Dynamic obstacle avoidance for path planning and control on intelligent vehicle based on the risk of collision

LU YEQIANG[1] QIU FAJU[2] XIN JIANGHUI[3] SHANG WEIYAN[4]
1Department of Electromechanical Engineering,
Hangzhou Vocational Technology College,
CHINA
weiyan221@nuaa.edu.cn
2 Ningbo Special Equipment Inspection & Research Institute,
CHINA
qiufaju@163.com
3 Department of mechanical engineering
Nanjing University of Technology
CHINA
4Department of mechanical engineering,
Ningbo university of technology
CHINA
weiyan221@nuaa.edu.cn

Abstract：- To improve the autonomy of intelligent vehicle in complex or uncertain environment, a dynamic planning control method for obstacle avoidance has been studied. By leading the degree of risk collision into control system as the input and using an improved fuzzy control algorithm, the input and output of the fuzzy controller can be considered as extract amount. The concept of patterns and pattern matching was used, and according to the matching degree of each rule, the weighted average algorithm was applied to determine the output of control action, this method can avoid a dynamic obstacle in a timely manner and shorten the computing time. At last, by simulation, it was verified that the dynamic control method can make intelligent vehicle avoid the dynamic obstacles independently, and walking toward the target point exactly. And the results also provide a theoretical basis for the realization of the intelligent vehicle moving independently and safely in the complex and dynamic environment.

Key-Word: -Intelligent vehicle, Degree of risk collision, Fuzzy control, Dynamic obstacle avoidance

1 Introduction

When executing a task, the intelligent vehicle always runs in uncertain environment, so the path planning in the exploration task is a very important link[1,2]. The environment information can be described into two types, one is the global path planning[3,4] that knows the whole information of environment, and the other is local path planning[5,6] that knows only a part or know nothing about the environment. In the global path planning, the robot can program a safety and optimal path from the starting point to the target point, and the information always comes from global map database information. In the local path planning, the information always comes from sensors that detect the environment online and provide the information of obstacle shape, size, speed and so on. As most of obstacles in the environment are dynamic, on line local path planning method is needed to avoid dynamic obstacles and reach the target at last. Therefore, it is important to find an effective method to avoid dynamic obstacles in the exploration task. The fuzzy control method, the artificial potential field method, the rolling window method, and RL method and so on are used widely at present[7-10]. The above methods do not consider about the motion information comprehensively, so a new method considering fully information about dynamic obstacles are needed to plan a safe path in non-structural road environment and to change the control strategy timely.

2 Establishment of dynamic obstacle avoidance model and fuzzy control algorithm for intelligent vehicle

2.1 kinematic equations for intelligent vehicle

Considering about the distance between intelligent vehicle and obstacles as well as the velocity information, the collision risk can be established. Assuming the intelligent vehicle moving towards the target point G, the coordinate system can be established as figure 1. In this coordinate system, the location coordinate of the intelligent vehicle is (x_g,y_g), and the movement towards the target point can be described by the line speed V_R and angular velocity ω .

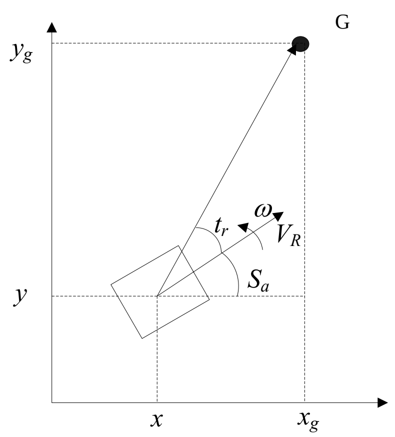

Fig.1. Diagram for intelligent vehicle location and movement

From figure 1 we know that the following equations exist, that is $\dot{x} = V_R \cos S_a$, $\dot{y} = V_R \sin S_a$ and $\dot{S}_a = \omega$. From the equations it can be concluded that the obstacle avoidance action of intelligent vehicle can be achieved by changing the line speed V_R and direction angle S_a(or ω), and by controlling these parameters the path planning or navigation can be realized.

2.2 The establishment of dynamic obstacle avoidance model

In dynamic environment, the intelligent vehicle, obstacles and target are always changing, so if more object motion information included in the environment model, more efficient path planning will be carried out. In this paper, by using the principle of the speed barrier, the collision risk conception was introduced to make path planning of the intelligent vehicle[11,12]. In order to simplify the problem, the model of intelligent vehicle can be regarded as a point, and the obstacle can be treated expansively according to the dimension of intelligent vehicle and safety requirements.

The spatial location diagram of the intelligent vehicle and obstacle is showed in figure 2. In this figure, B_i represents the ith obstacle and its velocity is V_i, and V_R is the relative speed between the intelligent vehicle and obstacles.

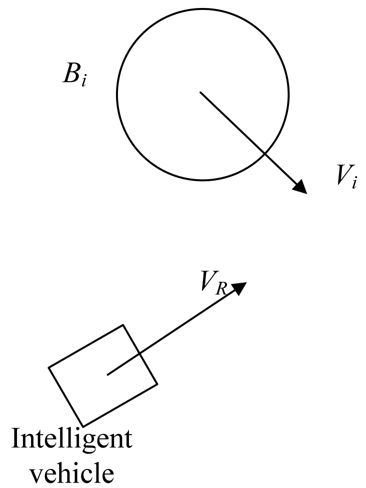

Fig.2. Spatial location diagram of the intelligent vehicle and obstacle

The principle of the speed and barriers is showed in figure 3. In this figure, CC_i represents the collision region, VO_i is the next collision region in the next time when moving on current speed. In order to carry on the effective collision avoidance, the intelligent vehicle should select its speed in the next time when the speed can be reached.

