The Third AFSS(1998), 607-612

A Design of Controller for 4-Wheeled 2-D.O.F. Mobile Robot Using Fuzzy-Genetic algorithms

Sangwon Kim[a] , Sunghoe Kim[a] , Sunho Cho[a] , Chongkug Park[a]

a. Dept. of Electronic Engineering, Kyung Hee University
Yongin, Kyunggi, 449-701 Korea
Tel:+82-201-2523 Fax:+82-203-4968 E-mail:kswfuzzy@hitel.net

Abstract

In this paper, a controller using fuzzy-genetic algorithms is proposed for path-tracking of WMR. A fuzzy controller is implemented so as to adjust appropriate crossover rate and mutation rate. A genetic algorithms is also implemented to have adaptive adjustment of control gain during optimizing process. To check effectiveness of this algorithms, computer simulation is applied.

1. Introduction

In recent, manipulator plays a important role in factory automation. However, manipulator is fixed in working space, it inevitably has some imitation of operational. To surmount it, a wheeled-mobile robot (WMR) was introduced. But, conventional control algorithms is applied for WMR driving, some difficulties are appealed. Moreover, many trial and errors are also faced and it have to be carried out to find pertinent gain. The fuzzy-genetic algorithms could be a good solution for improving such defectiveness of conventional control algorithms.. The main role of fuzzy-genetic algorithms is to adjust gains of control rule in order to follow reference trajectory well without making any error.

2. The Structure of WMR and Position Error

2.1 Coordinate Assignment of WMR

In Figure 1, the coordinate assignment of 4-wheeled 2-D.O.F. mobile robot is shown. Two front wheels of body are driving wheels actuated by independent motors and another two rear wheels are casters for stabilizing of WMR. Direction changing is achieved by velocity difference of two driving wheels.

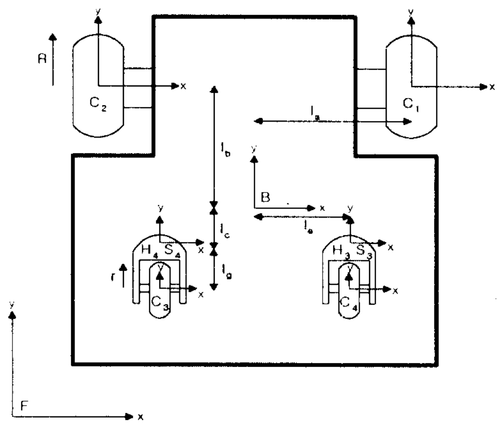

Fig. 1 The coordinate assignment for 4-wheeled 2-D.O.F. WMR..

The location of WMR is represented by $X = (x, y, \theta)$. The (x, y) is the control point and θ is the angle for the reference X axis.

WMR moves on a planar surface and there is no slippage at point-of-contact between wheel and surface.

2.2 Error Posture

The position error $P_e = (x_e, y_e, \theta_e)$ between reference position P_r and current position P_c is described as Eq. 1(Fig. 2).

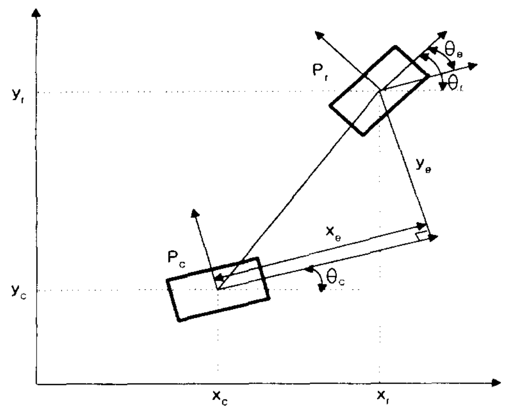

Fig. 2 Position error

$$P_e=\begin{bmatrix} x_e \\ y_e \\ \theta_e \end{bmatrix} = T(P_r-P_c)=\begin{bmatrix} \cos\theta_c & \sin\theta_c & 0 \\ -\sin\theta_c & \cos\theta_c & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} x_r-x_c \\ y_r-y_c \\ \theta_r-\theta_c \end{bmatrix} \quad (1)$$

where, T is the transformation matrix of position error in reference coordinate.

