WP5-5 5:40

Proceedings of the 1999 IEEE
International Conference on Control Applications
Kohala Coast-Island of Hawai'i, Hawai'i, USA • August 22-27, 1999

Parallel Parking a Car-Like Robot Using Fuzzy Gain Scheduling[1]

Kuang-Yow Lian[2], Chian-Song Chiu, and Tung-Sheng Chiang
Department of Electrical Engineering
Chung-Yuan Christian University
Chung-Li 32023, Taiwan

***Abstract*—In this study, a fuzzy gain scheduling controller is proposed to parallel park the car-like robot. At first, a fuzzy sliding mode controller (FSMC) embedded by driving experience is developed to locally track a most typical path for the parallel parking. In order to extend the controlled region, several typical paths are formed and can be pieced together to constitute a complete parking path in a large region. Then, a fuzzy gain scheduler with knowledge-based structure is proposed to decide the best parking path and to generate the proper control gains.**

***Index Terms*—Mobile robots, fuzzy gain scheduling, FSMC.**

I. INTRODUCTION

Parking problem of a car-like robot has been investigated immensely throughout the past few years. Many researches on this topic have been presented based on various control strategies, such as using fuzzy logic control (FLC) [1, 10], adaptive fuzzy control [4] and several neural network methods [8], with each having its own advantages. Most of the works have been done for rear parking a car based on a simplified model of steering a car. In this study, a more precise and complete model is used to well depict the kinematic nonholonomic constraints during the parking procedure and dynamic characteristics between the control force and velocity. The motion of a car-like robot is constrained by a set of non-integral kinematic equations and thus the steering model is a nonholonomic system. In [5, 7], the path programming methods using sinusoidal signals have been proposed for the motion planning of nonholonomic systems. Inspired by these methods, the typical paths for the dynamic parallel parking problem are obtained by using sinusoidal driving signals. Our goal is to take the predetermined path as a reference path for a path tracking controller. A fuzzy sliding mode controller (FSMC) including driving experience is proposed to solve the path tracking problem. The method using FSMC is to combine merits of both FLC and classical sliding-mode controllers (SMC). We introduce FSMC, where the FLC will adjust the magnitude of the control force and SMC will determine the sign of the control. The proposed controller generally leads to a suitable transient response and a good stability result. The path tracking controller, however, only suitably works in a local region. From experience, steering a car is confined with the conditions of the car's own capability of mechanism and the environment of parking space. Therefore it is usually impossible to park the car in one try. In many cases, moving forward and backward for several times is needed to complete the parking process. Due to this reason, we do not expect to find a continuous global controller to perform parking for general initial states. Alternatively, how to integrate several local controllers, which work well in a respected region, as a whole controller is our objective.

For the control design of using a conventional gain scheduler [2, 3], the reference trajectory must be slowly time-varying and the linearized system for an operating point must be controllable (or at least stabilizable). These two conditions, however, are restrict to the parallel parking problem considered here. Specifically, the linearized systems for the steering model is uncontrollable. This property is a common character of nonholonomic control systems. In this study, a new fuzzy gain scheduling controller is integrated by a family of local nonlinear controllers, whereby, as a by-product of this structure, the slowly time-varying constraint can be relaxed to a satisfied level. In this extended gain scheduler, the reference path mentioned above is regarded as the gain scheduling path when compared to the operational point in the typical gain scheduler. Accordingly, the reference paths are regarded as the extended gain scheduling variables. Moreover, based on the time-frozen concept, a zero-order hold sampler is introduced to transmit the fuzzy gain scheduler with the measured position coordinates at each sampling period. Once the sampling period can match the time response of the path tracking controller, then the fuzzy gain scheduler can be extended to our interesting workspace to carry out parallel parking.

II. DYNAMIC MODEL OF THE CAR-LIKE ROBOT

Consider a car-like robot in a neighborhood of a parking space, as shown in Fig. 1. For simplicity of modeling the car-like robot, the front and rear pairs of wheels are modeled as single wheels at the midpoints of the axles. The front wheel can be steered about the vertical axis to some reasonable degree while the rear wheel orientation is fixed. The configuration variable (x, y, θ, ϕ) are defined in Fig. 1. Thus the speed of the car, v, can be expressed as $(\dot{x}^2+\dot{y}^2)^{1/2}$. By allowing the wheels to roll without slipping on the road, then it is obtained the following kinematic constraints:

$$\dot{x}\sin(\theta+\phi)-\dot{y}\cos(\theta+\phi)-l\dot{\theta}\cos\phi=0\,; \tag{1}$$

