

Research Article

Feasible Path Planning for Autonomous Vehicles

Vu Trieu Minh¹ and John Pumwa²

¹ Department of Mechatronics, Tallinn University of Technology, Ehitajate tee 5, 19086 Tallinn, Estonia

² Department of Mechanical Engineering, Papua New Guinea University of Technology, Lae 411, Papua New Guinea

Correspondence should be addressed to Vu Trieu Minh; vutrieuminh@yahoo.com

Received 8 January 2014; Revised 10 February 2014; Accepted 11 February 2014; Published 18 March 2014

Academic Editor: Sivaguru Ravindran

Copyright © 2014 V. T. Minh and J. Pumwa. 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.

The objective of this paper is to find feasible path planning algorithms for nonholonomic vehicles including flatness, polynomial, and symmetric polynomial trajectories subject to the real vehicle dynamical constraints. Performances of these path planning methods are simulated and compared to evaluate the more realistic and smoother generated trajectories. Results show that the symmetric polynomial algorithm provides the smoothest trajectory. Therefore, this algorithm is recommended for the development of an automatic control for autonomous vehicles.

1. Introduction

Path planning is one of the most important parts of an autonomous vehicle. It deals with, searching a feasible path, taking into consideration the geometry of the vehicle and its surroundings, the kinematic constraints and others that may affect the feasible paths. Study of this paper can be used to develop a real-time control system for autonomous ground vehicles which can track exactly on any feasible paths from the global positioning system (GPS) maps or/and from unmanned aerial vehicle (UAV) images. This system can be also applied to autonomous unmanned ground vehicles (on road or off road) and autoparking and autodiving systems.

The main idea of this paper is a system which can automatically generate an optimal feasible trajectory and then control the vehicle to track exactly on this path from any given starting points to any given destination points from a map subject to the vehicle physical constraints on obstacles, speed, steering angle, and so forth.

From the existed pieces of literature on this research topic, there are still few research papers dealing with optimal trajectory generation subject to real automotive engineering constraints. Basic studies on the flatness, nonholonomic, and nonlinear systems can be read from the recent book of Lévine in [1] where the fundamental motion planning of a vehicle is presented. The flatness and controllability conditions for this nonlinear system are also well investigated and defined.

However, in this research, parking simulation of a two-trailer vehicle is also demonstrated but without the constraint of steering angle and steering angular velocity.

The problem of trajectory generation for nonholonomic system is also presented by Dong and Guo in [2] where two trajectory generation methods are proposed. The control inputs are the second-order polynomial equations. By integrating those control inputs, coefficients for those second-order polynomial equations are found. However this paper is lacking constraint analysis on the vehicle velocity and the steering angle.

The cell mapping techniques and reinforcement learning methods to obtain the optimal motion planning by Gomez et al. in [3] are tested for vehicles considering kinematics, dynamics, and obstacle constraints. This paper shows a simulation of a wheeled mobile robot moving on a path generated by the bang-bang trajectory generation. And the paper does not mention the algorithms used for generating the vehicle trajectory. Several other research papers on optimal trajectories and control of autonomous vehicles can be read in [4–7]; however, most of those studies are based on the real traffic flow roads and the control algorithms are to perform the maneuver tasks such as lane changing, merging, distance keeping, velocity keeping, stopping, and collision avoidance.

This paper, therefore, focuses on the applicable algorithms to generate feasible trajectories from a starting point

to any destination point subject to vehicle constraints. This paper refers to the vehicle sideslip model and estimation by Minh in [8, chapter 8]. The nonlinear computational schemes for the nonlinear systems are referred to by Minh and Afzulpurkur in [9]. Although the holonomic dynamical trajectories generation is not new, recent research articles have tried to develop those trajectories in different applications. Trajectory for autonomous ground vehicles based on Bezier curves operating under waypoints and corridor constraints can be referred to by Choi et al. in [10] and a real-time trajectory generation for car-like vehicles navigating dynamic environments can be read by Delsart et al. in [11]. A new paper of trajectory generation model-based IMM tracking for safe driving in intersection scenario can be read by Zhou et al. in [12] to reduce traffic accidents at intersections. Another paper of dynamic trajectory generation for wheeled robots is referred to by Missura and Behnke in [13] where a dynamic motion of wheeled robots can be determined in real-time onboard. The real vehicle models and conditions for stabilizability of dynamic systems are referred to in [14, 15].

This paper introduces three (flatness, polynomial, and symmetric) conventional holonomic trajectories generation subject to steering angle constraint for autonomous vehicles. Performances of these three algorithms are compared. Data taken from simulation on the vehicle longitudinal velocity and its steering angular velocity will be used to develop the control algorithms for the vehicle tracking on those trajectories in the next part of this paper. The outline of this paper is as follows. Section 2 describes the kinematic model; Section 3 presents the flatness method; Section 4 produces the trajectory generation subject steering angle constrained violation; Section 5 presents the polynomial method; Section 6 develops the symmetric polynomial method; Section 7 analyses the performance of the three methods; finally conclusion is drawn in Section 8. Simulations in this paper are conducted in the MATLAB Simulink R2009a.

2. Kinematic Model of a Vehicle

The constraints for this vehicle model are based on the assumption that the wheels are rolling without slipping and the steering angle is simplified as a single wheel in the midpoint of the two front wheels. Then, a kinematic model of a vehicle can be drawn in Figure 1.

The kinematic model of a forward rear-wheel driving vehicle can be written as

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \cos \theta \\ \sin \theta \\ \frac{\tan \varphi}{l} \\ 0 \end{bmatrix} r v_1 + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} v_2, \quad (1)$$

where $X = [x, y, \theta, \varphi]'$ is the system state variables, (x, y) are the Cartesian coordinates of the middle point of the rear wheel axis, θ is the angle of the vehicle body to the x -axis, φ is the steering angle, l is the vehicle wheel base, r is the wheel radius, v_1 is the angular velocity of the rear wheel, and v_2 is the angular steering velocity. Given the initial state $X(0) = [x_0, y_0, \theta_0, \varphi_0]'$ at time $t = 0$ and the final state

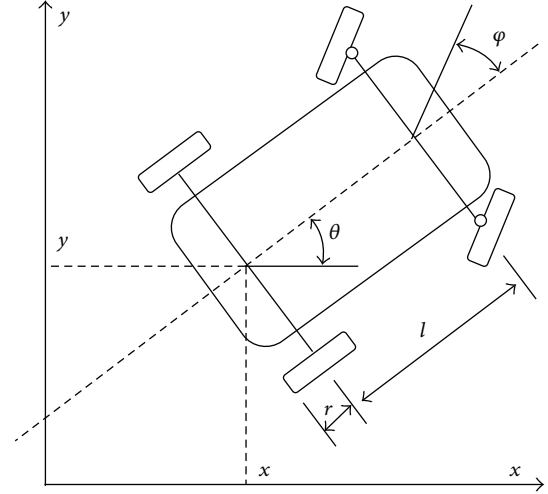


FIGURE 1: A simplified vehicle model.

$X(T) = [x_T, y_T, \theta_T, \varphi_T]'$ at time $t = T$, the paper generates a feasible trajectory for this vehicle.

Similarly, the model for a forward front-wheel driving vehicle is presented as

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \cos \theta \cos \varphi \\ \sin \theta \cos \varphi \\ \frac{\tan \varphi}{l} \\ 0 \end{bmatrix} r v_1 + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} v_2. \quad (2)$$

The model for a forward rear-wheel driving vehicle in (1) will be used for all calculations and simulations in this paper. From this kinematic model, flatness equations for the vehicle trajectory generation are presented in the next section.

3. Vehicle Flatness Trajectory Generation

The flatness system is the system of nonlinear differential equations in (1) whose movement curves are in smooth and flat space. From Figure 1, the vehicle angular velocity can be calculated as

$$v_1 = \frac{\sqrt{\dot{x}^2 + \dot{y}^2}}{r}. \quad (3)$$

Transforming from (1), the vehicle body angle is

$$\theta = \arctan\left(\frac{\dot{y}}{\dot{x}}\right). \quad (4)$$

From the derivative of the above trigonometric, θ , the body angular velocity, $\dot{\theta}$, is achieved as follows:

$$\dot{\theta} = \frac{\dot{y}\dot{x} - \dot{x}\dot{y}}{\dot{x}^2} \frac{1}{(\dot{y}/\dot{x})^2 + 1} = \frac{\dot{y}\ddot{x} - \dot{x}\ddot{y}}{\dot{x}^2 + \dot{y}^2} = \frac{\tan \varphi}{l} r v_1. \quad (5)$$

Therefore, θ and φ can be directly calculated from variables: \dot{x} , \ddot{x} , and \dot{y} , \ddot{y} . And it means that the above system is flat [1]. Thus, all state and input variables can be presented by the flat outputs x and y .

