Journal of Intelligent and Robotic Systems **34:** 175–194, 2002.

Design and Implementation of Fuzzy Parallel-Parking Control for a Car-Type Mobile Robot

SHIH-JIE CHANG and TZUU-HSENG S. LI ⋆
IC^{2}S Laboratory, Department of Electrical Engineering, National Cheng-Kung University, Tainan 70101, Taiwan, R.O.C.; e-mail: thsli@mail.ncku.edu.tw

(Received: 15 March 2001; in final form: 21 August 2001)

Abstract. The theme of this paper is to design and implement a car-type mobile robot (CTMR) that possesses autonomous parallel-parking capability. At first, we introduce the hardware architecture of the CTMR, which consists the robot mechanism, microcomputer part, electronic driver, and sensor. Two fuzzy parallel-parking controls (FPPC), the backward and forward parallel parking, are provided to maneuver the steering angle of the CTMR. Computer simulations are used to illustrate the effectiveness of the developed FPPC schemes. For real-time implementations, we utilize two FPPC methods that we proposed in simulation to back-drive or head-in the CTMR to the parking lot. Both simulation results and real-time experiments demonstrate the feasibility and effectiveness of the proposed intelligent parallel-parking control system.

Key words: fuzzy logic control, parallel parking, car-type mobile robot, real-time implementation.

1. Introduction

Design and implementation of intelligent parallel-parking or garage-parking techniques for automobiles have become an appealing challenge for both academics and industries worldwide. The CTMR is one of the most commonly used nonholonomic mobile robots to emulate the motion of a rear car. Parallel-parking control of the CTMR can be treated as one of the motion planning task [15, 16] of autonomous mobile robots.

In recent years, parallel-parking problems of CTMR have been rapidly attracted interest in [7–10, 12, 13, 18, 19]. Lyon [13] proposes the parallel-parking problem of curvilinear path generation for a car with nonholonomic constraints, where the slope and curvature constraints are used to derive a fifth-order polynomial equation that is denoted as the parallel-parking path. The relationship between the rear-path curvature and the steering angle is also developed. Ohkita et al. [18] derive fuzzy rules for traveling control of an autonomous mobile robot using DREAM-1 with six supersonic sensors, where three sorts of situations and their corresponding fuzzy control rules in flush parking are examined.

⋆ Author for correspondence.

Laumond et al. [9] present a complete motion planning algorithm, which includes three stages:

(1) plan a path for the geometric system;
(2) perform a subdivision on the path until all endpoints can be linked by optimal feasible paths;
(3) run an "optimization" routine to reduce the length of the path.

The algorithm has been implemented in the parallel-parking control in a complex environment with a number of polygonal obstacles. An iterative parallel-parking algorithm for a nonholonomic vehicle is proposed in [19], where the vehicle follows sinusoidal reference functions. The LIGIER electric autonomous vehicle with ultrasonic sensor system is used to verify the developed control.

Divelbiss and Wen [7] propose a method of trajectory tracking and parallel parking for a car-trailer, where the trajectory is generated off-line by using a path space iterative algorithm and a time-varying linear quadratic regulator is applied to track the trajectory. It also presents parallel parking experiments by utilizing CAT mobile robot, a tractor-trailer, and double tractor-trailer. Leitch and Probert [10] submit a novel and powerful genetic algorithm (GA) to design fuzzy systems. They exploit a new context-dependent coding technique, a chromosome reordering operator, and the coevolution of controller testing sets to improve the conventional GAs in designing controller. The algorithm is applied to parallel parking maneuver for mobile robots.

Jiang and Seneviratne [8] offer an automated parallel parking strategy for a CTMR. The strategy is considered in three phases: scanning phase, positioning phase, and maneuvering phase. The mobile robot follows a path by two circular arcs tangentially connected to each other. The strategy is implemented by modified B12 mobile robot. A fuzzy gain scheduling controller for parallel parking a CTMR is provided by [12], where a fuzzy sliding model controller is firstly proposed to locally track a typical path for the parallel parking. For the nonholonomic constraints and mechanical constraints, several typical paths are formed to complete the parallel parking procedure.

Among these literatures [7–10, 12, 13, 18, 19], one can find that about half the parallel parking are maneuvered by the fuzzy logic control (FLC) scheme. It is well known that the FLC has been successfully applied to numerous industrial appliances, consumer electronics, and autonomous mobile robots [1–6, 14, 17, 20–25, 27]. The decision-making logic (DML) rules of a FLC can be obtained from human experts, learning mechanisms, evolution algorithms [26], or some traditional control theories. In this paper, we will propound two FPPC methods for the CTMR in the relative coordinate systems. The DML are derived by the sliding-mode control and human driving skills.

This paper is organized as follows. Section 2 addresses the setup of the CTMR, where the system architecture of the CTMR includes the robot mechanism, microcomputer part, motor driver, and sensor. In Section 3, we first discuss the model of a CTMR and then investigate a suitable FPPC such that a CTMR can park in the

desired location appropriately and automatically. Computer simulation results are given to show the validity of the proposed FLC algorithms. Two FPPC methods are developed to implement the parallel-parking tasks. The real-time control of the CTMR is presented in Section 4. Section 5 concludes this paper.

2. Setup of the Car-Type Mobile Robot

In order to simulate the parallel parking of a real car, we adopt a 1/10th scale car-type model car on which a microcomputer system is mounted. The appearance of the real autonomous CTMR is shown in Figure 1. Figure 2 illustrates the hardware architecture of the CTMR. The CTMR consists of the following four parts, robot mechanism, microcomputer part, electronic driver, and sensor.

The robot mechanism manufactured by TAMIYA Co., Ltd, possesses car body, driving motor, steering structure, gear trains, and transmission systems. We fit the model car for our special application by modifying some mechanical structures. It is a four-wheeled vehicle with front wheel drive system and front steering wheels just like its full-size counterpart. Four-wheel independent, double wishbone suspension system is damped by large capacity, coil over oil-filled shock absorbers at corners for phenomenal traction. Front sealed gearbox incorporates precision ball type differential gearing for crisp cornering. Because the remote control (RC) model car is manufactured for sporting and racing, the driving motor runs at a high speed but low torque. In this study, the model car carries the CPU board, A/D-D/A card, sensors, and so on. Thus, it is much heavier than the original one. We replace the driving motor by a high torque and low speed DC motor. For details, please refer to [5].

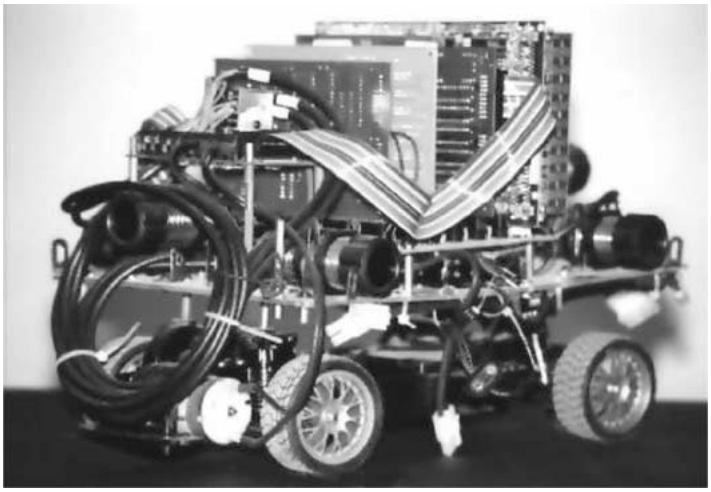

Figure 1. Appearance of the CTMR.

