Proceedings of the 1998 IEEE/RSJ
Intl. Conference on Intelligent Robots and Systems
Victoria, B.C., Canada • October 1998

Adaptive Fuzzy Controller Design for Trajectory Tracking of a 2 D.O.F. Wheeled Mobile Robot using Genetic Algorithm

Sung-Hoe Kim*, Chong-Kug Park*

* Department of Electronic Engineering,
Kyung Hee University, Korea
E-mail : ckpark@nms.kyunghee.ac.kr

Fumio Harashima**

** Institute of Industrial Science,
University of Tokyo, Japan
E-mail : f.harashima@ieee.org

Abstract

We dealt with the design of a WMR's fuzzy controller for trajectory tracking. The fuzzy controller uses a genetic algorithm to optimize membership function and reorganize the term set, and fuzzy rule. A similarity measure is taken to change the term set and rule correction is performed based on the changed term set. We propose the modified algorithm to take care of fine tuning. The searching region is gradually cut down while the modified GA is performed.

1 Introduction

Conventional control strategies have been developed using accurate mathematical models of control objects and their quantitative parameter values. However, they have some limitations on operating in real time on account of the nonlinear characteristics of the control object and the uncertainty of mathematical model. To solve such problems, many techniques have been researched to control the system without an accurate mathematical model or parameters. Intelligent control techniques such as fuzzy logic, neural networks and genetic algorithms were introduced and in recent work such techniques tend to be combined with each other to yield more efficiency and compensation [1][2][3]. A fuzzy controller makes a rule by language variables using expert knowledge and experience, then, performs system control by inference from these rules based on probability. But it can hardly be convincing if optimal setting up is made for the fuzzy system, and it has another problem that steady state error still exists though it has robustness. One of the methods which avoids such defects is to combine the neural network with a fuzzy system ; another one is to combine a genetic algorithm with a fuzzy system. The neural network has a distributively represented characteristic and learning ability. The fuzzy system has if-then type thinking and inference structure. The genetic algorithm searched approximate global solutions based on natural selection and genetics [4][5][6].

In this paper, optimization of the membership function for input and output variables, choice of appropriate term set for output variable and a self- tuning algorithm of rules in a fuzzy controller are proposed. All of them are performed by conventional GA and modified GA. To prove the effectiveness of the proposed algorithm, a computer simulation is performed for a 2-D.O.F. WMR which has many difficulties because of nonholonomic constraints. The results are shown below [7][8][9].

2 Basic Structure of Fuzzy Controller

The general configuration of the fuzzy controller, which is divided into four main parts, is shown in Fig. 1. The first part is fuzzification which normalizes input as the universe of discourse and transforms it into appropriate language parameters. The second one is the inference engine, which performs decision making by fuzzy logic and individual or combined inference as in a human being's brain activity. The third one is the fuzzy rule base which contains control rules, and the fourth one is defuzzification which transforms fuzzy output into plant input.

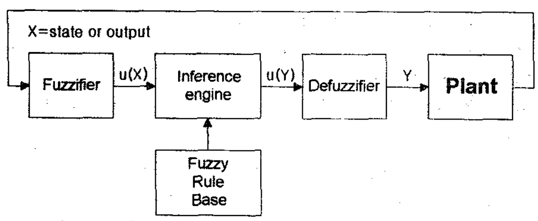


Fig. 1. The general structure of fuzzy controller

Generally, a fuzzy set F is represented as $F = \{(u, \mu_F(u)) | u \in U\}$ by membership function $\mu_F(u)$. U is the universe of discourse. The language variable X in the fuzzy controller is characterized by term set $T(x) = \{T_x^1, T_x^2, T_x^3 \cdots, T_x^k\}$ and membership function $\mu(x) = \{\mu_x^1, \mu_x^2, \mu_x^3 \cdots, \mu_x^k\}$. Input state vector X and output vector Y can be defined as Eq. (1).

