Article

A Study on Dynamic Motion Planning for Autonomous Vehicles Based on Nonlinear Vehicle Model

Xin Tang [1] **, Boyuan Li** [2,*] **and Haiping Du** [3,*]

1   Fok Ying Tung Research Institute, Hong Kong University of Science and Technology (HKUST),
     Guangzhou 511458, China
2   Research Centre for Intelligent Transportation, Zhejiang Lab., Hangzhou 311000, China
3   Faculty of Engineering and Information Science, University of Wollongong,
     Wollongong, NSW 2522, Australia
*   Correspondence: liboyuan@zhejianglab.com (B.L.); hdu@uow.edu.au (H.D.)

Abstract: Autonomous driving technology, especially motion planning and the trajectory tracking method, is the foundation of an intelligent interconnected vehicle, which needs to be improved urgently. Currently, research on path planning methods has improved, but few of the current studies consider the vehicle's nonlinear characteristics in the reference model, due to the heavy computational effort. At present, most of the algorithms are designed by a linear vehicle model in order to achieve the real-time performance at the cost of lost accuracy. To achieve a better performance, the dynamics and kinematics characteristics of the vehicle must be simulated, and real-time computing ensured at the same time. In this article, a Takagi–Sugeno fuzzy-model-based closed-loop rapidly exploring random tree algorithm with on-line re-planning process is applied to build the motion planner, which effectively improves the vehicle performance of dynamic obstacle avoidance, and plans the local obstacle avoidance path in line with the dynamic characteristics of the vehicle. A nonlinear vehicle model is integrated into the motion planner design directly. For fast local path planning mission, the Takagi–Sugeno fuzzy modelling method is applied to the modeling process in the planner design, so that the vehicle state can be directly utilized into the path planner to create a feasible path in real-time. The performance of the planner was evaluated by numerical simulation. The results demonstrate that the proposed motion planner can effectively generate a reference trajectory that guarantees driving efficiency with a lower re-planning rate.

Keywords: obstacle avoidance; path planning; Takagi–Sugeno fuzzy model; autonomous vehicle

1. Introduction

The path planning and path-tracking control strategies have attracted focused attention in recent years. The design of path-tracking control usually considers the detailed vehicle model to improve the control performance [1–3]. Particularly, a adaptive path-tracking control approach is proposed based on the lateral dynamics model with uncertain vehicle parameters [1]. Furthermore, a nonlinear sliding mode controller is proposed for the path-tracking control of an unmanned agricultural tractor by considering the kinematic model with wheel slip [2]. Yao et al. design a deep reinforcement training approach to learn the vehicle dynamics behaviour during the path-tracking [3]. However, considering the vehicle dynamics model in the path planning strategy design has been less addressed in the literature and will be the focused topic in our study.

The random search method based on vehicle dynamics model has been widely applied to conduct local path planning [4–6]. However, most of them are established based on a linear dynamics model to reduce the difficulty of controller design. The path planner using a complex nonlinear vehicle dynamics model [7,8] is still a big gap in the research. In order to optimize the speed of planning and convergence, and the stability of the fuzzy neural network, Xiong et al. [9] presented a fuzzy neural network controller based on a

Takagi–Sugeno (T–S) fuzzy model with simple membership function in the local obstacle avoidance path planning for autonomous mobile robots. However, the disadvantage of this type of algorithm is the planning of the obstacle avoidance path without consideration of the actual vehicle dynamics model. In vehicle attitude control, a high-speed vehicle in a complex dynamic environment has a limited degree of freedom compared to a low-speed robot with a high degree of freedom for rotation [10]. For this reason, it is difficult for an autonomous vehicle to avoid the dynamic obstacles quickly and ensure the stability of the vehicle dynamics simultaneously without considering the specific vehicle dynamic characteristics. More recently, Chen et al. presented an algorithm to optimize the path-planning performance based on the fourth-order Bezier curve [11]. However, the optimization of the feasible trajectory set in this research just considers the lateral slip of the vehicle body without the consideration of a vehicle dynamics model and constraints of vehicle actuators. Shiller and Gwo [12] presented a planner integrated with a single point dynamics model to analyze the alternative paths, and then the planner chose the feasible path selected from the ones that the vehicle does not slip on when steering. However, in this algorithm the trajectory generation should conduct the forward simulation on the vehicle's dynamic model continuously, which makes it hard to meet the real-time control requirement.

On the other hand, choosing an appropriate algorithm makes the planning results faster and more feasible. The traditional dynamic path-planning methods, such as the RRT algorithm [13,14] and road grid search algorithm [15], conduct a random search program to create a series of alternative paths in advance, and to compare the paths one by one to select the optimal path. In order to improve controller performance in real-time [16], and to make it more robust [17], some researchers are trying to optimize the method of random search in dynamic path planning. Frazzoli and Dahleh proposed an optimal path search method [18], which applies a closed-loop planning process (to compare possible local target points by random sampling method with the optimal cost function) instead of an open-loop calculation process (to conduct random sampling in existing routes with multiple iterations). In [19], Kuwata and Jonathan further proposed an RRT algorithm integrated with a closed-loop forward simulation to conduct real-time dynamic path planning. It is noted in the feasibility analysis of the optional path that the planner effectively considers the constraints of vehicle rollover limit and the safety zone limit for obstacle avoidance. With this preliminary consideration, the feasible obstacle avoidance path can be selected under the high dynamic environmental conditions. However, it is also noted from the experiment result that the high nonlinearity of the vehicle model greatly limits the real-time performance and accuracy of the controller's forward simulation process. Therefore, this controller is more suitable to be used in the situation of relative low vehicle speed rather than a high-speed situation in real-time dynamic path planning.

It can be seen from the abovementioned research that the influence of dynamic model nonlinearity on planner design has increasingly drawn attention, and how to simultaneously improve the planner's real-time performance and the applied dynamic model completeness should be further studied. Representing a nonlinear vehicle dynamic model by a T–S fuzzy model [20], the path planner can be directly applied to the vehicle model to plan the path with the consideration of the nonlinear vehicle's dynamic characteristics. In this paper, a Takagi–Sugeno fuzzy-model-based closed-loop rapidly exploring random tree (TS-RRT) planning algorithm is proposed. The nonlinear vehicle dynamics model is described by the T–S fuzzy modelling process. In particular, the dynamic response of the vehicle is precisely described by the T–S fuzzy model for the closed-loop search process and on-line re-planning program. Thus, the optimal local path can be obtained more directly by combining the vehicle dynamics and the environmental information into the path planner. After the uniform boundedness of systems and global asymptotic stability are proven in the controller design process, the steering and braking control output can be calculated by the TS-RRT planner in real-time, and then a proportional–integral–derivative (PID) controller is applied to conduct the trajectory tracking considering the command signal and the vehicle state.

In Section 2, the dynamics model is established and linearized by the T–S fuzzy modelling process. Section 3 presented the closed-loop RRT online planning program and the corresponding trajectory controller design. Section 4 presented the results of simulation evaluation. Finally, the conclusions are drawn in Section 5.

2. Modeling of the Vehicle System

The autonomous vehicle is a nonlinear system with strong coupling characteristics and uncertainty. Since the real vehicle dynamics model has high nonlinearity, it is very difficult to apply a linear path-planning algorithm in the practical experiment. For this reason, a T–S modelling method is utilized in this study to build the vehicle dynamic model and maintain the precision of the model. It is noted that this research focuses on the obstacle avoidance path planning in a highly dynamic environment, therefore, the applied dynamic model mainly takes the consideration of the coordinated control system of steering and braking, without a discussion of the driving force input system.

