

## Research Article

# Cloud Model Approach for Lateral Control of Intelligent Vehicle Systems

Hongbo Gao,<sup>1,2,3</sup> Xinyu Zhang,<sup>3</sup> Yuchao Liu,<sup>4</sup> and Deyi Li<sup>1,2,4</sup>

<sup>1</sup>State Key Laboratory of Software Development Environment, Beihang University, Beijing 100191, China

<sup>2</sup>State Key Laboratory of Automotive Safety and Energy, Tsinghua University, Beijing 100083, China

<sup>3</sup>Information Technology Center, Tsinghua University, Beijing 100083, China

<sup>4</sup>The Institute of Electronic System Engineering, Beijing 100039, China

Correspondence should be addressed to Xinyu Zhang; [xyzhang@tsinghua.edu.cn](mailto:xyzhang@tsinghua.edu.cn)

Received 6 June 2016; Revised 9 September 2016; Accepted 28 September 2016

Academic Editor: Xiong Luo

Copyright © 2016 Hongbo Gao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Studies on intelligent vehicles, among which the controlling method of intelligent vehicles is a key technique, have drawn the attention of industry and the academe. This study focuses on designing an intelligent lateral control algorithm for vehicles at various speeds, formulating a strategy, introducing the Gauss cloud model and the cloud reasoning algorithm, and proposing a cloud control algorithm for calculating intelligent vehicle lateral offsets. A real vehicle test is applied to explain the implementation of the algorithm. Empirical results show that if the Gauss cloud model and the cloud reasoning algorithm are applied to calculate the lateral control offset and the vehicles drive at different speeds within a direction control area of  $\pm 7^\circ$ , a stable control effect is achieved.

## 1. Introduction

In academic and industrial circles, studies on intelligent vehicles have drawn considerable attention. Such studies play an important role in the research on vehicles and intelligent transportation. Control methods are the key in the study of intelligent vehicles. Vehicle model parameters are extremely complex. The system model equation is nonlinear, and its system parameters constantly change over time. Research on vehicle control theory includes lateral tracking control and longitudinal tracking control. Lateral tracking control includes the support vector machine (SVM) method, the sub-level control method [1], the traditional PID (Proportional-Integral-Derivative) method [2], and the intelligent method. The latter includes the fuzzy control [3, 4] and neural network control methods [5, 6]. Longitudinal tracking control includes the coordination of the brake and the accelerator as well as the antijamming capability of the control accuracy. One important study on intelligent vehicle control is the Urban Challenge, which was organized by DARPA (Defense Advanced Research Projects Agency) in 2007. The champion, “BOSS,” used a road navigation and regional navigation

control strategy [7]. The third placer, the “Odin” team, used a control model [8] based on driver behavior. The “Talos” team applied the output path of the navigator and the speed commands of the low-level controller output, using the method based on a RRT (Rapidly-exploring-Random-Tree) [9], and generated a dynamic trajectory feasible tree through countless random samples, thereby expanding the typical RRT [10]. Current self-adaptive control methods of intelligent vehicles modify parameters of PID on the basis of changes in intelligent vehicle states and object properties, thereby improving control. They mainly include adaptive control reference models [11], adaptive control fuzzy models [12, 13], adaptive control neural networks models [14, 15], and adaptive control evolutionary models [16].

The current study aims to improve the accuracy, robustness, and adaptability to various road conditions of the vehicle control algorithm. First, the convergence of vehicles toward trajectory tracking errors is investigated from the perspective of nonlinear system stability, which is the premise of vehicle tracking trajectory. Subsequently, the robustness and control algorithm that can adapt to the environment is also considered, thereby ensuring control performance when

the running conditions of a vehicle are drastically changed. Finally, the function of vehicle motion control is expanded, which enables vehicles to complete automatic overtaking task, adaptive cruise task, automatic parking task, flowing into traffic task, and so on.

In most of the studies cited above, some researches only focused on lateral tracking control and some researches only focused on longitudinal tracking control, without considering driving speed and driving direction as input values. When intelligent driving tasks increase in complexity, the control systems cited earlier are unable to adapt to complex tasks. In addition, the control system should be able to guarantee stability. The main contributions of our study are as follows. (1) A new uncertainty control system according to the Gauss cloud model (GCM) and cloud reasoning is illustrated. (2) The new model considers both speed and direction, whereas velocity and direction are mutually constrained. (3) The speed control rules for intelligent driving vehicles are constructed, with reference to human driving experience.

This paper is organized as follows. Section 1 presents the lateral control of intelligent vehicle. Section 2 presents the GCM, the GCM algorithm, and cloud reasoning, including a preconditioned Gauss cloud generator (PGCG), a post-conditioned Gauss cloud generator (PCGCG), and a rule generator. Section 3 describes the lateral control algorithm for intelligent vehicle systems and cloud controller rules. Section 4 provides the results of the experiment and analysis performed using the cloud control algorithm. Finally, the results of experiment are illustrated in Section 5.

## 2. Model and Problem Formulation

**2.1. Gauss Cloud Model.** The Gauss distribution (GD) is one of the most important distributions in probability theory, in which the general characteristics of random variables are represented as means of the mean and variance of two numbers. As a fuzzy membership function, the bell-shaped membership function is mostly used in sets, which is typically expressed through the analytical expressions of  $m(x) = \exp\{-(x - a)^2/2b^2\}$ . This study presents a cloud model based on the GD, called the Gauss cloud model (GCM), which is defined as follows [17, 18].

*Definition 1.*  $U$  is expressed in a precise numerical quantitative domain.  $C(Ex, En, He)$  is a qualitative concept on  $U$ . If the value of  $x$  ( $x \in U$ ) is a random realization of the qualitative concepts of  $C$ , then the ‘‘expectation’’ of the GD  $x \sim N(Ex, En'^2)$  is denoted as  $Ex$ , and its ‘‘variance’’ is denoted as  $En'^2$ . Meanwhile, the ‘‘expectation’’ of GD  $En' \sim N(En, He^2)$  is denoted as  $En$ , and its ‘‘variance’’ is denoted as  $He^2$ .  $En'$  is the full form of GD  $En' \sim N(En, He^2)$  and is a random realization [19]. The certainty degree of  $x$  in  $C$  is satisfied via  $m(x) = \exp\{-(x - Ex)^2/2(En')^2\}$ . The distribution of  $x$  in the domain of  $U$  is called a Gauss cloud (GC) [20]. The GC algorithm is presented as follows [17, 20].

*The GC Algorithm*

*Input.* Three figures ( $Ex, En, He$ ) and the number of cloud drops  $n$ .



FIGURE 1: The GCG.

*Output.* A sample set that represents concept extension and its certainty  $(x_i, m_i)$ ,  $i = 1, \dots, n$ .

- (1) Generate Gauss random  $En' \sim N(En, He^2)$ .
- (2) Generate Gauss random  $x \sim N(Ex, En'^2)$ .
- (3) Calculate the certainty:  $m(x) = \exp\{-(x - Ex)^2/2(En')^2\}$ .
- (4) Repeat (1)–(4) until the number of cloud drops is  $n$ .

The algorithm causes distribution drops, called cloud distribution (CD). The algorithm of GCM can be obtained through a cloud generator (CG), which forms a forward Gauss cloud generator (GCG), as shown in Figure 1. The Gauss random number generation method is the foundation of the whole algorithm. It generates uniform random numbers in  $[0, 1]$  and uses them to calculate the Gauss random number. Random number sequences are determined through the uniform random function of a seed. The method of using uniform random numbers to generate a Gauss random number is described in detail in [21]. GC distribution (GCD) is different from the GD because the GCD algorithm uses the Gauss random number twice, in which one random number is the basis of another random number. Among these,

- (1) when  $He = 0$ , the algorithm generates a precise value of  $En$  and the value of  $x$  is transformed into a GD,
- (2) when  $He = 0$  and  $En = 0$ , the value of  $x$  of the algorithm generation is an exact value of  $Ex$ , and  $m \equiv 1$ .