$$X = \{(x_i, U_i, \{T_{x_1}^1, T_{x_2}^2, T_{x_3}^3, \cdots, T_{x_i}^k\}, \{\mu_{x_1}^1, \mu_{x_2}^2, \mu_{x_3}^3, \cdots, \mu_{x_i}^k\}) | i = 1..n\}$$
$$Y = \{(y_i, U_i, \{T_{y_1}^1, T_{y_2}^2, T_{y_3}^3, \cdots, T_{y_i}^l\}, \{\mu_{y_1}^1, \mu_{y_2}^2, \mu_{y_3}^3, \cdots, \mu_{y_i}^l\}) | i = 1..n\} \tag{1}$$

The ith fuzzy rule of rule set $R = \{R_{MIMO}^1, R_{MIMO}^2, \cdots, R_{MIMO}^n\}$ for MIMO system is defined as Eq. (2).

$$R_{MIMO}^i: IF\ x_i\ is T_{x_1}\ and \cdots and\ x_r\ is T_{x_r}, \\ THEN\ y_i\ is T_{y_1}\ and\ y_2\ is T_{y_2}\ and \cdots and\ y_s\ is T_{y_s} \tag{2}$$

Fuzzy output can be derived by implication rules such as max-min, max drastic and Goguen fuzzy logic. However, max-min and max product are mainly used because of its simplification of calculation and effectiveness. The defuzzification transforms fuzzy inference values into control input of plant. There are many defuzzifications such as center of area, center of sum, middle of maxima and max criterion, etc.. But, the center of area method is widely used. This method is described in Eq. (3).

$$y = \frac{\sum_J \bar{\mu}_y^J(\omega_J)\omega_J}{\sum_J \bar{\mu}_y^J(\omega_J)} \tag{3}$$

Where, the ω_J is support value and $\mu_y^J(\omega_i)$ is the maximum value of the membership function.

In this paper, the triangle membership function is applied and it is defined as in Fig. 2.

$$\mu^J(x_i) = 1 + \frac{x_i - b}{b - a} : a \le x_i < b \\ 1 + \frac{b - x_i}{c - b} : b \le x_i \le c \\ 0 \quad : elsewhere \tag{4}$$

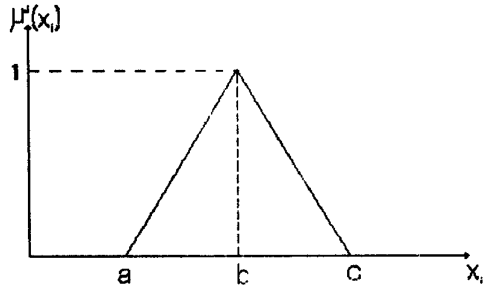


Fig. 2. Triangular fuzzy set

The type of membership function(width and center), choice of term set and structure of inference rule are main components, but the optimal choice for them still remains as a difficult problem. In this paper, we introduce a genetic algorithm. It has searching ability for a global optimal solution. The algorithm is used to optimize the membership functions for input and output variables. We can correct term set and self-tune fuzzy rule based on optimized membership functions. In the proposed algorithm, optimization of membership function in input variable is processed. after all components related with output variables are adjusted.

3 Reorganization of term set and inference

The genetic algorithm came from imitating the evolutionary process of nature. It is an algorithm which searches possible solutions of a problem with a representation of some predetermined material structure. The chromosome, gene, is represented as a string by integer, real number and binary number. It evolves by crossover, mutation and reproduction based on natural selection and genetics. Fitness function is used as a reference when an initial solution evolves into an optimum one and disappearance of a string in population of next generation is determined according to the results of computation of fitness function. Reproduction performs selection. While selection is under way, a string which has large fitness function value has high selection probability. Crossovers exchange genetic information by taking an arbitrary crossing point and interchanging literal arrays with each other. Mutations make it possible to get again latent and available genetic information, and prevents convergence of local minimum. The general flow chart of the genetic algorithm mentioned above is shown in Fig. 3 [11].