From (5), the boundary conditions for the outputs x and y are

$$\frac{\partial^2 y}{\partial x^2} = \frac{\tan \varphi}{l \cos^3 \theta}. \quad (6)$$

The initial state at time, $t = 0$, to the final state at time, $t = T$, for $x(t)$, is

$$x(0) = x_0, \quad x(T) = x_T. \quad (7)$$

The initial state for $y(t) = y(0)$ is

$$y(0) = y_0 = \tan \theta_0 \implies \left. \frac{\partial^2 y}{\partial x^2} \right|_{t=0} = \frac{\tan \varphi_0}{l \cos^3 \theta_0}. \quad (8)$$

The final state for $y(t) = y(T)$ is

$$y(T) = y_T = \tan \theta_T \implies \left. \frac{\partial^2 y}{\partial x^2} \right|_{t=T} = \frac{\tan \varphi_T}{l \cos^3 \theta_T}. \quad (9)$$

From the initial state $(x_0, y_0, \theta_0, \varphi_0)$ at the time $t = 0$ to the final state $(x_T, y_T, \theta_T, \varphi_T)$, under a real condition that $|\dot{x}(t)| \geq \varepsilon > 0$, if it is assumed that $|x_T - x_0|/2T > 0$, the trajectory of $x(t)$ can be written freely as

$$x(t) = \left(\frac{T-t}{T} \right) x_0 + \frac{t}{T} x_T + |x_T - x_0| \frac{t(t-T)}{2T^2}. \quad (10)$$

And the trajectory of $y(t)$ can be selected as

$$y(t) = y_0 + t \alpha_1 \tan \theta_0 + t^2 \frac{\alpha_2 \tan \varphi_0}{2l \cos^3 \theta_0} + t^3 b_1 + t^4 b_2 + t^5 b_3, \quad (11)$$

where $\alpha_1 = (2(x_T - x_0) - |x_T - x_0|)/2T$, $\alpha_2 = |x_T - x_0|/T^2$, and $\alpha_3 = (2(x_T - x_0) + |x_T - x_0|)/2T$; and $b = [b_1, b_2, b_3]^T = A^{-1}c$ with

$$A = \begin{bmatrix} T^3 & T^4 & T^5 \\ 3T^2 & 4T^3 & 5T^4 \\ 6T & 12T^2 & 20T^3 \end{bmatrix}, \quad (12)$$

$$c = \begin{bmatrix} y_T - y_0 - T \alpha_1 \tan \theta_0 - T^2 \frac{\alpha_2 \tan \varphi_0}{2l \cos^3 \theta_0} \\ \alpha_3 \tan \theta_T - \alpha_1 \tan \theta_0 - T \frac{\alpha_2 \tan \varphi_0}{l \cos^3 \theta_0} \\ \frac{\alpha_2 \tan \varphi_T}{l \cos^3 \theta_T} - \frac{\alpha_2 \tan \varphi_0}{l \cos^3 \theta_0} \end{bmatrix}.$$

From (10),

$$\theta = \arctan \left(2T^2 \left(\alpha_1 \tan \theta_0 + \frac{\alpha_2 \tan \varphi_0}{l \cos^3 \theta_0} t + 3b_1 t^2 + 4b_2 t^3 + 5b_3 t^4 \right) \times (2T(x_T - x_0) - T|x_T - x_0| + 2|x_T - x_0|t)^{-1} \right) \quad (13)$$

And, from (5),

$$\varphi = \arctan \left(\left(\frac{\alpha_2 \tan \varphi_0}{l \cos^3 \theta_0} + 6b_1 t + 12b_2 t^2 + 20b_3 t^3 \right) \times (2T^2)^2 l \cos^3 \theta \times (2T(x_T - x_0) - T|x_T - x_0| + 2|x_T - x_0|t)^{-1} \right). \quad (14)$$

The angular velocity of the vehicle in (1) can be calculated from (13) and (14).

For derivative of y ,

$$\dot{y}(t) = \left(\alpha_1 \tan \theta_0 + \frac{\alpha_2 \tan \varphi_0}{l \cos^3 \theta_0} t + 3b_1 t^2 + 4b_2 t^3 + 5b_3 t^4 \right) \quad (15)$$

and, then,

$$\ddot{y}(t) = \left(\frac{\alpha_2 \tan \varphi_0}{l \cos^3 \theta_0} + 6b_1 t + 12b_2 t^2 + 20b_3 t^3 \right) \quad (16)$$

and derivative of x is

$$\dot{x}(t) = \left(\frac{2T(x_T - x_0) - T|x_T - x_0| + 2|x_T - x_0|t}{2T^2} \right) \quad (17)$$

And, then,

$$\ddot{x}(t) = \left(\frac{|x_T - x_0|}{T^2} \right). \quad (18)$$

The absolute vehicle velocity can be calculated from (15) to (18) with

$$v_1(t) = \frac{\sqrt{\dot{x}^2 + \dot{y}^2}}{r} \quad (19)$$

or with another formula to calculate $v_1(t)$ is $v_1(t) = (\dot{x}(t) \cos \theta / r) + (\dot{y}(t) \sin \theta / r)$; then

$$v_1(t) = \left(\frac{(2T(x_T - x_0) - T|x_T - x_0| + 2|x_T - x_0|t)}{2rT^2} \times \cos \theta + \left(\left(\alpha_1 \tan \theta_0 + \frac{\alpha_2 \tan \varphi_0}{l \cos^3 \theta_0} + 3b_1 t^2 + 4b_2 t^3 + 5b_3 t^4 \right) \sin \theta \times (r)^{-1} \right) \right). \quad (20)$$

To calculate $\dot{\theta}$ from (5),

$$\dot{\theta} = \frac{\dot{y}\dot{x} - \ddot{y}\ddot{x}}{\dot{x}^2 + \dot{y}^2} = \frac{\tan \varphi}{l} r v_1. \quad (21)$$

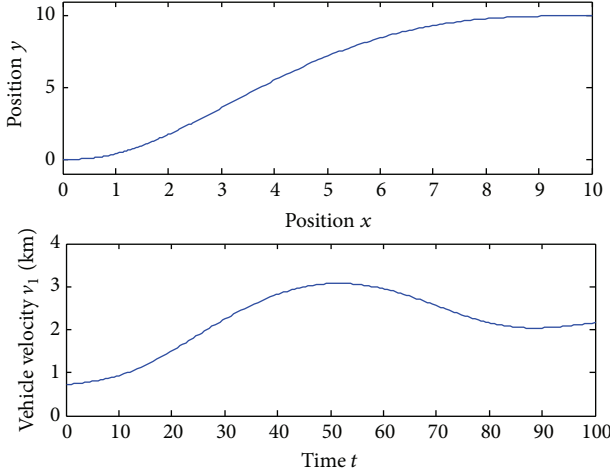


FIGURE 2: Trajectory and velocity.

To calculate $v_2(t) = \dot{\varphi}$, from (14), if $\varphi = \arctan(X)$, then,

$$\dot{\varphi} = \dot{X} \left(\frac{1}{1+X^2} \right) \quad \text{with}$$

$$X = \left(\left(\frac{\alpha_2 \tan \varphi_0}{l \cos^3 \theta_0} + 6b_1 t + 12b_2 t^2 + 20b_3 t^3 \right) (2T^2)^2 l \cos^3 \theta \right. \\ \left. \times \left((2T(x_T - x_0) - T|x_T - x_0| + 2|x_T - x_0|t)^2 \right)^{-1} \right). \quad (22)$$

Another formula to calculate $v_2(t) = \dot{\varphi}$ which is derived from (5) is

$$\dot{\varphi} = \partial \left(l \frac{\ddot{y}\dot{x} - \ddot{x}\dot{y}}{r(\dot{x}^2 + \dot{y}^2)^{3/2}} \right) \\ \times \left(\frac{1}{1 + \left(l(\ddot{y}\dot{x} - \ddot{x}\dot{y}/r(\dot{x}^2 + \dot{y}^2)^{3/2}) \right)^2} \right). \quad (23)$$

Simulation for this flatness technique and all other simulations for different path planning algorithms in this paper are conducted with the vehicle's parameters of $l = 2$ m and $r = 0.25$ m; the vehicle initial state at time $t = 0$ is $x(0) = [x_0, y_0, \theta_0, \varphi_0]' = [0, 0, 0, 0]'$; the vehicle final state at time $t = T = 100$ is $x(T) = [x_T, y_T, \theta_T, \varphi_T]' = [10, 10, 0, \pi/6]'$.