From (1) and (2), certainty can be concluded as a special case of uncertainty, and the GD is a special case of the GCD.

For a qualitative concept of a steering angle of positive and negative  $40^\circ$ , given that  $Ex = 80^\circ$ ,  $En = 1$ , and  $He = 0.1$ , 1000 cloud drops are generated. The distribution of drops and its certainty degree of  $C(x, m)$  are shown in Figure 2.

### 2.2. Cloud Reasoning

**2.2.1. Preconditioned Gauss Cloud Generators and Postconditioned Gauss Cloud Generators.** Knowledge forms a concept and its relationship with communicating and abstracting. The relationship among concepts forms certain rules, from which rules library and rules generator can be established through knowledge reasoning based on GC. Rules include preconditioned and postconditioned rules. Preconditioned rules include one or several rules, whereas postconditioned rules express the results and specific control actions generated by the preconditioned rules. In the control field, ‘‘perception-action’’ can establish the rule library based on the relationship among concepts, thereby realizing control of uncertainty.

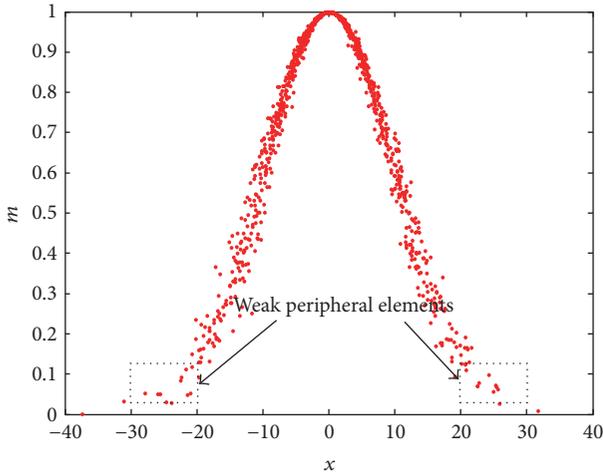


FIGURE 2: The distribution of 1000 drops.

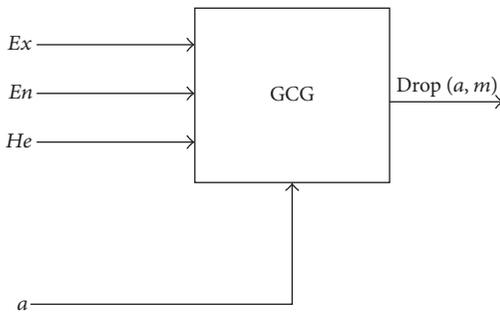


FIGURE 3: The PGCG.

A preconditioned Gauss cloud generator (PGCG) and a postconditioned Gauss cloud generator (PCGCG) are composed of the GCG, which is defined as follows.

*Definition 2.* Assume the following rule:

$$\text{If } A, \text{ then } B, \tag{1}$$

where  $A$  corresponds to concepts  $C_1$  in universal sets  $U_1$  and  $B$  corresponds to concepts  $C_2$  in universal sets  $U_2$ .  $a$  is a specific value in universal sets  $U_1$ , where the GCG generates a specific value of  $a$  based on the concept  $C_1$  of the certainty degree of  $m$  distribution, and  $m \in [0, 1]$ , which is called a PGCG [22], as shown in Figure 3.

The PGCG algorithm is presented as follows [23].

*The PGCG Algorithm*

*Input.* Three figures  $(Ex, En, He)$  and a specific value  $a$ .

*Output.* The distribution of drops  $(a, m)$ .

- (1) Generate Gauss random  $En' \sim N(En, He^2)$ .
- (2) Calculate the certainty:  $m(x) = \exp\{-\frac{(x - Ex)^2}{2(En')^2}\}$ .
- (3) Generate the distribution of drops  $(a, m)$ .

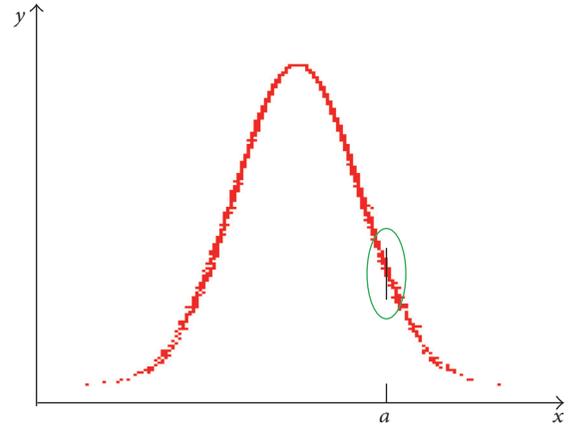


FIGURE 4: Cloud drop distribution of the PGCG.

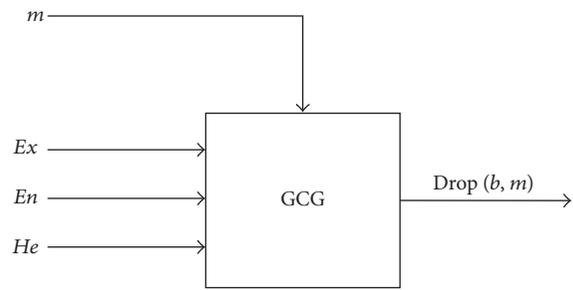


FIGURE 5: The PCGCG.

As shown in Figure 4, the distribution of drops  $(a, m)$  of the specific value of  $a$  and the certainty degree of  $m$  is on the line of  $x = a$ .

*Definition 3.* Assume the following rule:

$$\text{If } A, \text{ then } B, \tag{2}$$

where  $A$  corresponds to concepts  $C_1$  in universal sets  $U_1$  and  $B$  corresponds to concepts  $C_2$  in universal sets  $U_2$ . The certainty degree of  $m$  belongs to  $[0, 1]$ . The GCG generates the certainty degree of  $m$  drop distribution, which is satisfied by applying concepts  $C_2$  in universal sets  $U_2$ , called PCGCG [24], as shown in Figure 5.

The PCGCG algorithm is presented as follows [25, 26].

*The PCGCG Algorithm*

*Input.* Three figures  $(Ex, En, He)$  and certainty degree  $m$ .

*Output.* The drop distribution  $(b, m)$ .

- (1) Generate Gauss random  $En' \sim N(En, He^2)$ .
- (2) Calculate the certainty:  $b = Ex \pm En' \sqrt{-2 \ln m}$ .
- (3) Generate the distribution of drops  $(b, m)$ .

As shown in Figure 6, the drop distribution  $(b, m)$  of the cloud drop specific value of  $b$  and the certainty degree of  $m$  is on the line of  $y = m$ .

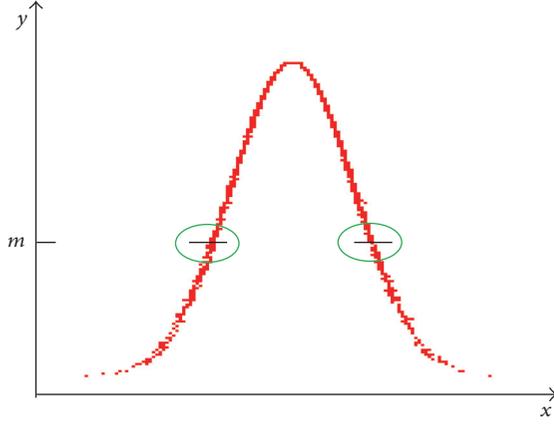


FIGURE 6: Cloud drop distribution of the PCGCG.

