TRANSPORT
ISSN 1648-4142 / eISSN 1648-3480
Special Issue on Smart and Sustainable Transport / 2015 Volume 30(3): 353–360
doi:10.3846/16484142.2015.1089322

INTEGRATION OF MOTION PLANNING AND MODEL-PREDICTIVE-CONTROL-BASED CONTROL SYSTEM FOR AUTONOMOUS ELECTRIC VEHICLES

Guodong Yin[1, 2], Jianghu Li[3], Xianjian Jin[4], Chentong Bian[5], Nan Chen[6]

[1, 3, 4, 5, 6] *School of Mechanical Engineering, Southeast University, Nanjing, China*
[2] *State Key Laboratory of Automotive Safety and Energy, Tsinghua University, Beijing, China*

Submitted 29 July 2014; resubmitted 9 October 2014, 18 January 2015; accepted 17 February 2015

Abstract. This paper introduces the development of an autonomous driving system in autonomous electric vehicles, which consists of a simplified motion-planning program and a Model-Predictive-Control-Based (MPC-based) control system. The motion-planning system is based on polynomial parameterization, which computes a path toward the expected longitudinal and lateral positions within required time interval in real scenarios. Then the MPC-based control system cooperates the front steering and individual wheel torques to track the planned trajectories, while fulfilling the physical constraints of actuators. The proposed system is evaluated through simulation, using a seven-degrees-of-freedom vehicle model with a 'magic formula' tire model. The simulations and validation through *CarSim* show that the proposed planner algorithm and controller are feasible and can achieve requirements of autonomous driving in normal scenarios.

Keywords: autonomous electric vehicles; motion planning; model predictive control.

Introduction

Autonomous driving has become a fast-developing and promising area in recent years (Urmson *et al.* 2008; Levinson *et al.* 2011). An autonomous vehicle is required to perform two important tasks coherently: a motion-planning program computes the desired trajectories in various scenarios and a control system manipulates actuators to track the planned path (Park *et al.* 2009).

A wide range of motion-planning approaches have been reported, including genetic algorithms, randomized path planners (randomized potential field, rapidly exploring random tree, etc.), heuristic searching algorithms (Dijkstra algorithm and D* algorithm) and the polynomial parameterization method. The genetic and neural network algorithms can solve complex actual problems within a reasonable accuracy, but the convergence time is problematic and it is difficult to determine a set of conditions systematically to ensure convergence (Phinni *et al.* 2008; Lafta, Hassan 2013). For the potential field approach, adjustments are made directly in the vehicle's operational space instead of preprocessing before the start of vehicle motion. Nevertheless, this method can easily be trapped into local minima especially in a complex environment with multiple obstacles (Lin, Chuang 2010). The sampling-based Rapidly-exploring Random Tree (RRT) enables the online use on robotic vehicles with complex dynamics and significant drift, but the difference between the original prediction and the re-propagation is difficult to converge in a dynamic and uncertain environment (Kuwata *et al.* 2009). The Dijkstra's algorithm finds the path with lowest cost between vertexes, but the planning problem for autonomous driving among dynamic obstacles is complicated (Yershov, LaValle 2011). Although the improved D* algorithm can update the path based on moving obstacles, the algorithm is difficult to deal with unfixed goals in on-road scenarios (Koenig, Likhachev 2002). Shim *et al.* (2012) developed a new polynomial parameterization method to obtain a collision-free path, which is optimized by a minimum detour, but the boundary conditions are complex and the utilization of parameter estimation is unaware. This paper will present a simplified polynomial parameterization method for motion planning in real scenarios and make the best use of initial and terminal motion states of the autonomous vehicle.

Once the motion-planning program determines a reference trajectory, proper control methodologies are chosen to control the steering and speed actuators to

Corresponding author: Guodong Yin
E-mail: *ygd@seu.edu.cn*

make the vehicle follow the generated trajectory. These control logics mainly include sliding-mode control, fuzzy logic and Model Predictive Control (MPC). Durali *et al.* (2006) designed a sliding mode controller to guarantee that the vehicle tracks that desired trajectory, which has few computational difficulties and is appropriate for real time implementations. Fuzzy logic was utilized to incorporate human procedural knowledge with control algorithms and enhance the performance of tracking trajectories and avoiding obstacles, but vast work needed to be done in order to ensure the precision of control and dynamic quality (Naranjo *et al.* 2007). Falcone *et al.* (2007) proposed a MPC approach to best follow a given path by controlling the front steering angle and the brakes at the four wheels independently. As many of the control logics assume that the computed inputs of control system never reach the physical saturation limit of the actuator, MPC has the property of considering constraints in computation of the optimal solution and makes itself preferred over other methods (Kim *et al.* 2004). In this paper, an MPC-based controller is designed for four-wheel-drive electric vehicles to follow the planned trajectory by actively cooperating the front steering and wheel torques at each wheel.