Figure 2 shows the trajectory (x, y) for the vehicle from the initial position to the final position and its velocity. Figure 3 shows the vehicle body angle, θ , and the steering angle, φ , corresponding to their angular velocities, $\dot{\theta}$ and $\dot{\varphi}$.

The maximum steering angle for this movement is $\varphi_{\max} = 64^\circ$; it exceeds the physical constraint of steering angle with $-45^\circ \leq \varphi_{\max} \leq 45^\circ$. Therefore, the above trajectory is not feasible. The problem is that the distance from the starting point to the destination point is too short subject to the real vehicle physical constraints. A solution for this problem is to lengthen the travelling distance until it meets the constraint on steering angle. This issue is discussed in the next section.

4. Trajectory Subject to Steering Angle Constraint

The maximum steering angle of a vehicle depends on its real structure space and design. In this paper, it is assumed that the real structure constraint for the vehicle is $(-\pi/4) \leq \varphi \leq \pi/4$ or the maximum steering angle for this vehicle is in a range of $-45^\circ \leq \varphi \leq 45^\circ$:

$$-\frac{\pi}{4} \leq \varphi \leq \frac{\pi}{4}. \quad (24)$$

Simulation in the previous part shows the maximum steering angle, $\varphi_{\max} = 64^\circ$, exceeding the limitation in (24). Therefore, a new technique for the vehicle trajectory generation subject to the constraint in (25) is proposed.

For time t from initial state, $t = 0$, to the final state, $t = T$, the feasible trajectories $x(t)$ and $y(t)$ subject to conditions in (10) and (11) can be generated. During the progress of timing, t , recalculate the steering angle, φ , in (14) and then check for the steering constraint in (24). If the constraint in (24) is violated, the distance from the initial position (x_0, y_0) to the final position (x_T, y_T) is too short for satisfying the steering angle limit.

In this case, it is proposed that to lengthen the distance $d_0 = \sqrt{(x_T - x_0)^2 + (y_T - y_0)^2}$ to a new distance,

$$d_{Ni} = \rho^i d_0 \quad \text{with } \rho > 1 \quad (25)$$

for $i = 1, 2, 3, \dots, n$ until the constraint in (24) is satisfied and with ρ being an amplification coefficient, $\rho > 1$.

Then the new coordinates are

$$x_{TN} = \rho^n (x_T - x_0), \quad y_{TN} = \rho^n (y_T - y_0). \quad (26)$$

The next simulation is done with the same parameters in the previous part and subject to the constraint in (24) with an amplification coefficient $\rho = 1.1$. The constraint in (24) will be satisfied after $n = 4$ iterations. The new coordinate is $x_{TN} = y_{TN} = 23.579$.

Figure 4 shows the new trajectory and new velocity of the vehicle with the new lengthened distance. The velocity of the vehicle is increased since the final time T is not changed.

Figure 5 shows the new vehicle body angle, θ , velocity, $\dot{\theta}$, steering angle, φ , and velocity, $\dot{\varphi}$, for the new trajectory. The new steering angle has now satisfied the constraint $\varphi = 40^\circ$.

Due to the size of this paper, the sideslip of the vehicle model is ignored. In reality, the sideslip of a vehicle depends on the tire stiffness and the cornering velocity. Then the trajectory generation in this study does not depend on the vehicle velocity. In the next part, a new vehicle trajectory generation based on polynomial equations is investigated.

5. Polynomial Trajectory Generation

For faster generation of a feasible vehicle tracking, a second-order polynomial for trajectory generation is presented. Equation (1) is separated into the following forms:

$$z_1 = x, \quad z_2 = \frac{\tan \varphi}{l \cos^3 \theta}, \quad z_3 = \tan \theta, \quad z_4 = y. \quad (27)$$

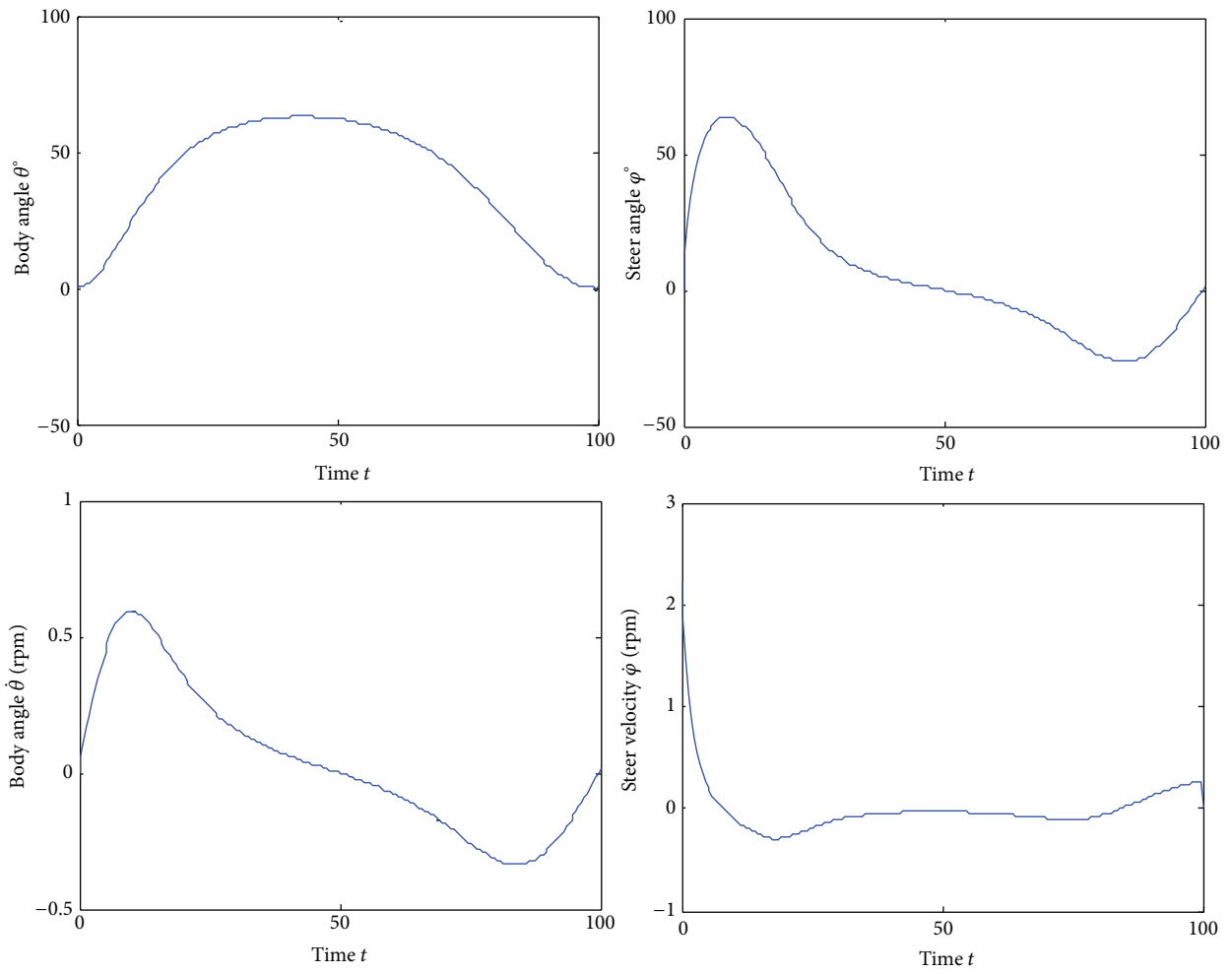


FIGURE 3: Body and steering angle.