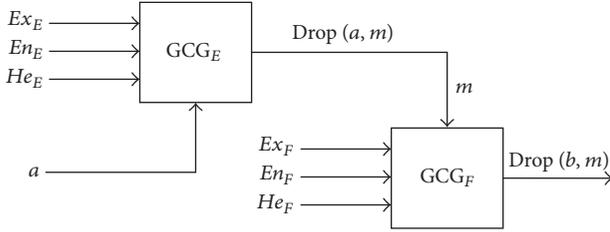


FIGURE 7: The SCSRGCG.

### 2.2.2. Rule Generator

**Definition 4.** Assume the following rule:

$$\text{If } E, \text{ then } F, \quad (3)$$

where  $E$  is the PGCG that generates the drop distribution  $(a, m)$  with a specific value of  $a$  and a certainty degree of  $m$ .  $F$  is the PCGCG that generates the drop distribution  $(b, m)$  of the cloud with a specific value of  $b$  and a certainty degree of  $m$ , which is called the single-condition single-rule GCG (SCSRGCG) [27, 28]. The composition diagrams of PGCG and PCGCG are shown in Figure 7.

The SCSRGCG algorithm is presented as follows.

#### The SCSRGCG Algorithm

**Input.** Three figures  $(Ex_E, En_E, He_E)$ , three figures  $(Ex_F, En_F, He_F)$ , and a specific value  $a$ .

**Output.** The drop distribution  $(b, m)$ .

- (1) Generate Gauss random  $En'_E \sim N(En_E, He_E^2)$ .
- (2) Calculate the certainty:  $m = \exp\{-(x - Ex_E)^2 / 2(En'_E)^2\}$ .
- (3) Generate Gauss random  $En'_F \sim N(En_F, He_F^2)$ .
- (4) If  $a < Ex$ , then calculate the certainty:  $b = Ex_F - En'_F \sqrt{-2 \ln m}$ .
- (5) If  $a > Ex$ , then calculate the certainty:  $b = Ex_F + En'_F \sqrt{-2 \ln m}$ .

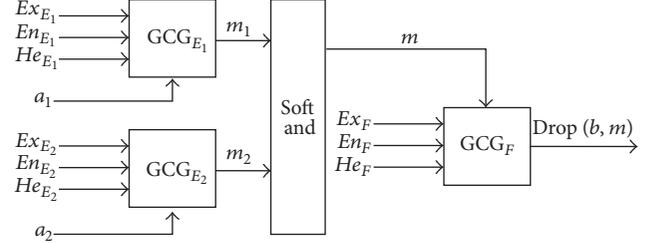


FIGURE 8: The DCSRGCG.

- (6) Generate the distribution of drops  $(b, m)$ .

The SCSRGCG implies an uncertainty transfer in the conceptual reasoning process. In the universal sets  $U_1$  of the PGCG, the distribution of the certainty degree of  $m$  belongs to the specific value of  $a$ , whereas the certainty degree of  $m$  is the input of the PCGCG that generates the drop distribution  $(b, m)$  of the cloud specific value of  $b$  and the certainty degree of  $m$ . The processing of the certainty value of  $a$  to the certainty value of  $b$  is uncertain.

**Definition 5.** Assume the following rule:

$$\text{If } E_1, E_2, \text{ then } F, \quad (4)$$

where  $E_1$  is the PGCG that generates the drop distribution  $(a_1, m_1)$  of the specific value of  $a_1$  and the certainty degree of  $m_1$ ,  $E_2$  is the PGCG that generates the drop distribution  $(a_2, m_2)$  of the specific value of  $a_2$  and the certainty degree of  $m_2$ , and  $F$  is the PCGCG that generates the drop distribution  $(b, m)$  of the cloud drop specific value of  $b$  and the certainty degree of  $m$ . The certainty degree  $m$  is obtained from the “soft and” of  $m_1$  and  $m_2$ , which is called a double-condition single-rule GCG (DCSRGCG). The composition diagrams of two PGCGs and one PCGCG are shown in Figure 8.

The “soft and” is expressed via 2D GCM  $(1, En_x, He_x, 1, En_y, He_y)$ , which expresses the uncertainty of “and” of  $m_1$  and  $m_2$ ; the result of “and” is expressed by the certainty degree of  $m$  [29]. The degree of “soft and” can be realized by adjusting the values of  $En_x, He_x, En_y,$  and  $He_y$  when  $En_x = En_y = 0$  and  $He_x = He_y = 0$ . Then, “soft and” becomes an “and” operation. The “soft and” output is presented in Figure 9, which shows the distribution of the drops and their certainty degree  $(x, y, m)$ , with  $x \in [0, 1]$ ,  $y \in [0, 1]$ .

The DCSRGCG can establish numerous conditions of single-rule GCG (MCSRGCG) based on its composition principle. The SCSRGCG and the MCSRGCG are stored in the rule library and applied in qualitative knowledge reasoning and intelligent control field.

### 3. Lateral Control Approach of an Intelligent Vehicle System

The control of an intelligent vehicle mainly comprises the control for speed and angle under conditions of car-following driving, lane-changing driving, and intersection driving, with car-following driving being the most common. Using

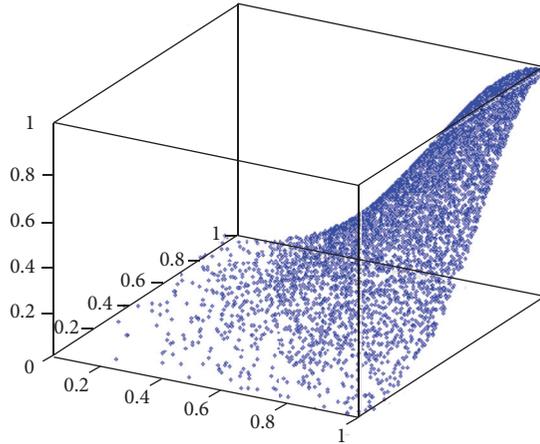


FIGURE 9: Quantitative transformation of the qualitative concept “soft and.”

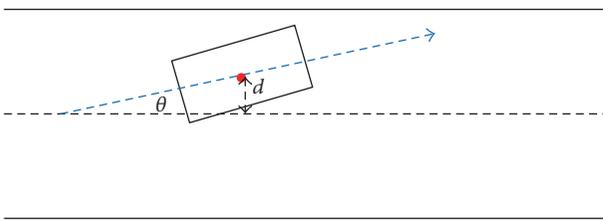


FIGURE 10: Distance  $d$  between the geometric center of a vehicle and the central axis of a lane, as well as the included angle  $\theta$  between heading direction and the central axis.

this state as an example, vehicle speed and angles can be intelligently controlled once cloud reasoning and cloud control are introduced.

Under the condition of car-following driving, an intelligent vehicle should constantly adjust its speed according to obstacles, such as vehicles and pedestrians, while driving efficiently and avoiding collisions. The angle control of an intelligent vehicle aims to keep the car in the middle of the road, with an equal distance between the left/right lane line and the center of the vehicle while driving. Furthermore, the heading direction should remain in accordance with the lane line.

The speed and angle controls of intelligent vehicles are both typical double-conditional and single-rule controllers. In Figure 10, the solid line represents the real lane line, whereas the thick dotted line represents the central axis of the lane calculated according to two lane markings. The black dots represent the geometric center of an intelligent vehicle, which is represented by a rectangle. Vehicles achieve angle control by calculating the distance  $d$  between the geometric center of an intelligent vehicle and the central axis of the lane, as well as the included angle  $\theta$  between heading direction and the central axis.