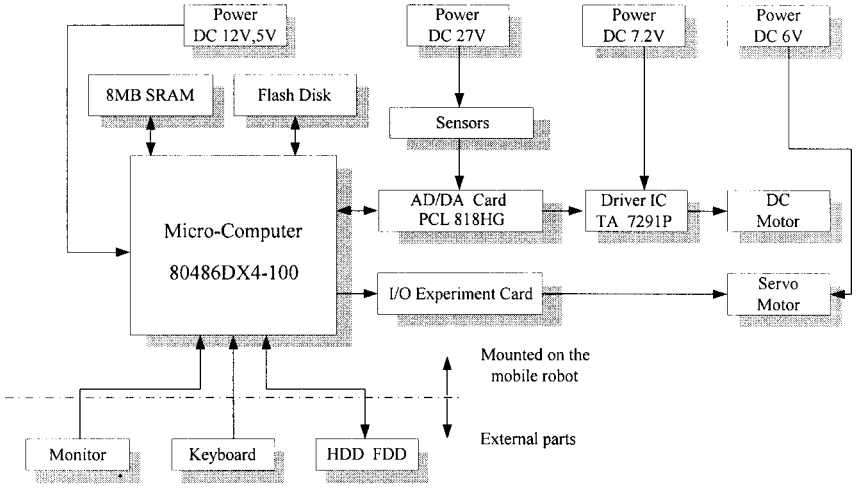

Figure 2. Hardware architecture of the CTMR.

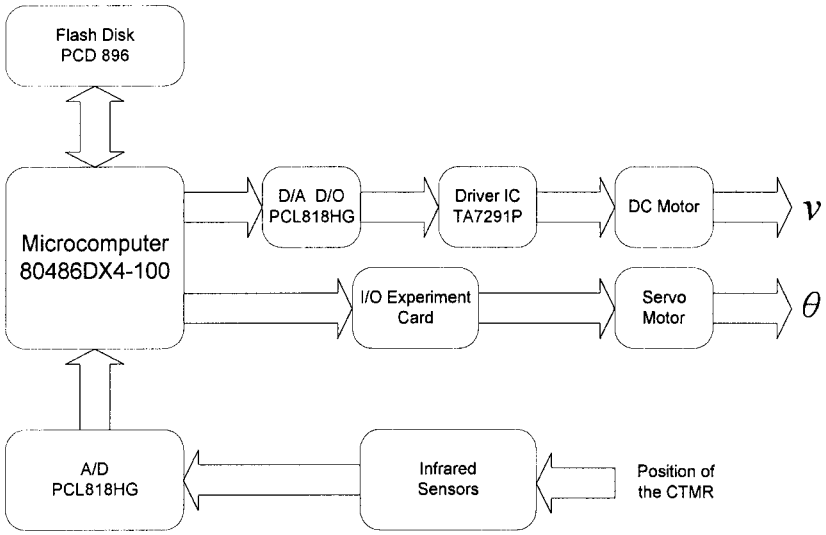

Figure 3. The signal processing block diagram of the CTMR.

The microcomputer part includes CPU board, booting board, and A/D-D/A card. The signal processing block diagram of a CTMR is shown in Figure 3. The CPU board (PCA-6143P) is operated by a 32 bit Intel 80486DX4-100 CPU. It is manufactured by Advantech Co., Ltd. The booting disk (PCD-896) manufactured by M-SYSTEMS Flash Disk Pioneers Ltd emulates a floppy disk drive by using solid-state memory chip to store program and data instead of the magnetic particles on the mechanical drive's disk. For the CTMR, there are no HDD and FDD connected with microcomputer system. Hence, we install Flash Disk as a bootable device.

The A/D-D/A card (PCL-818HG) acts a very important part in this control system. It is manufactured by Advantech Co., Ltd and is a high-gain, high-performance multifunction data acquisition card for IBM PC/XT/AT or compatible computers. It offers five most desired measurements and control functions: 12 bits A/D conversion, D/A conversion, digital input, digital output and timer/counter. The first function we adopted is A/D conversion. The CTMR recognizes the information of environments via the mounted sensors. The output of sensor is an analog signal and is not sent to computer directly. Hence, A/D-D/A card plays as the communicator between the sensors and computer. The second function we adopted is the digital output. We can control the rotating direction of the DC motor via connected digital output of the A/D-D/A card with TA7291P IC. Besides, we can connect the D/A conversion of A/D-D/A card with TA7291P IC to command the speed of DC motor.

The electronic driver consists of a TA7291P DC motor driver IC and an I/O experiment card. We have assembled the TA7291P IC in a circuit board to drive the DC motor to change the speed of the front wheels. The I/O experiment card is manufactured by ICC Information Co., Ltd. We employ the I/O card to generate PWM signal and to drive the DC servo motor to control the steering angle of the front wheels.

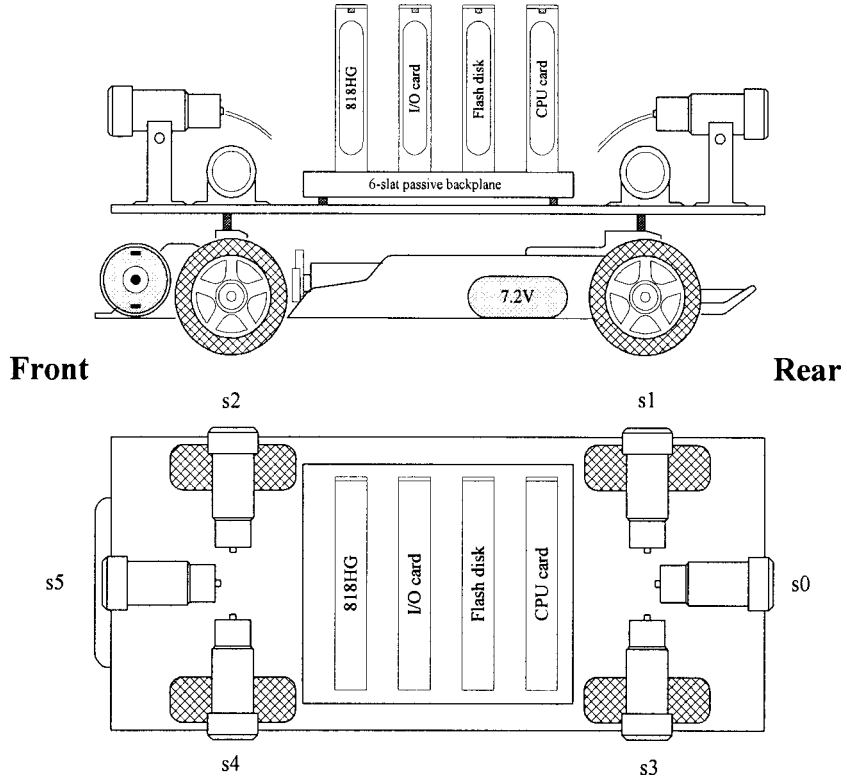

Figure 4. Microcomputer system arrangement and sensor arrangement.

The CTMR is able to navigate through unknown environment based on the information of infrared sensors. The sensor used in this study is the UF 55MG manufactured by TELCO International Ltd. The infrared sensor contains both a transmitter and a receiver and is exploited to detect the distance between the CTMR and obstacles. The output voltage of sensor UF 55MG is almost inverse proportional to the distance.

There are six infrared sensors placed on the CTMR. The sensor arrangement of the CTMR is illustrated in Figure 4. The microcomputer system arrangement is also presented. Four sensors are set up above the position of four wheels. Hence the CTMR can rapidly respond to a control command when sensors detect a change of environment. To avoid colliding with obstacles, we place one sensor in the front of the CTMR as well as the rear part.