2.3 Control Rule

Each element, P_e, P_r, P_c, in equation (1) is time function. Hence, it can be differentiated and arranged. Eq. (2) can be described as a reference input $q=(v_r, w_r)^T$ and an error compensation input $q=(v(p_e, q_r), w(p_e, q_r))^T$.

$$\dot{P}_e=f(t,p_e)=\begin{bmatrix} w(p_e,q_r)y_e - v(p_e,q_r) + v_r\cos\theta_e \\ -w(p_e,q_r)x_e + v_r\sin\theta_e \\ w_r - w(p_e,q_r) \end{bmatrix} \quad (2)$$

The Lyapunov function in Eq. (3) is introduced to determine input $q=(v(p_e, q_r), w(p_e, q_r))^T$, which makes system in Eq.(2) asymptotically stable.

$$V=\frac{1}{2}(k_1x_e^2+k_2y_e^2)+(1-\cos\theta) \quad (3)$$

where, k_1, k_2 are positive constants. The control rule should satisfy the condition in Eq. (4) to be stable.

$$V \geq 0 \quad \text{and} \quad \dot{V} < 0 \quad (4)$$

The control rule satisfied the condition in Eq. (4) is given as Eq. (5).

$$q = \begin{bmatrix} v(p_e, q_r) \\ \omega(p_e, q_r) \end{bmatrix} = \begin{bmatrix} v_r\cos\theta_e + k_1x_e \\ \omega_r + v_r(k_2y_e + k_3\cos\theta_e) \end{bmatrix} \quad (5)$$

where, k_1, k_2, k_3 are positive constants.

3. Controller Organization

3.1 Fuzzy Controller

Input variables of fuzzy controller are decide by position error and its change. The error is come from difference between reference position and current position in every sampling time. The output values of fuzzy controller decide crossover rate(Cr) and mutation rate(Mr) of genetic algorithms.

The input value of fuzzy controller is quantified as 5 levels.

$$P_{ef} = \{-2,\ -1,\ 0,\ 1,\ 2\}$$

$$\Delta P_{ef} = \{-2,\ -1,\ 0,\ 1,\ 2\}$$

Table 1 Nonlinear quantization level

P_{ej}	Range of Set	ΔP_{ej}	Range of Set
-2	$X \le -1.5$	-2	$Y \le -1.75$
-1	$-1.5 < X \le -0.12$	-1	$-1.75 < Y \le -0.3$
0	$-0.12 < X \le 0.12$	0	$-0.3 < Y \le 0.3$
1	$0.12 < X \le 1.5$	1	$0.3 < Y \le 1.75$
2	$1.5 < X$	2	$1.75 < Y$

P_{ef}, ΔP_{ej} are quantization levels for position error and its change. X, Y are input variables.

Fuzzy rule is designed as 25 rules for crossover rate and mutation rate.

Table 2 Fuzzy rule

ΔP_{ej} \ P_{ej}	NB	NS	ZO	PS	PB
NB	NB	NS	NS	NS	ZO
NS	NS	NS	NS	ZO	PS
ZO	NS	NS	PS	PS	PS
PS	NS	ZO	PS	PS	PB
PB	ZO	PS	PS	PB	PB

where, NB-Negative Big, NS-Negative Small, ZO-Zero, PS-Positive Small, PB-Positive Big

When position error and its change are increased, genetic algorithms is found as global minimum by increased crossover and mutation probability. However, if the position error and its change will decrease, then genetic algorithms will have optimal solution quickly by decreased crossover and mutation probability.

3.2 The Application of Genetic algorithms

Genetic algorithms is introduced for optimization problem, which is imitating the rule of survival of the fittest. This algorithms is probability optimization method.

The selection, crossover and mutation operator of genetic algorithms will increase average fitness of solution group.

A continuous operator in continuous space for genetic algorithms is used in this paper. Selection is done by adopting standard roulette wheel method. The crossover adopt simple crossover method, which do not consider the order of crossover. And it uses the output of fuzzy controller.. Mutation uses flip function and output of fuzzy controller.

