitrary individuals $a, b \in \mathbf{P}$. It is said that a dominates b , denoted by $a \succeq b$, if and only if:

$$f_i(a) \leq f_i(b), \forall i = \{1, \dots, m\},$$

with at least one strict inequality. Every individual that is not dominated by any other individual are called non-dominant set or Pareto-optimal set. For the car parking problem, a set of objective functions is defined as $f = (f_{err}, f_{intr}, f_{comp})$. f_{pen} should also be considered. If $f_{pen}(\mathbf{x}) \neq 0$, the individual $(\mathbf{x})$ is not feasible so that it should be dealt as an unfeasible solution. Pareto-based approach is used to find non-dominant set. The definition which has been mentioned above for the Pareto-optimality is adjusted to detect the non-dominant individuals as shown in the following equation. An individual $\mathbf{a}$ cannot be a member of the non-dominant set if and only if

$$\exists \mathbf{b} \in P (\text{iff } \mathbf{b} \neq \mathbf{a}), \\ f_i(\mathbf{a}) + f_{pen}(\mathbf{a}) \leq f_i(\mathbf{b}) + f_{pen}(\mathbf{b}), \forall i = \{\text{err, intr comp}\},$$

with at least one strict inequality.

In this case, only if there is no individual $\mathbf{b}$ satisfied with the above condition at all, individual $\mathbf{a}$ can become a member of the non-dominant set. After the individuals of the non-dominant set are evaluated, the individuals of the population set are also evaluated. The penalty fitness values of the individuals are used once more to guarantee that the solution does not violate the penalty constraints. Thus, the required condition is more strengthened. After all, the

Feasible individuals with a penalty value that is equal to zero are obtained. Those individuals become members of the non-dominant set, $\mathbf{NP}$. The fitness assignment of individual $\mathbf{x} \in \mathbf{NP}$ is as follows:

$$f(\mathbf{x}) = \frac{n_c(\mathbf{x})}{N_p + 1} + f_{pen}(\mathbf{x}) \quad (16)$$

where $n_c(\mathbf{x})$ represents the number of individuals which the individual $\mathbf{x}$ dominates in set $\mathbf{P}$. N_p is the number of individuals in P .

Figure 2 describes the fitness assignment for the non-dominant individuals. The fitness assignment of individual $\mathbf{y} \in \mathbf{P}$ is given by:

$$f(\mathbf{y}) = 1 + \sum_{\mathbf{x} \succeq \mathbf{y}} f(\mathbf{x}) + f_{pen}(\mathbf{y}) \quad (17)$$

The maximum number of non-dominated solutions are defined as N_{nondom} *a priori*. To prevent the number of non-dominated solutions from exceeding N_{nondom} , clustering algorithm is used [17] [18].

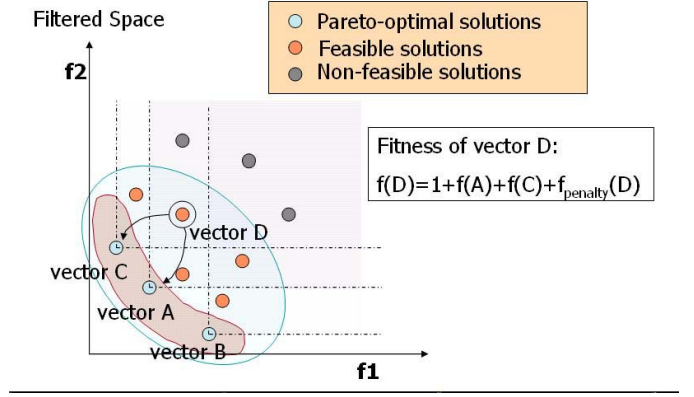


Fig. 2. The example of fitness assignment in P for the minimization problem.

V. SIMULATION

A. Parameter setting

In this section, simulations results are presented. As mentioned previously, the motion of a car is described by Eqs. (1)-(3). The linear velocity of car, v , is assumed constant and set as $0.5(m/s)$ in this paper. As shown in Table I, the constants and ranges of the other parameters are defined initially. ϵ in Eq. (9) was set to 0.5. ξ_D and ξ_C were 0.8 and 0.1, respectively. Predefined number of MFs for each input variable is five. Thus, a fuzzy controller can use up to 25 fuzzy rules. Population size was 50. After 500 generations, the evolution finishes. N_{nondom} was 20. Evolutionary operators are similar to those used in [15] while the evolution framework follows that of [17] [19]. Table II defines the initial and desired information of the position and posture of the car model.