Figure 4 represents the situation that there are several obstacles, and the quadrilateral PSFQ is the velocity range that the intelligent vehicle can reach. And the reachable avoidance velocities (RAV) composed by several distinction regions which shows in figure 4 is the region 1, 2 and 3. And region 1 is the reached region, where the intelligent vehicle should slow down. While in region 2 and region 3, the intelligent vehicle should speed up to surpass the obstacle, and we can know from figure 4 that the obstacle 2 is above B_i and the obstacle 3 is below B_i.

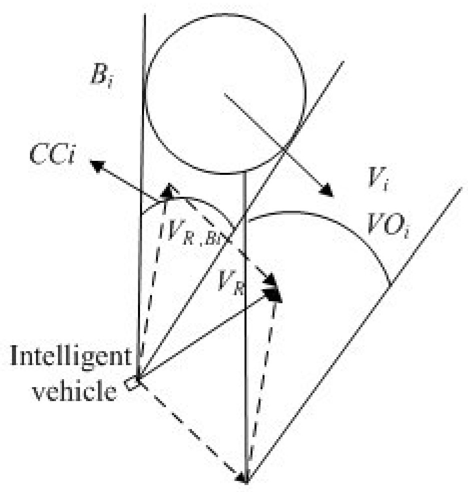

Fig.3. Speed barrier diagram

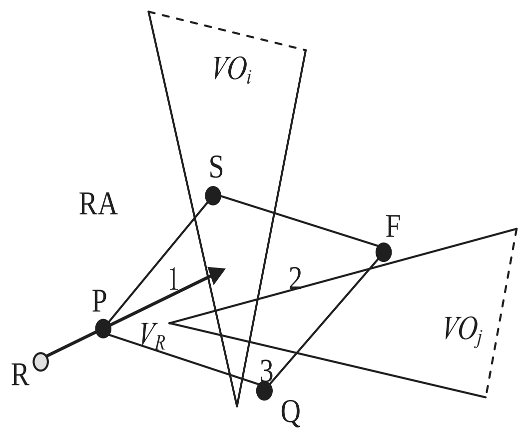

Fig.4. Diagram of preventing collisions under the situation of several obstacles

In order to deal with the problem, we introduce the concept of collision risk. The collision risk can be described as the following. When the risk of collision exists, the main factors depend on the following factors: the initial minimum determination distance, the initial nearest point position, and the orientation of intelligent vehicle that relative to obstacles. And the main index is the minimum safety distance between intelligent vehicle and the obstacle. The possibility of collision risks is defined by the main factor and the main index.

The schematic diagram of collision risk is showed in figure 5. Where d_i means the distance between intelligent vehicle and the obstacle i, V_R, B_i represents the relative velocity between intelligent vehicle and obstacles, A and B are the two edge points of obstacles perceived by sensors on R, and θ is the angle between relative velocities and center connection.

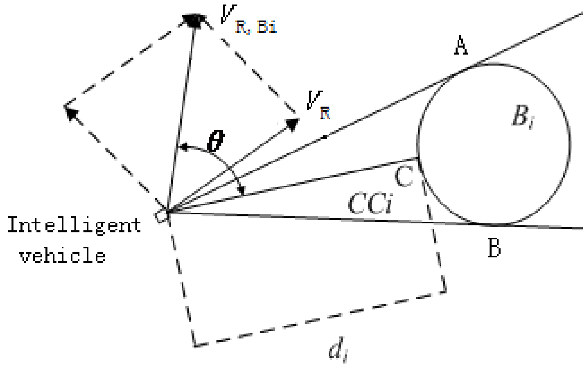

Fig.5. Schematic diagram of space collision risk

The Space collision risk (SCR) can be defined as the following,

$$SCR(t) = a \times D(t) + b \times F(t)$$

$$D(t) = \begin{cases} 0 & ; d_i \ge d_{\max} \\ \dfrac{d_{\max} - d_i}{d_{\max} - d_{\min}} & ; d_{\min} < d_i < d_{\max} \\ 1 & ; 0 \le d_i \le d_{\min} \end{cases}$$

（1）

$$F(t) = \begin{cases} 0 & ; \theta \ge \theta_{\max} \\ \dfrac{\theta_{\max} - \theta}{\theta_{\max} - \theta_{\min}} & ; \theta_{\min} < \theta < \theta_{\max} \\ 1 & ; 0 \le \theta \le \theta_{\min} \end{cases}$$

when $\theta_{\min} < \theta < \theta_{\max}$, $V_{R,\ Bi}$ is in CC_i;

when $0 \le \theta \le \theta_{\min}$, V_{R,B_i} is out of $CC_{i.}$ (2)

The a and b are impact factors, $D(t)$ reflects the relationship between distance and $SCR(t)$, and $F(t)$ reflect the relationship between the direction of relative velocity and $SCR(t)$. $d_{\max}$ is the maximum measuring distance, $d_{\min}$ is the safe distance, $\theta_{\max}$ is the maximum measuring angle, and $\theta_{\min}$ is the safe angle.