The main contribution of this paper lies in two aspects. First, this is among the first attempts to incorporate together with motion-planning and control system for four-wheel-drive autonomous electric vehicles. The motion-planner delivers real-time information (position, velocity and acceleration of vehicles) to control system and the controller calculates input of actuators to track the planned trajectory. The cooperation between motion-planning system and control system can enhance the ability to deal with complex driving requirements quickly and effectively in autonomous electric vehicles. Second, a new method is adopted to calculate the wheel torques indirectly in MPC-based control system, which chooses the wheel slip ratios as the control inputs and utilizes their closer relation with control outputs in order to reduce overall computational complexity of the model predictive control problem.

The rest of this paper is structured as follows. In Section 1 we introduce a simplified polynomial parameterization method for motion planning in common driving scenarios. The vehicle model and the MPC approach are formulated in Section 2. Section 3 presents simulative results and validation of two normal driving scenarios to evaluate the feasibility of the motion-planning program and MPC-based controller. Finally, the paper is closed with concluding remarks and ideas for future work.

1. Motion Planning

This section describes the motion-planning method with a simplified kinematic vehicle model, as shown in Fig. 1. δ is the front steering-wheel angle, ($X(t)$, $Y(t)$) are the longitudinal and lateral positions of the midpoint at the rear axle in a inertial frame, ψ is the heading angle, v is the longitudinal vehicle speed, and l is the wheel base.

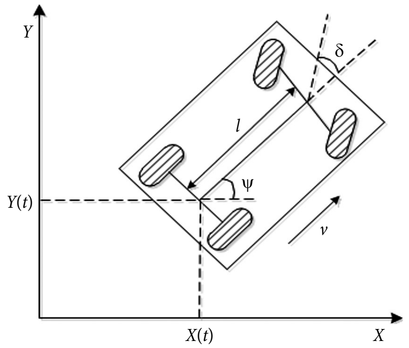

Fig. 1. The simplified vehicle model

The kinematic equations are given by:

$$\dot{X}(t) = v \cdot \cos\psi(t);$$
$$\dot{Y}(t) = v \cdot \sin\psi(t);$$
$$\dot{\psi}(t) = \frac{v \cdot \tan\delta(t)}{l}. \quad (1)$$

According to the polynomial parameterization method, let the initial state be $q_0 = \left(X_0, \dot{X}_0, \ddot{X}_0, Y_0, \dot{Y}_0, \ddot{Y}_0\right)$ at the initial time t_0 and the final state be $q_f = \left(X_f, \dot{X}_f, \ddot{X}_f, Y_f, \dot{Y}_f, \ddot{Y}_f\right)$ at the final time t_f. The initial state can be detected via various sensors and the terminal state can be estimated based on driving requirements. It shows that the initial and final configuration have six constraints in lateral and longitudinal directions respectively. Therefore, parameter expressions need at least six coefficients to accommodate these constraints. With above considerations, the planned path is defined by fifth-order polynomials that have six parameters given as:

$$X_d(t) = \sum_{i=0}^{5} a_i \cdot t^i;$$
$$Y_d(t) = \sum_{i=0}^{5} b_i \cdot t^i. \quad (2)$$

The coefficients $\vec{a}$ and $\vec{b}$ can be computed as:

$$\vec{a} = T^{-1} \cdot q_x;$$
$$\vec{b} = T^{-1} \cdot q_y, \quad (3)$$

where:

$$\vec{a} = \left[a_0, a_1, a_2, a_3, a_4, a_5\right]^T;$$
$$\vec{b} = \left[b_0, b_1, b_2, b_3, b_4, b_5\right]^T;$$
$$q_x = \left[X_d(t_0), \dot{X}_d(t_0), \ddot{X}_d(t_0), X_d(t_f), \dot{X}_d(t_f), \ddot{X}_d(t_f)\right]^T;$$
$$q_y = \left[Y_d(t_0), \dot{Y}_d(t_0), \ddot{Y}_d(t_0), Y_d(t_f), \dot{Y}_d(t_f), \ddot{Y}_d(t_f)\right]^T;$$
$$T = \begin{bmatrix} 1 & t_0 & t_0^2 & t_0^3 & t_0^4 & t_0^5 \\ 0 & 1 & 2 \cdot t_0 & 3 \cdot t_0^2 & 4 \cdot t_0^3 & 5 \cdot t_0^4 \\ 0 & 0 & 2 & 6 \cdot t_0 & 12 \cdot t_0^2 & 20 \cdot t_0^3 \\ 1 & t_f & t_f^2 & t_f^3 & t_f^4 & t_f^5 \\ 0 & 1 & 2 \cdot t_f & 3 \cdot t_f^2 & 4 \cdot t_f^3 & 5 \cdot t_f^4 \\ 0 & 0 & 2 & 6 \cdot t_f & 12 \cdot t_f^2 & 20 \cdot t_f^3 \end{bmatrix}.$$

Once polynomial parameters $\vec{a}$ and $\vec{b}$ are determined, the desired trajectory (X_d, Y_d) is generated and the desired heading angle can be calculated by the Eq. (1) simultaneously. In order to evaluate the effectiveness of the developed motion-planning approach, two normal driving scenarios will be discussed below.

The first scenario is a lane-change maneuver, as shown in Fig. 2. The vehicle needs to perform a lateral maneuver when changing lanes, but ends with a pure longitudinal motion. Suppose the initial longitudinal and lateral velocities of the vehicle are taken as 10 m/s and 0 m/s respectively, the time interval is 5 seconds, the initial way point is at (0, 0) m position and the final way point is at (50, 3) m. Since the longitudinal and lateral velocities need not change sharply in this scenario, it is assumed that the final velocities are the same as the initial state and both the initial and final accelerations are zero. In this way, the longitudinal state $q_x = \left(0,\ 10,\ 0,\ 50,\ 10,\ 0\right)$ and the lateral state $q_y = \left(0,\ 0,\ 0,\ 3,\ 0,\ 0\right)$ are obtained. The planned trajectory is shown in Fig. 3. Fig. 3a and 3b show the longitudinal and lateral positions respectively. The heading angle is shown in Fig. 3c and 3d depicts the relation between longitudinal and lateral positions.

The second scenario is a right-angle turn maneuver, as shown in Fig. 4. During a right-angle turn, the motion of vehicle switches from a longitudinal motion to a lateral motion in the inertial frame, along with significant changes of longitudinal and lateral velocities. Therefore, it is necessary to select proper accelerations for the initial and final states. Similar to the planning process of double-lane change case, suppose that the longitudinal state is $q_x = \left(0,\ 5,\ -2,\ 10,\ 0,\ 0\right)$, the lateral state is $q_y = \left(0,\ 0,\ -1,\ -10,\ -2,\ -1\right)$ and the time interval is 10 seconds. The planned trajectory is shown in Fig. 5.