The operating principle of genetic algorithms is as follows. We define parameters to find optimization solution in Eq. (6).

$$P = [K_{11}, K_{21}, \cdots, K_{1j}, K_{2j}, \cdots, K_{1n}, K_{2n}]^T \in \Xi_p \tag{6}$$

where, K is parameter, $j \in \{1, 2, \cdots, n\}$, $\Xi_p = [P_{\min}, P_{\max}]$ are beforehand settled space for real parameters vector.

The measure of quantification is fitness function (FF) for optimization in search space. FF is defined as Eq.(7).

$$FF(P) = \frac{1}{((P_e - P)^2 + 1)} + \frac{1}{((\Delta P_e - P)^2 + 1)} \tag{7}$$

P_e, ΔP_e are position error and change of position error between reference position and current position.

We are able to consider optimization problem that maximize fitness function from parameter vector P.

$$\underset{P \in \Xi_j}{MAX} \quad FF(P) \tag{8}$$

Genetic algorithms can be summarized to find out parameter vector P of optimization solution.

[step 1] Initial parameter construction

Random selected parent parameter vector is constructed $P_0^1, P_0^2, \cdots. P_0^l, \cdots, P_0^N$ at initial time $t=0$ in search space.

[step 2] Selection

It selects parameter vector $P'^1_k, P'^2_k, \cdots, P'^l_k, \cdots, P'^N_k$ when each parents vector $P_k^1, P_k^2, \cdots, P_k^l, \cdots, P_k^N$ is given in random time t.

[step 3] Crossover

It selects two vectors from $P'^1_k, P'^2_k, \cdots, P'^l_k, \cdots, P'^N_k$. One parameter vector P''^l_k is made by crossover using crossover rate and roulette wheel method.. This strategy can speed up information exchange between the parent samples.

[step 4] Mutation

The next generation P_{k+1}^l is made of parameter vector P''^l_k by mutation. Mutation operation is determined by flip function and mutation rate. The mutation bit is randomly performed. This strategy can take global minimum without local minimum.

[step 5] Fitness is calculated by fitness function $FF(P_{k+1})$.

It executes repeatedly [step 2]~[step 5] until convergence.

4. Position Estimation

The position of WMR is estimated using dead-reckoning algorithms. Dead-reckoning algorithms can decide present position of WMR in real time by adding up increased position data to the previous position data in each sampling period. If the increase of WMR moving distance and its variation of direction angle are ΔD_k and $\Delta\theta_k$, respectively, in each sampling period which is from t=(n-1)T to t=nT, we can arrive at below conclusion.

$$\Delta Dk = \frac{R\Delta\omega_{1k} + R\Delta\omega_{2k}}{2} T \quad (9)$$

$$\Delta\theta_k = \frac{r\Delta\omega_{1k} - r\Delta\omega_{2k}}{2l_b} T \quad (10)$$

WMR position is described as Eq. (11), (12) and (13) by Eq. (9), (10) at time t=nT.

$$x[k+1] = x[k] + \Delta D[k]\cos(\theta[k] + \Delta\theta[k]/2) \quad (11)$$

$$y[k+1] = y[k] + \Delta D[k]\sin(\theta[k] + \Delta\theta[k]/2) \quad (12)$$

$$\theta[k+1] = \theta[k] + \Delta\theta[k] \quad (13)$$

5. A Design of Controller

Design of controller is inherently aimed to converge quickly reference trajectory with minimum path error.

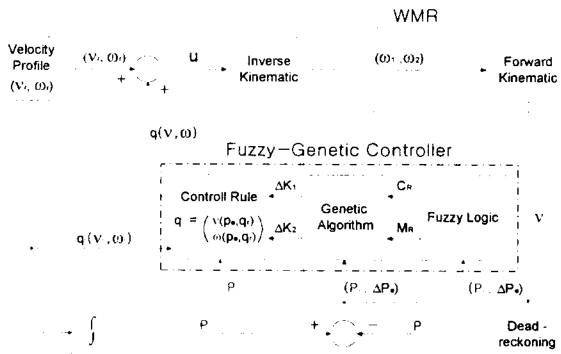

Fig. 3 Schematic Diagram of Controller

Input is position error P_e and change of position error ΔP_e between reference position and actual position through dead-reckoning algorithms.
If crossover rate and mutation rate become high, high fitness of search space can be found at initial step of evolution. However, it might reduce convergence speed after finding proper solution.. In contrast, if crossover rate and mutation rate are low, it can alleviate executing time for computing the fitness. But it might face a dangerous situation to

be fall in convergence before having optimal solution by losing the variety quickly between individuals. Fuzzy logic controller is implemented to the system to get adaptive adjustment for crossover rate and mutation rate in proportion to the position error and the change of position error while optimization is processed.