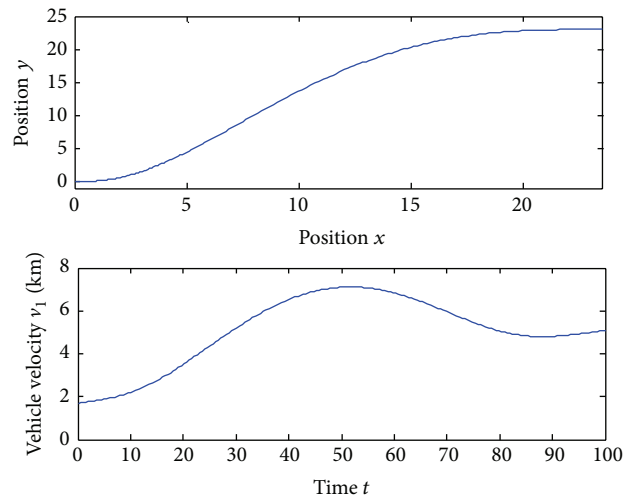


FIGURE 4: Trajectory and velocity.

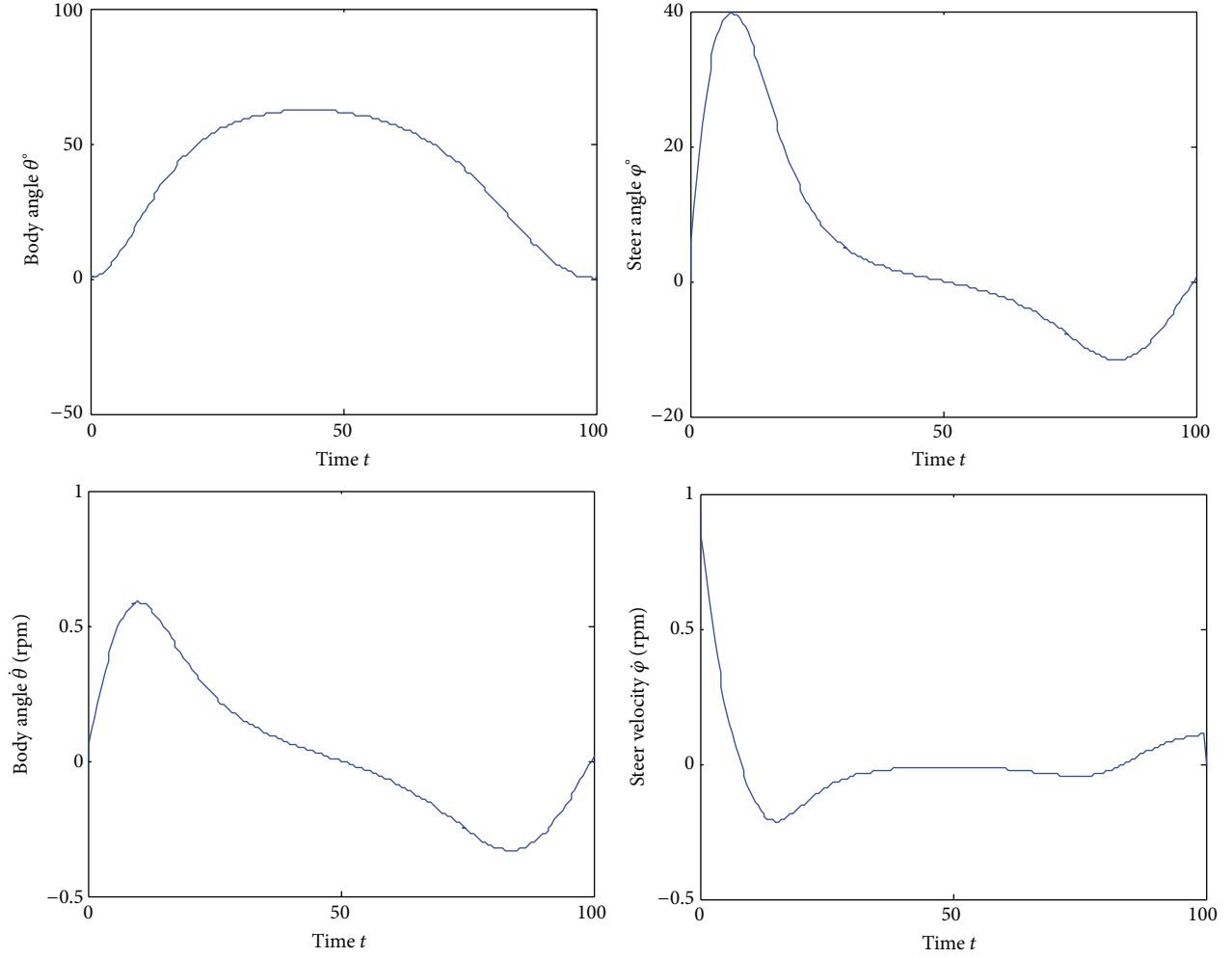


FIGURE 5: Body and steering angle.

Then,

$$\dot{z}_1 = \dot{x}, \quad (28)$$

$$\dot{z}_2 = \frac{v_2 l \cos^2 \theta + 3rv_1 \cos \theta \sin \theta \sin^2 \varphi}{l^2 \cos^5 \theta \cos^2 \varphi},$$

$$\dot{z}_3 = \frac{\tan \varphi}{l \cos^2 \theta} rv_1, \quad (29)$$

$$\dot{z}_4 = \dot{y} = \sin \theta rv_1.$$

The vehicle will move from the initial state $(x_0, y_0, \theta_0, \varphi_0)$ at time $t = 0$ to the final state $(x_T, y_T, \theta_T, \varphi_T)$ at time $t = T$ corresponding from the initial state at $(z_{1,0}, z_{2,0}, z_{3,0}, z_{4,0})$ to the final state at $(z_{1,T}, z_{2,T}, z_{3,T}, z_{4,T})$.

For $0 \leq t \leq T$, the calculation of $[z_1(t), z_2(t), z_3(t), z_4(t)]$ will be

$$z_1(t) = z_{1,0} + gt,$$

$$z_2(t) = z_{2,0} + h_1 t + \frac{1}{2} h_2 t^2 + \frac{1}{3} h_3 t^3,$$

$$z_3(t) = z_{3,0} + g z_{2,0} t + \frac{1}{2} g h_1 t^2 + \frac{1}{6} g h_2 t^3 + \frac{1}{12} g h_3 t^4,$$

$$z_4(t) = z_{4,0} + g z_{3,0} t + \frac{1}{2} g^2 z_{2,0} t^2 + \frac{1}{6} g^2 h_1 t^3 + \frac{1}{24} g^2 h_2 t^4 + \frac{1}{60} g^2 h_3 t^5 \quad (30)$$

with

$$g = \frac{z_{1,T} - z_{1,0}}{T},$$

$$[h_1, h_2, h_3]' = D^{-1} e,$$

$$D = \begin{bmatrix} T & \frac{1}{2} T^2 & \frac{1}{3} T^3 \\ \frac{1}{2} g T^2 & \frac{1}{6} g T^3 & \frac{1}{12} g T^4 \\ \frac{1}{6} g^2 T^3 & \frac{1}{24} g^2 T^4 & \frac{1}{60} g^2 T^5 \end{bmatrix}, \quad (31)$$

$$e = \begin{bmatrix} z_{2,T} - z_{2,0} \\ z_{3,T} - z_{3,0} - g z_{2,0} T \\ z_{4,T} - z_{4,0} - g z_{3,0} T - \frac{1}{2} g^2 z_{2,0} T^2 \end{bmatrix}.$$

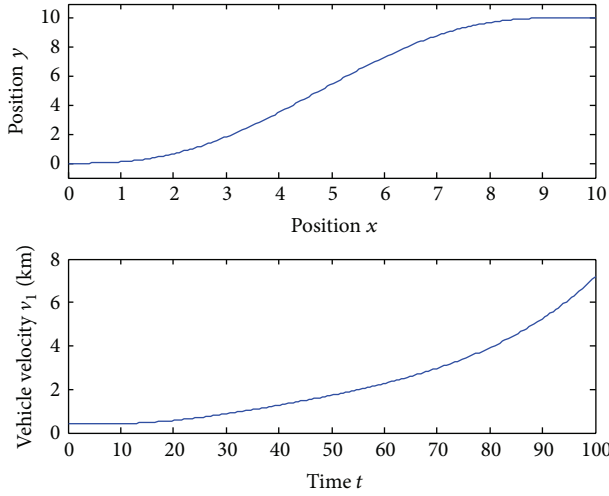


FIGURE 6: Trajectory and velocity.

