

## Research on a New Path Planning Method for Mobile Robot

Xiaoping Zhang, Xiaogang Ruan, Xiaoqing Zhu, Jing Huang

Institute of Artificial Intelligence and Robots, School of Electronic Information and Control Engineering  
Beijing University of Technology  
Beijing, China

zhangxiaoping369@163.com, adrxg@bjut.edu.cn, president2zhu@qq.com, aiandrobot@163.com

**Abstract**—Aim at the path planning problem of a mobile vehicle in an unknown environment, a new hybrid learning method is proposed, which is based on the fuzzy inference system and assisted by the combination of supervised learning and bionic automata, the bionic learning endows the robot the capabilities of self-learning and self-organization, the supervised learning is used to tune the input membership function of the fuzzy inference system, that avoid the expert knowledge's disadvantage of inaccurate estimation to the actual model. The simulation results show that the robot has learned the ability of obstacle avoidance and goal seeking and performs a better robustness in overcoming the local minima.

**Keywords**—hybrid learning; automata; fuzzy inference system; supervised learning; path planning

### I. INTRODUCTION

Mobile robot is a new integrated discipline in recent years, among which, path planning problem is one hot spot of the subject. Path planning aims to enable the robot to find an appropriate path from the starting point to the end point in an environment with obstacles[1]. At present, according to the different in information of the working environment, path planning is divided into global path planning of which the environment's information is totally known, local path planning of which the environment is unknown or partially unknown and the combination of both the methods [2, 3]. Since most of the actual training environment is unknown, so the research on the local path or dynamic path planning algorithm has an important and far-reaching significance.

In recent years, as mobile robot is more and more applied to different environment, local path planning becomes more and more complex, some hybrid learning strategies emerged. Based on fuzzy inference system, S. Kermiche combined artificial potential field theory with supervised learning to adjust the fuzzy controller, successfully realize the robot's obstacle avoidance and target navigation, the curve is closer to the optimal path[4], Tan and Lee applied genetic algorithm to regular fuzzy controller, also achieved good results[5, 6], meanwhile, there are many scholars combined reinforcement learning with fuzzy inference system to accomplish different navigation tasks[7-11], Meng improved reinforcement by proposing a dynamic fuzzy Q-learning, greatly improved the speed of operation and control accuracy[12, 13], in 2012, Gao based on the fuzzy inference, introduced bionics control mechanism, through continuous

interaction with the external environment enable the robot features of self-learning and adaptability [14], however, the fuzzy rule base established by expert knowledge increases the uncertainty and imprecision of the model.

Summarize the advantages of previous work, aim at solving the problems of obstacle avoidance and navigation, based on fuzzy inference system, this paper presents a new hybrid learning approach. First, build a fuzzy inference system with 8 rules based on expert knowledge, Second, introduce supervised learning and learning automata to optimize the input membership functions and the output conclusions of the fuzzy inference system respectively, the supervised learning is used to adjust the input fuzzy membership functions, after that, the functions remain unchanged, use learning automata to optimize the conclusions further more. We experiment on the designed algorithm, results show that this method not only makes the robot acquire the skills of obstacle avoidance and target searching faster, but also shows better robustness and ability to overcome the local minima.

### II. THE HYBRID LEARNING MODEL

The hybrid learning model designed in this paper includes three parts, as shown in Fig.1, the fuzzy controller contains 8 fuzzy rules established by expert knowledge and is used to control the output; the supervised learning adjusts the fuzzy inference system roughly; the learning automata optimizes the fuzzy rule conclusions further more and realizes the fine-tuning of the fuzzy inference system.