3. Modelling and FPPC of the CTMR

The infrared sensor UF66MG gives the output voltage inverse proportional to the reflection distance within 1.0 m. In this section, the modeling, wall following, and parallel-parking problems of the CTMR are examined by this kind of measurements. Finally, we address the design procedures of the FPPC and give computer simulations to show the feasibility.

3.1. MODELING OF A CTMR

Consider a kinematic model of the CTMR shown in Figure 5, where the rear wheels are fixed parallel to car body and allowed to roll or spin but not slip. The front wheels can turn to left or right, but the left and right front wheels must be parallel. All the corresponding parameters of the CTMR depicted in Figure 5 are defined as follows:

$F(x_f, y_f)$: position of the front wheel center of CTMR,
$R(x_r, y_r)$: position of the rear wheel center of CTMR,
ϕ: the orientation of the steering-wheels with respect to the frame of CTMR,
θ: the angle between vehicle frame orientation and X-axis,
l: the wheel-base of CTMR,
O: center of curvature,
r: distance from point O to point $F(x_f, y_f)$,
k: curvature of the fifth-order polynomial.

The CTMR motion is constrained by the following equation [9]

$$\dot{y}_r \cos\theta - \dot{x}_r \sin\theta = 0. \tag{1}$$

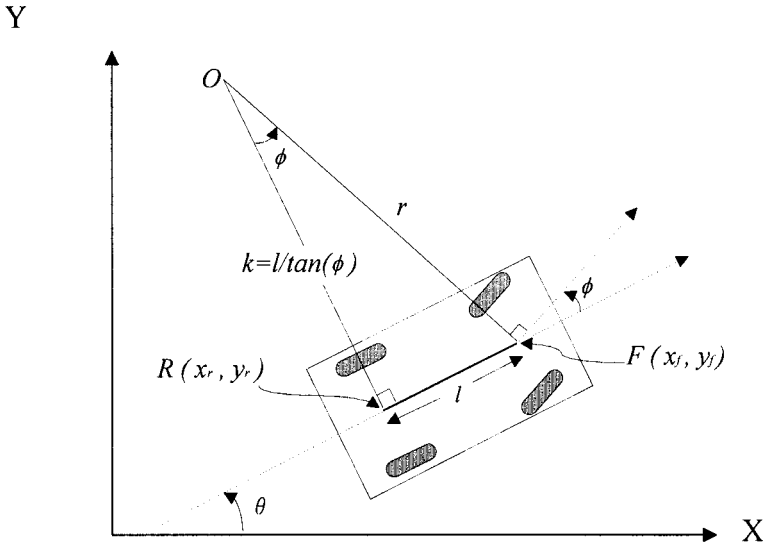

Figure 5. Kinematic model of a CTMR.

Equation (1) is the so-called nonholonomic constraint, which presents the tangent direction along any feasible path for a car and is bounded by the curvature of the path. The rear wheel kinematic equation of the CTMR is then described by

$$\dot{x}_{\mathrm{r}} = v \cos\theta \cos\phi, \tag{2a}$$

$$\dot{y}_{\mathrm{r}} = v \sin\theta \cos\phi, \tag{2b}$$

$$\dot{\theta} = v\frac{\sin\phi}{l}, \tag{2c}$$

where v is the speed of the front wheels. Equations (2) present the backward movement of the CTMR. Since the rear wheels are fixed relative to the car body and no slipping happens, the relationship of the coordinates $(x_{\mathrm{f}}, y_{\mathrm{f}})$ and $(x_{\mathrm{r}}, y_{\mathrm{r}})$ are

$$x_{\mathrm{r}} = x_{\mathrm{f}} - l\cos\theta, \qquad y_{\mathrm{r}} = y_{\mathrm{f}} - l\sin\theta. \tag{3}$$

Thus, differentiating (3) implies

$$\dot{x}_{\mathrm{r}} = \dot{x}_{\mathrm{f}} + \dot{\theta} l \sin\theta, \qquad \dot{y}_{\mathrm{r}} = \dot{y}_{\mathrm{f}} - \dot{\theta} l \cos\theta. \tag{4}$$

Substituting (4) into (2) results in

$$\dot{x}_{\mathrm{f}} = v\cos(\theta + \phi), \qquad \dot{y}_{\mathrm{f}} = v\sin(\theta + \phi), \qquad \dot{\theta} = v\frac{\sin\phi}{l}. \tag{5}$$

Equations (5) are the front wheel kinematic equations and denote forward motion of the CTMR.

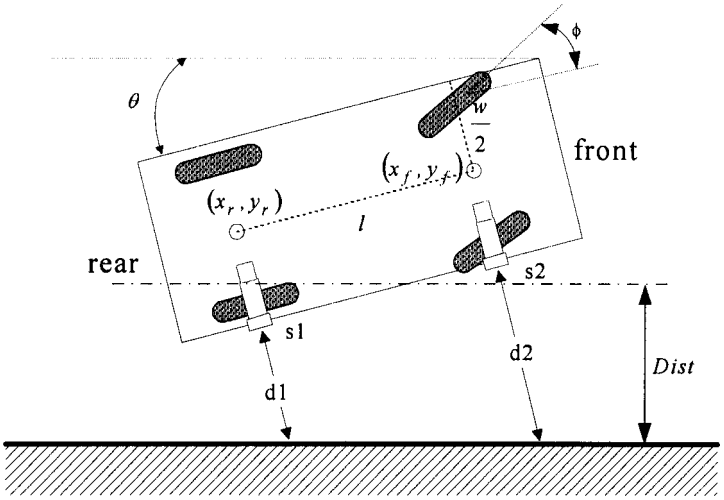

Figure 6. Definition of the input variables for wall following.

3.2. CONTROLLER FOR WALL FOLLOWING

The motivation of this task is to control the steering of a CTMR so that it smoothly follows a wall with a constant distance, i.e., θ is almost equal to zero. In other words, the CTMR must be always parallel to the wall. The input variables of the fuzzy logic controller for wall following problem are depicted in Figure 6, where d_1 is distance measured by the infrared sensor (s1) between the CTMR and wall, d_2 presents distance measured by the infrared sensor (s2) between the CTMR and wall, θ denotes angle of the CTMR with respect to the wall, and ϕ indicates the steering angle of the robot's front wheels.

In simulations, the kinematic Equations (2) and (5) are used to generate the next step of the CTMR. We can obtain the positions of each sensor from $(x_\mathrm{f}, y_\mathrm{f})$ and $(x_\mathrm{r}, y_\mathrm{r})$, which is shown in Figure 6, where

$$\text{s1:} \quad \left(x_\mathrm{f} + \frac{w}{2}\sin\theta,\, y_\mathrm{f} - \frac{w}{2}\cos\theta\right) \quad \text{and}$$

$$\text{s2:} \quad \left(x_\mathrm{r} + \frac{w}{2}\sin\theta,\, y_\mathrm{r} - \frac{w}{2}\cos\theta\right).$$

The distances can be calculated via the difference between the position of each sensor and wall, so the output values (d_1 and d_2) of each sensor can be obtained.