B. Simulation results

The proposed algorithm produced N_{nondom} fuzzy logic controllers which can perform the automated car parking efficiently. Among these solutions, Fig. 3 describes the trajectories of the car controlled by eight fuzzy controller. Figure 5 represents the optimized MFs for the first fuzzy logic controller. As shown in Fig. 3, the first fuzzy logic controller shows better trajectory than the last one while using more rules than the last one. The first fuzzy controller uses six rules shown below, which achieves the given task. It means that among 25 fuzzy rules, this fuzzy logic controller which uses only six fuzzy rules were able to achieve the given car parking task with corresponding optimized MFs. In the fuzzy rules obtained, consequent parameters is decoded into actual steering angle using Eq. (9). For example, $\theta = 0.87$ is equal to 'steering angle is 24 degrees'.

- R_1 : IF e_x is $A_1^{e_x}$ and e_ϕ is $A_3^{e_\phi}$ then $\theta_1 = 0.63$
- R_2 : IF e_x is $A_2^{e_x}$ and e_ϕ is $A_1^{e_\phi}$ then $\theta_2 = -0.72$
- R_3 : IF e_x is $A_3^{e_x}$ and e_ϕ is $A_2^{e_\phi}$ then $\theta_3 = 0.10$
- R_4 : IF e_x is $A_3^{e_x}$ and e_ϕ is $A_3^{e_\phi}$ then $\theta_4 = 0.76$

TABLE I
CONSTANTS AND RANGES IN A SIMULATION

Parameters	value
v	0.5 m/sec
l	3.0 m
t_s	0.1 sec
e_x	-50 to +50 m
e_ϕ	-180 to 180 deg
θ	-50 to +50 deg

TABLE II
INITIAL AND DESIRED POSTURE OF THE CAR

Test conditions	value
(x_0, ϕ_0)	$(5.0(m), 270(deg))$
(x_d^1, ϕ_d^1)	$(1.0(m), 315(deg))$
(x_d^2, ϕ_d^2)	$(1.0(m), 22.5(deg))$
(x_d^3, ϕ_d^3)	$(1.0(m), 90(deg))$
Maximum Time	2 min

$$R_5 : \text{IF } e_x \text{ is } A_4^{e_x} \text{ and } e_\phi \text{ is } A_2^{e_\phi} \text{ then } \theta_5 = 0.76$$

$$R_6 : \text{IF } e_x \text{ is } A_4^{e_x} \text{ and } e_\phi \text{ is } A_3^{e_\phi} \text{ then } \theta_6 = -0.30$$

The last controller of Fig. 3 uses only 5 rules among 25 rules as follows:

$$R_1 : \text{IF } e_x \text{ is } A_1^{e_x} \text{ and } e_\phi \text{ is } A_1^{e_\phi} \text{ then } \theta_1 = -0.66$$

$$R_2 : \text{IF } e_x \text{ is } A_1^{e_x} \text{ and } e_\phi \text{ is } A_3^{e_\phi} \text{ then } \theta_2 = -0.33$$

$$R_3 : \text{IF } e_x \text{ is } A_2^{e_x} \text{ and } e_\phi \text{ is } A_3^{e_\phi} \text{ then } \theta_3 = 0.76$$

$$R_4 : \text{IF } e_x \text{ is } A_3^{e_x} \text{ and } e_\phi \text{ is } A_2^{e_\phi} \text{ then } \theta_4 = 0.76$$

$$R_5 : \text{IF } e_x \text{ is } A_3^{e_x} \text{ and } e_\phi \text{ is } A_3^{e_\phi} \text{ then } \theta_5 = 0.76$$

Distribution of the antecedent MFs of the last fuzzy logic controller and parameters of the MFs are shown in Fig. 4 and in Table IV, respectively.

VI. CONCLUSION

This paper proposed an automatic method to design a fuzzy logic controller for the automated car parking problem within the framework of multi-objective evolutionary algorithm. The proposed algorithm mainly consists of a new encoding scheme and new evaluation criteria. The parameters of antecedent MFs, the number of fuzzy rules, and consequent parameters are explicitly represented in the chromosome, a population of which evolves to optimize the rule structure as well as parameters of MFs simultaneously. The proposed fitness functions consist of three objectives which helps to find compact and interpretable fuzzy logic controllers with good control performance based on Pareto optimality. Simulation results shows that the proposed algorithm was able to find a set of fuzzy logic controllers which are adequate for the automated car parking problem.