2.1. Takagi–Sugeno Fuzzy Model

A simplified three degree of freedom nonlinear vehicle dynamic model is applied in this research, as shown in Figure 1, which can be effectively described by longitudinal speed, lateral speed, and yaw rate. The dynamic model is derived under the following assumptions:

(1) The vertical, roll, and pitch motion is ignored;
(2) The braking and steering dynamic are approximated to a linear first-order system;
(3) The influence of suspension on tire axle is ignored [21]. The nonlinear model used in our work to represent the vehicle's dynamics is of the general form as follows:

$$
\begin{aligned}
\dot{v}_x &= -\frac{c_x}{m}v_x + \frac{2k_f\left(v_y + l_f\dot{\psi}\right)}{mv_x}\delta + \frac{1}{m}a + \tau(\Delta x), \\
\dot{v}_y &= -\frac{2\left(k_f + k_r\right)}{m}v_y - \left[v_x + \frac{2\left(k_f l_f - k_r l_r\right)}{m}\right]\dot{\psi} \\
&\quad + \frac{2k_f}{m}\delta - \frac{c_y}{m}v_y + \tau(\Delta y), \\
\ddot{\psi} &= -\frac{2\left(k_f l_f^2 - k_r l_r^2\right)}{I_z}\dot{\psi} - \frac{2\left(k_f l_f - k_r l_r\right)}{I_z}v_y \\
&\quad + \frac{2k_f l_f}{I_z}\delta + \tau(\Delta\psi), \\
\dot{\delta} &= \frac{1}{T_z}(\delta_c - \delta), \\
\dot{a} &= \frac{1}{T_a}(a_c - a),
\end{aligned}
\tag{1}
$$

where v_x, v_y, and ψ represent the longitudinal speed, lateral speed, and yaw angle, respectively; m represents the total mass of the vehicle; I_z is the yaw inertia; l_f and l_r the distance from the front axle and the rear axle to the center of gravity, respectively; c_x and c_y are the air resistance coefficients of longitudinal movement and lateral movement, respectively; f_r represents rolling resistance coefficient; k_f and k_r are the stiffness of front and rear tires, respectively; $\tau(\Delta x)$, $\tau(\Delta y)$, and $\tau(\Delta\psi)$ represent the external disturbances and uncertainties caused by time-varying parameters and unmodeled dynamics; T_z and T_a are the first-order hysteretic quantities of the relevant control output reference quantities; δ_c is the control input of steering angle; a_c is the acceleration

control input of vehicle body; δ is the actual steering angle of the vehicle; a is the actual longitudinal acceleration of the vehicle body, and the input signal constraints are

$$a_{\min} < a < a_{\max},$$

$$\delta_{\max} \geq \|\delta\|, \tag{2}$$

$$\dot{\delta}_{\max} \geq \|\dot{\delta}\|,$$

where $a_{\min}$ and $a_{\max}$ are the maximum and minimum acceleration of vehicle body, respectively; $\delta_{\max}$ represents the maximum steering angle of vehicle; and $\dot{\delta}_{\max}$ represents the maximum slew rate for steering.
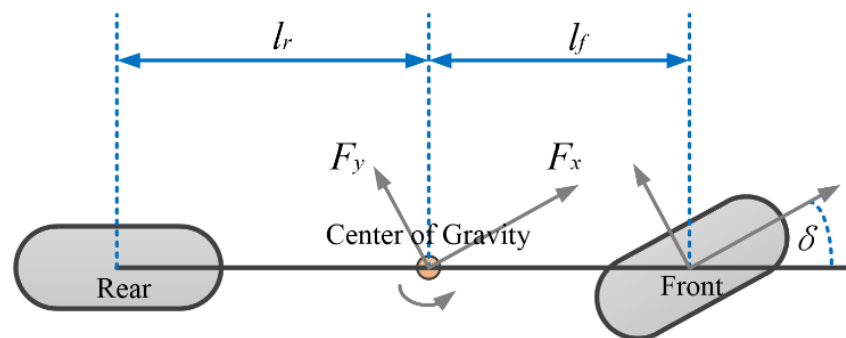

Figure 1. Vehicle dynamics model.

It is noted that the vehicle dynamics model in (1) is based on the assumption that the uncertainty and external disturbance of the model are limited in a certain range. Under this assumption, we have the continuous function $\prod \tau_i (i = 1,2,3)$ satisfying the following inequality conditions

$$\tau(\Delta x) \leq \overline{\tau}_1\left(v_x, v_y, \dot{\psi}\right),$$

$$\tau(\Delta y) \leq \overline{\tau}_2\left(v_x, v_y, \dot{\psi}\right), \tag{3}$$

$$\tau\left(\Delta\dot{\psi}\right) \leq \overline{\tau}_3\left(v_x, v_y, \dot{\psi}\right).$$

For the process of T–S fuzzy modelling, the state variables of the vehicle dynamics model can be defined as follows:

$$x_1 = v_x, \quad x_2 = v_y, \quad x_3 = \dot{\psi}, \quad x_4 = \delta, \quad x_5 = a, \tag{4}$$

and the state vector as:

$$x = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 & x_5 \end{bmatrix}^T. \tag{5}$$

We can write a state-space for system (1) as:

$$\dot{x} = Ax + B_1 w + B_2 u, \tag{6}$$

where w is the external disturbance of the vehicle, u is the input of the control system, and

$$w = \begin{bmatrix} w_1 & w_2 & w_3 \end{bmatrix}^T, u = \begin{bmatrix} u_1 & u_2 \end{bmatrix}^T$$

where $w_1 = \tau(\Delta x)$, $w_2 = \tau(\Delta y)$, $w_3 = \tau\left(\Delta\dot{\psi}\right)$, $u_1 = \delta_c$ and $u_2 = a_c$. Adding the state to Equation (1), we obtain

$$
\begin{aligned}
\dot{x}_1 &= -\frac{c_x}{m}x_1 + \frac{2k_f\left(x_2+l_fx_3\right)}{mx_1}x_4 + \frac{1}{m}x_5 + w_1, \\
\dot{x}_2 &= -\frac{2\left(k_f+k_r\right)}{m}x_2 - \left[x_1 + \frac{2\left(k_fl_f-k_rl_r\right)}{m}\right]x_3 \\
&\quad + \frac{2k_f}{m}x_4 - \frac{c_y}{m}x_2 + w_2, \\
\dot{x}_3 &= -\frac{2\left(k_fl_f^2-k_rl_r^2\right)}{I_z}x_3 - \frac{2\left(k_fl_f-k_rl_r\right)}{I_z}x_2 \\
&\quad + \frac{2k_fl_f}{I_z}x_4 + w_3, \\
\dot{x}_4 &= \frac{1}{T_z}(u_1 - x_4), \\
\dot{x}_5 &= \frac{1}{T_a}(u_2 - x_5).
\end{aligned}
\tag{7}
$$

In order to simplify the high nonlinearity of the proposed model, f_1 and f_2 are defined as:

$$
\begin{aligned}
f_1 &= \frac{2k_f\left(x_2+l_fx_3\right)}{mx_1}, \\
f_2 &= x_1 + \frac{2\left(k_fl_f-k_rl_r\right)}{m}.
\end{aligned}
\tag{8}
$$