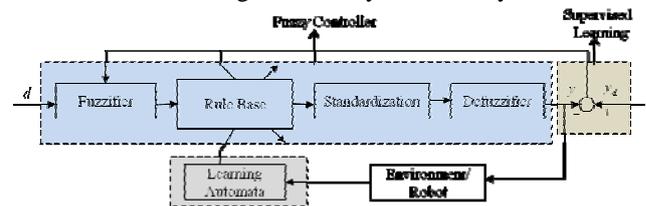


Figure 1. The hybrid learning model

In simulation, we use a simple circular symmetry robot [10], as depicted in Fig. 2, 24 ultrasonic sensors  $s_i (i = 1, \dots, 24)$  uniformly distribute on the circumference of the robot and check the distance from the robot to the obstacles  $l_i$ . In order to reduce the input dimension of the fuzzy controller, we divide the 18 sensors in front of the

robot into three groups  $SL, SF, SR$ , respectively detect the obstacle's distance  $dL, dF$  and  $dR$ , the radius of the robot is  $R_v$ , so the distance is expressed as follows:

$$\begin{cases} dL = R_v + \min(l_i) & i = 1, \dots, 6, \\ dF = R_v + \min(l_i) & i = 7, \dots, 12, \\ dR = R_v + \min(l_i) & i = 13, \dots, 18. \end{cases} \quad (1)$$

This paper uses two coordinate systems (Fig. 3): world coordinate system  $XWY$  and the robot's own coordinate system  $xOy$ . Robot moves by controlling the rotation  $\Phi$  and the speed  $v$ .

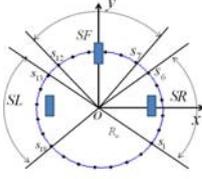


Figure 2. The robot model

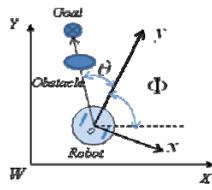


Figure 3. The system coordinate

### III. FUZZY CONTROLLER

The internal structure of the fuzzy controller consists of four layers as shown in Fig. 4.

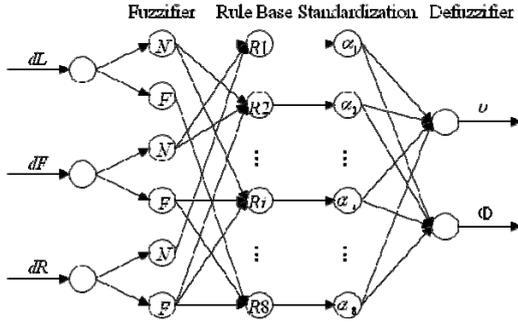


Figure 4. Fuzzy controller

#### A. Fuzzifier layer

First, fuzzifier the input  $d = (dL, dF, dR)$ , the fuzzy functions is as shown in Fig. 5,  $N$  and  $F$  represent near and far.  $x_{ri} (r = 1, \dots, 4, i = 1, 2, 3)$  can be get by the behind-supervised learning, whereby the input fuzzy function will be eq. (2) and eq. (3):

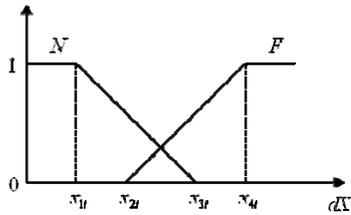


Figure 5. Input fuzzy function

$$\mu_N(d_i) = \begin{cases} 1 & d_i \leq x_{1i}, \\ (x_{3i} - d_i) / (x_{3i} - x_{1i}) & x_{1i} < d_i \leq x_{3i}, \\ 0 & d_i > x_{3i}. \end{cases} \quad (2)$$

$$\mu_F(d_i) = \begin{cases} 0 & d_i \leq x_{2i}, \\ (x_{2i} - d_i) / (x_{2i} - x_{4i}) & x_{2i} < d_i \leq x_{4i}, \\ 1 & d_i > x_{4i}. \end{cases} \quad (3)$$

#### B. Rule Base layer

For the tasks to be completed, the fuzzy rule base that established by expert knowledge will greatly reduce the search space, the expert knowledge in this paper is as follows:

(1) If the robot is close to the obstacles in all three directions, then steers to the goal and walks along the nearest obstacle;

(2) If the robot is far from the obstacles in only one direction, then moves to this direction;

(3) If the robot is far from the obstacles in two directions, moves to the direction both no obstacles and near the goal;

(4) If the robot is far from the obstacles in all three directions, then moves to the goal with its maximum speed.