$$\dot{x}\sin\theta-\dot{y}\cos\theta=0\,. \tag{2}$$

The kinematics (1) and (2) are not integrable and thus is a type of nonholonomic constraint. Here it is considered in this paper to take $\tan\phi$ as the control input u_1 and the driving force (or torque) u_2 is used to control the speed. For the sake of ride quality and handling performance, we also considering the effect due to road disturbance, like road bumps, and some kind of model uncertainty rising from complex mechanism. To model these situations in a simplified method, two external disturbances $w_1(v,t)$ and $w_2(v,t)$ are, respectively, used to specify the disturbance existing in the driving dynamics and steering kinematics. Then the control system for the car-like robot considered in this study is assumed with the following form:

$$\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{\theta}\\ \dot{v}\end{bmatrix}=\begin{bmatrix}v\cos\theta\\ v\sin\theta\\ 0\\ 0\end{bmatrix}+\begin{bmatrix}0\\ 0\\ \frac{1}{l}v(u_1+w_1(v,t))\\ u_2+w_2(v,t)\end{bmatrix}. \tag{3}$$

This nonholonomic control system not only exhibits the system kinematics but also the dynamic characteristics between the control force and the speed response. Here and through the end of this paper, the specifications of the car-like robot are set as follows: the length is 3 M; the width is 1.5 M; and the distance between the wheels is l=2 M. Reasonably, the magnitudes of u_1 and u_2 are required to satisfy saturation constraints, which are assumed to be the limitation as below:

---

[1]This work was supported by the National Science Council, R.O.C., under Grant NSC-88-2213-E-033-027.
[2]Correspondence addressee

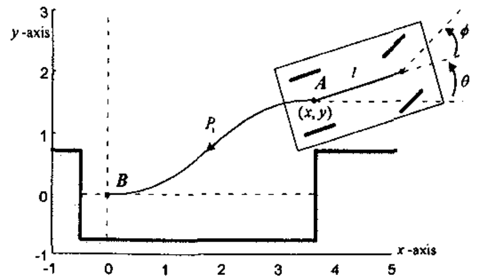

Fig. 1 A typical parking space and parallel parking path for a car-like robot.

$$|u_1| \le 1 \text{ (or } |\phi| \le 45^0\text{) and } |u_2| \le 5\ m/s^2 . \tag{4}$$

III. LOCAL TRACKING FOR A CHOSEN PATH

In this section, the most common parallel parking problem is considered as shown in Fig. 1, where the car-like robot will be parked in one try. For this purpose, a typical parallel parking path will be specified and the associated controller will be addressed.

A. A Typical Parallel Parking Path

Consider the parallel parking problem illustrated in Fig. 1. It is assumed that point A is the initial position of movement for the rear wheel. The goal of this problem is to drive the rear wheel to move from point A to point B, i.e. $(x, y) = (0, 0)$, and to satisfy $v(t_f) = \theta(t_f) = 0$, by following a path like P_1 shown in Fig. 1. From parallel parking experiences the path P_1 is a typical path for most drivers. To find this reference path, let us consider an ideal steering model:

$$\begin{bmatrix} \dot{x}_d \\ \dot{y}_d \\ \dot{\theta}_d \\ \dot{v}_d \end{bmatrix} = \begin{bmatrix} v_d \cos\theta_d \\ v_d \sin\theta_d \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{l} v_d r_1 \\ r_2 \end{bmatrix}, \tag{5}$$

where r_1 and r_2 are called as reference inputs. The ideal model for the car-like robot will be assumed with zero initial speed and steering angle. Inspired by the driving experience and research results in the nonholonomic kinematics [7], we first consider a type of inputs such as $r_1 = -k_1 \sin 2\pi(t - t_0)/T$, $r_2 = k_2 \cos \pi(t - t_0)/T$ to generate a suitable reference path similar to P_1. Using the sinusoid signals as the inputs has the following properties:

Property 1: The car-like robot can be steered to an arbitrary configuration by repeatedly using $r_1 = -k_1 \sin 2\pi(t - t_0)/T$, and $r_2 = k_2 \cos \pi(t - t_0)/T$ with suitable values of k_1, k_2 and T.

Property 2: Let the reference inputs be the sinusoidal signals in Property 1. If the angle $\theta_d(t)$ could be kept small, then the solution of x_d and y_d at $t_f = t_0 + T$ can be approximated by $x_d(t_f) \approx x_d(t_0) + \frac{2k_2T^2}{\pi^2}(1 - \frac{3k_1^2k_2^2T^4}{32l^2\pi^4})$ and $y_d(t_f) \approx y_d(t_0) - \frac{k_1k_2^2T^4}{4l\pi^3}$.

The first property is concerned with the controllability for nonholonomic systems and lies in the issue of motion planning. We refer it to [7]. The second property can be verified by a direct integrating technique. These properties will be exploited to generate the reference paths, although the paths may not be optimal. By Property 1 and 2, one can suitably assign k_1, k_2 and T to obtain a smooth parking path, whereas both the mechanical constraints (4) and obstacle avoidance are satisfied.