Then, Equation (7) can be rewritten as

$$
\begin{aligned}
\dot{x}_1 &= -\frac{c_x}{m}x_1 + f_1\cdot x_4 + \frac{1}{m}x_5 + w_1, \\
\dot{x}_2 &= -\frac{c_y+2\left(k_f+k_r\right)}{m}x_2 - f_2\cdot x_3 + \frac{2k_f}{m}x_4 + w_2, \\
\dot{x}_3 &= -\frac{2\left(k_fl_f-k_rl_r\right)}{I_z}x_2 - \frac{2\left(k_fl_f^2-k_rl_r^2\right)}{I_z}x_3 \\
&\quad + \frac{2k_fl_f}{I_z}x_4 + w_3, \\
\dot{x}_4 &= \frac{1}{T_z}(-x_4 + u_1), \\
\dot{x}_5 &= \frac{1}{T_a}(-x_5 + u_2).
\end{aligned}
\tag{9}
$$

Therefore, the matrices A, B_1 B_2 can be written as

$$
A_{(5\times5)} =
\begin{bmatrix}
-\frac{c_x}{m} & 0 & 0 & f_1 & \frac{1}{m} \\
0 & -\frac{c_y+2\left(k_f+k_r\right)}{m} & -f_2 & \frac{2k_f}{m} & 0 \\
0 & -\frac{2\left(k_fl_f-k_rl_r\right)}{I_z} & -\frac{2\left(k_fl_f^2-k_rl_r^2\right)}{I_z} & \frac{2k_fl_f}{I_z} & 0 \\
0 & 0 & 0 & -\frac{1}{T_z} & 0 \\
0 & 0 & 0 & 0 & \frac{1}{T_a}
\end{bmatrix},
$$

$$
B_{1(5\times3)} =
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0
\end{bmatrix}^T,\quad
B_{2(5\times2)} =
\begin{bmatrix}
0 & 0 & 0 & \frac{1}{T_z} & 0 \\
0 & 0 & 0 & 0 & \frac{1}{T_a}
\end{bmatrix}^T.
$$

The T–S fuzzy modelling method is applied here to approximate the high nonlinearity of the system (6). It is noted that the state variables x_1, x_2, and x_3 are actually limited for a

stable system, and that the nonlinear f_1 and f_2 should also be bounded. We represent f_1 and f_2 using their minimum values and maximum values by following a "sector nonlinearity" approach:

$$f_1 = M_1 f_{1\text{max}} + M_2 f_{1\text{min}},$$
$$f_2 = N_1 f_{2\text{max}} + N_2 f_{2\text{min}}, \tag{10}$$

where $f_{(i)max}$ ($i = 1, 2$) represents the maximum values and $f_{(i)min}$ ($i = 1, 2$) is the minimum value of the nonlinear $f_{(i)}$ ($i = 1, 2$). $M_{(i)}$ and $N_{(i)}$ ($i = 1, 2$) are fuzzy membership functions and satisfy:

$$M_1 + M_2 = 1,$$
$$N_1 + N_2 = 1, \tag{11}$$

and the member functions are defined as:

$$M_1 = \frac{f_1 - f_{1\text{min}}}{f_{1\text{max}} - f_{1\text{min}}}, \quad M_2 = \frac{f_{1\text{max}} - f_1}{f_{1\text{max}} - f_{1\text{min}}},$$
$$N_1 = \frac{f_2 - f_{2\text{min}}}{f_{2\text{max}} - f_{2\text{min}}}, \quad N_2 = \frac{f_{2\text{max}} - f_2}{f_{2\text{max}} - f_{2\text{min}}}. \tag{12}$$

Then, the nonlinear vehicle model system can be described by the above linear subsystems. For each possibility, there is a corresponding state–space equation:

$$\text{if } f_1 = M_1, \ f_2 = N_1, \ \text{then } \dot{x} = A_{(1)}x + B_1 w + B_2 u,$$
$$\text{if } f_1 = M_1, \ f_2 = N_2, \ \text{then } \dot{x} = A_{(2)}x + B_1 w + B_2 u,$$
$$\text{if } f_1 = M_2, \ f_2 = N_1, \ \text{then } \dot{x} = A_{(3)}x + B_1 w + B_2 u,$$
$$\text{if } f_1 = M_2, \ f_2 = N_2, \ \text{then } \dot{x} = A_{(4)}x + B_1 w + B_2 u,$$

where $A_{(i)}(i = 1, 2, 3, 4)$. ($i = 1, 2, 3, 4$) are obtained by replacing $f_{(i)}(i = 1, 2)$ in matrix A of (6) with $f_{(i)max}$ and $f_{(i)min}$. Then, the T–S fuzzy model for the nonlinear vehicle model under the bounded state variables is obtained as:

$$\dot{x} = \sum_{i=1}^{4} h_i \left[A_{(i)}x + B_1 w + B_2 u \right]$$
$$= A_h x + B_1 w + B_2 u, \tag{13}$$

where $A_h = \sum_{i=1}^{4} h_i \cdot A_i$, and $h_1 = M_1 N_1$, $h_2 = M_1 N_2$, $h_3 = M_2 N_1$, $h_4 = M_2 N_2$, and $h_{(i)}(i = 1, 2, 3, 4)$ satisfy $\Sigma_{i=1}^{4} h_1 = 1$.

2.2. State Observer

In practice, not all the state variables are available to be measured in real-time. In particular, the velocities are nearly unmeasurable since the direct integration from acceleration to estimate the vertical velocities means the accuracy of the estimation deteriorates as a consequence. To meet input requirements, a planner must be constructed using the estimated state variables and premise variables; that is, to estimate the state variables in real-time, a state observer is designed and integrated with the planner. In terms of the vehicle model, both the steering angle, δ, and the longitudinal acceleration of vehicle body, a, can be measured by sensors. Therefore, the observer measurement is defined as:

$$Y = \begin{bmatrix} \delta & a \end{bmatrix}^T$$
$$= C_1 x, \tag{14}$$

where

$$C_1 = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

To effectively estimate the state by using the easily measured signals, the estimation error can be defined based on the observer measurement as

$$e = x - \hat{x}. \tag{15}$$

Therefore, the state observer can be designed as

$$\begin{aligned} \dot{\hat{x}} &= \sum_{i=1}^{4} h_i \left[A_{(i)}\hat{x} + L(Y - \hat{Y}) + B_2 u \right] \\ &= A_h \hat{x} + L(Y - \hat{Y}) + B_2 u, \end{aligned} \tag{16}$$

where L are the state observer gains to be designed. Rearrangement of (16) provides:

$$\dot{\hat{x}} = (A_h - LC_1)\hat{x} + LY + B_2 u. \tag{17}$$

In Equation (15), the error vector of real and estimated states is defined as $e = x - \hat{x}$., then the dynamic behavior of error can be deduced as:

$$\begin{aligned} \dot{e} &= \dot{x} - \dot{\hat{x}} \\ &= A_h x + B_1 w + B_2 u - (A_h - LC_1)\hat{x} - B_2 u - LY \\ &= A_h x - (A_h - LC_1)\hat{x} - LC_1 x + B_1 w \\ &= (A_h - LC_1)e + B_1 w. \end{aligned} \tag{18}$$