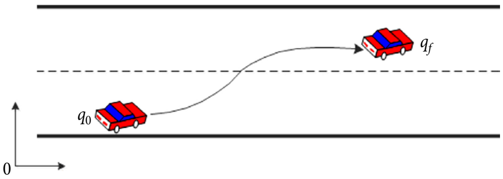

Fig. 2. The lane change maneuver

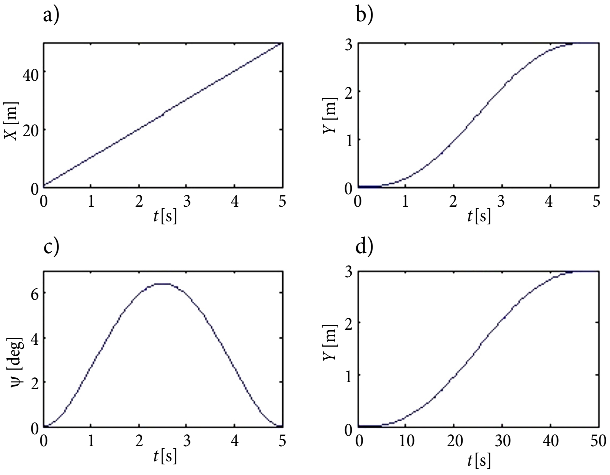

Fig. 3. The planned path for lane change

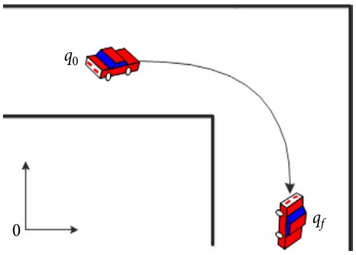

Fig. 4. The right-angle turn maneuver

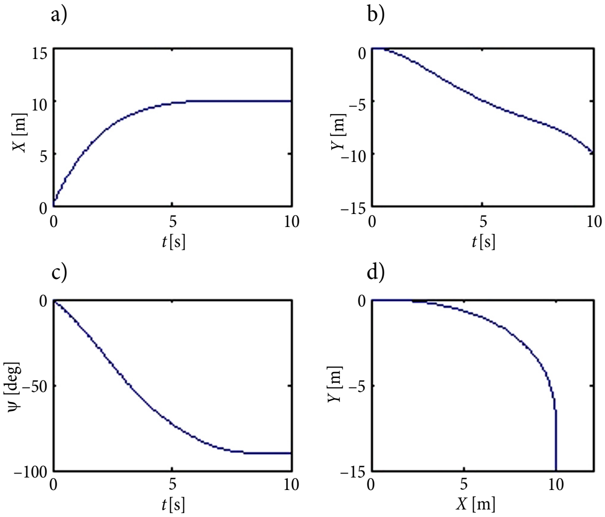

Fig. 5. The planned path for right-angle turn

From the generated trajectories of two specified scenarios, it is shown that the autonomous vehicle correctly reach the expected longitudinal and lateral positions within required time interval. In addition, these trajectories and heading angles transmit smoothly and steadily, without break points. Therefore, the motion-planning algorithm can effectively produce desired trajectories in normal scenarios.

2. Vehicle Control System

This section introduces the development of a vehicle control system to track the desired trajectories in different scenarios. The controller mainly includes two parts, vehicle modeling and control methodologies. It is preferred that the selected vehicle model is dynamically close to those of an actual vehicle. The control methodology should be well suited to the system complexity and computational burden. The vehicle model and control logic are presented in following sections.

2.1. Vehicle Model

It is widely recognized that an ideal vehicle model, producing reasonable computation burden, has dynamic characteristics close to those of an actual vehicle. The choice of a seven-degrees-of-freedom vehicle model meets those requirements well (Falcone *et al.* 2008), and the schematic diagram of vehicle model is depicted in Fig. 6. We denote by F_l, F_c the longitudinal and lateral tire forces, respectively, *a*, *b*, *c* the vehicle geometry parameters, $\dot{\psi}$ the yaw angular velocity, v_l, v_c the longitudinal and lateral wheel velocities respectively, and δ the wheel steering angle. The subscript symbols denote variables related to the four wheels: the first subscript denotes the front and rear axles, and the second denotes the left and right sides of the vehicle. As example, F_{fl} is referred to the front left wheel force.

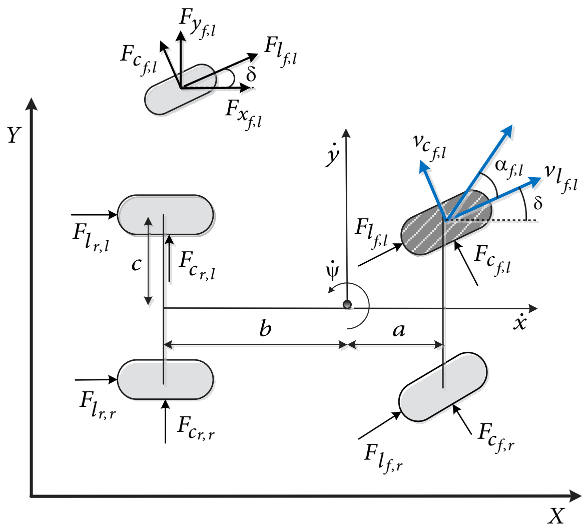

Fig. 6. The schematic diagram of vehicle model

This vehicle model has three DOFs for the chassis velocities and one DOF at each of the four wheels, representing the wheel spin dynamics. The chassis velocities include the longitudinal velocity $\dot{x}$, the lateral velocity $\dot{y}$ and the yaw angular velocity $\dot{\psi}$. The vehicle dynamic model is built as follows:

$$\begin{gathered} m\cdot\ddot{x}=m\cdot\dot{y}\cdot\dot{\psi}+F_{x_{f,l}}+F_{x_{f,r}}+F_{x_{r,l}}+F_{x_{r,r}}; \\ m\cdot\ddot{y}=-m\cdot\dot{x}\cdot\dot{\psi}+F_{y_{f,l}}+F_{y_{f,r}}+F_{y_{r,l}}+F_{y_{r,r}}; \\ I\cdot\ddot{\psi}=\left(F_{y_{f,l}}+F_{y_{f,r}}\right)\cdot a-\left(F_{y_{r,l}}+F_{y_{r,r}}\right)\cdot b+ \\ \left(-F_{x_{f,l}}+F_{x_{f,r}}-F_{x_{r,l}}+F_{x_{r,r}}\right)\cdot c, \end{gathered} \tag{4}$$

where: m is the vehicle mass; I is the yaw moment of vehicle inertia. The forces acting on the center of gravity F_x and F_y are computed by resolving the longitudinal force F_l and lateral force F_c as:

$$\begin{gathered} F_x=F_l\cdot\cos\delta-F_c\cdot\sin\delta; \\ F_y=F_l\cdot\sin\delta+F_c\cdot\cos\delta. \end{gathered} \tag{5}$$

A 'magic formula' model (Pacejka, Bakker 1992) is used to generate tire longitudinal and lateral forces. Tire forces for each tire are obtained by:

$$\begin{gathered} F_l=f_l\left(\alpha,\mu,s,F_z\right); \\ F_c=f_c\left(\alpha,\mu,s,F_z\right), \end{gathered} \tag{6}$$

where: α is the slip angle; μ is the road friction coefficient; F_z is the normal tire load and s is the slip ratio defined as:

$$s=\begin{cases} 1-\dfrac{v_l}{\omega\cdot r} & \text{if } v_l<\omega\cdot r, w\neq 0 \text{ (driving);} \\ \dfrac{\omega\cdot r}{v_l}-1 & \text{if } v_l>\omega\cdot r, v\neq 0 \text{ (braking),} \end{cases} \tag{7}$$

where: v_l is the longitudinal wheel speed, r and w is the wheel radius and angular speed respectively. The wheel's angular speed is calculated as:

$$\dot{\omega}_{ij}=\frac{1}{I_w}\cdot\left(T_{dij}-T_{bij}-F_{lij}\cdot r\right), \tag{8}$$

where: I_w is the wheel inertia; T_{dij} and T_{bij} are the driving and braking torques respectively at each wheel, and the subscript ij denotes the four independent wheels.

The lateral and longitudinal motion equations of vehicle in an absolute inertial frame are:

$$\begin{gathered} \dot{Y}=\dot{x}\cdot\sin\psi+\dot{y}\cdot\cos\psi; \\ \dot{X}=\dot{x}\cdot\cos\psi-\dot{y}\cdot\sin\psi. \end{gathered} \tag{9}$$

With above equations, the vehicle model can be rewritten as:

$$\begin{gathered} \dot{\xi}=f\left(\xi,u\right); \\ \eta=h\left(\xi\right), \end{gathered} \tag{10}$$

where the state and input vectors are $\xi=\left\{X,Y,\dot{x},\dot{y},\psi,\dot{\psi}\right\}$ and $u=\left\{\delta,s_{f,l},s_{f,r},s_{r,l},s_{r,r}\right\}$ respectively, and the output η is obtained by:

$$\eta\left(\xi\right)=\begin{bmatrix} 0&1&0&0&0&0 \\ 0&0&0&0&1&0 \\ 0&0&0&0&0&1 \end{bmatrix}\xi. \tag{11}$$

2.2. MPC-Based Controller

Next an MPC controller calculating the steering angle and the slip ratios is established in order to track a desired path, while keeping the longitudinal and lateral positions as close as possible to a given reference. The slip ratios can be further delivered to compute wheel torques by resolving the wheel angular speed algorithm at each sampling instant. Since the outputs of model predictive controller will be conducted a series of discretization, it is beneficial to calculate the derivative of Eq. (7) numerically and obtain the wheel's angular acceleration. The driving or braking wheel torques can be figured out finally by substituting previous-derived wheel's angular acceleration into the differential Eq. (8). Fig. 7 depicts the control scheme of autonomous electric vehicle.