B. Fuzzy Sliding Mode Controller for Path Tracking

In this subsection a nonlinear tracking control scheme shall be proposed. This scheme drives the car-like robot near point A to track the reference path P_1, thus also fulfilling the parking goal. By virtue of practical driving experience, one can derive the following experience rules:

***If** the distance between car and parking space is large,*
***then** the front wheels at big angle and increasing driving force. Or,*
***If** the distance between car and parking space is small,*
***then** the front wheels at small angle and decreasing driving force, etc.* (6)

To include these precious experiences in the controller, a fuzzy logic controller (FLC) with linguistic rules is naturally adopted and will be implemented as the core of the path tracking controller. Moreover, combining sliding mode control (SMC) to enhance robustness, the proposed controller structure is an SMC-type control law whose magnitude is adjusted by using FLC. This controller will be called the fuzzy sliding mode controller (FSMC).

Let $q = [x \quad y]^T$ and $q_d = [x_d \quad y_d]^T$ be the position coordinate of the car-like robot and denote the reference path, respectively. The position tracking error is thus defined as $\tilde{q} = q - q_d$. We also define an error measure $s = \dot{\tilde{q}} + \lambda\tilde{q} = \dot{q} - q_r \in R^{2\times1}$ with positive constant λ, where q_r, an auxiliary signal vector, is denoted by $q_r = \dot{q}_d - \lambda(q - q_d)$. Based on this setting, the motion tracking can be insured once the error measure s can be steered to zero. To obtain the control law satisfying this condition, we first differentiate the error measure and have:

$$\dot{s} = H\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} + \dot{q}_r + W, \tag{7}$$

where $H \equiv \begin{bmatrix} -(v^2/l)\sin\theta & \cos\theta \\ (v^2/l)\cos\theta & \sin\theta \end{bmatrix}$ and $W \equiv H[w_1 \quad w_2]^T$. If the matrix H is invertible, the control law can be set in the form:

$$u \equiv \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = -H^{-1}\left(K\frac{s}{\|s\|} + \dot{q}_r\right). \tag{8}$$

The control gain K is obtained by letting $K = amp \cdot abs(Kfuzzy)$, where *Kfuzzy* is the inferential result of FLC, *amp* denotes the scaling factor from the universal set to the operating domain and *abs* stands for the absolute value. By substituting (8) into (7), the resulting error equation of the closed-loop system can be written as

$$\dot{s} = -K\frac{s}{\|s\|} + W. \tag{9}$$

The stability of the closed-loop system can be easily shown by choosing the Lyapunov function candidate as $V = \frac{1}{2}s^Ts$. By applying error equation (9), the time derivative of V leads to

$$\dot{V} = s^T\dot{s} = s^T\left(-K\frac{s}{\|s\|} + W\right) = -K\|s\| + s^TW.$$

To guarantee that the states of the system move to the stable surface with $s(t) = 0$, the magnitude of K is requested to satisfy the following inequality:

$$K \ge (\|W\| + \eta). \tag{10}$$

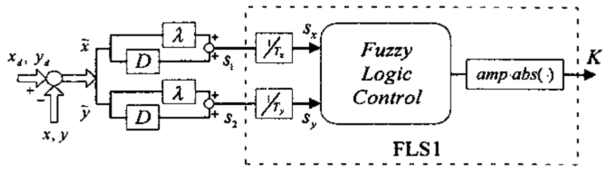

Fig. 2 The fuzzy sliding mode controller.

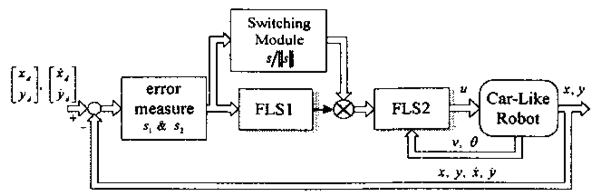

Fig. 3 Entire system configuration of the FSMC.

Thus, we obtain $\dot{V} \le -\eta\|s\|$. This means the error measure will exponentially converge to zero and thus the robot's trajectory and its velocity will move towards the desired trajectory and desired velocity, respectively. Moreover, $\theta(t_f)$ will approach to $\theta_d(t_f)$ because $\theta = \tan^{-1}(\dot{y}/\dot{x})$ and $\theta_d = \tan^{-1}(\dot{y}_d/\dot{x}_d)$. The stability results are summarized in the following property:

Property 3: Consider the car-like robot system (3) with the control law (8). Suppose that during the operating interval the matrix H is invertible, also the driving constraint (4) and the inequality on the control gain are satisfied. Then the states of the closed-loop system $q(t)$, $\dot{q}(t)$, $\theta(t)$ will asymptotically track $q_d(t)$, $\dot{q}_d(t)$ and $\theta_d(t)$, respectively.