Here, we have two assumptions that the external disturbance w is a Gaussian white noise, whose mean value tends to be zero, and the values of (A_h^T, C_1^T) are fully controllable and fully measurable. In system (18), the attenuation of the error is determined by the poles' locations of the matrix $(A_h - LC_1)$. For a known system, A_h and C_1 are determined by the system characteristics. Therefore, the gain matrix L of the designed observer should be chosen to maintain the stability of the system, which is degenerated into a pole assignment problem. Specifically, if the matrix $(A_h - LC_1)$ has an appropriate eigenvalue, the error of the system has a certain decay rate to make the system stable. We have

$$\det[\lambda I - (A_h - LC_1)] = \det\left[\lambda I - (A_h - LC_1)^T\right] = \det\left[\lambda I - \left(A_h^T - L^T C_1\right)\right]$$

and since the object control model has been linearized by the Takagi–Sugeno modelling process, the observer design problem can be transformed into a pole assignment problem of (A_h^T, C_1^T). To set $L = \begin{bmatrix} 0 & 0 & 0 & l_1 & 0 \\ 0 & 0 & 0 & 0 & l_2 \end{bmatrix}^T$, then $(A_h - LC_1)$ can be represented as

$$(A_h - LC_1) =$$

$$\begin{bmatrix} -\frac{c_x}{m} & 0 & 0 & f_1 & \frac{1}{m} \\ 0 & -\frac{c_y + 2(k_f + k_r)}{m} & -f_2 & \frac{2k_f}{m} & 0 \\ 0 & -\frac{2(k_f l_f - k_r l_r)}{I_z} & -\frac{2(k_f l_f^2 - k_r l_r^2)}{I_z} & \frac{2k_f l_f}{I_z} & 0 \\ 0 & 0 & 0 & -\frac{1}{T_z} + l_1 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{T_a} + l_2 \end{bmatrix}.$$

In order to ensure that the system is approaching zero, the eigenvalue set λ of the system need to all be negative values. Then, the corresponding matrix L content (l_1 and l_2) can be calculated to complete the observer design. Thus, the T–S fuzzy model and the corresponding state observer is established and can be applied in the closed-loop RRT planning program.

3. Control System Design

To determine the vehicle control input, the existing stochastic planning algorithms generally apply a look-up table to perform the reverse calculation based on the sampled control input value. In this section, the T–S fuzzy-model-based closed-loop RRT algorithm is proposed, which is integrated with a low-order controller to expand the RRT and conduct the on-line re-planning process by considering the closed-loop dynamics. Different from the existing work [15], the proposed TS-RRT samples the input of the stable closed-loop system composed of the controller and the T–S fuzzy dynamic state space. The complete planning and control system is shown in Figure 2.
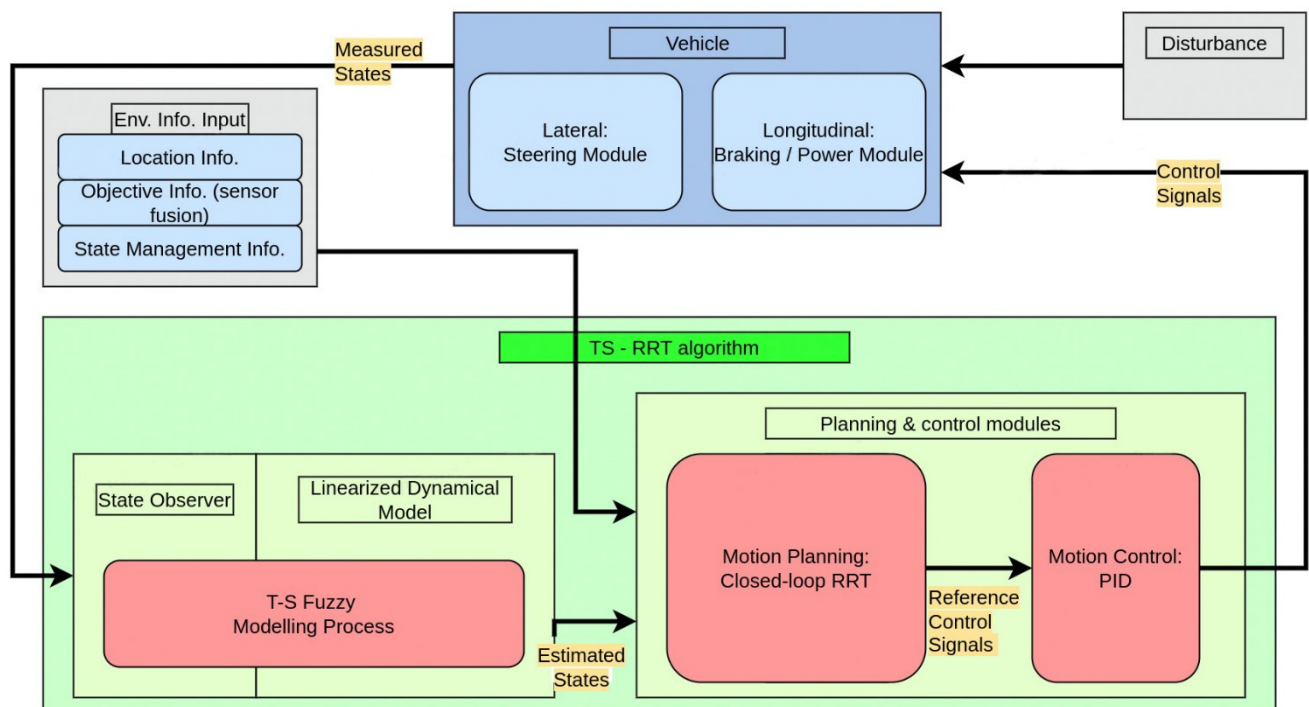

Figure 2. The control flow of the closed-loop system.

3.1. Takagi–Sugeno Fuzzy-Model-Based Path Planner

In this study, the controller output consists of a series of tuples, which include the steering angle profile of the steering controller and the speed command profile of the speed controller. The TS-RRT uses the controller outputs and the T–S fuzzy dynamics model to conduct a forward simulation process, to calculate the predicted trajectory, and then to check the feasibility of the controller output signal according to the vehicle and environmental boundaries including rollover and obstacle avoidance constraints.

The main idea of this TS-RRT algorithm is to rapidly reduce the distance between a randomly selected node and the tree until all nodes meet the planning requirements. The goal is to find a feasible path from the start point (x_m, y_m) to the end point (x_{goal}, y_{goal}). Note that the term q show below is equivalent to (x, y). The exploration process of the RRT planner is shown as follow:

1. Generate a random pot q_{rand};
2. Find the node $q_{nearest}$ to q_{rand} on the tree;

3.   Connect q_{rand} and $q_{nearest}$;
4.   Search for nodes on the tree with q_{rand} as the center and r_c as the radius;
5.   Find out the potential set of parent nodes $q_{parent\_potential}$. The purpose is to update q_{rand} to find if there are any better parent nodes;
6.   Start to evaluate a random note of potential parent $q_{parent\_potential}$;
7.   Calculate the cost of $q_{parent\_potential}$ as a parent node;
8.   Before the detection of collision, connect $q_{parent\_potential}$ with q_{child} (that is, q_{rand}) first;
9.   Calculate the cost of this path $\Omega(t)$, $t \in [t_1, t_2]$;
10. Compare the cost of the new path to the cost of the original path. If the cost of the new path is less, conduct the collision detection on it, or it should be replaced by the next potential parent node;
11. Collision detection failed, the $q_{parent\_potential}$ is not a new parent;
12. Start to evaluate the next potential parent;
13. Connect potential parent nodes to q_{child};
14. Calculate the cost of this path $\Omega(t)$, $t \in [t_3, t_4]$;
15. Again, compare the cost of the new path to the cost of the original path. If the cost of the new path is less, conduct the collision detection on it, or it should be replaced by the next potential parent node;
16. Collision detection passed;
17. Delete the previous edge in the tree;
18. Add the new edge in the tree, and make $q_{parent\_potential}$ as q_{parent}.