We will propose one two-input-single-output FLC scheme to command the steering angle of the front wheels for the wall following task. We introduce $u1$ and $u2$ as the input linguistic variables, when the robot is moving forwards, variable $u1$ is defined as $d_2 - \mathit{Dist}$ and variable $u2$ is defined as $d_2 - d_1$; when the robot is moving backward, variable $u1$ is defined as $d_1 - \mathit{Dist}$ and variable $u2$ is defined as $d_1 - d_2$. Variable $u1$ presents the distance from CTMR to the wall. Variable $u2$ denotes the orientation of CTMR parallel to the wall. Where *Dist* is the desired

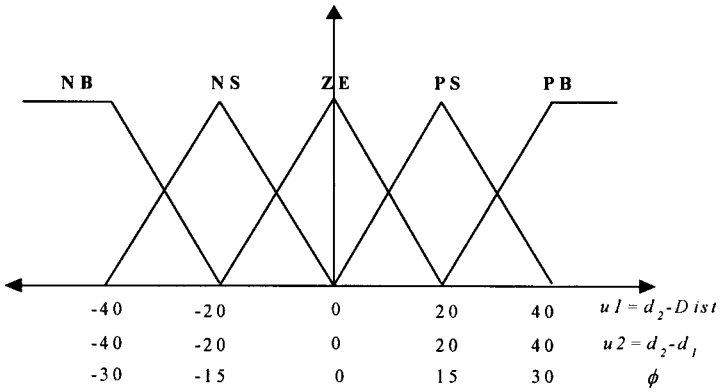

Figure 7. Fuzzy membership functions for the CTMR wall following.

constant distance between the CTMR and the wall. The output of the FLC is the steering angle ϕ. We exploit the kinematic Equations (2) or (5) of the CTMR and the relationships among the CTMR, the wall, and the information of sensors to derive the two- input-and-single-output FLC scheme.

Input variables $u1$, $u2$, and output variable ϕ are decomposed into five fuzzy partitions with triangular membership functions. We do not apply any special tuning algorithm to determine membership functions. We just equalize the universes of discourses to form five membership functions. The partitions and the shapes of the membership functions are shown in Figure 7, where the $u1$, $u2$, and ϕ are divided into five term-sets, denoted by NB (Negative Big), NS (Negative Small), ZE (Zero), PS (Positive Small), and PB (Positive Big). In fact, if we change the support regions of the membership functions, the simulation results are not obviously different. The control input ϕ presents the steering angle of front wheels. According to the fuzzy sliding-mode control (FSMC) [11], fuzzy reasoning rules for the wall following can be expressed in the following linguistic forms:

R1: if $u1$ is Positive Big and $u2$ is Negative Big, then ϕ is Zero.
R2: if $u1$ is Positive Big and $u2$ is Negative Small, then ϕ is Positive Small.
⋮
R13: if $u1$ is Zero and $u2$ is Zero, then ϕ is Zero.
⋮
R24: if $u1$ is Negative Big and $u2$ is Positive Small, then ϕ is Negative Small.
R25: if $u1$ is Negative Big and $u2$ is Positive Big, then ϕ is Zero.

For FSMC, we first have to choose the sliding surface (in fact, a sliding line in this application) that represents the control purpose. For instance, the sliding line is defined as

$$s = u1 + u2. \tag{6}$$

All the control effort is to drive the input variables to the sliding line $s = 0$. Once the CTMR is on this condition, then the steering angle ϕ should be zero. One of the

Table I. Fuzzy rule table for wall following

u1 \ *u2* / ϕ	**NB**	**NS**	**ZE**	**PS**	**PB**
PB	ZE	PS	PM	PB	PB
PS	NS	ZE	PS	PM	PB
ZE	NM	NS	ZE	PS	PM
NS	NB	NM	NS	ZE	PS
NB	NB	NB	NM	NS	ZE

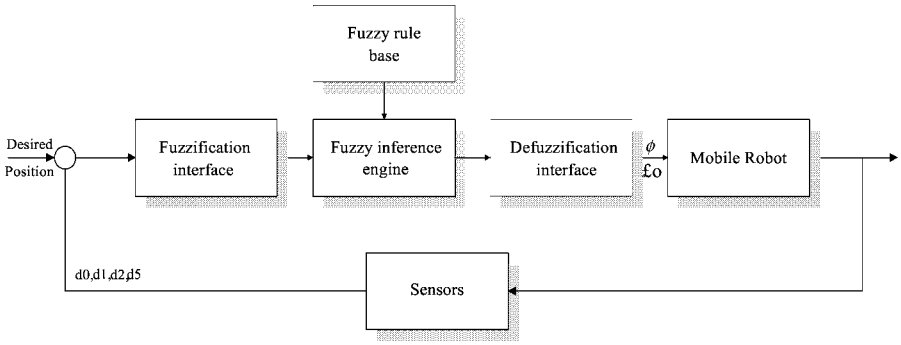

Figure 8. Basic configuration of the fuzzy logic system.

corresponding linguistic meanings is that "if $u1$ is NB and $u2$ is PB, then ϕ is ZE". Similar ϕ can be determined when $u1$ and $u2$ are in opposite state pair, such as (NS, PS), (PS, NS), (PB, NB) and so on. Suppose $s > 0$ (for example, $u1$ is PS and $u2$ is PS), this phenomenon denotes the CTMR is just above the desired distance, we have to turn the steering angle to the right to decrease d_2 and increase d_1. Figure 6 depicts this situation. Similarly, if $u1$ is PB and $u2$ is PB, then ϕ should be PB. On the other hand, if $s < 0$, then $\phi < 0$. We can conclude that if $u1 + u2$ is closed to the sliding line, the control action is smaller than those are far from the line. The rule table for wall following is summarized in Table I.

The basic configuration of the FLC for CTMR is depicted in Figure 8. The defuzzification strategy can be described as

$$\phi_{\text{crisp}} = \frac{\sum_{j=1}^{n} \mu_j(\phi_j) \cdot \phi_j}{\sum_{j=1}^{n} \mu_j(\phi_j)}, \tag{7}$$

where ϕ_j is the mean value that supports each fuzzy set j and n is the number of fuzzy sets, and $\mu_j(\phi_j)$ is the membership function value of the control input.

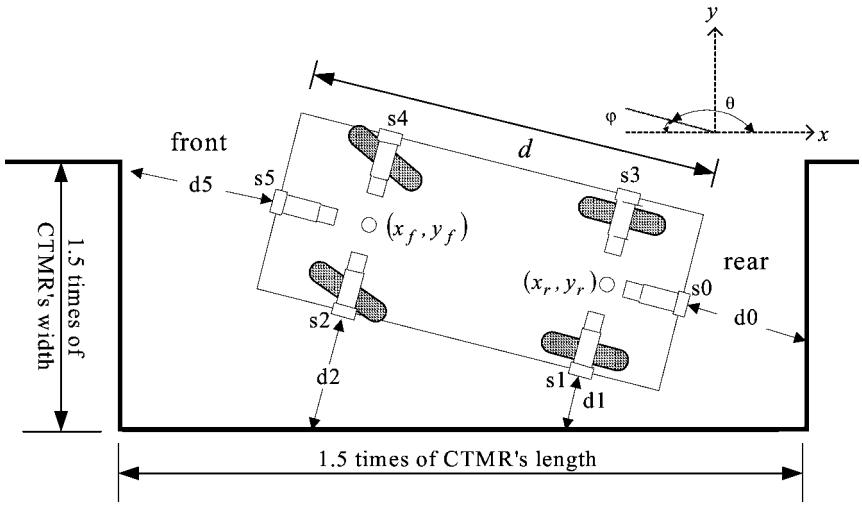

Figure 9. Definition of input variables for backward FPPC.