Notice that Property 3 are sustained only if (i) the mechanical constraints (4); (ii) the condition of control gain (10); and (iii) the boundedness of H^{-1} are all satisfied. For the sake of condition (i), the path tracking controller only works in a local region where the initial configurations of the car-like robot lies near the start point of the reference path. To cope with this constraint, we introduce a fuzzy gain scheduling structure such that the parallel parking can be carried out in a large region. The detailed description is addressed in Section 4. The strategies for dealing with Conditions (ii) and (iii) are described in the following paragraphs, respectively.

Strategy for tuning the control gain : From (10), the right hand side composes of the system disturbances and partial states, thus in turn makes it difficult to choose an appropriate magnitude of K. Therefore, an FLC called FLS1 is applied to generate the gain K in (8). The gain tuned by FLC can have a better transient response. The structure of the fuzzy logic controller is sketched in Fig. 2, where where s_1 and s_2 are the error measure of $\tilde{x}$ and $\tilde{y}$; T_x and T_y are the scale parameters of error measure s_1 and s_2; s_x and s_y are scaled error measure, respectively.

Strategy for confirming the boundedness of H^{-1} : Notice that if instant velocity is zero then the matrix H is noninvertible. To render the singularity problem, we can use T.-S-type [3, ,10, 11] fuzzy model $\overline{H}$ instead of using H^{-1}. The fuzzy model $\overline{H}$ is implemented by the following fuzzy rules:

$$\text{If } v \text{ is } Zero\text{, then } \overline{H} = \begin{bmatrix} -\xi & \xi \\ \cos\theta & \sin\theta \end{bmatrix};$$

$$\text{If } v \text{ is } Not\ Zero\text{, then } \overline{H} = H^{-1},$$

where "*Zero*" is a fuzzy set defined in the neighborhood of v=0 and the region is specified according to the performance requirement ; "*Not Zero*" is the reciprocal fuzzy set of *Zero* ; and ξ is the factor of the maximum steering angle. The fuzzy logic system will be called as FLS2.

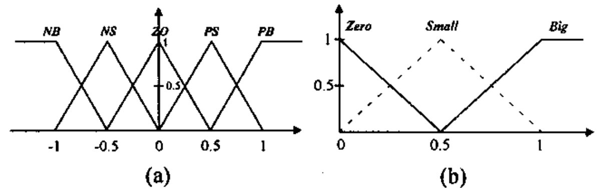

Fig. 4 (a) Membership functions for F_1^r and F_2^r. (b) Membership function for E^r.

s_x \ s_y	NB	NS	ZO	PS	PB
NB	Big	Small	Small	Small	Big
NS	Big	Small	Zero	Small	Small
ZO	Small	Small	Zero	Small	Small
PS	Small	Small	Zero	Small	Big
PB	Big	Small	Small	Small	Big

Table 1 Fuzzy rulebase of FSMC.

Based on the design concept mentioned above, the entire system configuration of the FSMC is sketched in Fig. 3, where the magnitude of the control gain, K, is determined by FLS1 and the sign of the control force is decided by a switching module $s/\|s\|$.

C. *Fuzzy Realization for a Path Tracking Controller*

To establish a knowledge base in FLS1, the linguistic variables in (6) must be transformed to the new fuzzified variables, s_x and s_y, and the result of inference is the magnitude of the control gain. The fuzzy rule-base for the FLS1 is composed by the following fuzzy rules, for $r = 1, 2, \cdots, m$:

$$R^r: IF\ (s_x \text{ is } F_1^r) \text{ and } (s_y \text{ is } F_2^r) \text{ THEN } (Kfuzzy \text{ is } E^r).$$

where F_1^r, F_2^r and E^r are fuzzy sets, and m denotes the total number of rules. According to the driving experience for a parallel parking process, one can obtain many linguistic rules like (6). Therefore, one can possibly conclude five common rules:

Rule1 : ***IF*** *s_x is positive big and s_y is negative big* ***THEN*** *Kfuzzy is big.*

Rule2 : ***IF*** *s_x is positive big and s_y is small* ***THEN*** *Kfuzzy is small.*

Rule3 : ***IF*** *s_x is negative big and s_y is small* ***THEN*** *Kfuzzy is small.*

Rule4 : ***IF*** *s_x is small and s_y is zero* ***THEN*** *Kfuzzy is zero.*

Rule5 : ***IF*** *s_x is zero and s_y is zero* ***THEN*** *Kfuzzy is zero.*

Since the sign of the input for FLS2 is determined by the switching module, and only the magnitude of the input for FLS2 is concerned in FLS1, all fuzzy sets in the *THEN* part are set with positive values. This characteristic is to our benefits when constructing fuzzy rules. Table 1 contains of the overall 25 rules (i.e., $m = 25$), which is expanded from the five basic rules mentioned above. It shows the fuzzy rule base composed of linguistic terms such as zero (*ZO*), positive small (*PS*), negative small (*NS*), positive big (*PB*) and negative big (*NB*). These membership functions for F_1^r, F_2^r and E^r are shown in

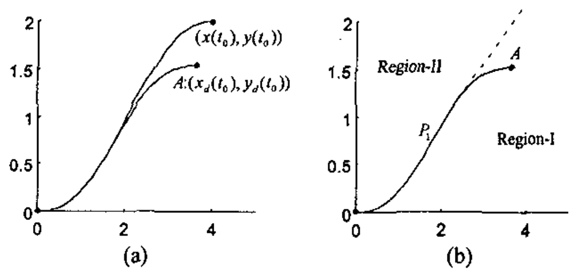

Fig. 5 (a) The result of path tracking. (b) The good and poor tracking areas.