Simulation of this method is conducted with the same parameters in Section 2 and is shown in Figures 6 and 7 using the following equations:

$$\begin{aligned}
 \theta &= \arctan(z_3), \\
 \varphi &= \arctan(z_2 \cos^3 \theta), \\
 \dot{x}(t) &= \dot{z}_1(t) = g, \\
 \dot{y}(t) &= \dot{z}_4(t) = gz_{3,0} + g^2 z_{2,0} t \\
 &\quad + \frac{1}{2} g^2 h_1 t^2 + \frac{1}{8} g^2 h_2 t^3 + \frac{1}{12} g^2 h_3 t^4, \\
 v_1(t) &= \frac{\sqrt{\dot{x}^2(t) + \dot{y}^2(t)}}{r}, \\
 \dot{\theta} &= \dot{z}_3 \cos^2 \theta = \left(gz_{2,0} + gh_1 t + \frac{1}{2} gh_2 t^2 + \frac{1}{3} gh_3 t^3 \right) \cos^2 \theta,
 \end{aligned} \tag{32}$$

And, from (28),

$$\dot{\varphi} = \dot{z}_2(t) \cos^3 \theta \cos^2 \varphi - 3\dot{\theta} \tan \theta \sin \varphi \cos \varphi. \tag{33}$$

Figure 6 shows the trajectory (x, y) for the vehicle from initial point to the final point and the velocity of the vehicle.

Figure 7 shows the vehicle body angle, θ , and the steering angle, φ , corresponding to the angular velocity, $\dot{\theta}$ and $\dot{\varphi}$, of the vehicle.

As shown in Figure 7, the maximum steering angle for this trajectory generation is $\varphi = 41.5736^\circ$ and satisfied the constraint $-45^\circ \leq \varphi \leq 45^\circ$. Therefore this trajectory generation is better than the flatness method in the previous part.

From Figure 6, it is really not realistic when the vehicle speed is increasing exponentially. It is expected that, when a vehicle is moving from one point to another point, the speed will increase gradually at the starting point and decrease gradually to the destination point. Therefore, in the next part, new symmetric polynomial equations in third order are investigated.

6. Symmetric Polynomial Trajectory Generation

Since the system is flat and each flat output can be parameterized by a sufficiently smooth polynomial, symmetric third-order polynomial equations are tried for this trajectory generation. Because the sideslip is ignored, the vehicle trajectory does not depend on its speed, v_1 , and on the travelling time, T . A new time variable, therefore, for this system is applied with a ratio of t/T for $t = 0 \div T$:

$$\begin{aligned}
 x(t) &= -\left(\frac{t}{T} - 1\right)^3 x_0 + \left(\frac{t}{T}\right)^3 x_T \\
 &\quad + a_x \left(\frac{t}{T}\right)^2 \left(\frac{t}{T} - 1\right) + b_x \frac{t}{T} \left(\frac{t}{T} - 1\right)^2
 \end{aligned} \tag{34}$$

and the trajectory of y is

$$\begin{aligned}
 y(t) &= -\left(\frac{t}{T} - 1\right)^3 y_0 + \left(\frac{t}{T}\right)^3 y_T \\
 &\quad + a_y \left(\frac{t}{T}\right)^2 \left(\frac{t}{T} - 1\right) + b_y \frac{t}{T} \left(\frac{t}{T} - 1\right)^2.
 \end{aligned} \tag{35}$$

Derivation of $x(t)$ is

$$\begin{aligned}
 \dot{x}(t) &= -3\left(\frac{t}{T} - 1\right)^2 x_0 + 3\left(\frac{t}{T}\right)^2 x_T + a_x 2 \frac{t}{T} \left(\frac{t}{T} - 1\right) \\
 &\quad + a_x \left(\frac{t}{T}\right)^2 + b_x \left(\frac{t}{T} - 1\right)^2 + b_x 2 \frac{t}{T} \left(\frac{t}{T} - 1\right)
 \end{aligned} \tag{36}$$

and derivation of $y(t)$ is

$$\begin{aligned}
 \dot{y}(t) &= -3\left(\frac{t}{T} - 1\right)^2 y_0 + 3\left(\frac{t}{T}\right)^2 y_T + a_y 2 \frac{t}{T} \left(\frac{t}{T} - 1\right) \\
 &\quad + a_y \left(\frac{t}{T}\right)^2 + b_y \left(\frac{t}{T} - 1\right)^2 + b_y 2 \frac{t}{T} \left(\frac{t}{T} - 1\right).
 \end{aligned} \tag{37}$$

Then,

$$\begin{aligned}
 \ddot{x}(t) &= -6\left(\frac{t}{T} - 1\right) x_0 + 6\frac{t}{T} x_T + a_x 2 \left(2\frac{t}{T} - 1\right) \\
 &\quad + a_x 2 \frac{t}{T} + b_x 2 \left(\frac{t}{T} - 1\right) + b_x 2 \left(2\frac{t}{T} - 1\right),
 \end{aligned} \tag{38}$$

$$\begin{aligned}
 \ddot{y}(t) &= -6\left(\frac{t}{T} - 1\right) y_0 + 6\frac{t}{T} y_T + a_y 2 \left(2\frac{t}{T} - 1\right) \\
 &\quad + a_y 2 \frac{t}{T} + b_y 2 \left(\frac{t}{T} - 1\right) + b_y 2 \left(2\frac{t}{T} - 1\right).
 \end{aligned} \tag{39}$$

The constraint on speed is

$$r v_1 = \frac{\dot{x}}{\cos \theta} = \frac{\dot{y}}{\sin \theta}. \tag{40}$$

The constraint at starting point $t = 0$ is

$$\dot{x}(0) = k_0 \cos \theta_0, \quad \dot{y}(0) = k_0 \sin \theta_0. \tag{41}$$

The constraint at destination point $t = T$ is

$$\dot{x}(T) = k_T \cos \theta_T, \quad \dot{y}(T) = k_T \sin \theta_T. \tag{42}$$