The input of the cloud controller is the distance  $d$  (in meters) between the geometric center of an intelligent vehicle and the direction of the central axis of the lane and the

included angle  $\theta$  (in degrees) between heading directions and lane line. The output is the steering wheel angle  $\delta$  (in degrees). On the basis of a brief summary of an actual driving experience, several qualitative conclusions are drawn:

- (1) If the intelligent vehicle does not veer off the middle of the lane and the heading direction of that vehicle remains in accordance with the axis of the lane, then the steering wheel should be returned to the zero position to keep the car moving straight forward. That is, if  $d$  and  $\theta$  are near 0,  $\delta$  should be 0, thereby enabling the vehicle to proceed normally.
- (2) If the vehicle offsets to the right, then turn the wheel to the left to try and return the vehicle to the center of the lane. For a higher offset value, a greater adjustment angle of the steering wheel is necessary. That is, if  $d$  is more than 0, then  $\delta$  is less than 0. A positive correlation exists between  $\delta$  and  $d$ 's absolute value.
- (3) If the vehicle offsets to the left, then turn the wheel to the right and try to return the vehicle to the center of the lane. For a greater offset value, a greater adjustment angle of the steering wheel is necessary. That is, if  $d$  is less than 0, then  $\delta$  is more than 0. A positive correlation exists between  $\delta$  and  $d$ 's absolute value.
- (4) If the included angle between the heading direction and the central axis of the lane is greater than 0, which indicates that the vehicle is drifting toward the right front of the axis, then turn the wheel to the left and try to return the vehicle to the center of the lane. For a higher offset value, a larger adjustment angle of the steering wheel is necessary. If  $\theta$  is more than 0, then  $\delta$  is less than 0. A positive correlation exists between  $\delta$  and  $\theta$ 's absolute value.
- (5) If the included angle between the heading direction and the central axis of the lane is less than 0, which indicates that the vehicle is drifting toward the left front of the axis, then turn the wheel to the right and try to return the vehicle to the center of the lane. For a higher offset value, a larger adjustment angle of the steering wheel is necessary. If  $\theta$  is less than 0, then  $\delta$  is more than 0. A positive correlation exists between  $\delta$  and  $\theta$ 's absolute value.

In the next section, we will describe the linguistic value sets of the input and the output, define the range of different linguistic values, and establish the cloud controller and its control rules based on the aforementioned five qualitative rules.

**3.1. Cloud Controller Rules.** The variables  $d$ ,  $\theta$ , and  $\delta$  can be described using five qualitative concepts, namely, “positive more,” “positive less,” “near-zero,” “negative less,” and “negative more.” The input and output variables define the five qualitative concepts and construct a corresponding cloud regulation generator.

The detailed car-following state and the speed control rules for intelligent vehicles are shown in Table 1. The rule set  $RS(d)$  of distance  $d$  between the geometric center of an

TABLE 1: (a) Rule sets  $RS(d)$  of distance  $d$  between the geometric center of an intelligent vehicle and the central axis of the lane. (b) Rule sets  $RS(\theta)$  of included angle  $\theta$  between heading direction and the central axis of the lane.

(a)		
	Axis-line-distance $d$	Steering wheel angle $\delta$
If	Positive more	Negative more
	Positive less	Negative less
	Zero	Zero
	Negative less	Positive less
	Negative more	Positive more
	Then	
(b)		
	Axis-line-angle $\theta$	Steering wheel angle $\delta$
If	Positive more	Negative more
	Positive less	Negative less
	Zero	Zero
	Negative less	Positive less
	Negative more	Positive more
	Then	

TABLE 2: Parameter setting of the qualitative concepts of the composition of the speed control rules of an intelligent vehicle.

Parameter	Positive greater	Positive less	Zero	Negative less	Negative greater
$RS(d)$	(1.5, 0.2, 0.004)	(0.5, 0.15, 0.003)	(0, 0.08, 0.001)	(-0.5, 0.15, 0.003)	(-1.5, 0.2, 0.004)
$RS(\theta)$	(10, 1.2, 0.02)	(5, 1, 0.02)	(0, 1, 0.01)	(-5, 1, 0.02)	(-10, 1.2, 0.02)
$\delta$	(20, 3, 0.005)	(10, 2, 0.02)	(0, 2, 0.008)	(-10, 2, 0.02)	(20, 3, 0.05)

intelligent vehicle and the central axis of the lane is shown in Table 1(a). The rule set  $RS(\theta)$  of the included angle  $\theta$  between heading direction and the central axis of the lane is shown in Table 1(b). The parameter settings of the qualitative concepts in the rules are shown in Table 2.

**3.2. Lateral Control Algorithm.** The GCM and cloud reasoning can express human inference and decision and both exhibit strong robustness in solving the control problems of complicated systems. This study applies the GCM and the steering behavior imposed by cloud reasoning on drivers to build models. The model is shown in Figure 11.

The flowchart of the steering control algorithm shown in Figure 11 consists of two modules: steering wheel adjustment angle decision and steering wheel adjustment speed decision. The former comprises double-condition single-input and single cloud controllers, which are used to independently estimate the preview drift angle and the preview cornering distance. These controllers are called controller  $a$  and controller  $d$ , where  $u_a$  refers to the output of controller  $a$  and  $u_d$  refers to the output controller  $d$ .  $u_a, u_d \in [-1, 1]$  denotes adapting the expectation direction of the adjustment angle of the steering wheel. The left represents the negative values, whereas the right represents the positive values. A value stands for an expectation degree. When the value is closer to 1, the expectation degree is higher, and vice versa. The adjustment angle decision will constantly determine the adjustment of the steering wheel in terms of controller output. The module of the steering wheel adjustment speed decision, which consists of double-condition single-input and

single-cloud controllers, adopts the waterfall structure connection. It inputs the variables for the controller, which outputs the longitudinal velocity of the vehicle, and obtains the information of the adaptation speed of the wheel after adjusting speed. Negative values indicate adjusting the wheel to the left, whereas positive values indicate adjusting the wheel to the right. When the absolute value is closer to 1, the steering speed is faster. Conversely, the steering speed is lower when the absolute value is farther from 1.

## 4. Experiment Result and Analysis

### 4.1. Experiment Setup

**4.1.1. Hardware Architecture of an Intelligent Vehicle System.** The on-board sensor configuration of an intelligent vehicle comprises a radar sensor, a vision sensor, and a positioning sensor. The radar sensor consists of two separate Universal Transverse Mercator (UTM) single laser radars on the left and right of the body, a forward SICK single laser radar, a forward four-layer laser radar, and a backward millimeter wave radar. The vision sensor comprises three front-facing cameras, two rear-facing cameras, and two lateral cameras set in both rear-view mirrors. The positioning sensor consists of the Global Positioning System (GPS) and an inertial measurement unit (IMU), as is shown in Figure 12. This study is based on a ‘‘MengShi’’ intelligent vehicle, as is shown in Figure 13. All types of sensors are mainly applied to sense the surroundings of the vehicle for real-time acquisition of its location, posture, speed, and time.

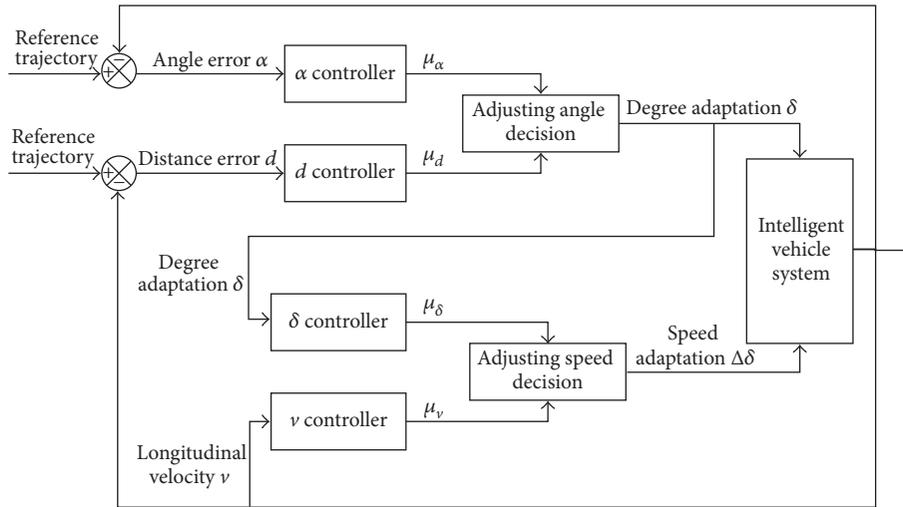


FIGURE 11: Lateral control algorithm flowchart.