Fig. 4, respectively. The fuzzy logic system is with singleton fuzzifier, product inference, and it is proper in this situation to use the method of max-product composition with center of gravity defuzzification.

D. Simulation for Tracking Path P_1

As the controller structure shown in Fig. 2 and 3, we now consider the path tracking for P_1 by applying FSMC. The simulation is carried out by setting the initial states of the car-like robot at $(x(t_0), y(t_0), \theta(t_0), v(t_0)) = (4,\ 2,\ 0,\ 0)$ and parameters of the path tracking controller at *amp*=15, $\lambda = 4$, and $(T_x, T_y) = (1,\ 1.5)$. This set of control parameters can achieve satisfactory results as shown in Fig. 5(a). Based on the same setting, the simulation also reveals some properties, which are helpful in choosing the control parameter *amp*. First, if the magnitude of *amp* is larger, then it tracks a path faster, and also the error is much smaller. However, poor tracking will occur if the initial position of the car-like robot is not near the reference path P_1. This is due to the restrictions $\text{Max}\,|u_2| \le 5$ and $\text{Max}\,|u_1| \le 1$. Therefore, a suitable value of choosing *amp* is important in obtaining good tracking results. Another important conclusion is roughly made based on region-I and region-II sketched in Fig. 5(b), which are separated by the dotted line. Generally, the initial states lying in area-II results in a poor tracking, whereas the initial states in area-I is with suitable satisfactory tracking results. To overcome the restriction and obtain a good tracking in a large region, a fuzzy gain scheduler with knowledge-based structure will be introduced in next section.

IV. Fuzzy Gain Scheduling

Due to the nonholonomic constraints and mechanical constraints, it is often necessary to steer the car-like robot back and forth to complete the parallel parking procedure. In light of this concept, several typical paths will be generated for specific regions in the workspace. These paths will be considered as the reference paths for local path tracking controllers. If the controller corresponded to each reference path has suitable control parameters, then the parking objective will be achieved by programming suitable reference paths in the parking procedure. Thus the parking control can be regarded as an integrated problem of path programming and control gain scheduling.

A. Reference Paths in a Larger Area

The workspace is assumed in the area of $\{(x, y, \theta, v) |\ -3 \le x \le 9,\ 0 \le y \le 7,\ -45^0 \le \theta \le 45^0,\ -5 \le v \le 5\}$, where the environmental obstacle must also be taken into consideration. To deal with the parking problem for the cases of arbitrary initial states in the workspace, we will design a family of reference paths which can

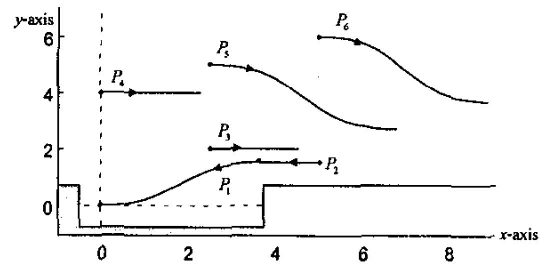

Fig. 6 Some typical reference paths.

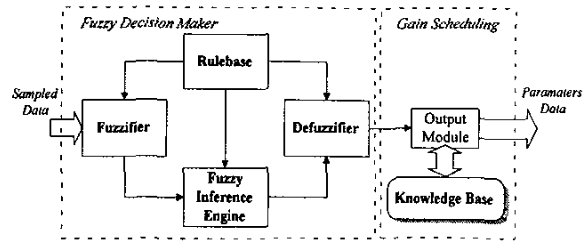

Fig. 7 The structure of the fuzzy decision maker and gain scheduling.