Thus, the trajectory of a finite period, which depends on the cycle of calculation, can be predicted by the expansion process of the random tree shown above. However, in a dynamic and uncertain environment, trees need to grow continuously in the process of execution, due to the continuously updated vehicle dynamic status and environmental information. Therefore, real-time planning requires the vehicle dynamic model state and the reuse of the information from previous calculation cycles [22,23]. The re-planning program of the TS-RRT planner, which is also demonstrated in Figure 3, is designed as follows:

1.   Open the re-planning program;
2.   Update the current vehicle T–S fuzzy states $x_{TS}(t_0)$;
3.   Update the environmental constraints $\Gamma_{\text{feasible}}(t)$ from the obstacle configuration space;
4.   Apply the state observer to propagate the states by a computational time step Δt and obtain $x_{TS}(t_0 + \Delta t)$;
5.   Conduct the random tree exploring process;
6.   Until calculation time limit Δt is reached;
7.   If no such sequence exists, then send emergency stop to controller and return to step 2;
8.   End if;
9.   Choose the best safe node sequence in the tree;
10. Re-propagate the latest T–S fuzzy state $x_{TS}(t_0 + \Delta t)$ using the $\Omega(t)$ with the best node sequence, and then obtain $x(t)$;
11. If $x_{TS}(t) \in \Gamma_{\text{feasible}}(t)$, then send the best potential path $\Omega(t)$ to the controller;
12. If anything else, delete the previous infeasible path in the tree and return step 9;
13. End if;
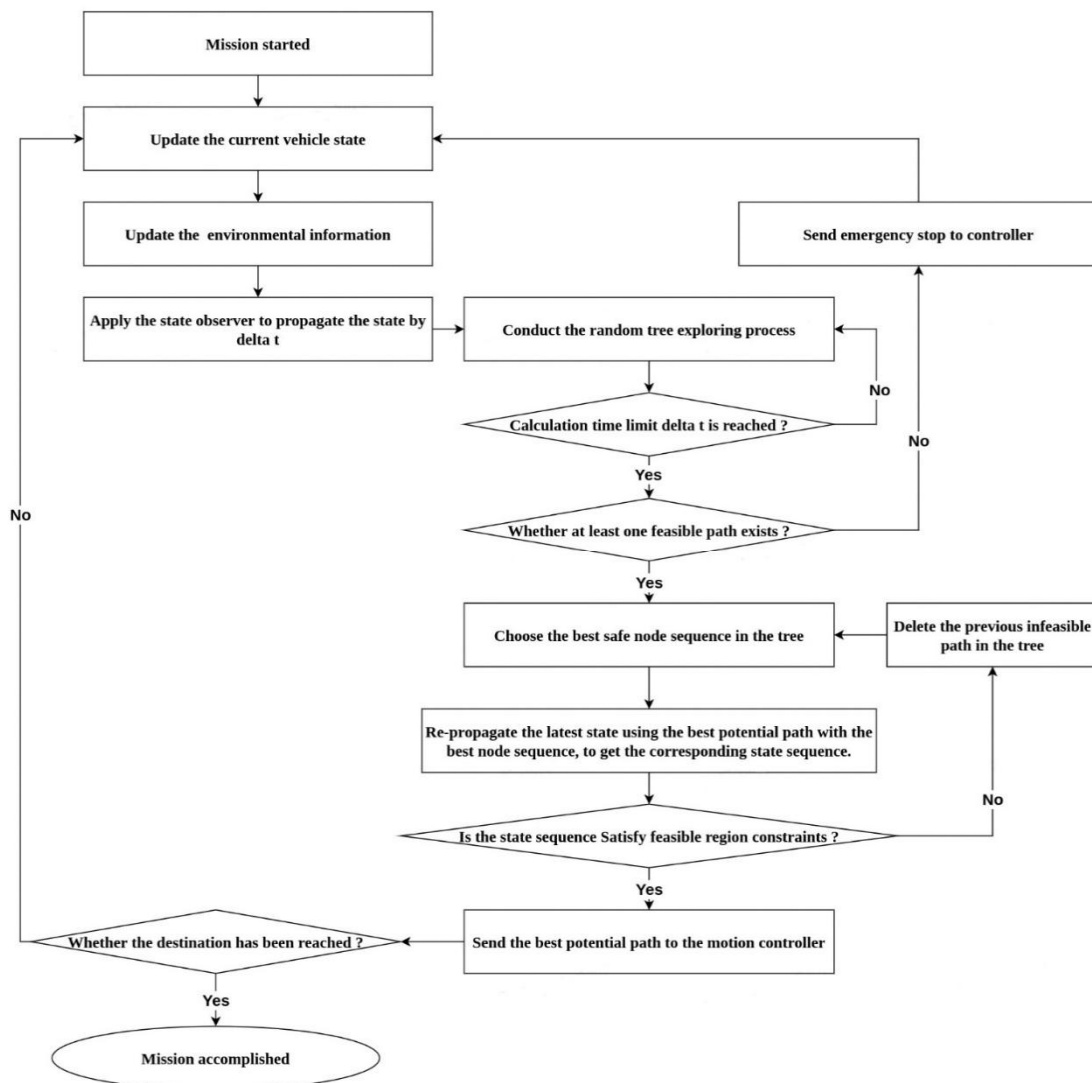14. Until the vehicle reaches goal.

Figure 3. The pipeline of the TS-RRT planner.

The algorithm shown above illustrates the program of the TS-RRT algorithm to execute the part of tree exploring and to grow the tree while the controller executes the path planning in real-time. The planner sends the input to the controller at a fixed rate per second and the extension of the tree continues until the time limit (line 7) is reached. After each computation cycle, the best track is selected as the node sequence with control input $U_{cmd} = [\delta_{cmd} \quad a_{cmd}]^T$. These signals are sent to the controller to be added to the reference path for trajectory control (line 13).

It is noted that when selecting the best path, only the sequence of nodes that end in a safe state is considered. If not, the planner commands the controller to perform an emergency brake to stop the car as soon as possible for security.

3.2. Trajectory Controller Design

During the TS-RRT planner conducting the path planning program, we use a simple and effective PID controller to track the designed trajectory in real-time.

For acceleration tracking, a simple PID controller is considered. However, due to the inherent speed damping of the vehicle and the noise of the acceleration signal, a PID

controller has no obvious advantage to a PI controller, which has fewer parameters to be designed. Therefore, a PI controller is applied here and shown as follows:

$$a_c = K_{p(a)}(a_{cmd} - a) + K_{i(a)} \int_0^t (a_{cmd} - a) dr \tag{19}$$

where u is the dimensionless speed control signal; K_p and K_i are proportional gain and integral gain, respectively.

Similarly, our steering controller is designed as:

$$\delta_c = K_{p(\delta)}(\delta_{cmd} - \delta) + K_{\delta(\delta)} \int_0^t (\delta_{cmd} - \delta) dr \tag{20}$$

After constructing the planner and the trajectory controller, the T–S fuzzy model is left aside in the simulation work, and the complete closed-loop control system is shown in Figure 2.

For vehicles with complex dynamics, the dimension of vehicle state might be established with a very high dimension. However, the control signal output from the applied PID controller has a lower dimension, which can effectively guarantee the real-time performance of trajectory tracking.