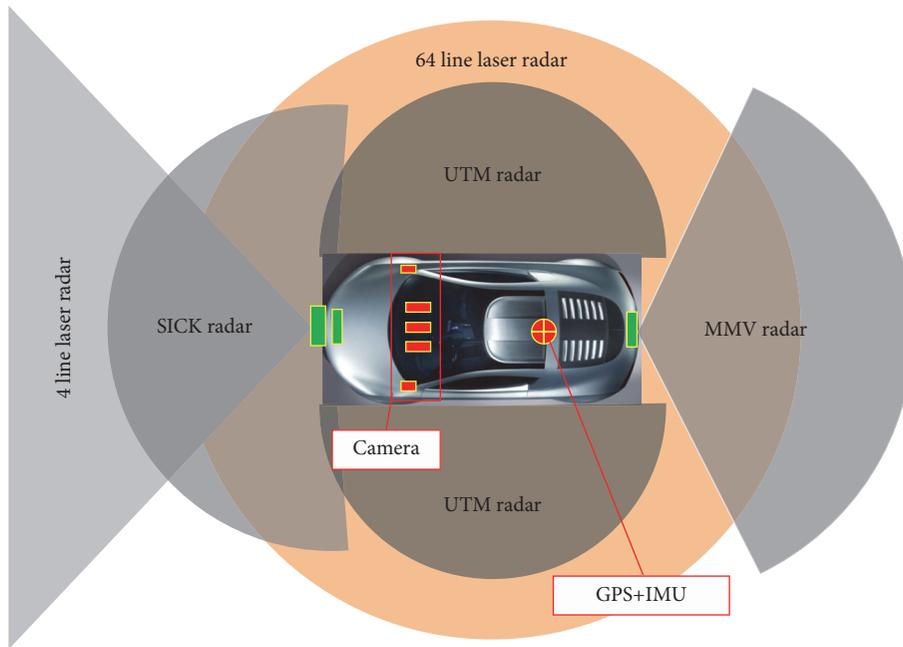


FIGURE 12: Experiment sensor configuration.



FIGURE 13: “MengShi” intelligent vehicle.

4.1.2. Software Architecture of an Intelligent Vehicle System.

The design and development of intelligent vehicles are aimed at studying the key techniques of multi-interaction and collaborative driving based on visual and auditory information. The software architecture of intelligent vehicle systems is shown in Figure 14. This architecture comprises a human computer interaction (HCI) layer, a sensor and sensing layer, a planning and decision layer, and a control layer.

*HCI Layer.* This layer receives the touch commands and emergency braking instructions of the driver and relays them

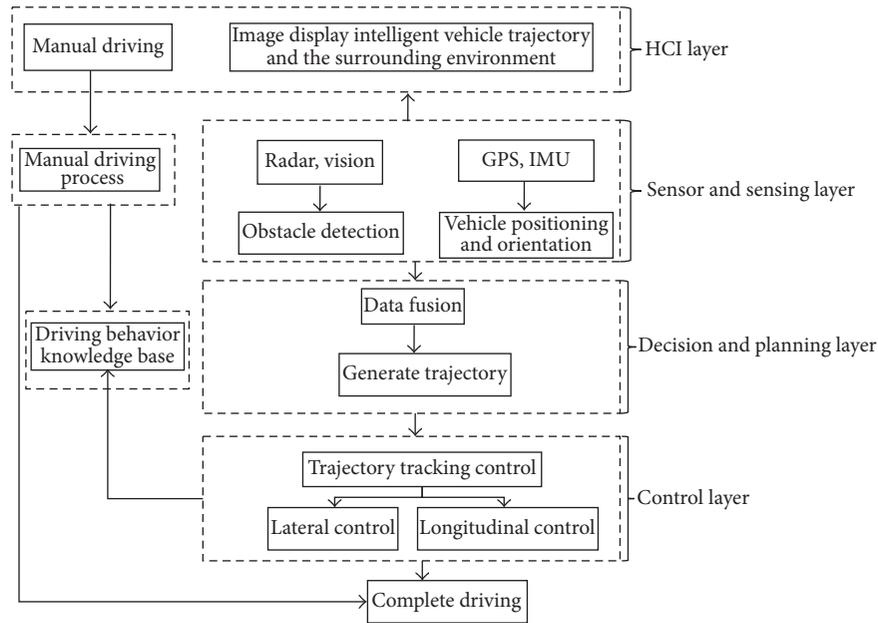


FIGURE 14: Software architecture of an intelligent vehicle system.

to the control layer. It simultaneously provides the driver with feedback information from the surroundings and other vehicles through sounds and images.

*Sensor and Sensing Layer.* This layer consists of a radar sensor, a vision sensor, a GPS sensor, and an IMU sensor. It focuses on completing the collection of sensor data. To realize the “plug and play” feature of the sensor, the standard data format of various sensors should be normative, which requires transforming the specific data format of the sensor to the standard format understood by an intelligent vehicle. The sensor data collected in this layer is delivered to the sensory module. The sensing layer focuses on sensor data analysis, road edge identification, obstacle detection, traffic sign detection, and body state estimation, which can facilitate the planning and decision of an intelligent vehicle.

*Decision and Planning Layer.* This layer focuses on path planning and navigation, which determine the driving pattern of an intelligent vehicle by analyzing environment data and vehicle data from the sensory module. This layer also determines the position of the vehicle in a detailed electronic map and generates the traveling track according to the coordinates of the target point. Human intervention and obstacles also influence the track.

*Control Layer.* This layer controls vehicles to enable them to proceed based on track data and current vehicle state. It also receives human instructions and performs acceleration/deceleration and steering operations. This layer directly outputs the control order to the accelerator, as well as the braking and steering controller, of the vehicle.

*4.1.3. Experimental Environment.* The Beijing-Tianjin Expressway, which spans the Taihu Toll Station and the Dongli

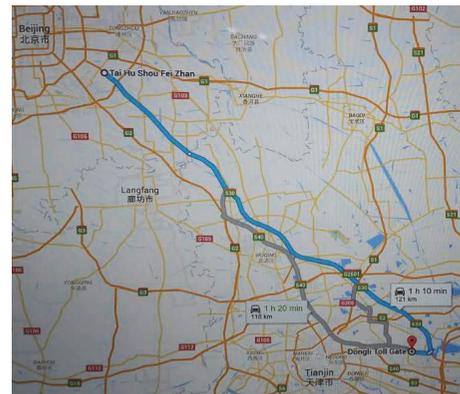


FIGURE 15: Experiments paths.

Toll Station, covers 121 km of shuttle distance. Rain is moderate rain in Tianjin, with a small amount of water on the ground. The weather is rainy in the Tianjin section of the Beijing-Tianjin Expressway. When the sun occasionally shines, the weather remains sunny until reaching Beijing, where it is cloudy. The temperature outside the vehicle is 32°C, and that on the road is 40°C. Visibility is over 200 m. The experiment path is designated by the blue line in Figure 15.