$p$  is introduced to enable the robot navigate automatically:

$$p = \begin{cases} 0 & 0 \leq \theta \leq \pi, \\ 1 & -\pi \leq \theta \leq 0, \end{cases} \quad (4)$$

$\theta(-\pi, \pi)$  is the angle between the robot's direction and the target. The symbolic language of fuzzy rules' conclusions is [10, 14]:  $PB$  (Positive Big),  $PM$  (Positive Medium),  $PS$  (Positive Small),  $ZR$  (Zero),  $NS$  (Negative Small),  $NM$  (Negative Medium),  $NB$  (Negative Big). we establish  $8(2^3)$  rules as follows:

$R1$  : IF  $(dL, dF, dR)$  is  $(N, N, N)$  THEN

$$v_1 \text{ is } ZR \text{ and } \Phi_1 \text{ is } p \times NB + (1-p) \times PB = (1-2p) \times PB$$

$R2$  : IF  $(dL, dF, dR)$  is  $(N, N, F)$  THEN

$$v_2 \text{ is } ZR \text{ and } \Phi_2 \text{ is } NB$$

$R3$  : IF  $(dL, dF, dR)$  is  $(N, F, N)$  THEN

$$v_3 \text{ is } V_{\max} \text{ and } \Phi_3 \text{ is } ZR$$

$R4$  : IF  $(dL, dF, dR)$  is  $(F, N, N)$  THEN

$$v_4 \text{ is } ZR \text{ and } \Phi_4 \text{ is } PB$$

$R5$  : IF  $(dL, dF, dR)$  is  $(N, F, F)$  THEN

$$v_5 \text{ is } V_{\max} \text{ and } \Phi_5 \text{ is } p \times NS + (1-p) \times ZR = p \times NS$$

$R6$  : IF  $(dL, dF, dR)$  is  $(F, F, N)$  THEN

$$v_6 \text{ is } V_{\max} \text{ and } \Phi_6 \text{ is } p \times ZR + (1-p) \times PS = (1-p) \times PS$$

$R7$  : IF  $(dL, dF, dR)$  is  $(F, N, F)$  THEN

$$v_7 \text{ is } ZR \text{ and } \Phi_7 \text{ is } p \times NB + (1-p) \times PB = (1-2p) \times PB$$

$R8$  : IF  $(dL, dF, dR)$  is  $(F, F, F)$  THEN

$$v_8 \text{ is } V_{\max} \text{ and } \Phi_8 \text{ is } p \times NM + (1-p) \times PM = (1-2p) \times PM$$

$V_{\max}$  is the maximum speed of the robot, conclusions of the fuzzy rules can be obtained by the learning automata. This paper mainly learns the conclusions of the motion angle.

### C. Standardization layer

We assume  $\mu_j$  is the truth value of rule  $R_j$ :

$$\mu_j(dL, dF, dR) = \mu_{D_{j1}}(dL)\mu_{D_{j2}}(dF)\mu_{D_{j3}}(dR) \quad (5)$$

$D_{ji}$  is the fuzzy set of the input  $d_i = (dL, dF, dR)$  in rule  $R_j$ , so the standardization process is as follows:

$$\alpha_i = \frac{\mu_i}{\sum_{j=1}^8 \mu_j}. \quad (6)$$

### D. Defuzzifier layer

Sugeno method is used to get the output of the system:

$$\Phi = \sum_{j=1}^8 \alpha_j \Phi_j. \quad (7)$$

$$\nu = \sum_{j=1}^8 \alpha_j \nu_j. \quad (8)$$

## IV. SUPERVISED LEARNING

In this paper, supervised learning is used to adjust the input fuzzy function of the fuzzy system, the actual output is  $y$  and the desired output is  $y_d$ , then the learning process is to minimize the objective function:

$$J = \frac{1}{2} (y - y_d)^2. \quad (9)$$

According to principles of the gradient descent method, the learning process is designed as follows:

$$\begin{cases} x_{ri}(k+1) = x_{ri}(k) - \delta \nabla J(x_{ri}(k)), \\ \Phi_j(k+1) = f(\Phi_j(k) - \delta \nabla J(\Phi_j(k))), \end{cases} \quad (10)$$

$\delta > 0$  is the learning rate,  $\nabla J(x_{ri}(k))$  and  $\nabla J(\Phi_j(k))$  means the objective function' gradient at  $x_{ri}(k)$  and  $\Phi_j(k)$  respectively, and can be get by chain rule.