3.3. BACKWARD FPPC

For back-drive parallel-parking control, the input variables of the FPPC are defined in Figure 9, where d_0 presents the distance measured by the infrared sensor (s0) from the center of the rear body to the wall; d_1 is the distance measured by the infrared sensor (s1) from the rear wheel of the CTMR to the wall; d_2 denotes the distance measured by the infrared sensor (s2) from the steering wheel of the robot to the wall; and d_5 represents the distance measured by the infrared sensor (s5) from the center of the front body to the wall. The speed control v has two commands, constant speed and stop. Input variables $u1$, $u2$, and output variable ϕ are the same definitions as those in wall following case.

In backward parallel-parking simulations, the coordinates of each sensor can be calculated from the relative coordinates as depicted in Figure 9, where

$$\text{s0:} \quad \left(x_{\mathrm{r}} + \frac{d-l}{2}\cos\varphi,\, y_{\mathrm{r}} - \frac{d-l}{2}\sin\varphi\right),$$

$$\text{s1:} \quad \left(x_{\mathrm{r}} - \frac{w}{2}\sin\varphi,\, y_{\mathrm{r}} - \frac{w}{2}\cos\varphi\right),$$

$$\text{s2:} \quad \left(x_{\mathrm{f}} - \frac{w}{2}\sin\varphi,\, y_{\mathrm{f}} - \frac{w}{2}\cos\varphi\right), \quad \text{and}$$

$$\text{s5:} \quad \left(x_{\mathrm{r}} - \frac{d-l}{2}\cos\varphi,\, y_{\mathrm{f}} + \frac{d-l}{2}\sin\varphi\right).$$

Then we can obtain the output values (d_0, d_1, d_2, and d_5) of each sensor in simulations.

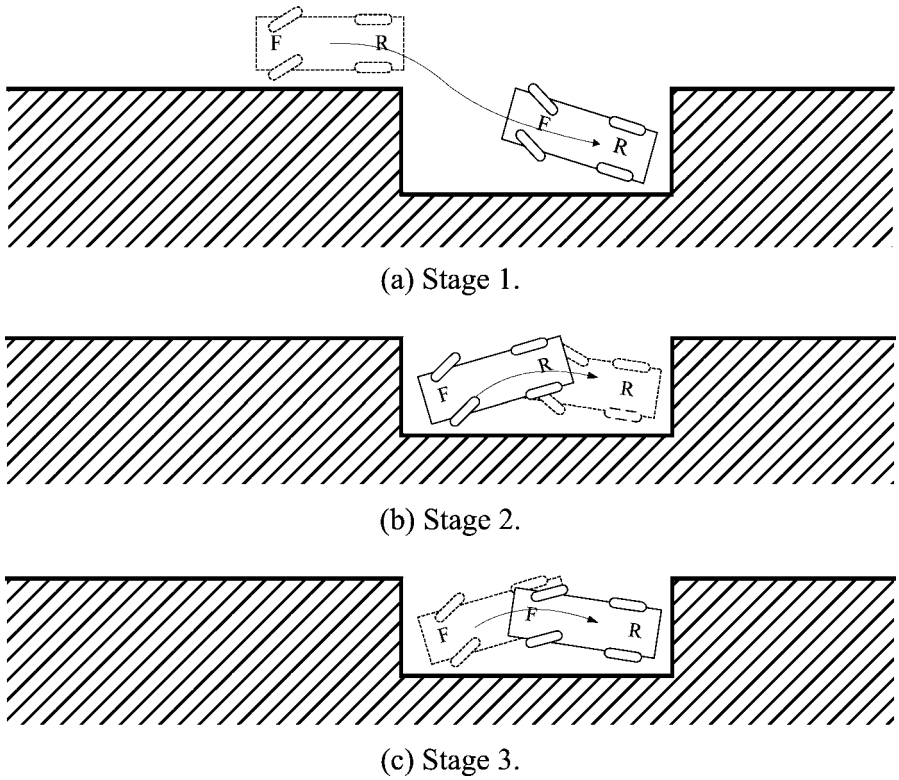

Figure 10. Three stages for backward parallel parking.

The FPPC should have the ability not only to park the car correctly but also not to collide with the wall of the parking lot. To determine the fuzzy control rules, we decompose the process of parking into following three stages:

- **Stage 1**. The CTMR is backed into the parking lot along the wall until the whole or most part enters the parking place and stops at the other side of parking lot as d_0 is almost zero.
- **Stage 2**. Examine the position of the robot to see whether it is properly staying in the parking lot via sensors. If the position of the robot does not meet the required condition, it moves forwards in the parking lot to correct its parking position.
- **Stage 3**. Examine the position of the robot to check whether it is properly staying in the parking lot via sensors. If the position of the robot does not meet our goal, it moves backwards in parking lot to correct its parking position.

The movements of stages 2 and 3 are repeated until the robot reaches the desired destination. The desired destination is that $d_1 - d_2$ and $d_0 - d_5$ are almost zero. Figure 10 illustrates these control stages for backward parallel parking.

It is well known that the rules of a FLC are usually determined by the human operator's behavior. In essence, a FLC is an algorithm that can convert the linguistic

control strategy based on the knowledge of expert or operator into an automatic control strategy. In this paper, the backward FPPC strategy is in fact obtained by the human driving skill and the fuzzy wall following control. The DML rules for each stage are addressed as follows:

- **Rules I** (for stage 1). In this stage, we perform the backward wall following control, where the fuzzy rule table for ϕ is the same as that in Table I, where $u1 = d_1 - Dist$ and $u2 = d_1 - d_2$. The speed v of the CTMR is constant and the CTMR stops when d_0 is almost zero. The desired distance parallel to the wall of the parking zone is $Dist = 15$ cm.
- **Rules II** (for stage 2). In this stage, the forward wall following control is utilized and the fuzzy rule table for ϕ is also the same as that in Table I, where $u1 = d_2 - Dist$, $u2 = d_2 - d_1$, and $Dist = 10$ cm. The CLMR will stop when d_5 is almost zero or $d_2 - d1$ and $d_0 - d_5$ are almost zero.
- **Rules III** (for stage 3). The rules are the same as those in stage 1, except $Dist = 10$ cm and the CTMR will also stop at $d_1 - d_2$ and $d_0 - d_5$ are almost zero.

3.4. FORWARD FPPC

Suppose the length of the parking area is almost 2 times the length of the CTMR, the car can directly head in the parking lot as the front sensors detect this lot. Figure 11 shows the definition of input variables for the forward parallel parking. In fact, all the input and output variables of the forward FPPC are the same as those of the backward one. The parking procedure is depicted in Figure 12, where three stages are also included:

- **Stage 1**. The CTMR moves forwards along the wall of the parking lot until the whole or most part of the robot enters the parking lot and stops at the other side of parking region as d_5 is almost zero.

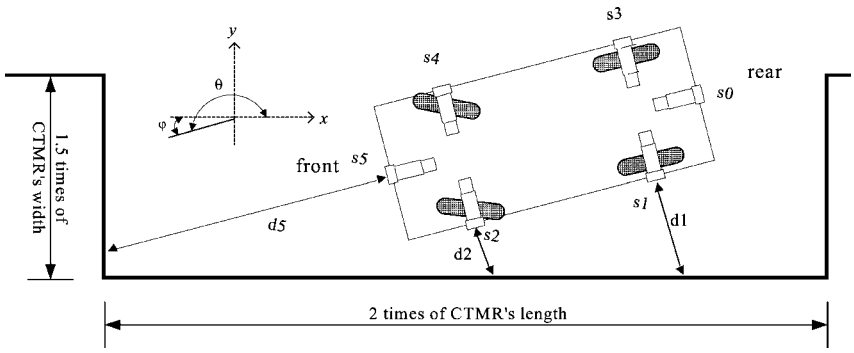

Figure 11. Definition of input variables for forward FPPC.