*4.2. Experiment Result and Analysis.* When the intelligent vehicle proceeds,  $\theta$  marks the included angle between the intelligent vehicle and the lane line. Negative and positive values refer to drifting to the left and right, respectively.  $d$  marks the distance between the geometric center of the vehicle and the lane line. The instant velocity is obtained using GPS. The control angle of the steering wheel target is calculated through a decision algorithm and recorded.

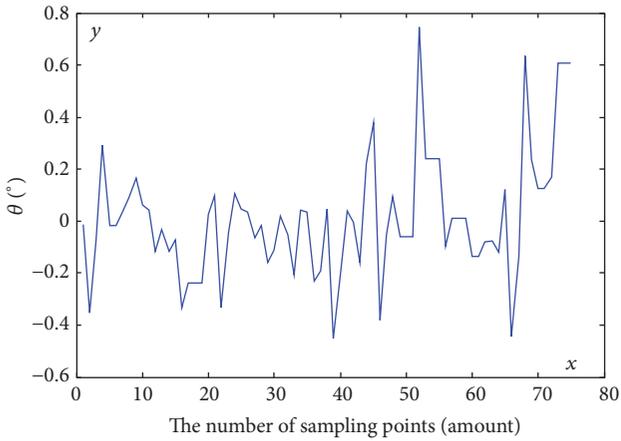


FIGURE 16: Variation curve of the included angle  $\theta$  between the vehicle body and the lane line.

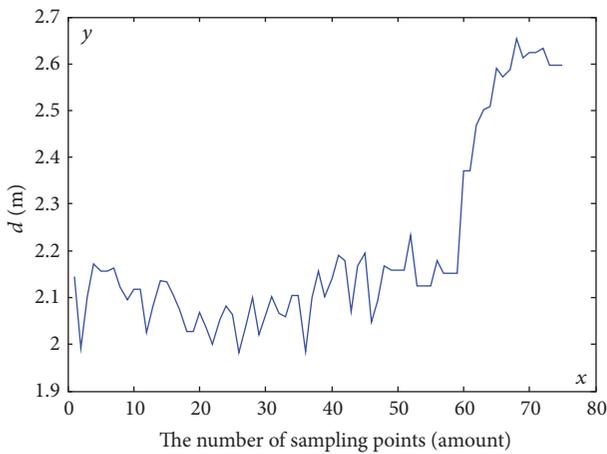


FIGURE 17: Variation curve of the distance  $d$  between the vehicle body and the lane line.

#### 4.2.1. Analysis of Maintenance Situations on Lanes under Different Speeds

(1) *Vehicle Speed Lower Than 80 km/h.* When speed is lower than 80 km/h, a section of real-time data (75 in total) is randomly selected for analysis. Figure 16 shows the variation curve of the included angle  $\theta$  between the vehicle body and the lane line. The  $x$ -axis represents the number of sampling points (unit: amount). The  $y$ -axis represents the included angle  $\theta$  (unit: degree). The value of the included angle  $\theta$  between the lane line and the heading direction of the vehicle body should range from  $-0.5^\circ$  to  $0.8^\circ$ , with a fluctuation range within  $1.3^\circ$ . Figure 17 shows the variation curve of distance  $d$  between the vehicle body and the lane line. The  $x$ -axis represents the number of sampling points (unit: amount). The  $y$ -axis represents the distance  $d$  (unit: m). The distance  $d$  should range from 0.2 m to 0.8 m, with a fluctuation range within 0.6 m. The data further demonstrate that the second half of the driving drifts toward the left side of the lane line by a wide margin, but the overall situation remains good.

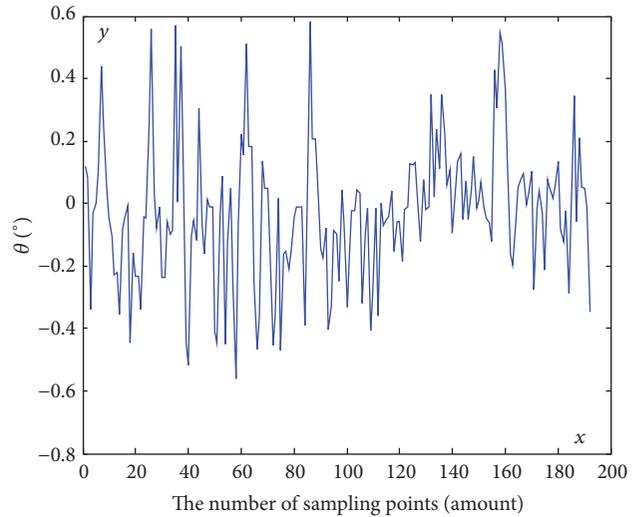


FIGURE 18: Variation curve of the included angle  $\theta$  between the vehicle body and the lane line.

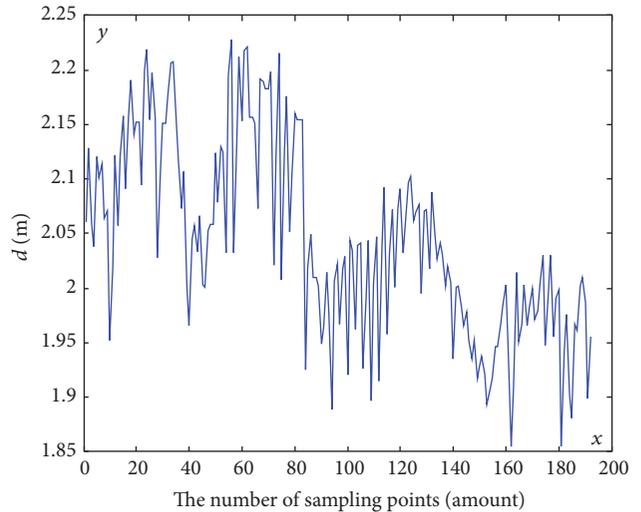


FIGURE 19: Variation curve of distance  $d$  between the vehicle body and the lane line.

(2) *Vehicle Speed between 80 km/h and 90 km/h.* When speed is between 80 km/h and 90 km/h, a section of real-time data (192 in total) is randomly selected analysis. Figure 18 shows the variation curve of the included angle  $\theta$  between the vehicle body and the lane line. The  $x$ -axis represents the number of sampling points (unit: amount), The  $y$ -axis represents the included angle  $\theta$  (unit: degree). The value of the included angle  $\theta$  between the lane line and the heading direction of the vehicle body should range from  $-0.6^\circ$  to  $0.6^\circ$ , with a fluctuation range within  $1.2^\circ$ . Figure 19 shows the variation curve of distance  $d$  between the vehicle body and the lane line. The  $x$ -axis represents the number of sampling points (unit: amount). The  $y$ -axis represents the distance  $d$  (unit: meter). The distance  $d$  should range from  $-0.1$  m to 0.4 m, with a fluctuation range within 0.5 m. The data show that the situation of the lane remains good.

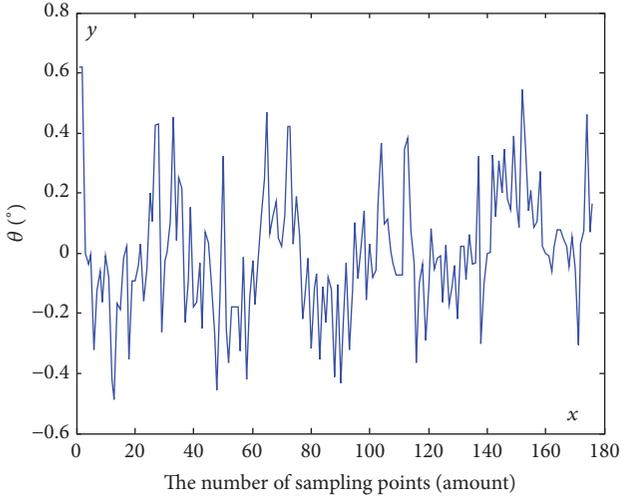


FIGURE 20: Variation curve of the included angle  $\theta$  between the vehicle body and the lane line.