Since the gradient algorithm "rock" downing too fast will cause the algorithm oscillations decreases and too slowly will reduce the speed of convergence, in order to make the "rock" move with a relatively stable speed, a momentum is introduced to keep the direction of "rock" in a certain extent, so we get the learning process based on improved BP algorithm as follows:

$$\begin{cases} x_{ri}(k+1) = x_{ri}(k) + \beta \Delta x_{ri}(k-1) - \delta(1-\beta) \nabla J(x_{ri}(k)), \\ \Phi_j(k+1) = f(\Phi_j(k) + \beta \Delta \Phi_j(k-1) - \delta(1-\beta) \nabla J(\Phi_j(k))). \end{cases} \quad (11)$$

$\beta \Delta x_{ri}(k-1)$  and  $\beta \Delta \Phi_j(k-1)$  are the momentums,  $\beta \in [0, 1]$  is the momentum parameter, it always chooses 0.9.

The conclusions of the fuzzy rules in the paper is discrete, to solve this problem,  $f(x)$  is designed as follows ( $[x]$  means the greatest integer less than  $x$ ):

$$f(x) = \begin{cases} 10 \times \lfloor \frac{x}{10} \rfloor & 0 \leq x - 10 \times \lfloor \frac{x}{10} \rfloor < 2.5, \\ 10 \times \lfloor \frac{x}{10} \rfloor + 5 & 2.5 \leq x - 10 \times \lfloor \frac{x}{10} \rfloor < 7.5, \\ 10 \times \lfloor \frac{x}{10} \rfloor + 10 & 7.5 \leq x - 10 \times \lfloor \frac{x}{10} \rfloor < 10. \end{cases} \quad (12)$$

## V. LEARNING AUTOMATA

In last part the input membership function of the fuzzy system have been adjusted, in this part the conclusions ( $PB, PM, PS, ZR, NS, NB$ ) of the fuzzy controller will be fine-adjusted by learning automata, in this paper, we use a learning automaton based on skinner operant conditioning[14] (Operant Conditioning Learning Automata, OCLA).

$$OCLA = \langle S, O, Z, \varphi, \delta \rangle$$

(1)  $S = \{s_i \mid i = 1, 2, \dots, 8\}$  is the input state set;

(2)  $O = \{o_k \mid k = 1, 2, \dots, 5\}$  is the internal operation set;

(3)  $Z = \{z_m \mid m = 1, 2, \dots, 5\}$  is the output symbol set;

(4)  $\varphi_i = \varphi(s_i)$  is the state orientation function of OCLA, and is designed as follows:

$$\varphi(s_i) = \begin{cases} -1 & \text{if } \min(dL, dF, dR) < d_{\min} + V_{\max} \times \Delta T, \\ 0.05 & \text{otherwise.} \end{cases} \quad (13)$$

$d_{\min}$  is the sum of the robot's radius and the minimum distance that the sensors can detect from the obstacles,  $\Delta T$  is the interval time between two learning steps.

(5)  $\delta$  is the operant conditioning learning rate of OCLA, we assume that at time  $t$  the state of OCLA is  $s(t) = s_a \in S$ , the operation to be selected by "operant conditioning" rules is  $o(t) = o_c \in O$ , and the state at time  $t+1$  is  $s(t+1) = s_d \in S$ , based on the principle of operant conditioning, the probability  $p_{ac}$  of the rules in the operator sets  $R$  is:

$$\delta: \begin{cases} p_{ak}(t+1) = p_{ak}(t) - \xi(\bar{\varphi}_{ac}) \cdot p_{ak}(t), \forall k \neq c, \\ p_{ak}(t+1) = \max \min(p_{ak}(t+1), 0, 1), \\ p_{ac}(t+1) = 1 - \sum_{k \neq c} p_{ak}(t+1), \text{ otherwise.} \end{cases} \quad (14)$$