3 Conventional fuzzy control algorithm

Typical operations throughout the sampling period on conventional fuzzy controller include measuring input sampling, accurate input fuzzified and to be expressed with the fuzzy sets, and at last, the fuzzy output can be transferred into accurate value and then the whole course can be controlled. Fuzzy inference involves two aspects: firstly, the matching degree between fuzzy input and the rule conditions (IF) should be calculated, secondly, the rules that be activated should be determined, and at last, the final results of these activated rules (THEN) should be averaged by its weight to form the final control set[13-15].

In fuzzy control, for the low compatibility between the value (or accurate) control environment and language inference algorithm, the processing becomes complicated and which usually requires two interfaces to link the two part[16-18]. To simplify this process and strengthen the connection of the various parts, an improved fuzzy control algorithm has been applied in the paper. Considering the input/output to be accurate values in the fuzzy controller, the improved algorithm includes two points: one is pattern matching and the other is weighted average. Which have removed the tedious fuzzy course and exact process. The starting point of the algorithm is to introduce the concept of patterns and pattern matching. The model includes the input mode and regular mode, and the matching degree between them can be expressed by the norm. Then according to each rule's matching degree, the output of the control action can be determined by the weighted average algorithm. In this process, the IF part in the pattern matching of the processing should find the active unit, and the THEN part of the weighted average processing rules will form the control output.

3.1 Input and output

Assuming that the number of inputs and outputs are both m in the multi-variable controller process, and the input of fuzzy controller can be various combinations of control error (E), the change rate of error (C), and the sum of error (S). The *k*th sampling period can be expressed as $e_{ci}(kT_s)$, where is the sampling period, *r* and $y_p \in Rm$ is respectively represent the setpoint and process output (i=1,2, ... ,m). Then we can get $c_c(kT_s)$ and $s_c(kT_s)$ from $e_c(kT_s)$, that is the error change rate is $c_c(kT_s) = e_c(kT_s) - e_c[(k-1)T_s]$, and the sum of error is

$$s_c(kT_s) = \sum_{i=1}^{k} e_c(iT_s) .$$

Then three kinds of fuzzy controller input mode can be structured as the following,

The first input mode: $e_{c1}, c_{c1}, e_{c2}, c_{c2}, \cdots, e_{cm}, c_{cm}$,

The second input mode:

$e_{c1}, s_{c1}, e_{c2}, s_{c2}, \cdots, e_{cm}, s_{cm}$,

The third input mode:

$e_{c1}, c_{c1}, s_{c1}, e_{c2}, c_{c2}, s_{c2} \cdots, e_{cm}, c_{cm}, s_{cm}$.

And the three input modes can be abbreviated as EC, ES and ECS. Where E,C,S respectively represent the error, the error change rate and error sum. One input mode can be selected from the three modes, and it can be expressed as ui, where i=1,2,…,n. And the output of fuzzy controller can be expressed as v_k, where k=1,2,…,m.

3.2 Rule pattern and input mode

Assuming that the number of rules in the control rule base is M, and each form of the rule is as the following,

IF ($A_1{}^j = U_1$) AND ($A_2{}^j = U_2$) …AND ($A_n{}^j = U_n$),

THEN ($B_1{}^j = V_1$) AND ($B_2{}^j = V_2$) AND ($B_n{}^j = V_n$).

Where U_i and V_k is the linguistic variables corresponding to the numeric variable u_i and v_k. $A_i{}^j$ and $B_k{}^j$ is the fuzzy subset to express the language items, and $A_i{}^j \in \overline{U_i}$, $B_k{}^j \in \overline{V_k}$ .The fuzzy subset $A_i^{\,j}(u_i)$ and $B_k^{\,j}(v_k)$ can be expressed as normalized fuzzy subset by membership function $A_i^{\,j}(u_i):\overline{U_i}\to[0,1]$ and $B_k^{\,j}(v_k):\overline{V_k}\to[0,1]$. The definition of membership function is as the following,

$$A_i^{\,j}(u_i)=\begin{cases}1-(\dfrac{\left|M_{u,i}^{\;j}-u_i\right|}{\delta_{u,i}^{\;j}}), & \left|M_{u,i}^{\;j}-u_i\right|\le\delta_{u,i}^{\;j}\\ 0, & \left|M_{u,i}^{\;j}-u_i\right|>\delta_{u,i}^{\;j}\end{cases}\qquad(3)$$

$$B_k^{\,j}(v_k)=\begin{cases}1-(\dfrac{\left|M_{v,k}^{\;j}-u_k\right|}{\delta_{v,k}^{\;j}}), & \left|M_{v,k}^{\;j}-u_k\right|\le\delta_{v,k}^{\;j}\\ 0, & \left|M_{v,k}^{\;j}-u_k\right|>\delta_{v,k}^{\;j}\end{cases}\qquad(4)$$

Where $M_{u,i}^{\;j}\in\overline{U_i}$, $M_{v,k}^{\;j}\in\overline{V_k}$, $\delta_{u,n}^{\;j}>0$ and $\delta_{v,k}^{\;j}>0$.

In formula (3) and formula (4), there are only two parameters for each membership function, that are $M_{u,i}^{\;j}$ and $\delta_{u,i}^{\;j}$ (or $M_{v,k}^{\;j}$ and $\delta_{v,k}^{\;j}$), where $M_{u,i}^{\;j}(M_{v,k}^{\;j})$ is the center of the domain $Ai^{j}(B_k^{\,j})$.