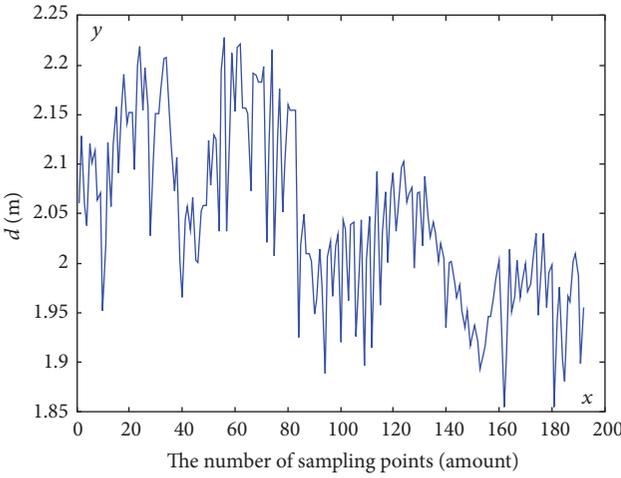


FIGURE 21: Variation curve of distance  $d$  between the vehicle body and the lane line.

(3) *Vehicle Speed between 90 km/h and 100 km/h.* When speed is between 90 km/h and 100 km/h, a section of real-time data (176 in total) is randomly selected for analysis. Figure 20 shows the variation curve of the included angle  $\theta$  between the vehicle body and the lane line. The  $x$ -axis represents the number of sampling points (unit: amount). The  $y$ -axis represents the included angle  $\theta$  (unit: degree). The value of the included angle  $\theta$  between the lane line and the heading direction of the vehicle body should range from  $-0.5^\circ$  to  $0.6^\circ$ , a range of  $1.1^\circ$ . Figure 21 shows the variation curve of the distance  $d$  between the vehicle body and the lane line. The  $x$ -axis represents the number of sampling points (unit: amount). The  $y$ -axis represents the distance  $d$  (unit: meter). The distance  $d$  should range from  $-0.25$  m to  $0.05$  m, with a fluctuation range of  $0.3$  m. The data show that the situation of the lane remains good.

(4) *Vehicle Speed Greater Than 100 km/h.* When speed is greater than 100 km/h, a section of real-time data (96 in

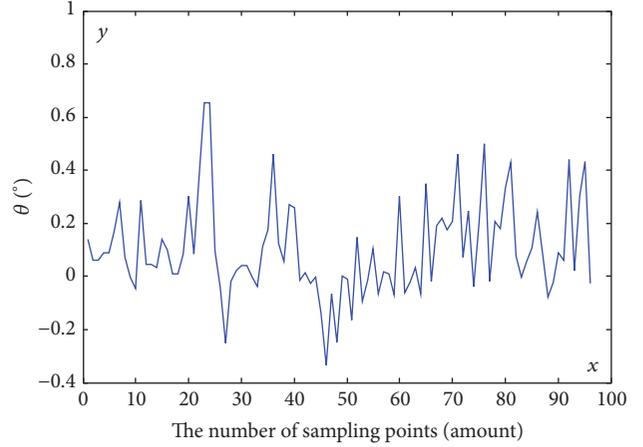


FIGURE 22: Variation curve of the included angle  $\theta$  between the vehicle body and the lane line.

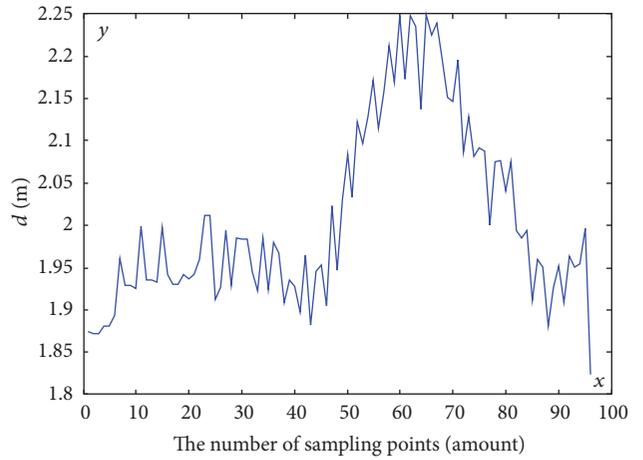


FIGURE 23: Variation curve of distance  $d$  between the vehicle body and the lane line.

total) is randomly selected for analysis. Figure 22 shows the variation curve of the included angle  $\theta$  between the vehicle body and the lane line. The  $x$ -axis represents the number of sampling points (unit: amount). The  $y$ -axis represents the included angle  $\theta$  (unit: degree). The value of the included angle  $\theta$  between the lane line and the heading direction of the vehicle body should range from  $-0.4^\circ$  to  $0.7^\circ$ , a fluctuation range within  $1.3^\circ$ . Figure 23 shows the variation curve of the distance  $d$  between the vehicle body and lane line. The  $x$ -axis represents the number of sampling points (unit: amount). The  $y$ -axis represents the distance  $d$  (unit: meter). The distance  $d$  should range from  $-0.6$  m to  $-0.2$  m, with a fluctuation range within  $0.4$  m. The data show that the lane situation of the lane remains good.

4.2.2. *Analysis of the Control Angle of the Steering Wheel.* The curve graph of the control angle of the steering wheel is shown in Figure 24. The  $x$ -axis represents the traveled mileage of the vehicle (unit: km). The data recording interval is 50 ms. The  $y$ -axis represents the control angle of the steering wheel (unit: degree).

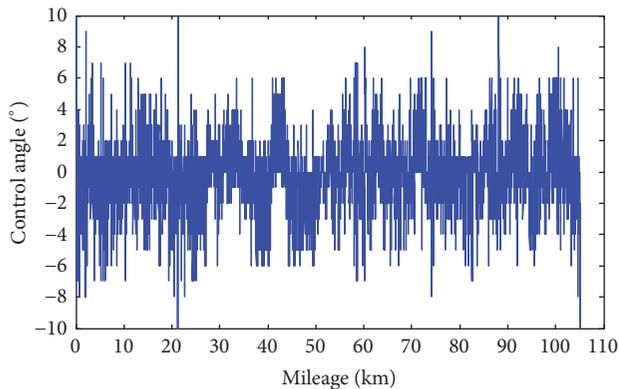


FIGURE 24: Curve graph of steering wheel control.

Steering to the right creates a negative value, whereas steering left creates a positive value. Approximately 81% of the angles of the steering wheel range from  $-3^\circ$  to  $3^\circ$ , 3% range from  $-6^\circ$  to  $6^\circ$ , and the maximum angle ranges from  $-7^\circ$  to  $7^\circ$ . According to relevant laws, the floating range of a manually operated steering wheel ranges from  $-7.5^\circ$  to  $7.5^\circ$ , which is a stable operation.

## 5. Conclusion

This study proposes a novel type of lateral control migration algorithm for intelligent vehicles. On the basis of the GCM and cloud reasoning, it also presents the qualitative concept cloud parameterization of the speed control rule for a vehicle on an expressway, designs a lateral control algorithm for an intelligent vehicle, and provides the speed control rules for different car-following conditions. The lateral controller of the vehicle, which is based on the GCM and the cloud reasoning algorithm, can be adapted to various speeds. Therefore, 81% of the angles of the steering wheel range from  $-3^\circ$  to  $3^\circ$ , 3% range from  $-6^\circ$  to  $6^\circ$ , and the maximum angle, which can achieve stable control, is within the range of  $-7^\circ$  to  $7^\circ$ .