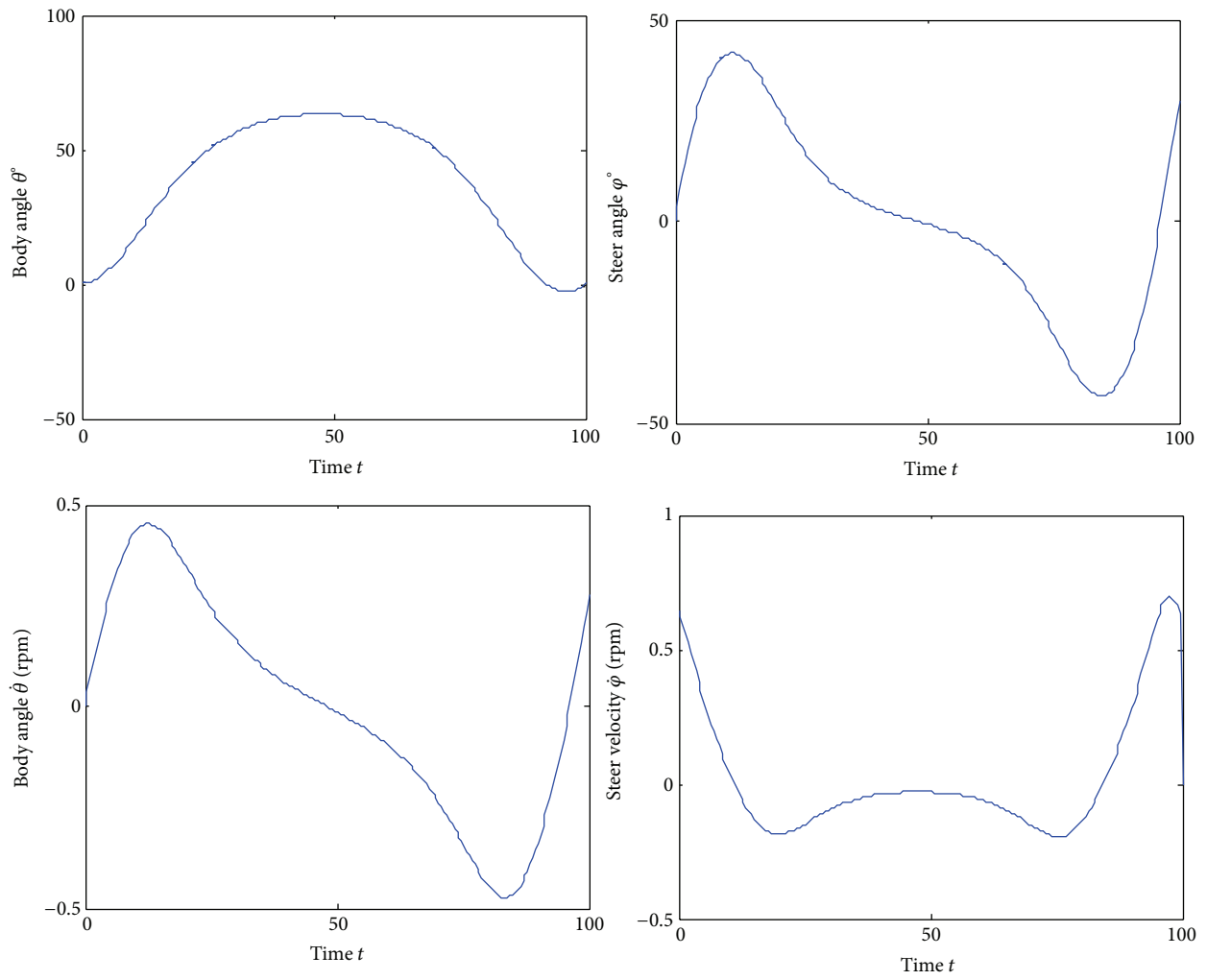


FIGURE 7: Body and steering angle.

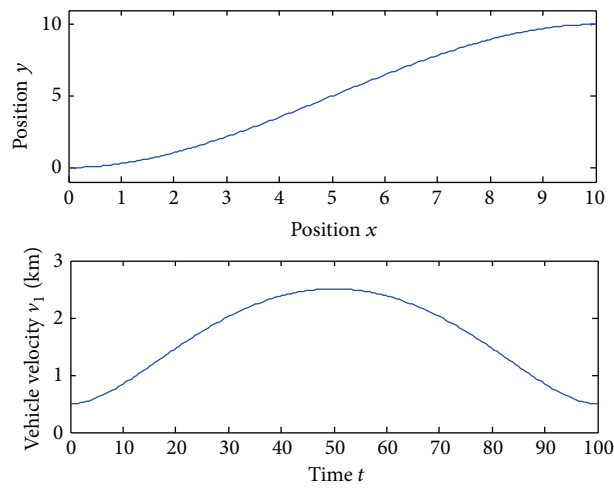


FIGURE 8: Trajectory and velocity.

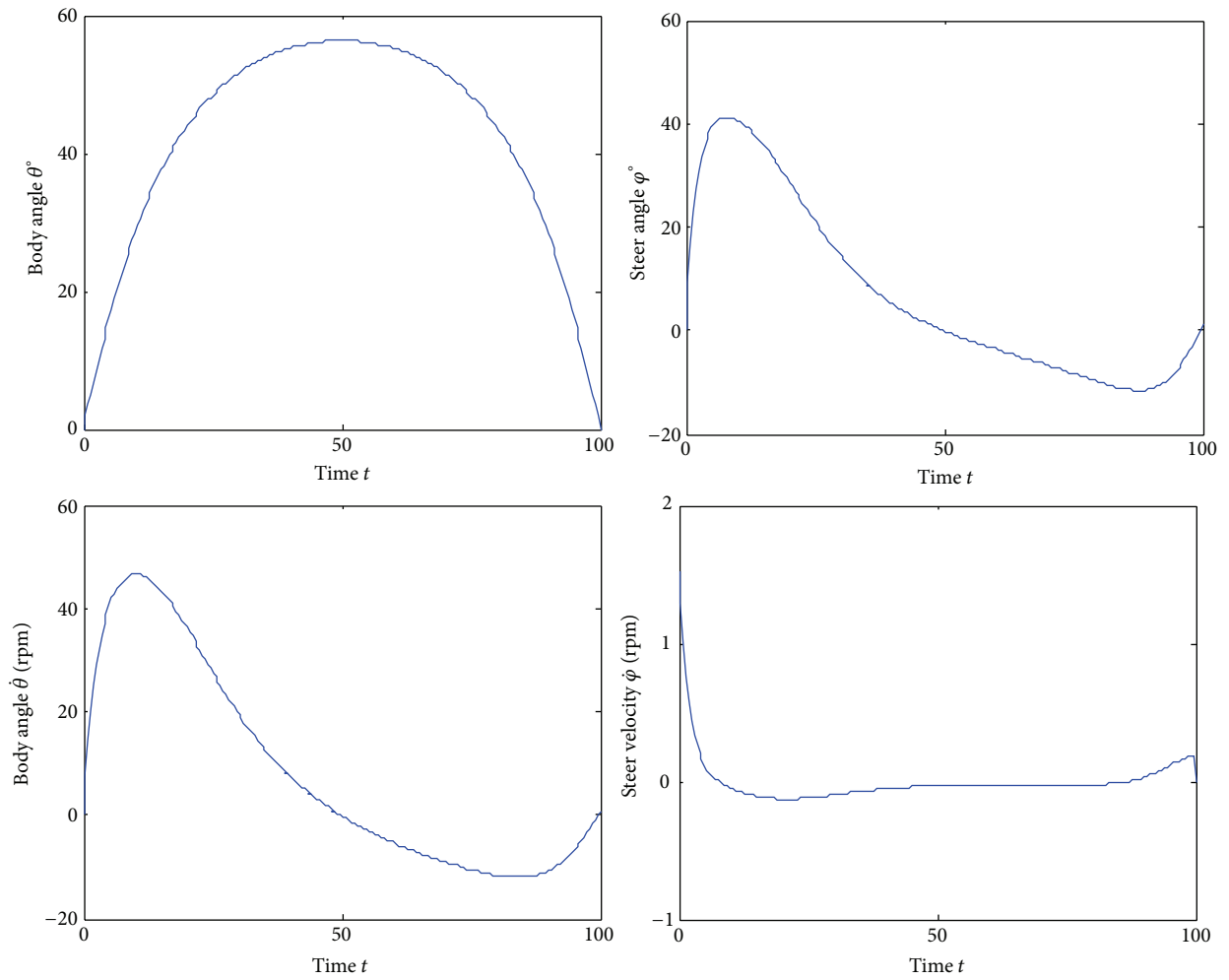


FIGURE 9: Body and steering angle.

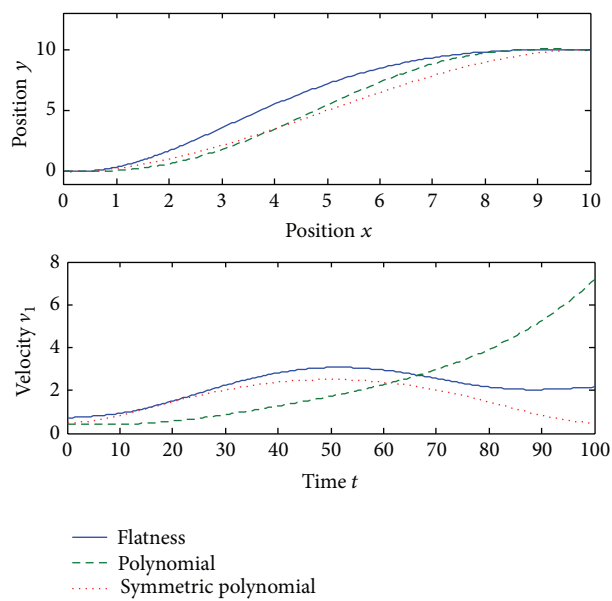


FIGURE 10: Comparison of trajectory and velocity.