$\bar{\varphi}_{ac} = \varphi(s_d) - \varphi(s_a)$  is the change of the orientation function from state  $s_a$  to  $s_d$  (after operation  $o_c$ ), it is used to evaluate the quality of the operation,  $\xi(\bar{\varphi}_{ac}) = \xi(\varphi(s_d) - \varphi(s_a)) = \frac{\lambda * \bar{\varphi}_{ac}}{r}$  is a monotonically increasing function,  $r = 5$  is the total number of operation rules,  $\lambda$  is the learning rate, we select 0.3 in this paper, larger  $p_{ac}(t)$  means better  $o_c$  in state  $s_a$ .

Assume that  $S(t) = (\theta_o(t), X_o(t), Y_o(t))^T$ , then the state  $S(t+1)$  after learning can be:

$$S(t+1) = \begin{pmatrix} \theta_o(t) + \Phi(t) \\ X_o(t) + v(t)\Delta T \cos(\theta_o(t) + \Phi(t)) \\ Y_o(t) + v(t)\Delta T \sin(\theta_o(t) + \Phi(t)) \end{pmatrix} \quad (15)$$

## VI. RULES LEARNING AND RESULTS ANALYSIS

### A. Supervised learning phase

Suppose that  $R_v = 20cm$ ,  $V_{max} = 0.5m/s$ ,  $\Delta T = 0.6s$ , the range that the ultrasonic sensors can measure is  $10cm - 210cm$ , the maximum angular velocity is  $100^\circ/s$ , at beginning, the parameter of the input function is  $x_{1i} = x_{2i} = 0.6m$  which is the minimum distance that allowed between the robot and the obstacles,  $x_{3i} = x_{4i} = 2m$  is the minimum distance when the robot can run at its maximum speed. Each fuzzy rule has 5 fuzzy conclusions, the minimum value in the positive direction is  $ZR = 0$ , the maximum value is  $PB_{max} = 100^\circ/s \times \Delta T = 60^\circ$ , for the convenience of division, let  $PB_{max} = 65^\circ$ , Table I. shows the maximum value and the minimum value of each rule, for example, the maximum value of  $NB$  is  $-45^\circ$  and the minimum is  $-65^\circ$ , then the conclusions that can be selected of  $NB$  will be  $-65^\circ, -60^\circ, -55^\circ, -50^\circ, -45^\circ$ . The initial value of each fuzzy conclusion is its median value, as shown in Table II, the learning rate is  $\delta = 2$ , the convergence criteria is  $J \leq 1.52 \times 10^{-6}$  when the output error is  $0.1^\circ$ , we give the training pair  $(d^n, y_d^n)$  by expert knowledge, the parameter  $x_{ri}$  after learning is shown in Table III:

TABLE I. THE CONCLUSIONS OF THE FUZZY RULES

	NB	NS	ZR	PS	PM	PB
MIN	$-65^\circ$	$-25^\circ$	0	$5^\circ$	$25^\circ$	$45^\circ$
MAX	$-45^\circ$	$-5^\circ$	0	$25^\circ$	$45^\circ$	$65^\circ$

TABLE II. THE INITIAL CONCLUSIONS OF THE FUZZY RULES

NB	NS	ZR	PS	PM	PB
$-55^\circ$	$-15^\circ$	0	$15^\circ$	$35^\circ$	$55^\circ$

TABLE III. THE VALUE  $x_{ri}$  AFTER SUPERVISED LEARNING

i/r	1	2	3	4
1	60.1259	65.1871	200.0046	200.1264
2	60.1724	66.1620	200.0064	200.1725
3	59.8296	59.9937	193.9323	199.8297

### B. The twice learning by learning automata

After the supervised learning phase, we put the robot in actual environment and use the learning automata to train the rules' conclusions further more. In primary learning stage, we use a simple channel environment as Fig.6 to train the model[11], as regular as channel environment is, the robot's track keeps unchanged in the learning process, so we can set a stopping learning standards for it.