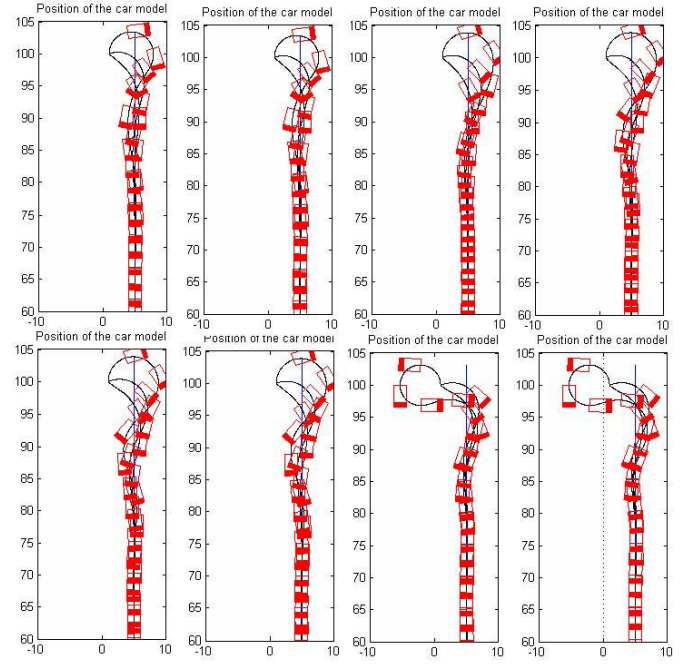


Fig. 3. The resulting eight trajectories of the car controlled by one of the obtained fuzzy logic controller.

TABLE III
CENTER AND WIDTH OF MFs SHOWN IN FIG. 4

MF	Center	Width
$A_1^{e_x}$	0.96 (m)	1.68 (m)
$A_2^{e_x}$	-1.49 (m)	2.40 (m)
$A_3^{e_x}$	0.30 (m)	1.58 (m)
$A_4^{e_x}$	3.26 (m)	2.46 (m)
$A_1^{e_\phi}$	-135.25 (deg)	122.13 (deg)
$A_2^{e_\phi}$	68.83 (deg)	60.05 (deg)
$A_3^{e_\phi}$	118.76 (deg)	74.30 (deg)

REFERENCES

- [1] K. Tanaka and M. Sano, "Trajectory stabilization of a model car via fuzzy control," *Fuzzy Sets Syst.*, vol. 70, pp. 155-170, 1995.
- [2] A. Homaifar and Ed. McCormick, "Simultaneous design of membership functions and rule sets for fuzzy controllers using genetic algorithms," *IEEE Trans. on Fuzzy Syst.*, vol. 3, no. 2, pp. 129-139, 1995.
- [3] Y. C. Chiou and L.W. Lan, "Genetic fuzzy logic controller: an iterative evolution algorithm with new encoding method," *Fuzzy Sets and Syst.*, vol. 152, no. 3, pp. 617-635, 2005.
- [4] D. Gorinevsky, A. Kapitanovsky, and A. Goldenberg, "Neural network architecture for trajectory generation and control of automated car parking," *IEEE Trans. Contr. Syst. Tech.*, vol. 4, no. 1, pp. 50-56, 1996.
- [5] T. L. Seng, M. B. Khalid, and R. Yusof, "Tuning of a neuro-fuzzy controller by genetic algorithm," *IEEE Trans. Syst., Man, Cybern. pt. B*, vol. 29, no. 2, pp. 226-236, 1999.
- [6] L. A. Zadeh, "Fuzzy algorithm," *Information Control*, vol. 12, pp. 94-102, 1968.
- [7] M. Sugeno and K. Murakami, "An experimental study of fuzzy parking control using a model car," *Industrial Applications of Fuzzy Control*, pp. 125-138, 1985.
- [8] C. V. Altrock et al., "Advanced fuzzy logic control technologies in

TABLE IV
CENTER AND WIDTH OF MFs SHOWN IN FIG. 5

MF	Center	Width
$A_1^{e_x}$	-2.06 (m)	2.52 (m)
$A_2^{e_x}$	-0.02 (m)	1.50 (m)
$A_3^{e_x}$	2.68 (m)	2.02 (m)
$A_1^{e_\phi}$	-124.31 (deg)	116.37 (deg)
$A_2^{e_\phi}$	58.96 (deg)	61.81 (deg)
$A_3^{e_\phi}$	127.13 (deg)	82.45 (deg)