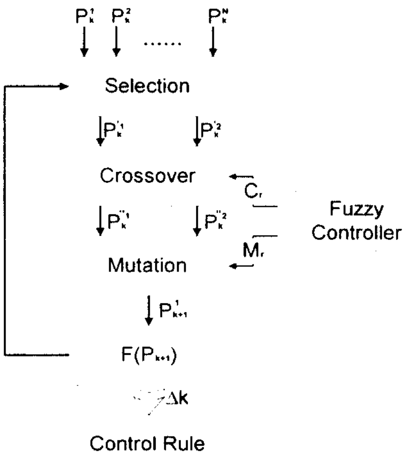

Fig. 4 The flowchart of fuzzy-genetic algorithms controller

The gain of control rule is achieved by adding increased value, which is get from output of genetic algorithms, to previous gain. It will be supplied to the system. That is, $k_1 = k_1 + \Delta k_1$, $k_2 = k_2 + \Delta k_2$, $k_3 = 2\sqrt{k_2}$.

6. Simulation

The computer simulation is processed to prove the efficiency of the algorithms and to get the result of path tracking. The reference trajectory of WMR is straight line from (0, 0, 45°) to (10, 10, 45°). The simulations are carried out at two case, which have different initial positions. The first initial position of the WMR is (0.5, 0, 0°) and the other is (0, -0.5, 0°). The overall parameters of WMR are shown in Table. 3.

Table 3 Overall Parameters of WMR

symbol	value	unit	detail
l_a	0.22	m	body width/2
l_b	0.25	m	body length/2
l_c	0.22	m	y displacement of support wheel
l_d	0.11	m	z displacement of driving wheel
l_e	0.12	m	x displacement of support wheel
l_f	0.15	m	z displacement of support wheel
R	0.05	m	the radius of driving wheel
r	0.025	m	the radius of support wheel

7. Conclusions

In this paper, fuzzy logic controller is proposed so as to modify crossover rate and mutation rate automatically while optimization is processed. The computer simulation result of control process, which is offered for fuzzy-genetic algorithms, shows remarkably improved performances.. It points out that the proposed fuzzy-genetic control algorithms can be applied to a actual system.

Reference

1. G. Campion, G. Bastin, and B. D'Andrea-Novel, "Structural Properties and Classification of Kinematic and Dynamic Models of Wheeled Mobile Robots," *IEEE Trans. on Robotics and Auto.* vol. 12, pp.47-62, 1996.
2. P. Y. Wang, G. S. Wang, Y. H. Song and A. T. John, "Fuzzy Logic Controlled Genetic algorithms", *IEEE Inter. Conf. Fuzzy System*, vol. 2, pp. 972-979, 1996.

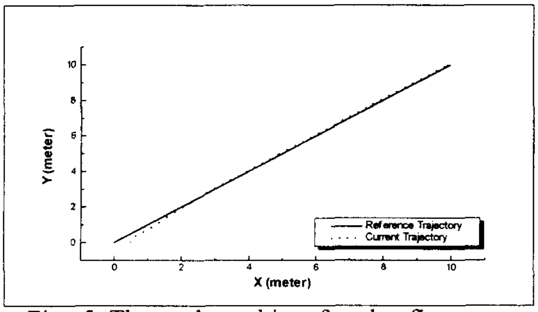

Fig. 5 The path-tracking for the first case using settled gain of the first case

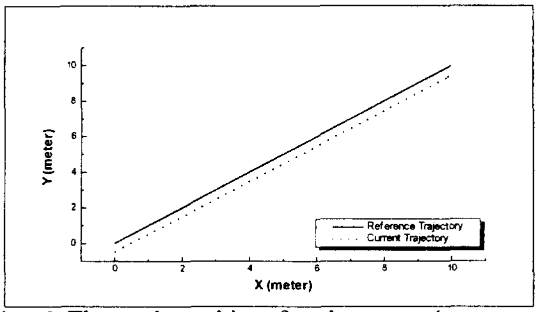

Fig. 6 The path-tracking for the second case using settled gain of the first case