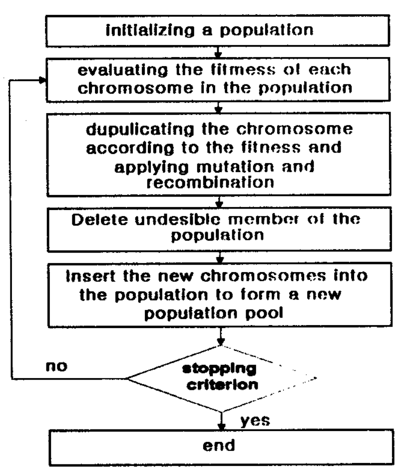


Fig. 3. General flowchart of genetic algorithm

By the genetic algorithm mentioned above, chromosome structure is set up to get an optimal fuzzy membership function for width and center of membership function as shown in Fig. 4.

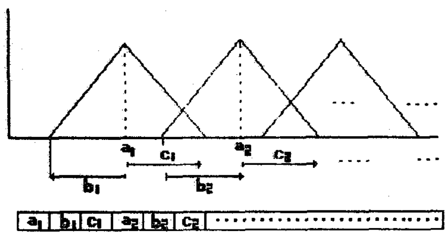


Fig. 4. The structure of chromosomes for triangular membership function

The strings assigned at input and output variables of a fuzzy controller approach to the optimum solution by repeated reproduction process, crossover and mutation. They will stop when they pass over the maximum number of generation or have a satisfied result. The term set and inference are reorganized when optimal membership function of output variable is taken. To execute such a process, the similarity measure of a fuzzy set is used. The similarity measure of fuzzy set E(A, B) is an index which shows the similarity between two fuzzy sets and it can be defined as Eq. (5) for arbitrary fuzzy sets A and B as shown in Fig. 5.[10].

$$E(A,B)=\frac{M(A\cap B)}{M(A\cup B)}=\frac{M(A\cap B)}{\sigma_1\sqrt{\pi}+\sigma_2\sqrt{\pi}-M(A\cap B)} \quad (5)$$

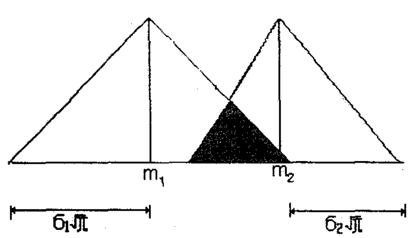


Fig. 5. Two isosceles triangular membership function

If the similarity index is less than the set up threshold, two membership functions are regarded as not having similarity with each other and a new membership function should be added up to the existing functions. When this new membership function is added, a new relationship between output of antecedent and input of consequence should be built up. An output of antecedent which have higher value than the critical value, (β), belongs to a new added membership function. It means that the relationship between output of antecedent and input of consequent is not set up appropriately. The flowchart of the overall procedure is shown in Fig. 6. O(i) is a firing value of ith rule and $f(a_i,b_i)$ is a membership function where a_i and b_i are a center and a width, respectively. And a_{iu} and b_{iu} are optimal values of a_i and b_i.

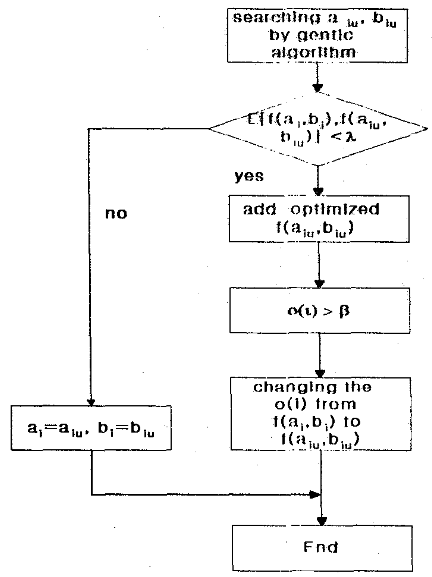


Fig. 6. Flowchart for overall procedure

4 Fine tuning by modified genetic algorithm

The fuzzy system that has its structure reorganized by genetic algorithm goes through evolutionary process again through the modified genetic algorithm. The membership functions obtained through previous procedure are represented as a chromosome. This chromosome can build up a population by mutation processing within a bounded region that converges gradually to zero while the mutation is repeated. After the crossover that is performed for new group including the source chromosome and the population that came from mutation, a new optimal chromosome is selected. That is shown in Fig. 7.