Thus Ai^{j} and $B_k^{\,j}$ can be expressed as

$$A_i^{\,j}=(M_{u,i}^{\;j},\delta_{u,i}^{\;j}),B_k^{\,j}=(M_{v,k}^{\;j},\delta_{v,k}^{\;j})\qquad(5)$$

By the above formula, the *j*th rule can be written as

IF $(M_{u,l}^{\;j},\delta_{u,l}^{\;j})$ AND …AND $(M_{u,n}^{\;j},\delta_{u,n}^{\;j})$, THEN $(M_{v,l}^{\;j},\delta_{v,l}^{\;j})$ AND … AND $(M_{v,m}^{\;j},\delta_{v,m}^{\;j})$.

In the IF section of the rules, the input space can be assumed as $\Omega=(\overline{U_1}\times\overline{U_2}\times\cdots\times\overline{U_n})\in R^n$, $M_u^{\,j}=(M_{u1}^{\;j},M_{u2}^{\;j},\cdots,M_{un}^{\;j})\in\Omega$ and $\Delta_u^{\,j}=(\delta_{u1}^{\;j},\delta_{u2}^{\;j},\cdots,\delta_{un}^{\;j})$. So the condition part of the jth rule can be regarded as a sub space $\Omega^j=\Omega$ or a hyperplane, the center and Radius is respectively $M_u^{\,j}$ and $\Delta_u^{\,j}$, which can be simplified as IF $M\Delta_u(j)$, where $M\Delta_u(j)=(M_u^{\,j},\Delta_u^{\,j})$.

3.3 Pattern matching

In the control process, as there is improved fuzzy algorithm based on the predetermined control process, appropriate control action can be gotten by the current input and the M rules. Taking the concept of distance matrix theory, the similarity between the two modes can be measured. And the algorithm to measuring the distance can be defined. Assuming that the current input is $u_0=[u_{01},\delta_{02},\cdots,\delta_{0n}]$, then the *j*th rule mode between $M\Delta_u(j)$ can be expressed as $S^j\in[0,1]$, which can be defined as,

$$S^j=1-D^j(u_0,M\Delta_u(j))\qquad(6)$$

Where D^j represents the relative distance from u_0 to $M\Delta_u(j)$. Here are three commonly used calculation method to calculate D^j.

(1) Relative Ohm distance of

$$D_E^{\,j}=\begin{cases}\dfrac{\left\|M_u^{\,j}-u_0\right\|}{\left\|\Delta_u^{\,j}\right\|} & ,\left\|M_u^{\,j}-u_0\right\|\le\left\|\Delta_u^{\,j}\right\|\\ 1 & ,\text{others}\end{cases}\qquad(7)$$

Where $\|\cdot\|$ expresses the norm.

(2) Relative Hamming moment distance

$$D_H^{\,j}=\begin{cases}\dfrac{\sum_{i=1}^{n}\left|M_{u,i}^{\;j}-u_{0i}\right|}{\sum_{i=1}^{n}\delta_{u,i}^{\;j}} & ,\sum_{i=1}^{n}\left|M_{u,i}^{\;j}-u_{0i}\right|\le\sum_{i=1}^{n}\delta_{u,i}^{\;j}\\ 1 & ,\text{others}\end{cases}\qquad(8)$$

(3) The relative maximum distance

$$D_M^{\,j}=\begin{cases}\max\limits_{1\le i\le n}\dfrac{\left|M_{u,i}^{\;j}-u_{0i}\right|}{\delta_{u,i}^{\;j}} & ,\left|M_{u,i}^{\;j}-u_{0i}\right|\le\delta_{u,i}^{\;j}\\ 1 & ,\text{others}\end{cases}\qquad(9)$$

From formula (6) to (9) we can know that, if u_0 and $M\Delta_u(j)$ matched exactly, that means u_0 is completely the same with center vectors M_u^j, then we get $D^j=0$ and $S^j=1$. Otherwise, if u_0 and $M\Delta_u(j)$ don't matched, that means u_0 is out of $M\Delta_u(j)$, then we get $S^j=0$. And in other conditions, we get $0 < D^j < 1$ and $0 < S^j < 1$, which means partial matches.

3.4 The weighted average

As the membership function has been defined, the rules can be expressed as the following,

IF $M\Delta_u(j)$ THEN $(M_{v,1}{}^j, \delta_{v,1}{}^j)$ AND $(M_{v,m}{}^j, \delta_{v,m}{}^j)$,

Assuming that when k and j takes any value, the value of $\delta_{v,k}{}^j$ will be the same. When the current input mode j is given and the match degree is $S^j = 1$, the control action will be deduced as $v_k = M_{v,m}{}^j (k = 1,2,\cdots,m)$. And this result is meeting with the maximum degree of decision-making program and meanwhile this method will be the same with the center of gravity method(COG). If the membership function is on the centrosymmetric, the above conclusion will be proved. In addition, if $S^j = 0$, the jth rules will have no effect on the control output; and when $0 < S^j < 1$, there will be several rules effect on the control output.