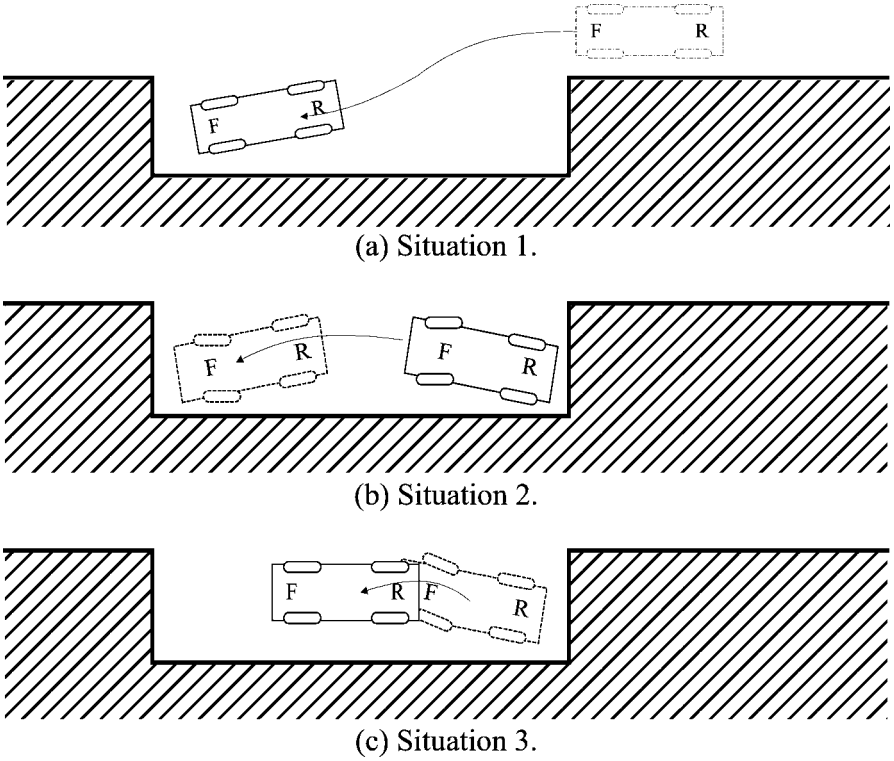

Figure 12. Three stages for forward parallel parking.

- **Stage 2**. Evaluate the position of the robot to see whether it is properly staying in the parking lot via sensors. If the position of the robot does not meet our requirement, it moves backwards in parking lot to correct the parking position.
- **Stage 3**. Evaluate the position of the robot to check whether it is properly staying in the parking lot via sensors. If the position of the robot does not meet our goal, it moves forwards in parking lot to correct the parking position.

The movements of stages 2 and 3 are repeated until the robot reaches the desired destination. The desired destination is the same as those in the backward FPPC. Similarly, the fuzzy inference rules for these stages are almost the same as those in backward FPPC and the descriptions are omitted here for the sake of saving the paper length.

3.5. SIMULATION RESULTS

The computer simulation results of the CTMR are given to demonstrate the effectiveness of the proposed control schemes. For wall following, the initial state $(x_{f_i}, y_{f_i}, \theta_i)$ of the CTMR is placed at $(120, 150, 30°)$. Simulation result for this initial is displayed in Figure 13, where the CTMR are able to follow the wall. For backward parallel parking, suppose the initial state $(x_{f_i}, y_{f_i}, \theta_i)$ of the CTMR

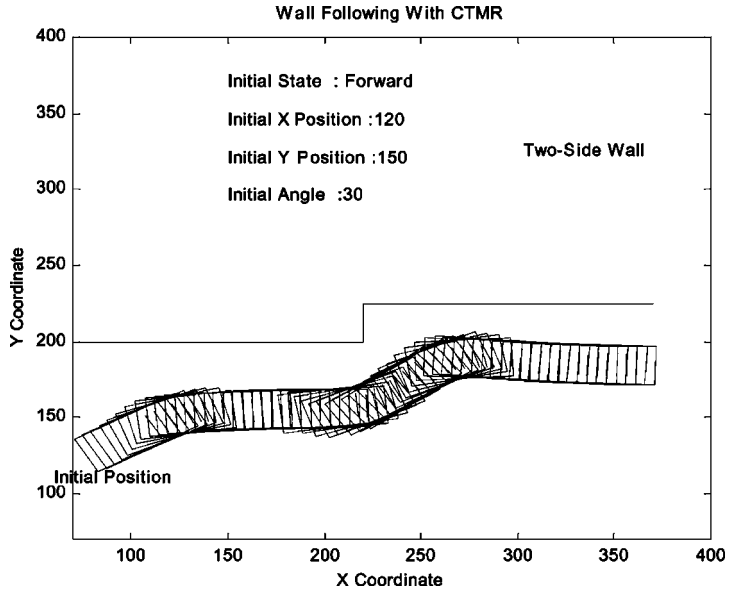

Figure 13. Simulation result of wall following: initial location $(x_{f_i}, y_{f_i}, \theta_i) = (120, 150, 30°)$.

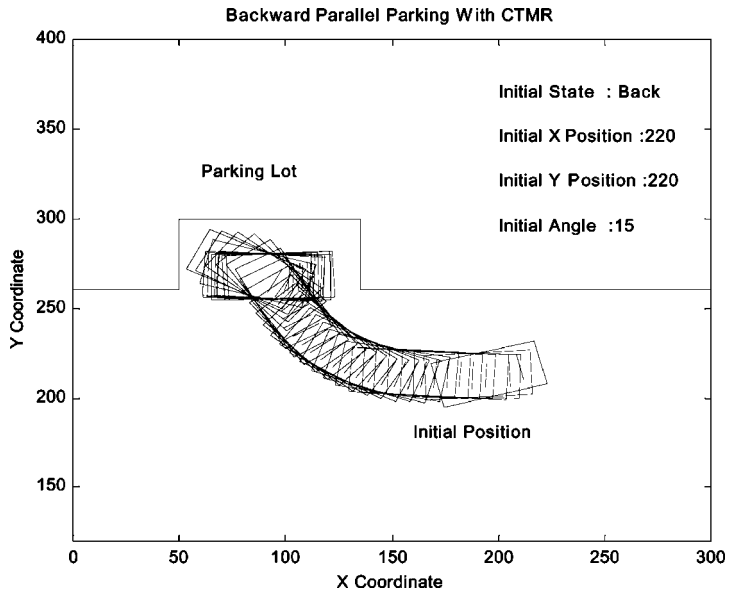

Figure 14. Simulation result of backward parallel parking: initial location $(x_{f_i}, y_{f_i}, \theta_i) = (220, 220, 15°)$.

are located at $(220, 220, 15°)$ and $(250, 200, 15°)$. Simulation results for these two initials are illustrated in Figures 14 and 15, respectively. For forward parallel parking, the initial states $(x_{f_i}, y_{f_i}, \theta_i)$ of the CTMR are placed at $(175, 225, 30°)$ and $(195, 205, -20°)$. The resulting trajectories are shown in Figures 16 and 17, respectively.