There are two stages in this process: first, let the robot learning in counter-clockwise (clockwise) from any point of the channel, it will be end until the robot can keep continuous collision-free movement, then comes the second stage, let the robot learning in clockwise(counter-clockwise) from any point of the channel until the robot can keep continuous collision-free movement then this process end.

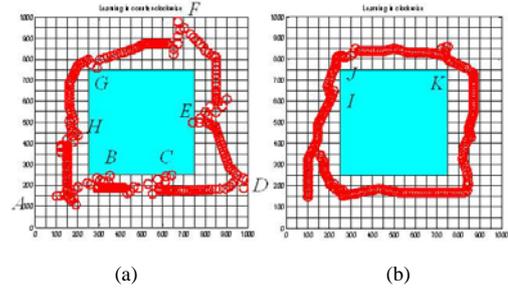


Figure 6. The channel environment

Fig. 6(a) is the result of the first circle when the robot moves in counter-clockwise, we can see the robot moves from the starting point  $A(0^\circ, 100, 150)$  to the right and takes a try at  $B$  but gets collision, then it starts to move line to the right, and takes the second try at  $C$ , also gets collision, then it dose not try again and keeps move line to the right until hits the wall at  $D$  then moves up. In the process moving up, the robot only tries one time at  $E$ , after collision it keeps moving up until get collision at  $F$ , in the later process to left and down, the robot dose not try any more, only two collisions happen, one is at the corner  $G$ , the other is at  $H$  because of little shocks. 17 collisions happen in the process and during it the effectiveness of learning can be reflected, let it go on, while the robot can move without collision, stop learning, the conclusions that get from the process can be used in the second learning stage. Fig. 6(b) is the result of the first clockwise circle based on the conclusions from the previous stage, we can see that the robot moves up from the starting point  $(\pi/2, 100, 150)$  with no wrong try, the

collisions happen only at the corner at the beginning, the curve is smooth with no collision even at the corner, there are 4 collisions this time, significantly reduces, this proves the effectiveness of the learning again.

While the robot can move without collision in both clockwise and count-clockwise, the primary stage end, the best operation of each state is the operation which with maximum probability, the conclusions without supervised learning (consuming 234 s) and the conclusions with supervised learning (consuming 192 s, the performance improved by 22% ) is shown in Table IV and Table V :

TABLE IV. CONCLUSIONS WITHOUT SUPERVISED LEARNING

R1	R2	R3	R4	R5	R6	R7	R8
PB	NB	ZR	PB	NS	PS	PB	PM
45°	-60°	0	50°	-10°	10°	55°	25°

TABLE V. CONCLUSIONS WITH SUPERVISED LEARNING

R1	R2	R3	R4	R5	R6	R7	R8
PB	NB	ZR	PB	NS	PS	PB	PM
55°	-50°	0	50°	-10°	15°	55°	25°

### C. Results and analysis

#### Experiment 1: Target navigation:

After the primary stage, we put the robot in a complex environment with many obstacles, let the robot move to the goal from different starting point. The result is shown in Fig. 7, in the two pictures, the red line is the track with supervised learning, the blue line is the track without supervised learning, the turning point means the robot chooses a new operation. In Fig. 7(a), the robot moves up and to the right respectively from (100, 150), in the process moving up, the blue line chooses twice and consumes 124 ms to get to the goal, the red line chooses only once and consumes 108 ms, the performance is improved by 15%, in the process moving to the right, the blue line chooses seven times and consumes 120 ms, the red line chooses five times and consumes 96 ms, the performance is improved by 25%. In Fig. 7(b), the robot moves up and to the right respectively from (550, 400), while moving up, the blue line get to the goal 59 ms later, and the red line 44 ms later, the performance is improved by 34%, while moving to the right, the blue line get to the goal 220 ms later, and the red line 190 ms later, the performance is improved by 16%. The results show that the robot can get to the goal from any point, what's more, with supervised learning, the operation that the robot takes reduces, so it takes less time to fulfill a task and get better performance.