Assuming that there are existing input u_0 and the number of the rule is P, and after the pattern matching, the number of match degree will be Q which meets the condition of $0 < S^j < 1$. The match degree can be expressed as $S^1, S^2, \cdots, S^Q$, which will correspond with Q group center of the control rule THEN, that is as the following,

$$\{M_{v,1}{}^1, M_{v,2}{}^1, \cdots M_{v,m}{}^1\} \cdots \{M_{v,1}{}^Q, M_{v,2}{}^Q, \cdots M_{v,m}{}^Q\}$$

Then we can deduce that the control action v_k of the kth components should be expressed as

$$v_k = \frac{\sum_{q=1}^{Q} S^q M_{vk}{}^q}{\sum_{q=1}^{Q} S^q} = \sum_{q=1}^{Q} \overline{S^q} M_{vk}{}^q \tag{10}$$

Where $$\overline{S^q} = \frac{S^q}{\sum_{q=1}^{Q} S^q}$$

In the above formula(10), the weighted average of the activated rule THEN is given, and the proportion of each rule in the control action is determined by their matching degree. So only the central element of the THEN part whose maximum membership degree is 1 has been used, and we can regard that it is the changing form of the maximum degree of decision-making program.

4 Improvement fuzzy neural network dynamic path planning based on the risk of collision

As the accurate mathematical model is difficult to be established in the practice, and meanwhile the environment for intelligent vehicle is also changing, improved fuzzy control algorithm combining with the advantages of neural networks will be applied for dynamic path planning. By applying this method, the fuzzy control can have the ability of self-learning, the neural network can be given the ability of induction and inference, and meanwhile the network structure and weights will have clear physical meaning to make the design and initialization of the network easier.

4.1 The establishment of the dynamic environment for intelligent vehicle

Fuzzy neural network topology is showed in Figure 6. The first layer is to input values, it can sent the input values directly to the next layer. There are four input variables, which are the left obstacle collision risk L_{SCR}, front obstacle collision risk C_{SCR}, the right obstacle collision risk R_{SCR} and targeting T_r.

Each node in layer 2 represents a linguistic variable values, such as NB, PS and so on. It is to calculate the membership function of each input component which belongs to the fuzzy sets of language variable value. By the definition of the above collision risk, the range of SCR is 0～2, where a=1 and b=1. Then the fuzzy subset of L_{SCR}, C_{SCR}, R_{SCR} can be defined as ｛Z, S, M, B｝, which represent the collision risk can be described as zero, few, medium and large. And it also means that the collision risk can be described into 4 levels, that is $m_1=m_2=m_3=4$. While the target location T_r can be described into 5 degrees, which are｛TLB, TLS, TZ, TRS, TRB｝, and they are respectively represent left large, left few, zero, right few, and right large. So the partition number of navigation angle input fuzzy is 5, that is m_4=5. The above membership functions is using the Gaussian function as

$$\mu_i^j = e^{\frac{-(x_i-c_{ij})^2}{\sigma^2_{ij}}}$$

Where c_{ij} and σ_{ij} represent the center and width of membership function. In dynamic local path planning, the total number of layer nodes would use the speed information of the intelligent vehicle and obstacles. In the simulation, collision risk from each direction will be regarded as the inputs for the fuzzy Neural Network. And considering that the intelligent vehicle will eventually have to reach the target point, the target location T_r is also regarded as an input. And correspondingly, the output of the network for the intelligent vehicle is the speed V and the turning angle S_a.

Each node in layer 3 represents a mode of rule, it is used to match the fuzzy rules, and calculate the match degree between each rule and the set of points in the space of Ω. The total number of nodes in the layer is $N_3 = m_1 \times m_2 \times m_3 \times m_4 \times m_5$.

The number of nodes in layer 4 is the same with layer 3, it is to realize the normalized calculation, that is $\overline{\alpha_j} = \frac{\alpha_j}{\sum_{i=1}^{m}\alpha_i}; j = 1,2,\cdots,m$

Layer 5 is to output the accurate information by weighted average, that is $y_1 = \sum_{j=1}^{m}\omega_{1j}\overline{\alpha_j}$ and $y_2 = \sum_{j=1}^{m}\omega_{2j}\overline{\alpha_j}$.

Where y_1 repesent the speed V_R and y_2 represent the turning angle of S_a。

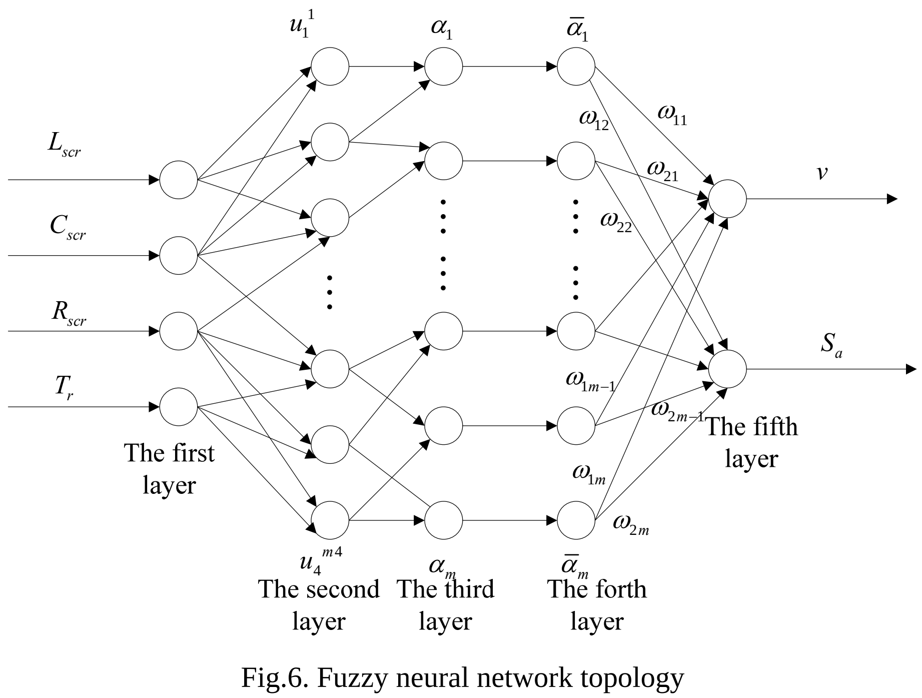

Fig.6. Fuzzy neural network topology

4.2Fuzzy neural network learning algorithm

Basing on the above structure of fuzzy neural network, the fuzzy partition number of the input component is pre-determined. So only the connection weights of the last layer($\omega_{ij}(i=1,2; j=1,2,\cdots,m)$), the second layer of the center value of membership function c_{ij}, and the width $\sigma_{ij}(i=1,2,3,4; j=1,2,\cdots,m)$ should be studied.