be pieced together into a complete parking path beginning at the initial states and ending at the parking space. Using the tracking properties of the path tracking controller mentioned in Section 3, it is possible to decide the location of each reference path. These paths can be generated by specifying the magnitude of reference inputs and the initial states. Here the parameters and initial condition of each reference path will be chosen accordingly to different situations. For convenience, we denote the i-th path in the form:

$$P_i : \begin{bmatrix} k_{1i} & k_{2i} & x_{di}(t_0) & y_{di}(t_0) \end{bmatrix},\ i=0,\dots,6,$$

where k_{1i} and k_{2i} are the magnitude of reference inputs, r_{1i} and r_{2i}, respectively. The initial position for each path is denoted by $x_{di}(t_0)$ and $y_{di}(t_0)$, whereas the initial velocity and the steering angle are commonly set at zero and are neglected in expressing P_i. Some of the paths are sketched in Fig. 6. Notice that P_5, P_6 denote the paths of driving the car-like robot towards front right. This is the case when the lateral distance to the parking space becomes too large when following path P_1. Additionally, P_2, P_3, and P_4, which can be generated by setting k_1 as zero, result in the car-like robot to either move forward or backward. These paths are followed in advance if the initial states become of a greater distance than the parking space. According to Properties 1 and 2, the family of the reference paths may be constituted by the following path parameter vectors:

$$\begin{aligned}
P_0 &: [0 \quad 0 \quad 0 \quad 0];\\
P_1 &: [1 \quad -1.25 \quad 3.64 \quad 1.53];\\
P_2 &: [0 \quad -0.42 \quad 5 \quad 1.5];\\
P_3 &: [0 \quad 0.625 \quad 2.5 \quad 2];\\
P_4 &: [0 \quad 0.7 \quad 0 \quad 4];\\
P_5 &: [1 \quad 1.56 \quad 2.5 \quad 5];\\
P_6 &: [1 \quad 1.56 \quad 5 \quad 6],
\end{aligned}$$

where path P_0, chosen as the desired parking point, will figure a motion to stop in parking procedure. Strictly, P_0 is not a path, it is more like an operational point. In other words, the goal of tracking this path is a set-point control.

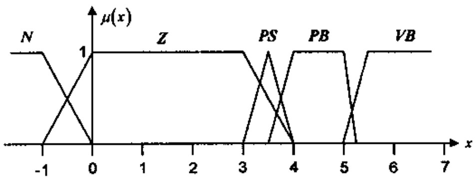

Fig. 8 The membership function for x_f^h.

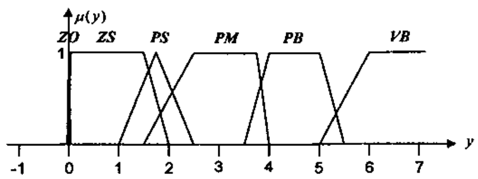

Fig. 9 The membership function for y_f^h.

In the case of tracking P_1, the nonlinear controller FSMC can steer the car-like robot close to the nearby reference path satisfactorily. Accordingly, one can choose a proper reference path and then obtain the corresponding control parameters after a series of tuning. In our research, each path tracked is terminated at the end of a time interval T. Thus the parallel parking goal will be completed by repeatedly and correctly switching the path to be tracked at the end of each time interval. How to properly choose reference paths and piece them together will be made by a fuzzy gain scheduler addressed in the next subsection.

B. Fuzzy Decision Maker

According to different initial positions of a car-like robot at every interval, the gain scheduler, designed later, will determine a proper reference path and give proper corresponding control parameters for the path tracking controller. If the initial states are far from the parking space, our strategy is to steer the car-like robot to approach the location A (the starting point of P_1) at the first stage of the parking procedure. Then, one can follow path P_1 to back the car-like robot into the parking space and, finally, stop there by following path P_0. This means that according to different time and different positions the corresponding proper reference path and the associated control parameters should be determined.

The structure of fuzzy decision maker and a gain scheduling module is shown in Fig. 7. The fuzzy decision maker exploits the fuzzy rule base and fuzzy reasoning to decide a reference path P_i, which accompanied with control parameters is then fed to the path tracking controller. The control parameters generated by the knowledge base consist of the concerned data of the chosen path and the corresponding control gains. We first describe the structure of the fuzzy decision maker in detail.

Based on the typical concept of gain scheduling variables, each operated reference path denoted by P_i will be considered as the extended gain scheduling variable, namely, a *gain scheduling path*. The idea of fuzzy cluster is facilitated in programming the gain scheduling path. According to Fig. 6, the workspace is classified into several fuzzy regions. In x-axis, there are five fuzzy sets shown in Fig. 8. Similarly, the y-axis are divided into six fuzzy sets defined in Fig. 9. Because path P_1 effects the control performance the most region near the initial position of P_1 is partitioned down to smaller regions.