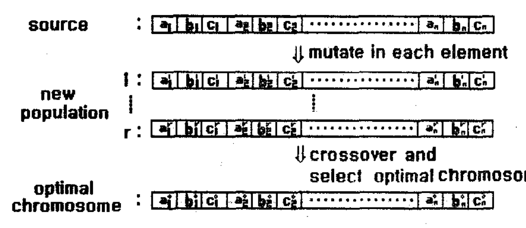


Fig . 7. Modified searching algorithm

As shown in Fig. 7, membership functions become a source of genetic information during the mutation process. This mutation builds up a population which is composed of chromosomes of r each. Two chromosomes are selected for crossover and one of the chromosome elements is selected for crossover. Selected elements are exchanged with each other and each element has equivalent selection probability in contrast to chromosome selection. The chromosome which has the highest fitness is selected in a group. This group is composed of the source chromosome and the populations which came from mutation and crossover. If this chromosome is not chosen in the population generated by crossover. then. the crossover repeats again. This procedure will be repeated continuously till it arrives at a set number of times. When it arrives at a goal without finding optimal chromosome, the optimal chromosome will be selected from overall chromosomes. By such a process, a detailed search can be achieved by cutting down the searching region generation by generation. Hence, an initial approximate solution will converge to optimal point step by step. An overall block diagram is shown in Fig. 8.

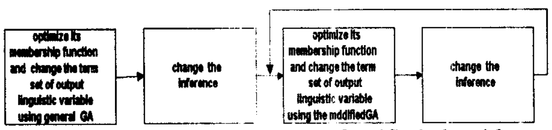


Fig. 8. Overall block diagram of unified algorithm.

5 Computer simulation and results

The computer simulation for the proposed algorithm is processed for 2-D.O.F. WMR shown in Fig. 9, 10[12].

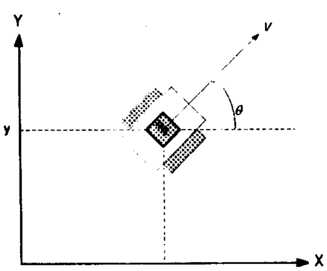
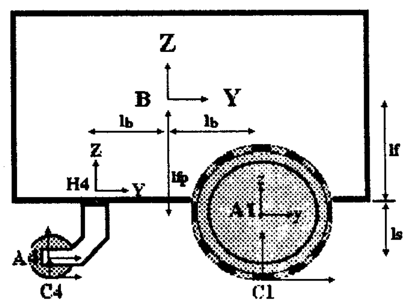


Fig. 9. Coordinate assignment for WMR's position and orientation.

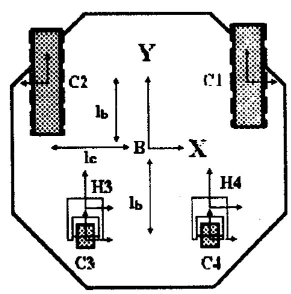
The controller design of WMR has many difficulties on account of nonholonomic constraints. P=[x, y, θ] in Fig. 9 is WMR's actual position and rotation angle on WMR's reference coordinate, X. Y. The dynamic equation of WMR used in computer simulation is Eq. (6) and the parameters are shown in Table 1.

Table 1. Parameters of WMR

Name	Value	units	Meaning
l_{Bh}	0.3	m	Body height
l_{Lh}	0.3	m	Load height
l_a	0.26	m	Half body width
l_b	0.26	m	Half body length
l_c	0.26	m	Caster y-displacement
l_d	0.15	m	Wheel z-displacement
l_e	0.26	m	Caster x-displacement
l_f	0.15	m	Caster z-displacement
l_g	0.02	m	Caster steering link length
R	0.11	m	Wheel radius displacement
r	0.03	m	Caster radius displacement
m_B	90.72	kg	Body mass



(a)



(b)

Fig. 10. Configuration of 2-D.O.F. WMR
(a) Side view, (b) Top view

$$N \cdot \tau_a + F \cdot f_e + M \cdot a_B = C_r \qquad (6)$$

Where, τ_a and f_e are driving torque and coulomb friction. a_B is body acceleration and C_r is Coliories force and centrifugal force.