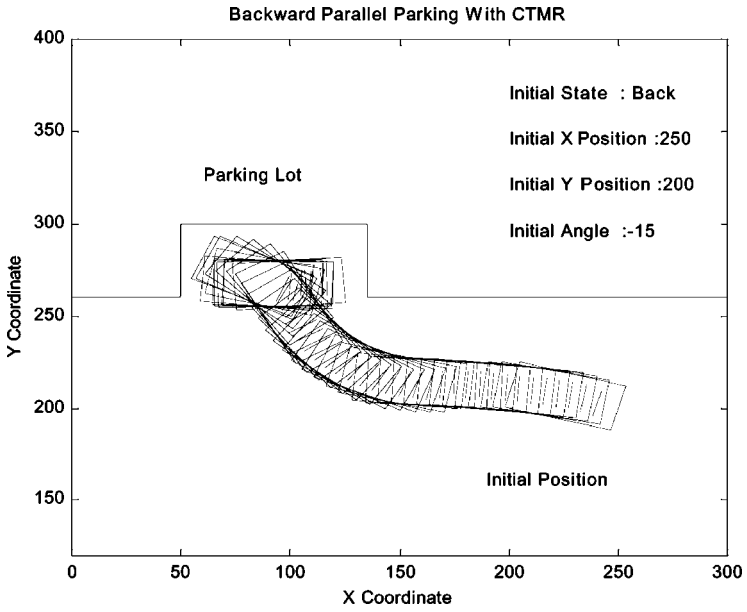

Figure 15. Simulation result of backward parallel parking: initial location $(x_{f_i}, y_{f_i}, \theta_i) = (250, 200, 15°)$.

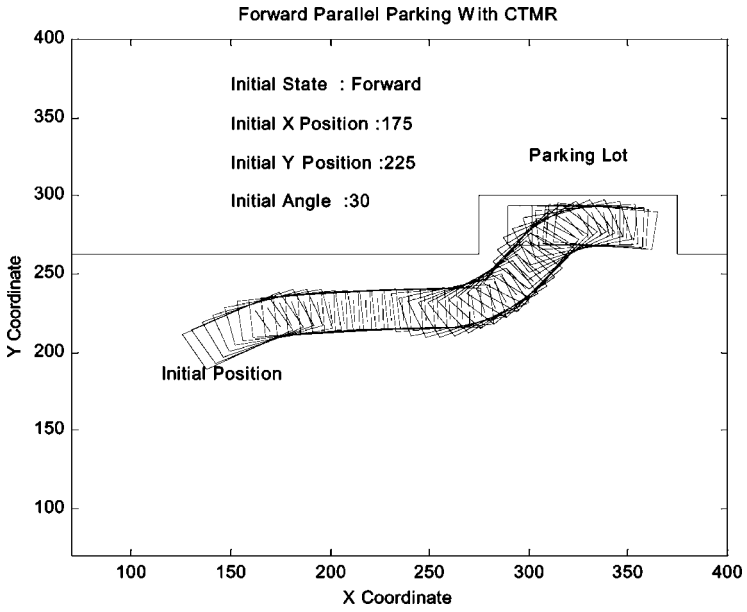

Figure 16. Simulation result of forward parallel parking: initial location $(x_{f_i}, y_{f_i}, \theta_i) = (175, 225, 30°)$.

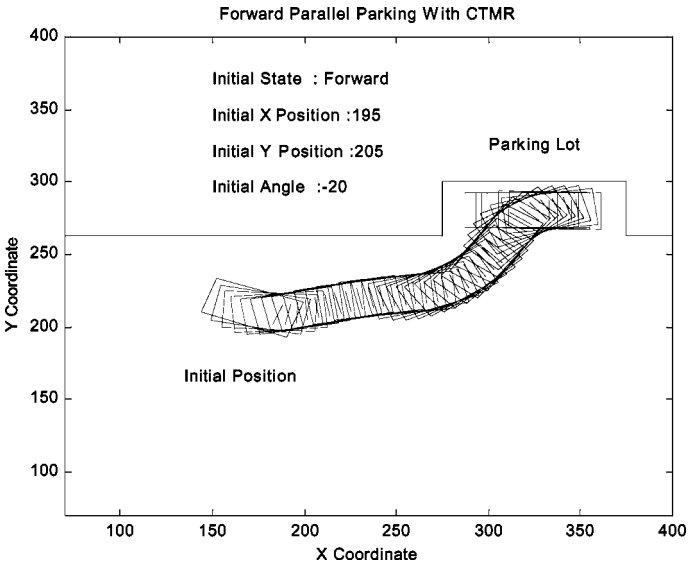

Figure 17. Simulation result of forward parallel parking: initial location $(x_{f_i}, y_{f_i}, \theta_i) = (195, 205, -20°)$.

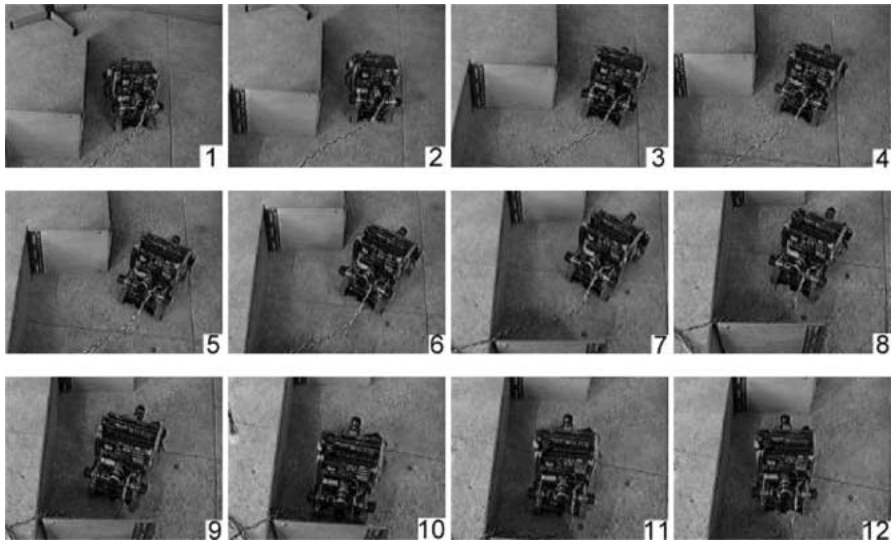

Figure 18. Experimental pictures of backward FPPC.

Figures 14–17 illustrate some typical examples of the CTMR by applying the FPPC schemes from different initial locations. It is seen very clearly from these figures that we can successfully control the vehicle to park at the desired position.

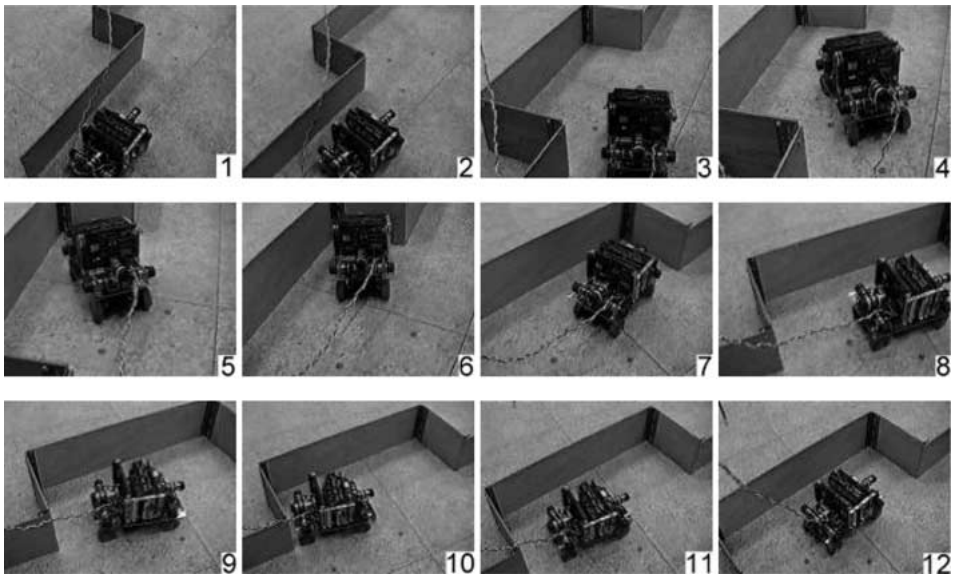

Figure 19. Experimental pictures of forward FPPC.