The fuzzy rulebase of the maker is devoted to determine a proper path P_i among the path family. Fuzzy rules provide a natural framework for characterization of human behavior and decision analysis. In fuzzy decision maker of this scheduler, the fuzzy rulebase consists of a collection of *IF-THEN* rules, for h=1, 2,..., 26:

$$R^h:\ IF\ x(t_s)\ is\ x_f^h\ and\ y(t_s)\ is\ y_f^h\ THEN\ \hat{P}\ is\ P^h,$$

where $(x(t_s), y(t_s))$ denotes the sampling position of the car-like robot at the sampling time $t_s = t_0 + T\cdot k$, k=0, 1, 2...; $\hat{P}$ is the output of the fuzzy decision maker; the fuzzy sets for x_f^h, y_f^h are defined in Fig. 8 and 9, respectively; and P^h is the fuzzy set denoting the chosen reference path from the inferential result of the h-th rule. Since fuzzy *IF-THEN* rules are used to determine a proper path to track, the linguistic variable in the *THEN* part actually denotes the path number. This is shown as singleton membership functions in Fig. 10. There exists 7 reference paths labeled as fuzzy sets $P_0 \sim P_6$ and the universe of the their own discourses is denoted at 0~6.

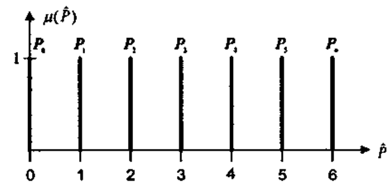

Fig. 10 The membership function for P^h.

P_r (x \ y)	ZO	ZS	PS	PM	PB	VB
N		P_3	P_3	P_4	P_4	P_4
Z	P_0	P_3	P_3	P_5	P_5	P_6
PS		P_1	P_3	P_5	P_6	P_6
PB		P_1	P_1	P_5	P_6	P_6
VB		P_2	P_2	P_2	P_6	P_6

Table 2 The rulebase of the fuzzy decision maker.

To judge whether the parking process has been done, the following stopping rule is designed:

$$R^{26}:\ IF\ x(t_s)\ is\ Z\ and\ y(t_s)\ is\ ZO\ THEN\ \hat{P}\ is\ P_0,$$

where the fuzzy set ZO of y_f^h is defined as

$$\mu(y) = \begin{cases} 20(y+0.05) & -0.05 \le y \le 0 \\ 20(0.05-y) & 0 \le y \le 0.05 \\ 0 & otherwise \end{cases}$$

When the stopping rule is fired, the control goal begins to enter the stopping motion, as to steer the car-like robot to a more exact position and then turn off the control inputs.

Combining the emphasis mentioned above, the fuzzy rulebase can be constructed in Table 2. Based on the position of the car-like robot, each rule fires an operational reference path whose starting point lies in the nearest neighborhood. The operation method of the fuzzy decision maker uses the Max-Min fuzzy inference and the result is obtained by choosing the Max fuzzy value in the defuzzifier. Then the output of the fuzzy inference is the number of the selected reference path.

C. Fuzzy scheduling

The gain scheduling works as a coordinator to transfer the control parameters into the path tracking controller. The control parameters for each gain scheduling path are previously obtained by adjusting the parameters to satisfy the needed tracking performance in the near partition area. The control parameters consist of the following data:

$$SP_i = \begin{bmatrix} amp_i & \lambda_i & T_{xi} & T_{yi} \end{bmatrix},\ i\text{=0, 1,…6.}$$

Each gain scheduling path P_i will be accompanied with corresponding control parameters vector SP_i in itself working region. Then the parameter vector SP_i can be regarded as a *scheduling parameter* (or *scheduling gain*). By careful test in the simulation, the scheduling parameter vectors can be properly chosen as follows:

$$\begin{aligned}
&SP_0: [3 \quad 4 \quad 1 \quad 20];\\
&SP_1: [15 \quad 4 \quad 1 \quad 1.5];\\
&SP_2: [5 \quad 2 \quad 5 \quad 1];\\
&SP_3: [8 \quad 2 \quad 5 \quad 2];\\
&SP_4: [10 \quad 3 \quad 10 \quad 2];\\
&SP_5: [5 \quad 4 \quad 10 \quad 2];\\
&SP_6: [5 \quad 3 \quad 5 \quad 2].
\end{aligned}$$

Notice that because P_1 is the most important path in this parking system, the path tracking controller for reference path P_1 may be designed to have a larger *amp*. This implies the system to have better tracking performance and enhance the stability of this path tracking. Conversely, the other reference paths may be designed to have smaller *amp*, this will in turn enlarge the area that is suitable for parking process.

The parameter vectors of the gain scheduling paths and the scheduling parameters construct the knowledge base of the scheduler. Combined with the fuzzy decision maker, this structure is called *fuzzy gain scheduler with knowledge-based structure* as shown in Fig. 7. The overall system including the car-like robot and fuzzy gain scheduler are shown in Fig. 11.

When the i-th reference path is chosen, the gain scheduling will transfer the parameter vector of i-th gain scheduling path P_i and corresponding scheduling parameters SP_i from the knowledge base to the path tracking controller. Then the path tracking controller begins to work based on the obtained data. At the same time, the scheduler holds on the parameter vectors P_i and SP_i until the selected path tracking is accomplished. This means that the needed parameters of path tracking controller does not change until the next sampling period. To this end, a zero-order-hold sampler with the same sampling period as the cost time of one reference path, i.e. T=4 sec, is adopted. The sampler sends the measured data to the fuzzy decision maker and keeps them fixed at a period interval. Thus, the scheduling variables and, in turn, the path tracking controller are kept fixed in the sampling period. If the parking goal is not achieved when the current path ends, then the scheduler will continue to decide next proper path once again. This procedure, will be executed over and over until the parking goal is achieved.