#### Experiment 2: Analysis of the local minima:

We put the robot in a local minima environment as Fig. 8, Fig. 8(a) is the result of traditional artificial potential field method, it shows that the robot is trapped into a local minima, in Fig. 8(b), the red line is the result with supervised learning and the blue line is the result without supervised learning, we can found that both of them can resistance local minima, but the curve with supervised learning is more smooth, and the operation is less, the time is 30.5 ms and 33.9 ms respectively, the performance is improved by 10%, the result we get proves the effectiveness of the algorithm again.

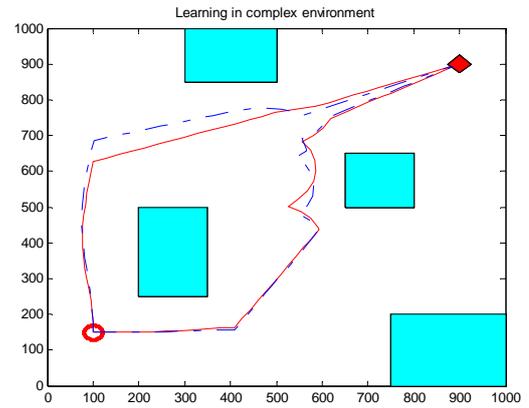


Figure 7(a). Starting point(100,150)

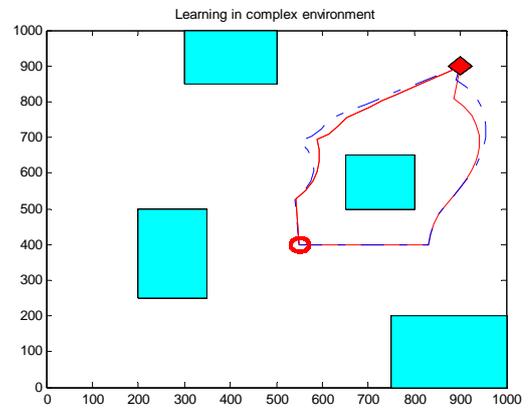


Figure 7(b). Starting point(550,400)

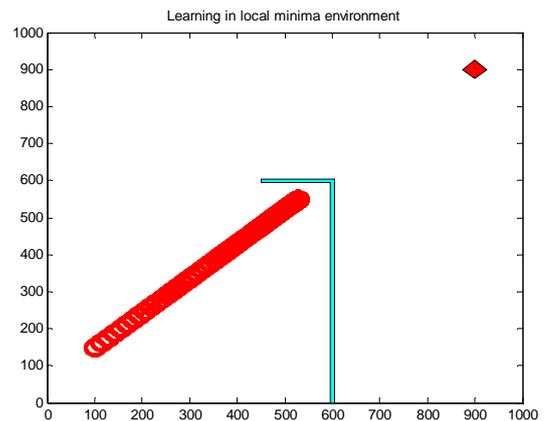


Figure 8(a). Artificial potential field method

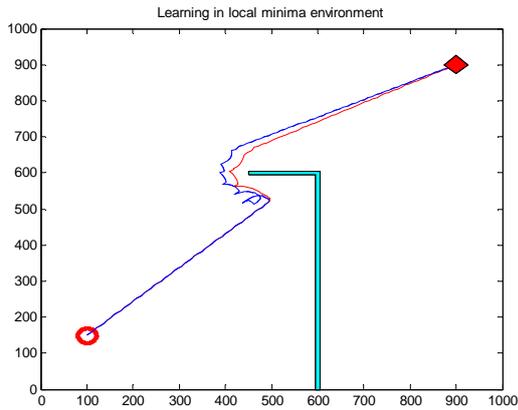


Figure 8(b). Blended method

## VII. CONCLUSION

The path planning problem of mobile robot in unknown environment is a focus of robotics, this paper presents a new hybrid learning method to solve this problem, based on fuzzy inference, this paper combines the traditional supervised learning and learning automata with bionic biological characteristics, finally realized the effective control of the robot. Firstly, adjust the input fuzzy function of the fuzzy system and both the conclusion roughly. Secondly, fixed the input fuzzy function, learning automata is used to tune the conclusions future more. Simulation results show that the learning automata makes the robot have the ability of autonomous learning and adaptability to the new environment, the introduce of supervised learning not only reduced the time that the learning automata used, but also get the automata better conclusions, the combination of this two methods improved the robot's robustness and its ability to overcome the local minima, in the task, it takes less time and performs better. At present, there are many methods of path planning, each has its own advantages, the combination of classical path planning methods and cognitive science methods will provide us a new research direction.