As the network given above is essentially a multilayer feedforward network, so it can be modeled on the BP neural network with error back propagation method to adjust the parameters of the learning algorithm. The error cost function can be written as

$$E_f = \frac{1}{2}\sum_{i-1}^{2}(y_{di} - y_i)^2$$

Where, y_{di} and y_i are respectively denote the desired output and actual output value.

Using error back propagation algorithm to calculate $\partial E_f / \partial \omega_{ij}, \partial E_f / \partial c_{ij}$ and $\partial E_f / \partial \sigma_{ij}$. And then we can adjust the three parameters ω_{ij}, c_{ij} and σ_{ij} by a ladder degree of optimization algorithm. Formalized description of the input and output of the neurons of the network should be carried out, each neuron input defined in the network is $f^{(q)}(x_1^{(q-1)},\cdots,x_n^{(q-1)};\omega_{j1}^{(q)}\cdots,\omega_{jn}^{(q)})$, and the output of the node is $x_j^{(q)} = g^{(q)} f^{(q)}$. Then in the fuzzy neural network of fig.3, the node function of each layer is given as the following,

The first layer,

$$f_i^{(1)} = x_i, x_i^{(1)} = g_i^{(1)} = f_i^{(1)}; i = 1,2,3,4$$

The second layer,

$$f_{ij}^{(2)} = \frac{(x_i^{(1)} - c_{ij})^2}{\delta_{ij}^2}$$

$$x_{ij}^{(2)} = u_i^j = g_{ij}^{(2)} = \exp(f_{ij}^{(2)}) = \exp\left[-\frac{(x_i^{(1)} - c_{ij})^2}{\delta_{ij}^2}\right],$$

$$i = 1,2,3,4; j = 1,2,\cdots,m_i$$

The third layer,

$$f_j^{(3)} = \min\left\{x_{1i_1}^{(2)}, x_{2i_2}^{(2)}, x_{3i_3}^{(2)}, x_{4i_4}^{(2)}\right\} = \min\left\{u_1^{i_1}, x_2^{i_2}, x_3^{i_3}, x_4^{i_4}\right\}$$

$$j = 1,2,\cdots,m; m = \prod_{i=1}^{n} m_i$$

The forth layer,

$$f_j^{(4)} = x_j^{(3)} \Big/ \sum_{i=1}^{m} x_i = \alpha_j \Big/ \sum_{i=1}^{m} \alpha_i$$

$$x_j^{(4)} = g_j^{(4)} = f_j^{(4)} = \alpha_j; j = 1,2,\cdots,m$$

The fifth layer,

$$f_i^{(5)} = \sum_{j=1}^{m} x_j^{(4)} = \sum_{j=1}^{m} \omega_{ij}\overline{\alpha_j}$$

$$x_i^{(5)} = y_i = g_i^{(5)} = f_i^{(5)} = \alpha_j; i = 1,2$$

Then by calculation, the algorithm for parameter adjustment can be obtained as,

$$\omega_{ij}(l+1) = \omega_{ij}(l) - \beta_1 \frac{\partial E_f}{\partial \omega_{ij}};$$
$$i = 1,2; j = 1,2,\cdots,m \qquad (11)$$

$$c_{ij}(l+1) = c_{ij}(l) - \beta_2 \frac{\partial E_f}{\partial c_{ij}};$$
$$i = 1,2,3,4; j = 1,2,\cdots,m \qquad (12)$$

$$\delta_{ij}(l+1) = \delta_{ij}(l) - \beta_3 \frac{\partial E_f}{\partial \delta_{ij}};$$
$$i = 1,2,3,4; j = 1,2,\cdots,m \qquad (13)$$

Where, β_1, β_2 and β_3 are the learning rate.

5 Simulation

Based on the above method, the MATLAB software has been used to make simulation. And the result has been compared with the ordinary fuzzy control algorithm. Considering to the learning convergence rate, we set $\beta_1 = \beta_2 = \beta_3 = 0.7$. And two methods (the improved and the forward fuzzy neural network control method) have been adopted to make path planning in a static environment, shown in figure 7 and figure 9 The starting point of intelligent vehicle is S (10,0) and the target point is (35,35). It can be seen from Figure 7 and figure 9 that slight difference existed in the planning path between the algorithm of improved fuzzy neural network control and ordinary fuzzy control, and in the static environment, the algorithm of ordinary fuzzy algorithm consuming shorter time in the process of fuzzification and defuzzification.