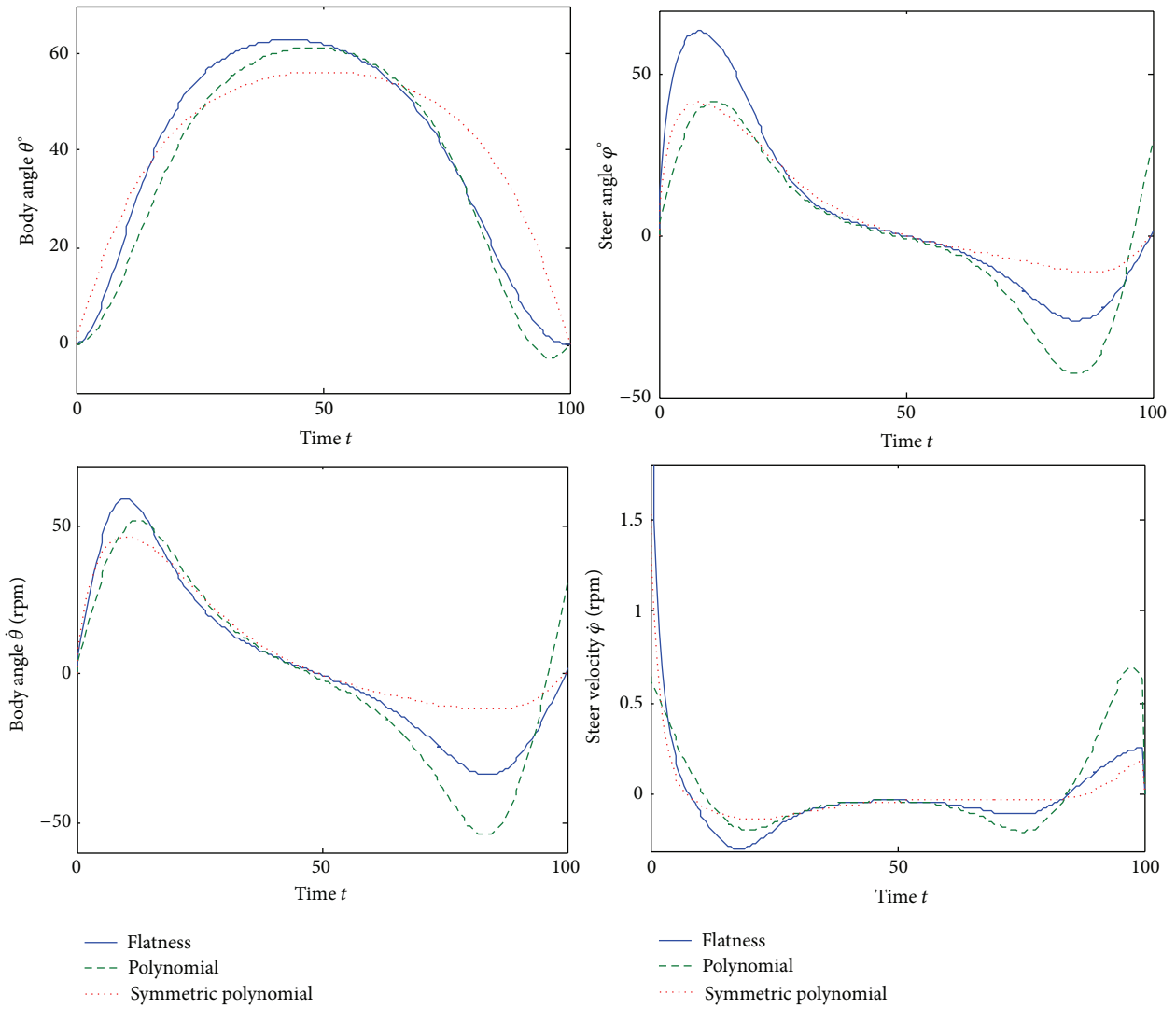


FIGURE 11: Comparison of body and steering angle.

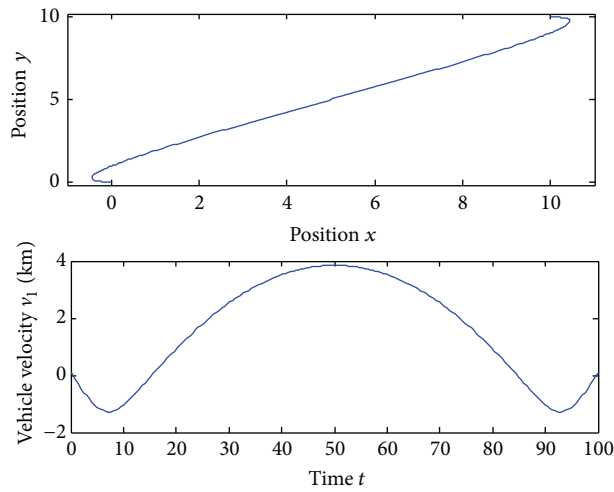


FIGURE 12: Trajectory and velocity in reverse speed.

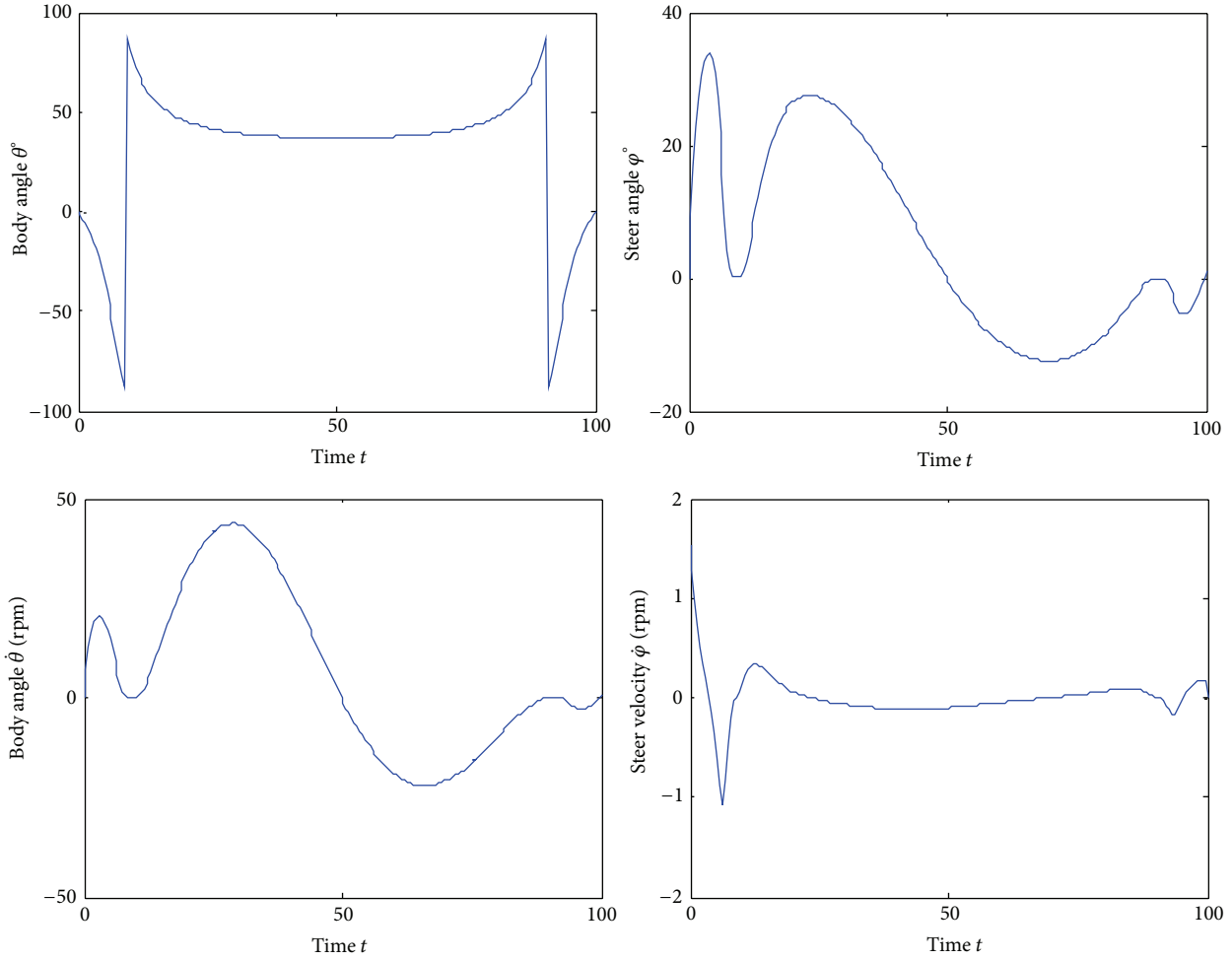


FIGURE 13: Body and steering angle in reverse speed.