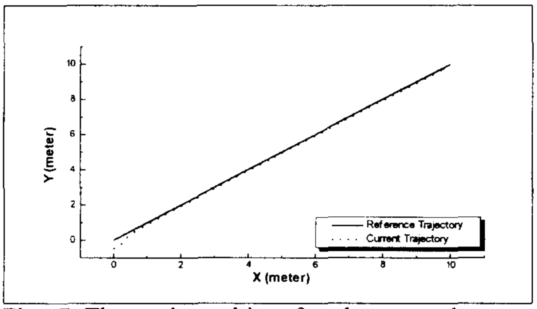

Fig. 7 The path-tracking for the second case using settled gain of the second case

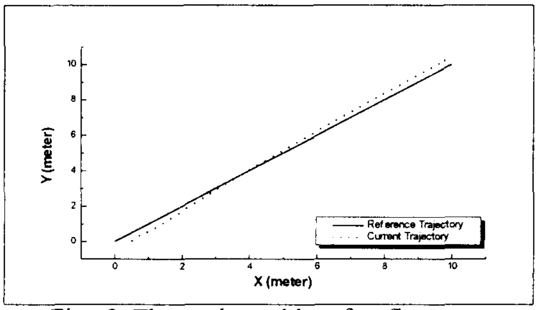

Fig. 8 The path-tracking for first case using settled gain of the second case

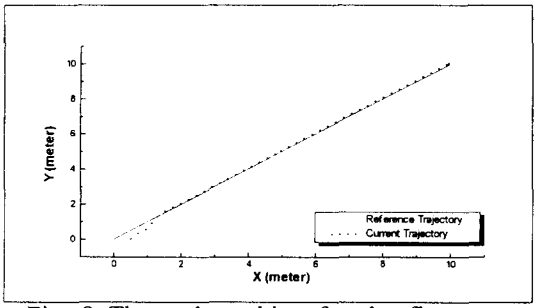

Fig. 9 The path-tracking for the first case using fuzzy-genetic algorithms controller

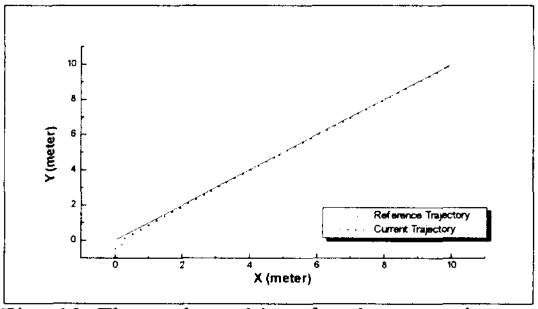

Fig. 10 The path-tracking for the second case using fuzzy-genetic algorithm controller

3. B. G. Hu, G. K. I. Mann and R. G. Gosine, "Theoretic and Genetic Design of a Three-Rule Fuzzy PI Controller," *IEEE Inter. Conf. Fuzzy System*, vol. 1, pp. 489-496, 1997.
4. Daihee Park, Abraham Kandel, Gideon Langholz, Genetic-based new fuzzy reasoning models with application to fuzzy control," System, Man, Cybemetics, 1994, pp. 34-47.
5. C. L. Karr, D. A. Stanley, "Fuzzy logic and genetic algorithms in time-varying control problem," Proc. NAFIPS-91, 1991, pp. 285-290.
6. D. L. Meredith, K. K. Kumar, C.L. Karr, "The use of genetic algorithms in the design of fuzzy logic controllers," Proc. WNNAIND 91, 1991, pp.695-702.
7. Abdollah Homaifar, Ed. McCormick,"Simultaneous design of membership function and rule sets for fuzzy controller using genetic algorithms,"IEEE Transactions on Fuzzy Systems, Vol 3, No. 2, 1995, pp. 129-139.