The overall block diagram is shown in Fig. 11. Inverse dynamic input a_t to generate deriving torque for WMR is derived from reference acceleration $\ddot{P}_r$ and error compensation value $G_f\left(G_f=\left[x_f, y_f, \theta_f\right]^T\right)$. In this computer simulation, error $(e=P_r-P)$ and error deviation ($\Delta e=e_n-e_{n-1}$) are used as an input parameter and five initial membership functions are chosen as each input and output variable.

We choose the population size, N, as 30 and the crossover probability, P_c, as 0.8. The mutation rate is chosen as 0.001. Choice of membership function for control variable, x, y, θ, and tuning of rule are carried out individually.

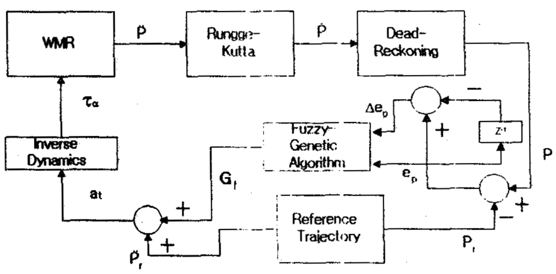


Fig. 11. Overall block diagram of WMR controller.

As shown in Fig. 11, WMR is accelerated by torque generated by inverse dynamic. Velocity($\dot{P}$) of body can be derived from acceleration($\ddot{P}$) through the Rungge-Kutta method. Position(P) also can be derived from velocity($\dot{P}$) through the dead-reckoning process. Computer simulation results are shown in Fig. 12, 13, 14, and 15, respectively. Reference position and actual position of WMR in x-y direction are shown in Fig. 12 and 14, Rotation angle is shown in Fig. 12 and 14. Total driving time is 13 seconds including 0.5 seconds acceleration and deceleration time for each. In Fig. 12 and 13, the initial position of body is set up as P(0, 0, 0°) and it's final position in desired trajectory is set up as P(4.65, 3.8, -90°). In Fig. 14. and 15., the initial position of the body is set up as P(0, 0, 0°) and it's final position in the desired trajectory is set up as P(4.65, 2.8, -90°).
As shown in computer simulation results, the actual trajectory of WMR is asymptotically converged to the reference trajectory and orientation is also asymptotically converged to reference orientation. Therefore, we can prove the feasibility of our proposed algorithm through these simulation results.

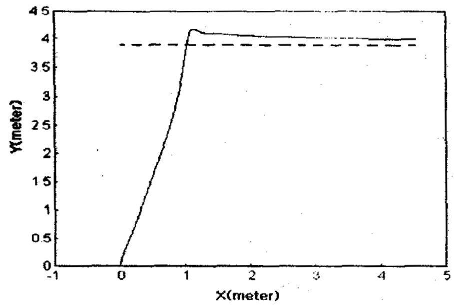


Fig. 12. Real trajectory of WMR for X, Y (X_d=4.65, Y_d=3.8)

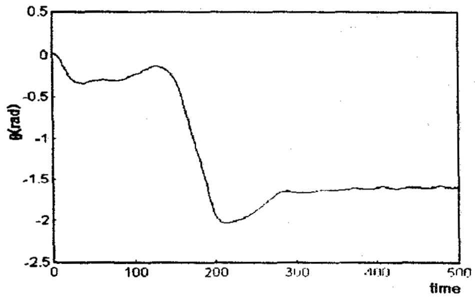


Fig. 13. Rotation angle of WMR (θ_d=-1.57)

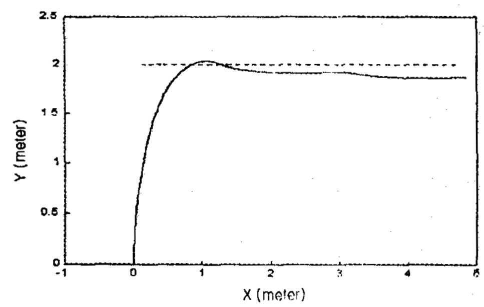


Fig. 14. Real trajectory of WMR for X, Y (X_d=4.65, Y_d=2.0)