4. Real-time Implementation and Experimental Results

Since the computer simulations are successful, we want to use the developed CTMR to realize the FPPC schemes. The established CTMR is shown in Figure 1, its corresponding dimensions are length 380 mm, width 240 mm and weight 4 kg. The dimensions of the parking zone are length 570 mm and width 360 mm for backward parallel parking, and are length 760 mm and width 360 mm for forward parallel parking.

The actual experimental photographs for backward and forward parallel-parking tasks are demonstrated in Figures 18 and 19, respectively. One can find that the proposed FPPC can successfully accomplish the parallel-parking missions.

5. Conclusion

In this paper, we have designed and implemented an autonomous CTMR control system, where the chassis of the CTMR, microcomputer unit, motor drive unit, AD/DA unit, and sensor unit have been successfully set up. We have proposed fuzzy wall following control and two FPPC schemes for the CTMR. All the computer simulations and practically experimental results demonstrate that the propounded FPPC methods are indeed effective and feasible.

Acknowledgements

The authors wish to thank the National Science Council, Taiwan, Republic of China, for their support of the studies under grants NSC 89-2213-E006-186 and

NSC 89-2213-E006-187. The authors are also grateful to the anonymous reviewers for their valuable comments, which improved the contents of this paper considerably.

References

1. Altrock von, C. and Krause, B.: Fuzzy logic and neurofuzzy technologies in embedded automotive applications, in: *IFI'93, 3rd Internat. Conf. on Industrial Fuzzy Control and Intelligent Systems*, New York, 1993, pp. 55–59.
2. Altrock von, C., Krause, B., and Zimmermann, H.-J.: Advanced fuzzy logic control of a model car in extreme situations, *Fuzzy Sets Systems* **48** (1992), 41–52.
3. Barraquand, J. and Latombe, J. C.: On nonholonomic mobile robots and optimal maneuvering, in: *Proc. of '89 Symposium on Intelligent Vehicles*, Albany, New York, 1989, pp. 340–347.
4. Beom, H. R. and Cho, H.: A sensor-based navigation for a mobile robot using fuzzy logic and reinforcement learning, *IEEE Trans. Systems Man Cybernet.* **25**(3) (1995), 129–139.
5. Chang, S.-J.: Design and implementation of intelligent garage-parking control, Master Thesis, Dept. of E.E., N.C.K.U., Taiwan, 1997.
6. Cheng, C. W., Chang, S. J., and Li, T.-H. S.: Parallel-parking control of autonomous mobile robot, in: *IECON '97, Industrial Electronics, Control and Instrumentation*, Vol. 3, 1997, pp. 1305–1310.
7. Divelbiss, A. W. and Wen, J. T.: Trajectory tracking control of a car-trailer system, *IEEE Trans. Control Systems Technol.* **53** (1997), 269–278.
8. Jiang, K. and Seneviratne, L. D.: A sensor guided autonomous parking system for nonholonomic mobile robot, in: *Proc. of 1999 IEEE Internat. Conf. on Robotics and Automation*, Vol. 1, 1999, pp. 311–316.
9. Laumond, J. P., Jacobs, P. E., Taix, M., and Murray, R. M.: A motion planner for nonholonomic mobile robots, *IEEE Trans. Robots Automat.* **10**(5) (1994), 577–593.
10. Leitch, D. and Probert, P. J.: New techniques for genetic development of a class of fuzzy controller, *IEEE Trans. Systems Man Cybernet.* **28** (1998), 112–123.
11. Li, T. H. S. and Shieh, M. Y.: Switching-type fuzzy sliding model control of a cart-pole system, *Mechatronics* **10** (2000), 91–109.
12. Lian, K. Y., Chin, C. S., and Chiang, T. S.: Parallel parking a car-like robot using fuzzy gain scheduling, in: *Proc. of 1999 IEEE Internat. Conf. on Control Applications*, Vol. 2, 1999, pp. 1686–1691.
13. Lyon, D.: Parallel parking with curvature and nonholonomic constraint, in: *Proc. of '92 Symposium on Intelligent Vehicles*, Detroit, MI, 1992, pp. 341–346.
14. Maeda, M., Maeda, Y., and Murakami, S.: Fuzzy drive control of an autonomous mobile robot, *Fuzzy Sets and Systems* **39** (1991), 195–204.
15. Murray, R. M. and Sastry, S. S.: Nonholonomic motion planning: Steering using sinusoids, *IEEE Trans. Automat. Control* **4** (1993), 700–716.
16. Nelson, W.: Continuous curvature paths for autonomous vehicles, in: *Proc. of IEEE Internat. Conf. on Robotics and Automation*, 1989, pp. 1260–1264.
17. Nishimori, K., Hirakawa, S., and Tokutaka, H.: Fuzzification of control timing in driving control of a model car, in: *Proc. of the 2nd IEEE Conf. on Fuzzy Systems*, San Francisco, CA, 1993, pp. 297–302.
18. Ohkita, M., Mitita, H., Miura, M., and Kuono, H.: Travelling experiment of an autonomous mobile robot for a flush parking, in: *Proc. of the 2nd IEEE Conf. on Fuzzy Systems*, San Francisco, CA, Vol. 2, 1993, pp. 327–332.

19. Paromtchik, I. E. and Laugire, C.: Motion generation and control for parking an autonomous vehicle, in: *Proc. of '96 IEEE Conf. on Robotics and Automation*, Vol. 4, Minneapolis, MN, 1996, pp. 3117–3122.
20. Sugeno, M. and Murakami, K.: An experimental study on fuzzy parking control using a model car, in: M. Sugeno (ed.), *Industrial Applications of Fuzzy Control*, North-Holland, Amsterdam, 1985, pp. 105–124.
21. Sugeno, M., Murofushi, T., Mori, T., Tatematsu, T., and Tanaka, J.: Fuzzy algorithmic control of a model car by oral instructions, *Fuzzy Sets Systems* **32** (1989), 207–219.
22. Sugeno, M. and Nishida, M.: Fuzzy control model car, *Fuzzy Sets Systems* **16** (1985), 103–113.
23. Watanabe, K., Tang, J., Nakamura, M., Koga, S., and Fukuda, T.: A fuzzy-Gaussian neural network and its application to mobile robot control, *IEEE Trans. Control Systems Technol.* **4**(2) (1996), 193–199.
24. Wu, C. J.: On the representation and collision detection of robots, *J. Intelligent Robotic Systems* **16** (1996), 151–168.
25. Wu, C. J. and Huang, C. H.: Back-propagation neural networks for identification and control of a direct drive robot, *J. Intelligent Robotic Systems* **16** (1996), 45–64.
26. Wu, C. J. and Liu, G. Y.: A genetic approach for simultaneous design of membership functions and fuzzy control rules, *J. Intelligent Robotic Systems* **28** (2000), 195–211.
27. Yasunobu, S. and Murai, Y.: Parking control based on predictive fuzzy control, in: *Proc. of the 3rd IEEE Conf. on Fuzzy Systems*, Orlando, FL, 1994, pp. 1338–1341.