## Competing Interests

The authors would like to declare that there is no conflict of interests regarding the publication of this paper.

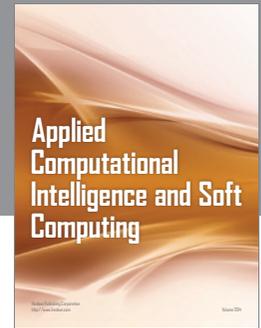
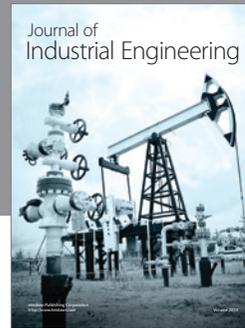
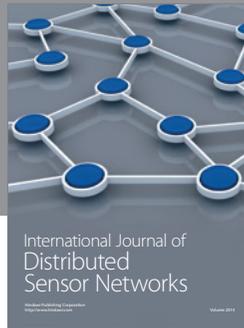
## Acknowledgments

This work is supported by National Natural Science Foundation of China under Grant nos. 61035004, 61273213, 61300006, 61305055, 90920305, and 61203366, the National Key Research and Development Program of China under Grant no. 2016YFB0100903, and the National High Technology Research and Development Program ("863" Program) of China under Grant no. 2015AA015401.

## References

- [1] D. J. Zhuang, Y. U. Fan, and Y. Lin, "The vehicle directional control based on fractional order PD $\sim$   $\mu$  controller," *Journal of Shanghai Jiaotong University*, vol. 41, no. 2, pp. 278–283, 2007.
- [2] Y. Li, K. H. Ang, and G. C. Y. Chong, "PID control system analysis and design," *IEEE Control Systems Magazine*, vol. 26, no. 1, pp. 32–41, 2006.
- [3] T. Hessburg and M. Tomizuka, "Fuzzy logic control for lateral vehicle guidance," *IEEE Control Systems*, vol. 14, no. 4, pp. 55–63, 1994.
- [4] R. Choomuang and N. Afzulpurkar, "Hybrid Kalman filter/Fuzzy logic based position control of autonomous mobile robot," *International Journal of Advanced Robotic Systems*, vol. 2, no. 3, pp. 207–213, 2005.
- [5] G. M. Scott, J. W. Shavlik, and W. H. Ray, "Refining PID controllers using neural networks," *Advances in Neural Information Processing Systems*, vol. 4, no. 5, pp. 746–757, 2008.
- [6] R. J. Wai, "Tracking control based on neural network strategy for robot manipulator," *Neurocomputing*, vol. 69, no. 7–9, pp. 425–445, 2003.
- [7] P.-J. He, K.-F. Ssu, and Y.-Y. Lin, "Sharing trajectories of autonomous driving vehicles to achieve time-efficient path navigation," in *Proceedings of the IEEE Vehicular Networking Conference (VNC '13)*, pp. 119–126, IEEE, Boston, Mass, USA, December 2013.
- [8] K.-W. Min and J.-D. Choi, "Design and implementation of autonomous vehicle valet parking system," in *Proceedings of the 16th International IEEE Conference on Intelligent Transportation Systems (ITSC '13)*, pp. 2082–2087, October 2013.
- [9] S. M. Lavelle and J. J. Kuffner, "Rapidly-exploring random trees: progress and prospects," in *Algorithmic & Computational Robotics: New Directions*, pp. 293–308, CRC Press, 2010.
- [10] E. Frazzoli, Z.-H. Mao, J.-H. Oh, and E. Feron, "Resolution of conflicts involving many aircraft via semidefinite programming," *Journal of Guidance, Control, and Dynamics*, vol. 24, no. 1, pp. 79–86, 2001.
- [11] R. H. Byrne and C. T. Abdallah, "Design of a model reference adaptive controller for vehicle road following," *Mathematical and Computer Modelling*, vol. 22, no. 4–7, pp. 343–354, 1995.
- [12] F. A. A. Cheein, C. De La Cruz, T. F. Bastos, and R. Carelli, "Slam-based cross-a-door solution approach for a robotic wheelchair," *International Journal of Advanced Robotic Systems*, vol. 7, no. 2, pp. 155–164, 2010.
- [13] A. Ghanbari and S. M. R. S. Noorani, "Optimal trajectory planning for design of a crawling gait in a robot using genetic algorithm," *International Journal of Advanced Robotic Systems*, vol. 8, no. 1, pp. 29–36, 2011.
- [14] P. C. Park and J. Miller, "Convergence properties of associative memory storage for learning control system," *Automation and Remote Control*, vol. 1989, no. 2, pp. 254–286, 1989.
- [15] S. Puntunan and M. Parnichkun, "Online self-tuning precompensation for a PID heading control of a flying robot," *International Journal of Advanced Robotic Systems*, vol. 3, no. 4, pp. 323–330, 2006.
- [16] M. Doi and Y. Mori, "Generalized minimum variance control for time-varying system," *Transactions of the Society of Instrument and Control Engineers*, vol. 45, no. 6, pp. 298–304, 2011.
- [17] D. R. Li, S. L. Wang, and D. Y. Li, *Cloud Model*, Spatial Data Mining, 2015.
- [18] D. Y. Li, Y. C. Liu, and Y. Du, "Artificial intelligence with uncertainty," *Journal of Software*, vol. 15, no. 11, pp. 1538–1594, 2004.
- [19] B. H. Cao, D. Y. Li, K. Qin et al., "An uncertain control framework of cloud model," in *Proceedings of the International Conference on Rough Set and Knowledge Technology (RSKT '10)*, pp. 618–625, Beijing, China, October 2010.

- [20] D. Li, C. Liu, and W. Gan, "A new cognitive model: cloud model," *International Journal of Intelligent Systems*, vol. 24, no. 3, pp. 357–375, 2009.
- [21] H. X. Gao, *Statistical Calculation*, Peking University Press, Beijing, China, 1995.
- [22] D. Y. Li and Y. Du, *Artificial Intelligence with Uncertainty*, National Defence Industry Press, Beijing, China, 2005.
- [23] H. Gao, J. Jiang, L. Zhang, L. Yuchao, and D. Li, "Cloud model: detect unsupervised communities in social tagging networks," in *Proceedings of the International Conference on Information Science and Cloud Computing Companion (ISCC-C '13)*, pp. 317–323, December 2013.
- [24] H. B. Gao, X. Y. Zhang, T.-L. Zhang, Y.-C. Liu, and D.-Y. Li, "Research of intelligent vehicle variable granularity evaluation based on cloud model," *Acta Electronica Sinica*, vol. 44, no. 2, pp. 365–373, 2016.
- [25] H. Liu and F. Sun, "Semi-supervised ensemble tracking," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '09)*, pp. 1645–1648, April 2009.
- [26] H. P. Liu, F. C. Sun, and M. Y. Yu, "Vehicle tracking using stochastic fusion-based particle filter," in *Proceedings of the International Conference on Intelligent Robots and Systems (IROS '07)*, pp. 2735–2740, San Diego, Calif, USA, 2007.
- [27] K. C. Di, *The Theory and Methods of Spatial Data Mining and Knowledge Discovery*, Wuhan Technical University of Surveying and Mapping, 1999.
- [28] Y. Du, *Research and Applications of Association Rules in Data Mining*, LA University of Science and Technology, 2000.
- [29] D. Y. Li, Y. Du, G. D. Yin et al., "Commonsense knowledge modeling," in *Proceedings of the 16th World Computer Congress*, pp. 34–45, Beijing, China, August 2000.



**Hindawi**

Submit your manuscripts at  
<http://www.hindawi.com>