4. Numerical Evaluation

In the section, an experiment in the loop simulation was conducted with a 1000 Hz real-time simulation frequency to verify the algorithm's effectiveness. The nonlinear vehicle dynamic model, considering the dug-off tire model [24], is applied as the plant model. Meanwhile, the designed TS-RRT planner runs at 20 Hz based on an AGX Xavier chip with the vehicle model parameters listed in Table 1. To validate the performance of the TS-RRT, a traditional RRT planner is adopted for a comparison. The simulation result in Figure 4 presents the planned path of the autonomous vehicle in the presence of three moving obstacles. The vehicle adjusts its speed and direction when it detects the moving obstacle in its path, and sometimes it slows down when making its decision. It is noted that only the results of TS-RRT are shown in Figure 4, and the results of the competing controller, traditional RRT, are omitted for clarity.

Table 1. Model parameters.

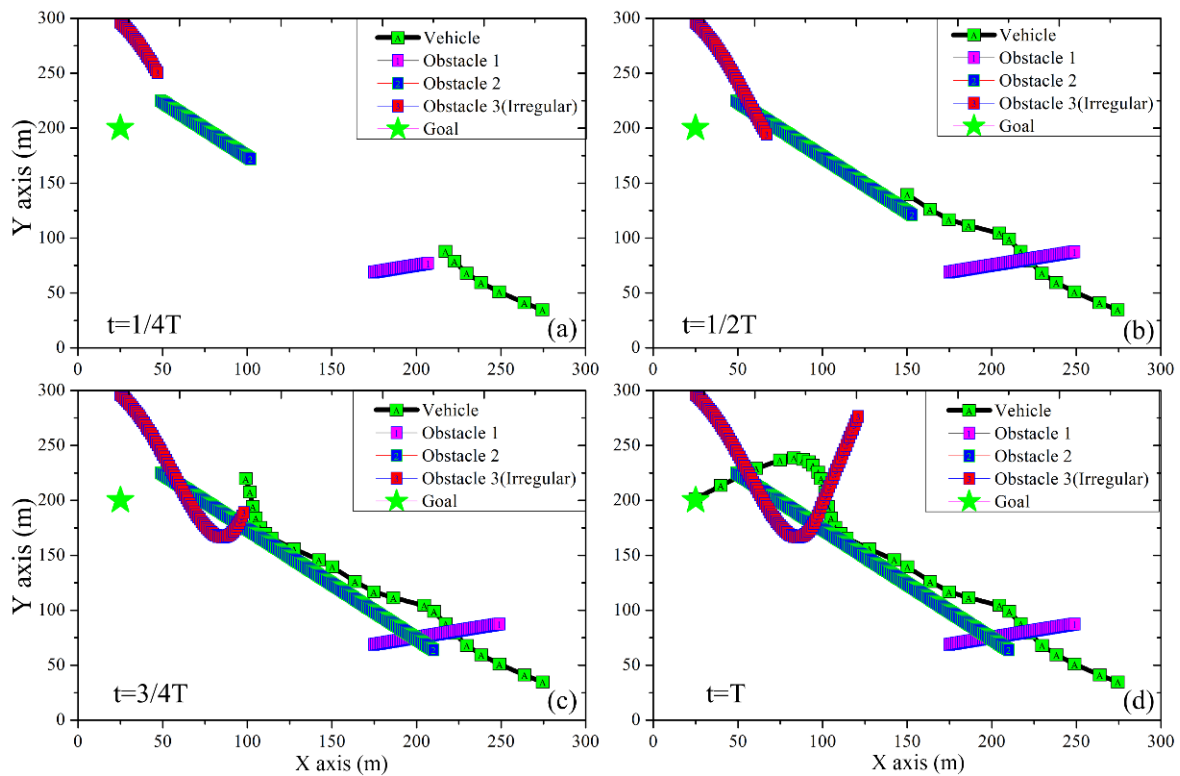| Parameter	Value	Parameter	Value
δ_{max}	0.539 rad	m	1589 kg
$\dot{\delta}_{max}$	0.331 rad/s	I_z	36,918 kg·m^2
a_{min}	-8.2 m/s^2	l_f	1.38 m
a_{max}	3.8 m/s^2	l_r	1.67 m
T_a	0.35 s	k_f	379 kN/m
T_z	0.35 s	k_r	388 kN/m

Figure 4. The planning results in dynamic environment: (**a**) at 1/4 of the total time, (**b**) at half of the total time, (**c**) at 3/4 of the total time, (**d**) at the end of the planned period.