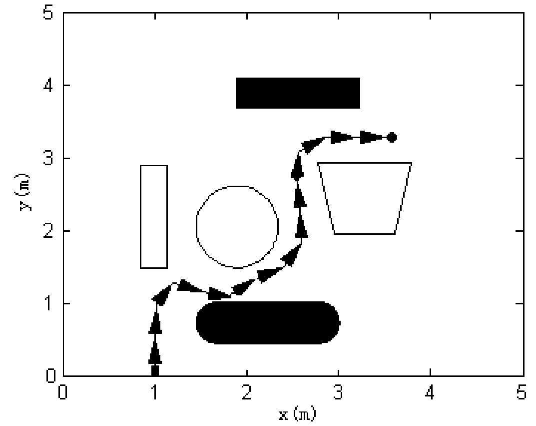

Fig.7 The ordinary fuzzy path planning algorithm in static environment

The speed curve for improved algorithm static path planning is showed in figure 11. From this figure it can be seen that the maximum speed is 0.09 m/s when there is no obstacle in the left and the right of intelligent vehicle. When there are obstacles in the front, the intelligent vehicle will avoid them and advance to the target point, and the vehicle’s velocity will be reduced. So when there are complicated conditions, more obstacles and large collision risk, there will be a small scale in the speed. And this condition is similar with humane driving, for the judgment process to complex environment will lead to the speed’s fluctuation.

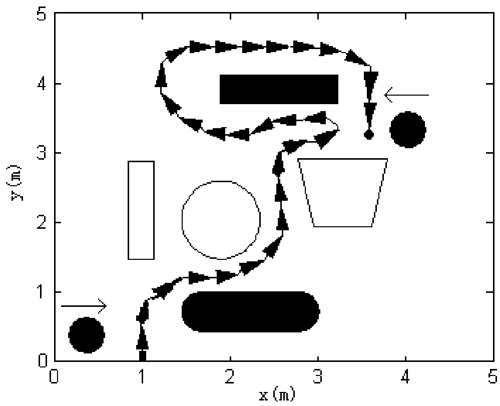

Fig.8. The ordinary fuzzy path planning algorithm in dynamic environment

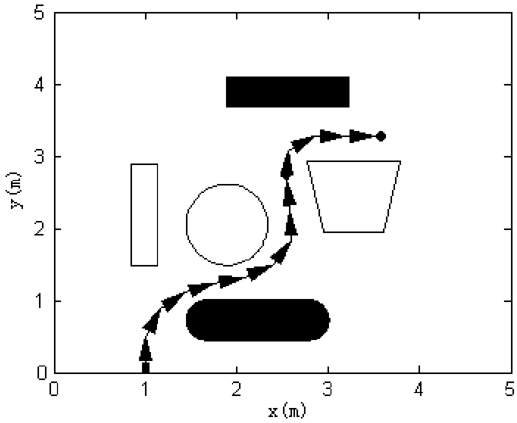

Fig.9. The improved path planning algorithm in static environment

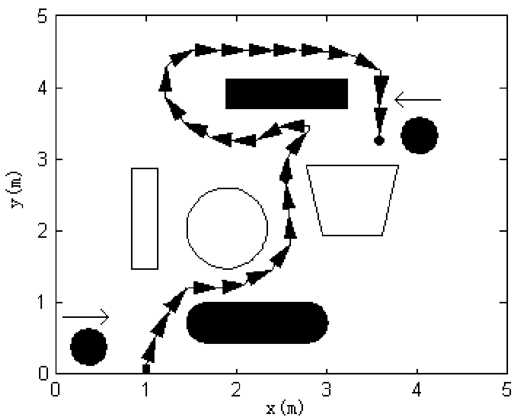

Fig.10. The improved path planning algorithm in dynamic environment

The path planning for intelligent vehicle adopting two algorithms in a dynamic environment is showed in figure 8 and figure 10. There are two moving obstacles in the environment and their initial position is respectively (3,3) and (40,35). And the two obstacles are moving respectively to the right and the left by the uniform velocities of 0.2m/s and 0.5m/s. There are also another five static obstacles in the environment. It can be seen from figure 8 and figure 10 that there are greater difference in the dynamic obstacle environment between the improved fuzzy neural network control algorithm and the ordinary fuzzy control algorithm. And it indicates that ordinary fuzzy algorithm consumes time in fuzzification and defuzzification which lengthened the planning path, while the improved algorithm is so simple to shorten the calculation time, and meanwhile the intelligent vehicle can avoid dynamic obstacles timely.

The speed curve for improved algorithm dynamic obstacles path planning is showed in figure 12. From the speed curve we can see that when the intelligent vehicle encountered dynamic obstacles, the planning path will change, and the intelligent vehicle will reach the crossing point where it meet the first dynamic obstacle after 9 sampling time. And the first moving obstacles will reach the crossing point after 12 sampling time. Then after another 14 sample time the intelligent vehicle will reach the crossing point where it meet the second dynamic obstacles. And we also can see that the second moving obstacle will reach the intersection through 8 sampling time. As the intelligent vehicle and the moving obstacles reach the same location at different times, the moving obstructions can thus be avoided.