V. Simulation Results

The simulation is carried out based on the control structure sketched in Fig.11. It is noted that the reference path is dynamically generated. Therefore when a new reference path is chosen, the initial state values for the reference model must be also set as new data to make sure that the reference path can be correctly generated. Moreover, to avoid the violent change of the control signal which occurs when the selected forward path and backward path are switched, the real control inputs replacing u_1, u_2 by $u_1/(1+\tau_1 p)$, $u_2/(1+\tau_2 p)$ respectively, where setting $\tau_1 = \tau_2 = 0.02$, is to smoothen the control inputs. Consider the proposed path tracking controller FSMC (c.f. Fig. 3) and using the fuzzy gain scheduling structure in Fig. 11, the simulation results are shown in Fig. 12 with initial states set at

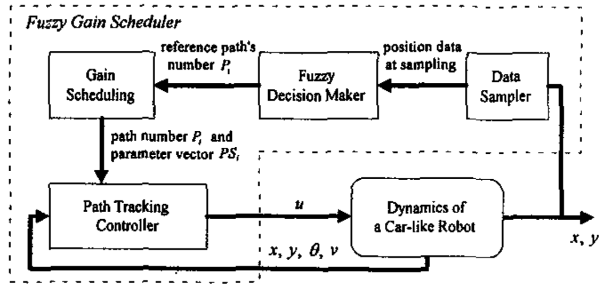

Fig. 11 The overall system of the fuzzy gain scheduler and a car-like robot.

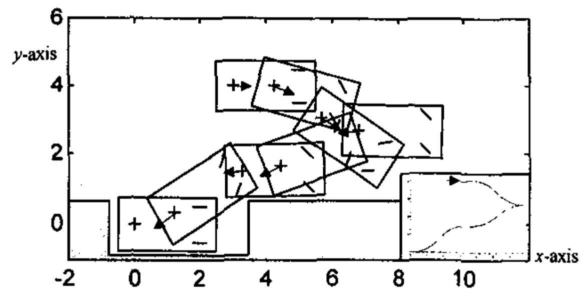

Fig. 12 The trajectory of the car-like robot. ' + ' is the center of the rear axis of the robot; '⟶' is the direction of the motion.

$(x, y, \theta, v) = (3, 4, 0, 0)$ and the unknown external disturbances are supposed as $w_1 = 0.02v\sin 5t$ and $w_2 = 0.2v\cos 8t$, respectively. Others results are not included here due to space limitation. The satisfactory results demonstrate the validity of using the proposed fuzzy gain scheduling structure on parallel parking process.

References

[1] C. von Altrock, B. Krause and H.J. Zimmermann, "Advanced fuzzy logic control of a model car in extreme situation," *Fuzzy Sets and Systems*, vol. 48, pp. 41-52, 1992.

[2] P. Apkarian and P. Gahinet, "A convex characterization of gain scheduled H^∞ controllers," *IEEE Trans. on Automatic Control*, Vol. 40, pp.853-864, May 1995.

[3] D. Driankov, R. Palm and U. Rehfuess, "A Takagi-Sugeno fuzzy gain scheduler," *Fuzzy-IEEE*'96, New Orleans Sept. 8-11, pp. 1053-1059, 1996.

[4] S. G. Kong and B. Kosko, "Adaptive fuzzy systems for backing up a truck-and-trailer," *IEEE Trans. Neural Networks*, vol. 3, pp. 211-223, 1992.

[5] J. C. Latombe. *Robot Motion Planning*, Kluwer, 1991.

[6] C. C. Lee, "Fuzzy logic in control system: fuzzy logic controller-part I & II," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-20, pp. 404-435, 1991.

[7] R. M. Murray and S. S. Sastry, "Nonholonomic motion planning: steering using sinusoids," *IEEE Trans. Automatic Control*, vol. 38, pp. 700-716, 1993.

[8] D. Nguyen and B. Widrow, "The truck back-upper: an example of self-learning in neural networks," *International Joint Conference on Neural Networks*, pp. 357-363, 1989.

[9] W. Pedrycz, *Fuzzy Control and Fuzzy Systems*, Wiley: New York, 1989.

[10] M. Sugeno and M. Nishida, "Fuzzy control of a model car," *Fuzzy Sets and Systems*, vol. 16, pp. 103-113, 1985.

[11] K. Tanaka and M. Sano, "Trajectory stabilization of a model car via fuzzy control," *Fuzzy Sets and Systems*, vol. 70, pp. 155-170, 1995.