To further illustrate the effectiveness of the TS-RRT algorithm, we constructed two test scenes including a highway case and a parking case, using a traditional artificial potential field (APF) method and an A* algorithm as a competitor for the simulation. The results of several typical scenarios are shown in Figures 5 and 6.
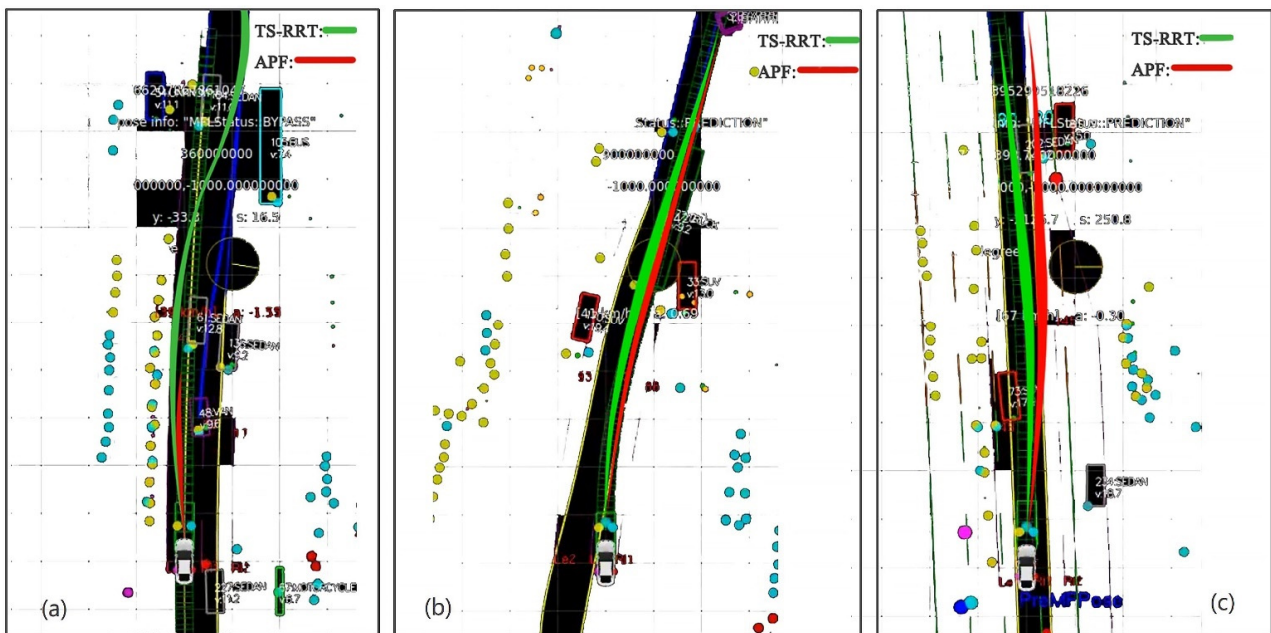
Figure 5. Comparison of algorithm effectiveness under motorway running: (**a**) bypassing in jam-packed case, (**b**) bypassing in normal case, (**c**) overtaking case.
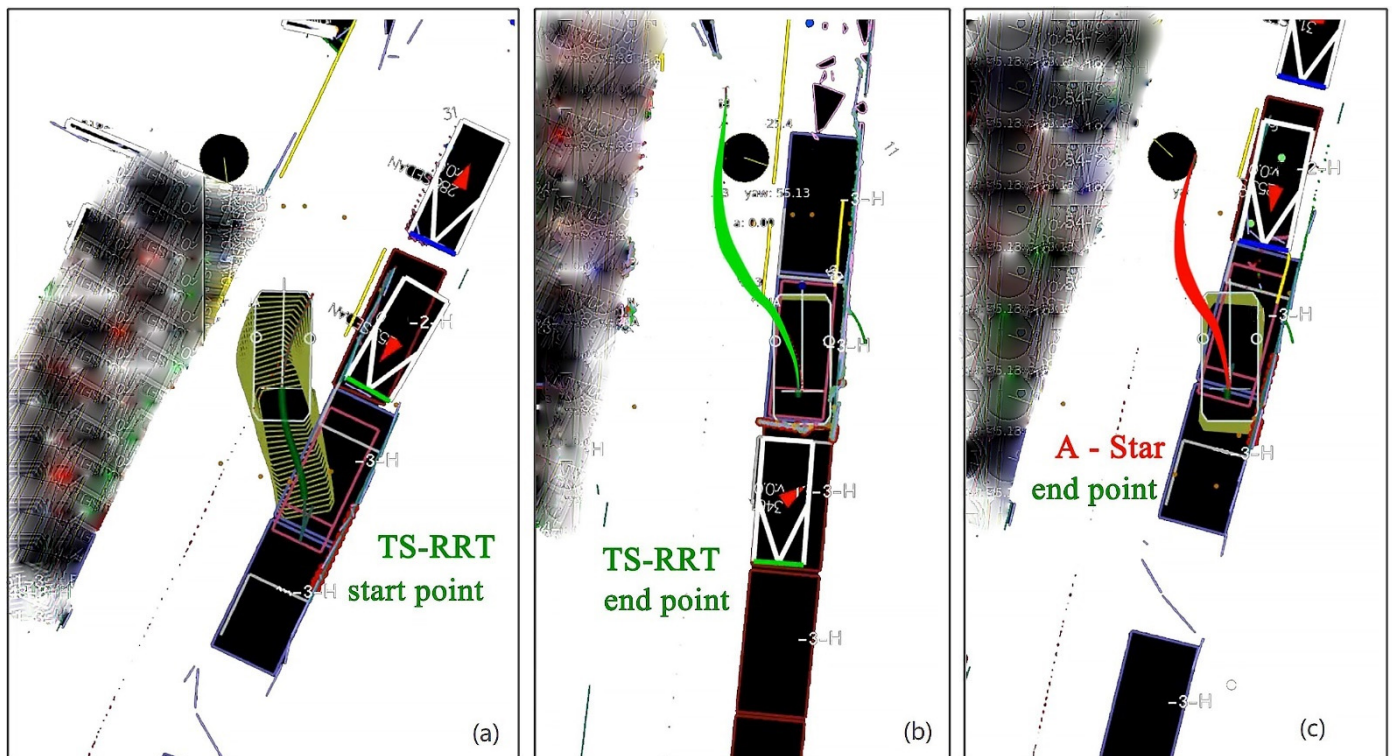
Figure 6. Comparison of algorithm effectiveness in parking planning: (**a**) start status of TS-RRT planning, (**b**) end status of TS-RRT with successful planning, (**c**) end status of A* planning with collision failure.

The simulation result in Figure 5 presents the planned paths of the two competing planners, TS-RRT and APF, in the scene of motorway driving. It can be detected from Figure 5a that the APF algorithm has the same planning effect as TS-RRT when the environment complexity is low. However, in a complex and jam-packed environment, as shown in Figure 5b, the potential field force of the APF algorithm is trapped in the dilemma of local optimum, which results in the planning failure and the vehicle's emergency stop. In Figure 5c, it can be seen that the front car is intending to slow down and merge to the left lane. In this case, with rolling optimization that includes re-planning and stitching, the TS-RRT planner generates a shorter and smoother trajectory than the APF one to complete the overtaking task.

The result in Figure 6 presents the planned paths of the two competing planners, TS-RRT and A*, in the scene of parallel parking. Comparing the results in Figure 6b,c, it can be found that the TS-RRT algorithm, which effectively considers the vehicle kinematics and dynamic characteristics, is appropriate for dealing with the narrow parking scenarios, and can complete the side parallel parking planning without collision. On the other hand, although the A* planner is improved with grid subdivision processing and accessibility path filtering, the final planned path is not available (a collision occurs) since the algorithm is lacking in dynamic characteristics consideration in the situation of tire slipping (simulated by the low road friction coefficient setting) and deceleration strip passing.

In order to illustrate the simulation results more clearly, the performance of the TS-RRT algorithm, the traditional RRT algorithm, and APF algorithm are compared in Table 2 in terms of the total planned distance, the time consumption, and the average vehicle speed. The value of the specific object is the average of ten simulation results. By the comparing the results, it can be seen that the proposed TS-RRT algorithm is optimized in terms of the total distance and the time consumption compared with the competing methods in the dynamic obstacle environment.

**Table 2.** Comparison of results (average value of 10 times).

Strategy	APF Algorithm	Traditional RRT	TS-RRT
Trajectory distance	338.30 m	325.93 m	304.23 m
Time consumption	24.09 s	25.11 s	21.38 s
Average vehicle speed	14.04 m/s	12.98 m/s	14.23 m/s

Table 3 illustrates the planning results and the time consumption of the two controllers in ten simulations. Under the dynamic obstacle environment, the search speed of TS-RRT is significantly faster compared with the traditional RRT search algorithm, and the search success rate is greatly improved with more propagations.

Table 3. Iterations and time consumed to generate the optimal track.

Strategy	Traditional RRT	TS-RRT
Potential trajectory	86	112
Total propagation	11,408	14,253
Time cost	11.85 s	10.24 s
Propagation/cycle	310	402
Time cost/cycle	0.32 s	0.29 s

Furthermore, the observer's effect that collected the simulation are organized and shown in Figure 7. Specifically, the four subfigures demonstrate the results of the two-observable states, steering angle and longitudinal acceleration, with measured inputs from sensors and the corresponding estimated value from the state observer, respectively. Moreover, the estimated input and output signals of the estimator calculate the delay as around 0.5–1 ms, which could be accepted within a 5 ms calculation period mission.
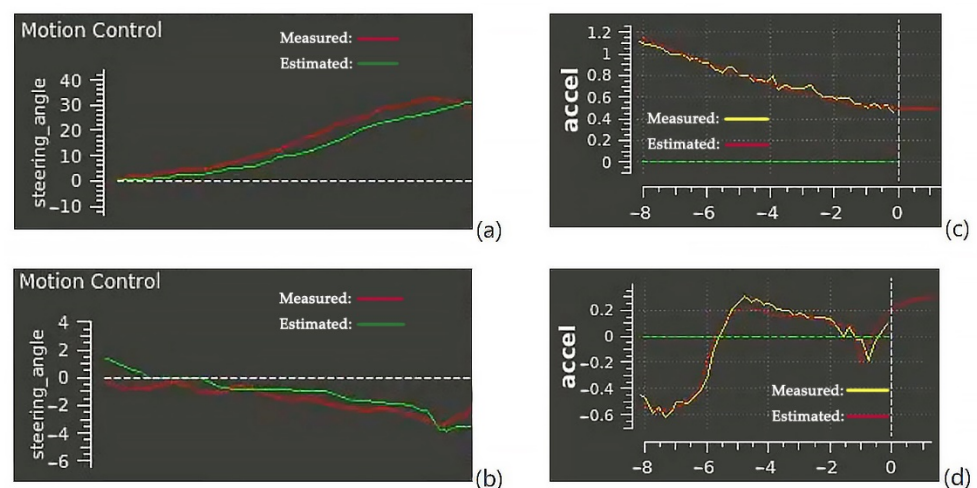
Figure 7. State monitoring and estimation: (**a**) steering angle with large amplitude, (**b**) steering angle with small amplitude, (**c**) longitudinal acceleration with large amplitude, (**d**) longitudinal acceleration with small amplitude.