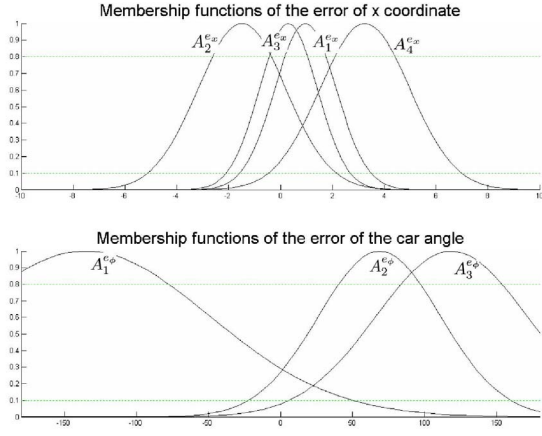


Fig. 4. Distribution of MFs of the first fuzzy logic controller of which controlled trajectory is shown in Fig. 3.

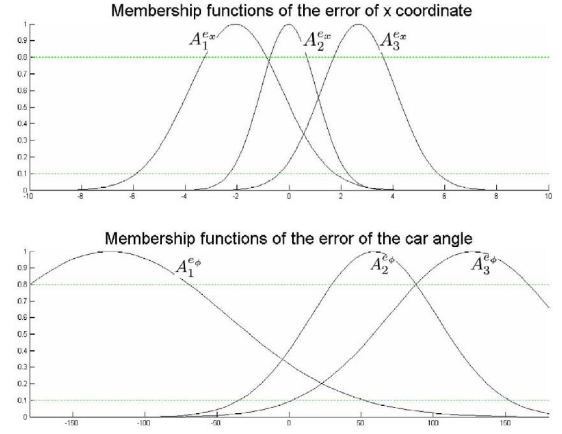


Fig. 5. Distribution of MFs of the last fuzzy logic controller of which controlled trajectory is shown in Fig. 3.

ACKNOWLEDGMENT

This paper was partially supported by ITRC-IRRC (Intelligent Robot Research Center), and also by a grant of Institute of Information Technology Assessment, Republic of Korea.

- automotive applications," *Proc. 1st IEEE Int. Conf. Fuzzy Syst.*, pp. 835-842, 1992.
- [9] F. Klawonn and R. Kruse, "Constructing a fuzzy controller from data," *Fuzzy Sets Syst.*, vol. 85, pp. 177-193, 1997.
- [10] O. Cordón, F. Herrera, F. Gomide, F. Hoffmann, and L. Magdalena, "Ten years of genetic fuzzy systems: current framework and new trends," *IFSA World Congress and 20th NAFIPS International Conference*, vol. 3, pp. 1241-1246, 2001.
- [11] P. Thrift, "Fuzzy logic synthesis with genetic algorithms," *Proc. 4th Int. Conf. Genetic Algorithms (ICGA)*, pp. 509-513, 1991.
- [12] C. L. Karr, "Adaptive control with fuzzy logic and genetic algorithms," *Fuzzy Sets, Neural Networks, and Soft Computing*, 1993.
- [13] C. L. Karr and E. J. Gentry, "Fuzzy control of pH using genetic algorithms," *IEEE Trans. Fuzzy Syst.*, vol. 1, pp. 46-53, 1993.
- [14] W. R. Hwang and W. E. Thompson, "Design of intelligent fuzzy logic controllers using genetic algorithms," *Proc. IEEE Int. Conf. Fuzzy Syst.*, pp. 1383-1388, 1994.
- [15] M.-S. Kim, C.-H. Kim, and J.-J. Lee, "Evolving structure and parameter of fuzzy models with interpretable membership functions," *Journal of Intelligent and Fuzzy Systems*, vol. 16, No. 2, pp. 95-105, 2005.
- [16] M.-S. Kim, C.-H. Kim, and J.-J. Lee, "Evolving compact and interpretable Takagi-Sugeno fuzzy models with a new encoding scheme," *IEEE Trans. Syst., man, Cyber., pt. B, to appear*.
- [17] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach," *IEEE Trans. Evol. Comp.*, vol. 3, no. 4, pp. 257-271, 1999.
- [18] K. Deb., *Multi-Objective Optimization using Evolutionary Algorithms*. Chichester, England: John Wiley & Sons, 2001.
- [19] J.-Y. Lee, M.-S. Kim, and J.-J. Lee, "Multi-objective walking trajectories generation for a biped robot," *IEEE/RSJ Int'l Conf. Intell. Robots and Syst.*, vol. 4, pp. 3853-3858, 2004.