## REFERENCES

- [1] Y. G. Xi, C. G. Zhang, "Rolling Path Planning of Mobile Robot in a Kind of Dynamic Uncertain Environment," *Acta Automatic Sinica*. Beijing, vol. 28, pp. 161-175, February 2002.
- [2] C. L. Zhong, S. R. Liu, "Mobile Robot Active Routing Navigation Based on Virtual Intermediate Goal," *Robot. Shenyang*, vol. 31, pp. 548-555, November 2009.
- [3] H. D. Zhang, R. Zheng, Y. W. Cen, "Present Situation and Future Development of Mobile Robot Path Planning Technology," *Journal of System Simulation*. Beijing, vol. 17, pp. 439-443, February 2005.
- [4] S. Kermiche, M. L. Saidi, H. A. Abbassi, "Gradient descent adjusting Takagi-Sugeno controller for a navigation of robot manipulator," *JEAS*. Taiwan, vol. 1, pp. 24-29, June 2006.
- [5] K. K. Tan, K. C. Tan, K. Z. Tang, "Evolutionary tuning of a fuzzy dispatching system for automated guided vehicles," *IEEE Trans Syst Man Cybern B Cybern*. New York, vol. 30, pp. 632-636, August 2000.
- [6] S. I. Lee, S. B. "Cho, Emergent behaviors of a fuzzy sensory-motor controller evolved by genetic algorithm," *IEEE Trans Syst Man Cybern B Cybern*. New York, vol. 31, pp. 919-929, December 2001.
- [7] Y. Dahmani, A. Benyettou, "Fuzzy Reinforcement Rectilinear Trajectory Learning," *J. Appl. Sci*. New York, vol. 4, pp. 388-392, September 2004.
- [8] P. Y. Glouennec, L. Jouffe, "A reinforcement learning method for an autonomous robot," *The Fourth European Congress on Intelligent Techniques and Soft Computing*. September 1996.
- [9] Y. Duan, X. H. Xu, "Reinforcement learning based on FNN and its application in robot navigation," *Control and Decision*. Shenyang, vol. 22, pp. 525-534, May 2009.
- [10] H. Boubertakh, M. Tadjine, P. Y. Glouennec, "A new mobile robot navigation method using fuzzy logic and a modified Q-learning algorithm," *J Intell Fuzzy Syst*. New York, vol. 21, pp. 113-119, January 2010.
- [11] N. H. C. Yung, Y. Cang, "An Intelligent Mobile Vehicle Navigator Based on Fuzzy Logic and Reinforcement Learning," *IEEE Trans Syst Man Cybern B Cybern*. New York, vol. 29, pp. 314-321, April 1999.
- [12] J. E. Meng, D. Chang, "Online Tuning of Fuzzy Inference Systems Using Dynamic Fuzzy Q-Learning," *IEEE Trans Syst Man Cybern B Cybern*. New York, vol. 34, pp. 1478-1489, June 2004.
- [13] Y. Zhou, J. E. Meng, "An Evolutionary Approach Toward Dynamic Self-Generated Fuzzy Inference Systems," *IEEE Trans Syst Man Cybern B Cybern*. New York, vol. 38, pp. 963-969, August 2008.
- [14] Y. Y. Gao, X. G. Ruan, H. J. Song, J. J. Yu, "Path planning method for mobile robot based on a hybrid learning approach," *Control and Decision*. Shenyang, vol. 27, pp. 1822-1827, December 2012.
- [15] H. Simon, *Neural Networks: A Comprehensive Foundation*. New York: Macmillan College Publishing Company, 1994, pp. 50-61.