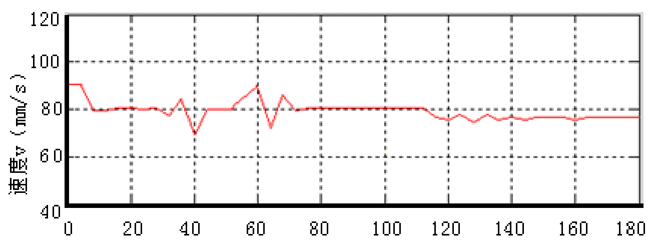

Fig.11 Speed curve in a static environment for the path planning of intelligent vehicle

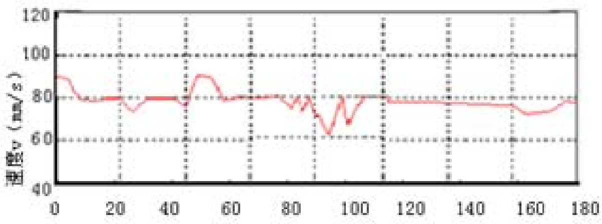

Fig.12 Speed curve in a dynamic environment for the path planning of intelligent vehicle

6 Conclusions

In the dynamic environments for path planning, it is necessary to consider both the distance information of obstacles and the motion information of obstacles. In this paper, a new fuzzy neural control algorithm has been used, and the collision risk has also been introduced to act as a control input, which provides a theoretical basis for the realization of the independent safety moving for intelligent vehicle in the complex and dynamic environment. By conclusion, it can be expressed as the followings,

（1） The inputs and outputs of fuzzy controller has been considered as exact amount directly in improved fuzzy control algorithm, which includes the two points of pattern matching and weighted average, and then the course of tedious fuzzy and precision has been removed.

（2） The patterns and pattern matching have been introduced into the study, and the weighted average algorithm has been used to determine the output of the control action by the match degree of each rule, which shortened the computing time.

（3） Controlling by fuzzy neural model based on the collision risk, the intelligent vehicle can avoid the complexity of the dynamic obstacles and move towards the target point.

References:

[1] Shang Weiyan, Li Shunming, Bao Qingyong.etd. Kinematics modeling and analysis of a military exploration vehicle with wheel-tracks [J]. Journal of Aerospace Power.2009,10： 2302-2307. (In

Chinese)

[2] Shang Weiyan, Li Shunming, Bao Qingyong. Structure Design and Kinematics Modeling of a New Compound Moving System of an Exploration Robot. [J]. Mechanical Science and Technology for Aerospace Engineering,2009,28(7):881-885. (In Chinese)

[3] Kennedy J., Eberhart RC. Particle Swarm Optimization[C].Proceedings of IEEE International Conference on Neutral Networks.Perth,1995:1942-1948.

[4] Qiaorong Zhang., Shuhong Li. Path planning based on improved particle swarm optimization for a mobile robot [J]. WSEAS Transactions on Systems and Control. 2007, 2 (5): p 347-52.

[5] Lahouar, S. , Zeghloul, S., Romdhane, L. Path planning for manipulator robots in cluttered environments[J]. WSEAS Transactions on Systems, 2005, 4(5): 555-60.

[6] Aoughellanet, S. Mohammedi, T. Bouterfa, Y. Neural network path planning applied to PUMA560 robot arm. WSEAS Transactions on Systems, 2005, 4(4): 446-50.

[7] Jianghui Xin, Shunming Li. Application of Fuzzy Neural Networks in the Intelligent Mobile Vehicle’s Steering Control. The 14th Asia Pacific Automotive Engineering Conference, Hollywood, USA, 2007: 5-8.

[8] Kuentai Chen, Hung-Chun Chen Che, Z.H. Simulation of production and transportation planning with uncertainty and risk. WSEAS Transactions on Computers, 2008, 7(10): 1535-1544.

[9] Shunming Li, Jianghui Xin, Weiyan Shang. The algorithm of obstacle avoidance based on improved fuzzy neural networks fusion for exploration vehicle. WSEAS Transactions on Systems and Control, 2009, 4(3): 140-150

[10] Duan qunjie. Real-time path planning for underwater robots.[Doctoral Dissertation]. Harbin: Harbin Engineering University, 2007.[In Chinese]

[11] Xin Jianghui, Li Shunming, Shang Weiyan, Shen Huan.The application of GA-support vector machine in exploration vehicle. Vehicle Power and Propulsion Conference (VPPC 2008) IEEE. Harbin, China, 3~5 Sept. 2008: p 4677592.

[12] Yan qingxin. Ship collision risk evaluation model [J]. Journal of Wuhan University of Technology,2002,26(2):220-222. [In chinese]

[13] Park J. , Kim J.S. ,Park D.LMI. Based design of stabilizing fuzzy controllers for nonlinear systems described by Takagi-Sugeno fuzzy model. Fuzzy Sets and Systems,2001, 122:73-82.

[14] Wai R.J. Robust fuzzy neural network control for linear ceramic motor drive via backstepping design technique. IEEE Transactions on Fuzzy Systems, 2002,10(1):102-112.

[15] F. Neri, "Traffic packet based intrusion detection: decision trees and genetic based learning evaluation" WSEAS Transactions on Computers, 4(9), 2005, pp. 1017-1024.

[16] F. Kimura,S. Wang, T. Tsuchiya. Fuzzy Modeling Using A Distance-type Fuzzy Reasoning Method. Memoirs of the Tomakomai National College of Technology.2002:81-86.

[17] X. Zhang, F. Mizuguchi, M. Mizumoto etal. Fuzzy Discriminant Analysis by Simplified Distance-type Reasoning Method. Fa ji Shisutemu Shinpojiumu Koen Ronbunshu.2006,22:6F2-1.

[18] J. P. Hwang, E. Kim. Robust tracking control of an electrically driven robot: Adaptive fuzzy logic approach. IEEE Transactions on Fuzzy Systems.2006,14(2):232-247.