5. Conclusions

In this paper, a nonlinear vehicle dynamic model is considered in the control system design. For local path planning, the T–S fuzzy modelling method is applied to the nonlinear dynamic model to help the path planner to create a feasible path. Then, a closed-loop RRT algorithm with an on-line re-planning process is applied to build the path planner, which

effectively improves the vehicle performance of dynamic obstacle avoidance, and plan the local obstacle avoidance path in line with the dynamic characteristics of the vehicle. Finally, the performance of the planner is evaluated by numerical simulation. The results demonstrate that the proposed controller can effectively plan the path and support a favorable trajectory.

References

1. Mohammadzadeh, A.; Taghavifar, H. A novel adaptive control approach for path tracking control of autonomous vehicles subject to uncertain dynamics. *Proc. Inst. Mech. Eng. Part D J. Automob. Eng.* **2020**, *234*, 2115–2126. [CrossRef]
2. Matveev, A.S.; Hoy, M.; Katupitiya, J.; Savkin, A.V. Nonlinear sliding mode control of an unmanned agricultural tractor in the presence of sliding and control saturation. *Robot. Auton. Syst.* **2013**, *61*, 973–987. [CrossRef]
3. Yao, J.; Ge, Z. Path-Tracking Control Strategy of Unmanned Vehicle Based on DDPG Algorithm. *Sensors* **2022**, *22*, 7881. [CrossRef] [PubMed]
4. Pepy, R.; Lambert, A.; Mounier, H. Reducing navigation errors by planning with realistic vehicle model. In Proceedings of the IEEE Intelligent Vehicles Symposium, Tokyo, Japan, 13–15 June 2006; pp. 300–307.
5. Pepy, R.; Lambert, A.; Mounier, H. Path planning using a dynamic vehicle model. In Proceedings of the 2006 2nd International Conference on Information & Communication Technologies, Berkeley, CA, USA, 25–26 May 2006; pp. 781–786.
6. Yang, S.; Wang, Z.; Zhang, H. Kinematic model based real-time path planning method with guide line for autonomous vehicle. In Proceedings of the 2017 36th Chinese Control Conference (CCC), Dalian, China, 26–28 July 2017; pp. 990–994.
7. Park, C.; Pan, J.; Manocha, D. ITOMP: Incremental trajectory optimization for real-time replanning in dynamic environments. In Proceedings of the Twenty-Second International Conference on Automated Planning and Scheduling, Atibaia, Sao Paulo Brazil, 25–29 June 2012.
8. Bouraine, S.; Fraichard, T.; Salhi, H. Provably safe navigation for mobile robots with limited field-of-views in dynamic environments. *Auton. Robot.* **2012**, *32*, 267–283. [CrossRef]
9. Xiong, K.; Hua, H. Path plan for mobile robot under dynamic environment based on the actual membership function. *Bull. Sci. Technol.* **2015**, *31*, 168–173.
10. ChaborAlwawi, B.K.; Roth, H.; Kazem, B.I.; Abdullah, M.W. Mobile robot motion planning and multi objective optimization using improved approach. *Int. J. Mech. Eng. Robot. Res.* **2015**, *4*, 325–330. [CrossRef]
11. Chen, C.; He, Y.; Bu, C.; Han, J. Feasible trajectory generation for autonomous vehicles based on quartic bezier curve. *Acta Autom. Sin.* **2015**, *44*, 486–496.
12. Shiller, Z.; Gwo, Y. Dynamic motion planning of autonomous vehicles. *IEEE Trans. Robot. Autom.* **1991**, *7*, 241–249. [CrossRef]
13. LaValle, S.; Kuffner, J.; Donald, B. *Algorithmic and Computational Robotics: New Directions 2000 WAFR*, 1st ed.; A K Peters/CRC Press: Boca Raton, FL, USA, 2001.
14. Zucker, M.; Kuffner, J.; Branicky, M. Multipartite RRTs for rapid replanning in dynamic environments. In Proceedings of the 2007 IEEE International Conference on Robotics and Automation, Roma, Italy, 10–14 April 2007; pp. 1603–1609.
15. Branicky, M.; LaValle, S.; Olson, K. Quasi-randomized path planning. In Proceedings of the 2001 ICRA IEEE International Conference on Robotics and Automation, Seoul, Korea, 21–26 May 2003; pp. 1481–1487.
16. An, Z.; Hao, L.; Liu, Y.; Dai, L. Development of mobile robot SLAM based on ROS. *Int. J. Mech. Eng. Robot. Res.* **2016**, *5*, 47–51. [CrossRef]
17. Widhiada, W.; Nindhia, T.; Budiarsa, N. Robust control for the motion five fingered robot gripper. *Int. J. Mech. Eng. Robot. Res.* **2015**, *4*, 226–232. [CrossRef]
18. Frazzoli, E.; Dahleh, M.; Feron, E. Real-time motion planning for agile autonomous vehicles. *J. Guid. Control. Dyn.* **2002**, *25*, 116–129. [CrossRef]
19. Kuwata, Y.; Teo, J.; Fiore, G.; Karaman, S.; Frazzoli, E.; How, J. Real-time motion planning with applications to autonomous urban driving. *IEEE Trans. Control. Syst. Technol.* **2009**, *17*, 1105–1118. [CrossRef]
20. Tang, X.; Du, H.; Sun, S.; Ning, D.; Xing, Z.; Li, W. Takagi-Sugeno fuzzy control for semi-active vehicle suspension with a magnetorheological damper and experimental validation. *IEEE/ASME Trans. Mechatron.* **2016**, *22*, 291–300. [CrossRef]

21. Guo, J.; Li, K.; Luo, Y. Coordinated control of autonomous four wheel drive electric vehicles for platooning and trajectory tracking using a hierarchical architecture. *J. Dyn. Syst. Meas. Control* **2015**, *137*, 101001. [CrossRef]
22. Stentz, A. Optimal and efficient path planning for partially known environments. In *Intelligent Unmanned Ground Vehicles*; Springer: Boston, MA, USA, 1997; pp. 203–220.
23. Petti, S.; Fraichard, T. Safe motion planning in dynamic environments. In Proceedings of the 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, Edmonton, AB, Canada, 2–6 August 2005; pp. 2210–2215.
24. Boada, B.; Boada, M.; Díaz, V. Fuzzy-logic applied to yaw moment control for vehicle stability. *Veh. Syst. Dyn.* **2005**, *43*, 753–770. [CrossRef]