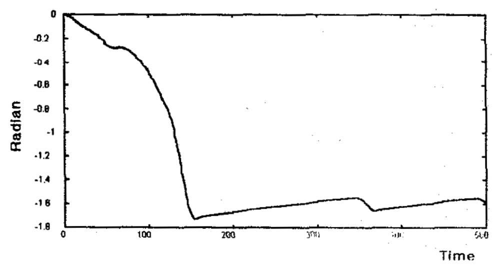


Fig. 15. Rotation angle of WMR (θ_d=-1.57)

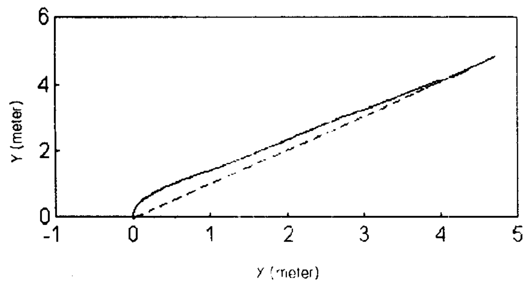


Fig. 16. Real trajectory of WMR for X, Y
(X_d=4.65, Y_d=4.65 θ_d=0.75)

6 Conclusion

In this paper, an optimization of membership function for input and output variables and self-tuning of fuzzy rule are proposed using a genetic algorithm. A modified genetic algorithm is proposed for fine tuning. The proposed algorithm is applied to a 2 D.O.F. WMR, then the feasibility of it is proved by computer simulation. However, steady state error and unexpected error which appears when the driving direction of WMR is changed drastically remains as problems for further study. And fast and effective convergence to a reference trajectory is also suggested as a further research project. Moreover, robustness in the initial setting up should be carefully considered because the system is affected by initial membership function and rule.

Reference

[1] George J. Klir and Bo Yuan. Fuzzy Sets and Fuzzy Logic Theory and Applications Prentice-Hall, 1995.

[2] D. Driankov and H. Hellendoorn. and M. Reinfrank, An Introduction To Fuzzy Control, Springer, New York, 1996.

[3] Madan M. Gupta and Naresh K.Sinha. Intelligent Control Systems, IEEE Press. 1996.

[4] J-S. R. JANG and T.-C. SUN. Neuro-Fuzzy and Softcomputing, Prentice Hall. 1997.

[5] Gregory V. TAN and Xiheng HU. On Design Fuzzy Controller using Genetic Algorithm, IEEE Int'l Conf. Fuzzy System, 1996.

[6] Chin-chih Hsu, Shin-ichi Yamada, Hideji Fujikawa, Koichiro Shida, A Multi-Operator Self-Tuning Genetic Algorithm for Fuzzy Control Rule optimization,IEEE,1996

[7] Bloch and N. H. McClamroch. "Control of mechanical systems with classical nonholonc constraints." Proceedings of the 28th IEEE conference on Decision and Control. pp. 201-205.1989.

[8] Bloch and N. H. McClamroch. and M.rey-hanoglu, "Controllability and stabilizability of nonholonomic control system." Proceedings of the 29th IEEE conference on Decision and Control. pp. 1312-1314, 1990.

[9] IlyaKolmanovsky and N. Harris McClamroch, "Developments in nonholonomic control problem."Journal of IEEE Control System. pp. 20-36, 1995.

[10] Chin-Teng Lin and C.S. George Lee. Neural Fuzzy Systems, Prentice Hall P T R. 1995.

[11] Zbigniew Michalewwicz. Genetic Algorithms + Data Structures = Evolution Programs, Springer-Verlag.

[12] Jong-Woo Moon and Chong-Kug Park. "Modeling and Path-tracking of Four Wheeled Mobile Robot with 2 D.O.F having the Limited Drive-Torques" Jurnal of The Korea Institute of Telematics and Electronics, vol 33-B. 1996.