From (38), (39), (41), and (42), for the calculation simplicity, it is assumed that the speed coefficients at the starting and destination points, $k_0 = k_T = k$; then

$$a_x = k \cos \theta_T - 3x_T, \quad b_x = k \cos \theta_0 - 3x_0. \quad (43)$$

Similarly,

$$a_y = k \sin \theta_T - 3y_T, \quad b_y = k \sin \theta_0 - 3y_0. \quad (44)$$

Simulation is conducted with the same parameters in the previous parts and is shown in Figures 8 and 9. The speed coefficients are $k_0 = k_T = k = 1$. Other parameters are calculated in the following equations:

$$v_1 = \frac{\sqrt{\dot{x}^2 + \dot{y}^2}}{r},$$

$$\theta = \arctan\left(\frac{\dot{y}}{\dot{x}}\right),$$

$$\varphi = \arctan\left(l \frac{\cos^3 \theta \ddot{y}}{(\dot{x})^2}\right)$$

$$\text{or } \varphi = \arctan\left(l \frac{\ddot{y}\dot{x} - \ddot{x}\dot{y}}{r(\dot{x}^2 + \dot{y}^2)^{3/2}}\right),$$

$$\dot{\theta} = \frac{\ddot{y}\dot{x} - \ddot{x}\dot{y}}{\dot{x}^2} \frac{1}{(\dot{y}/\dot{x})^2 + 1} = \frac{\ddot{y}\dot{x} - \ddot{x}\dot{y}}{\dot{x}^2 + \dot{y}^2} = \frac{\tan \varphi}{l} r v_1,$$

$$\dot{\varphi} = \frac{\partial \left(\arctan \left(l \left((\ddot{y}\dot{x} - \ddot{x}\dot{y}) / r(\dot{x}^2 + \dot{y}^2)^{3/2} \right) \right) \right)}{\partial t}. \quad (45)$$

It can be seen from Figure 8 that the trajectory of this symmetric polynomial is more realistic because the velocity increases from the starting point and decreases in the destination point as per the expectation.

The maximum steering angle for the symmetric polynomial method is $\varphi = 41.1622^\circ$ and satisfied the constraint $-45^\circ \leq \varphi \leq 45^\circ$. This steering angle is smaller than the steering angle in the second-order polynomial method. In the next part, a comparison of the previous three methods is presented.

7. Performance Analysis

Performances of the three methods on the trajectory generation and the vehicle velocity are shown in Figures 10 and 11.

It can be seen that the symmetric polynomial generation can produce a more realistic speed and a smoother trajectory since it allows the vehicle gradually to increase the speed at the starting point and reduce the speed at the destination point.

Figure 11 shows the symmetric polynomial method providing the lowest body angle, $\theta(t)$, the body angle velocity, $\dot{\theta}(t)$, the steering angle, $\varphi(t)$, and the steering angle velocity, $\dot{\varphi}(t)$. Therefore, this method is recommended for the development of an automatic control of tracking vehicles.

For the vehicle moving in reverse speeds, as mentioned in (1), the vehicle velocity, v_1 , will change the sign. The speed coefficients in (43) and (44) will be minus values. Simulations for the reverse speeds are conducted with the speed coefficients $k_0 = k_T = k = -1$. Results of the simulation are shown in Figures 12 and 13.

Since the vehicle is forced to reverse at the starting point with $k_0 = -1$ and at the destination point with $k_T = -1$, Figure 12 shows the vehicle reversing out at the starting point, going forward to the destination, and then again reversing to the parking space. The vehicle velocity is changing the direction in three times.

Figure 13 shows the body angle, θ , switching 180° in two times corresponding to each reverse speed. The maximum steering angle in this reverse speed is $\varphi_{\max} = 33.7097^\circ$. This number is much lower than the numbers of the forward speed movements. However the travelling distance is also longer in the reverse speed trajectory.

8. Conclusion

The paper has presented three methods of trajectory generation for autonomous vehicles subject to constraints. Regarding the real vehicle speed development, the third-order symmetric polynomial trajectory is recommended. Simulations and analyses are also conducted for vehicle moving in forward and in reverse speeds. Results from this study can be used to develop a real-time control system for autodiving and autoparking vehicles. The limitation of this study is the ignorance of the vehicle sideslip due to the cornering velocity. However this error can be eliminated with the feedback control loop and some offset margins of the steering angle constraint.

Conflict of Interests

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

References

- [1] J. Lévine, *Analysis and Control of Nonlinear Systems: A Flatness-Based Approach*, Springer, 1st edition, 2009.

- [2] W. Dong and Y. Guo, "New trajectory generation methods for nonholonomic mobile robots," in *Proceedings of the International Symposium on Collaborative Technologies and Systems*, pp. 353–358, Saint Louis, Mo, USA, 2005.
- [3] M. Gomez, V. González, T. Martínez-Marín, D. Meziat, and S. Sánchez, "Optimal motion planning by reinforcement learning in autonomous mobile vehicles," *Robotica*, vol. 30, no. 2, pp. 159–170, 2012.
- [4] J. Wang, Z. Lu, W. Chen, and X. Wu, "An adaptive trajectory tracking control of wheeled mobile robots," in *Proceedings of the 6th IEEE Conference on Industrial Electronics and Applications (ICIEA '11)*, pp. 1156–1160, Beijing, China, 2011.
- [5] M. Werling, S. Kammel, J. Ziegler, and L. Gröll, "Optimal trajectories for time-critical street scenarios using discretized terminal manifolds," *International Journal of Robotics Research*, vol. 31, no. 3, pp. 346–359, 2012.
- [6] K. Kanjanawanishkul, M. Hofmeister, and A. Zell, "Smooth reference tracking of a mobile robot using nonlinear model predictive control," in *Proceedings of the 4th European Conference on Mobile Robots (ECMR '09)*, pp. 161–166, Dubrovnik, Croatia, 2009.
- [7] G. Klančar and I. Škrjanc, "Tracking-error model-based predictive control for mobile robots in real time," *Robotics and Autonomous Systems*, vol. 55, no. 6, pp. 460–469, 2007.
- [8] V. T. Minh, *Advanced Vehicle Dynamics*, Malaya Press, Kuala Lumpur, Malaysia, 1st edition, 2012.
- [9] V. T. Minh and N. Afzulpurkar, "A comparative study on computational schemes for nonlinear model predictive control," *Asian Journal of Control*, vol. 8, no. 4, pp. 324–331, 2006.
- [10] J. Choi, R. Curry, and G. Elkaim, "Curvature-continuous trajectory generation with corridor constraint for autonomous ground vehicles," in *Proceedings of the 49th IEEE Conference on Decision and Control*, pp. 7166–7171, Atlanta, Ga, USA, 2010.
- [11] V. Delsart, T. Fraichard, and L. Martinez, "Real-time trajectory generation for car-like vehicles navigating dynamic environments," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '09)*, pp. 3401–3406, Kobe, Japan, 2009.
- [12] T. Zhou, M. Li, X. Mai, Q. Wang, F. Liu, and Q. Li, "Trajectory generation model-based IMM tracking for safe driving in intersection scenario," *International Journal of Vehicular Technology*, vol. 2011, Article ID 103696, 8 pages, 2011.
- [13] M. Missura and S. Behnke, "Efficient kinodynamic trajectory generation for wheeled robots," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 4883–4888, Shanghai, China, 2011.
- [14] V. T. Minh and A. A. Rashid, "Automatic control of clutches and simulations for parallel hybrid vehicles," *International Journal of Automotive Technology*, vol. 13, no. 4, pp. 645–651, 2012.
- [15] V. T. Minh, M. Awang, and S. Parman, "Conditions for stabilizability of linear switched systems," *International Journal of Control, Automation and Systems*, vol. 9, no. 1, pp. 139–144, 2011.



Hindawi

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