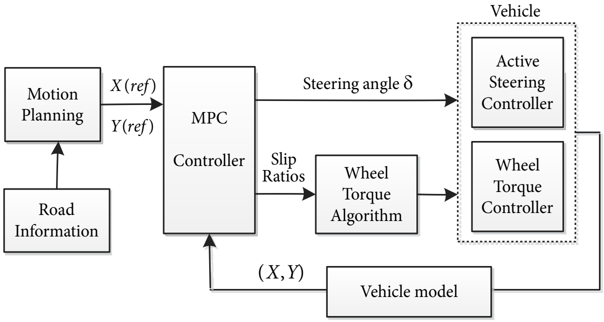

Fig. 7. The control scheme for autonomous electric vehicle

In order to obtain a finite dimentional optimal control problem, discretize system dynamics with the Euler approximation (Borrelli *et al.* 2005), and obtain:

$$\xi(k+1) = f\big(\xi(k), \Delta u(k)\big);$$
$$\eta(k) = h\big(\xi(k)\big);$$
$$\Delta u(k) = u(k) - u(k-1). \qquad (12)$$

Consider the subsequent cost function:

$$J\big(\xi(t), \Delta U_t\big) = \sum_{k=1}^{H_p} \left\| \hat{\eta}_{t+k,t} - \eta_{ref_{t+k,t}} \right\|_Q^2 + \sum_{k=0}^{H_c-1} \left\| \Delta u_{t+k,t} \right\|_R^2 + \sum_{k=0}^{H_c-1} \left\| u_{t+k,t} \right\|_S^2, \qquad (13)$$

where: $\Delta U_t = \Delta u_{t,t}, \ldots, \Delta u_{t+Hc+1,t}$ is the optimal vector at time t; $\hat{\eta}_{t+k,t}$ and $\eta_{ref_{t+k,t}}$ denote the predicted and desired output vectors at time t respctively. H_p is the prediction horizon and H_c is the control horizon with $H_p > H_c$. Q is the output weight matrix, R is the input weight matrix and S denotes the weights for control input rates.

The linearization of non-linear system (12) is carried out at general non-equilibrium points (Chang, Gordon 2007), and at each sampling instant the optimizatin problem is solved as:

$$\min_{\Delta U_t} J\big(\xi_t, U_t\big) \qquad (14a)$$

Subject to:

$$\xi_{k+1,t} = A_t \xi_{k,t} + B_t u_{k,t}, \qquad (14b)$$

$$\eta_{k,t} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \xi_{k,t} \qquad (14c)$$

with $k = t, \ldots, t + H_p - 1$;

$$u_{k,t} = u_{k-1,t} + \Delta u_{k,t}, \qquad (14d)$$

$$u_{\min} \le u_{k,t} \le u_{\max} \qquad (14e)$$

with $k = t, \ldots, t + H_c - 1$;

$$\Delta u_{\min} \le \Delta u_{k,t} \le \Delta u_{\max}. \qquad (14f)$$

Eqs (14b) and (14c) represent the system equations, where A_t and B_t are the parameter matrices at each instant. Eq. (14d) gives the control sequence of optimal steering and increments in the slip ratios computed at sampling instant k by optimizing the cost function (Eq. (14a)) for the state ξ_k and the input at the previous sample step $u(k-1)$. The first element of the obtained optimal control input sequence $u_{k,t}$ will be applied to control the vehicle. The constraint defined in Eq. (14e) limits the steering angle and wheel slip ratios. A limit on the input rates is defined in Eq. (14f).

3. Simulation and Analysis

To evaluate the feasibility of the motion-planning program and MPC-based controller, we consider two scenarios where the vehicle intelligently changes lane and accomplishes a right-angle turn respectively. The control goal is to track the trajectory as close as possible by minimizing the vehicle deviation from the expected path and the control inputs are the front tire steering angle and the wheel slip ratios.

For the lane-change case, the planned tajectory shown in Fig. 3 is tracked on the basis of MPC controller. The following control parameters have been used:

- sampling time:

  $T = 0.05$ s;
- horizons:

  $H_p = 12$, $H_c = 4$;
- bounds:

  $-10 \le \delta \le 10$ deg;

  $-1 \le \Delta\delta \le 1$ deg;

  $-3 \le s \le 3\%$;
- weights on tracking errors:

  $Q_Y = 20$;

  $Q_\psi = 100$;

  $Q_{\dot{\psi}} = 1$;
- weights on input rates:

  $R_\delta = 100$;

  $R_{s_{f,l}} = R_{s_{f,r}} = R_{s_{r,l}} = R_{s_{r,r}} = 0.01$;

  $R_{ij} = 0$ for $i \ne j$;
- weights on inputs:

  $S_\delta = 10^{-7}$;

  $S_{s_{f,l}} = S_{s_{f,r}} = S_{s_{r,l}} = S_{s_{r,r}} = 10^{-5}$;

  $S_{ij} = 0$ for $i \ne j$.

Fig. 8 shows the demanded steering angle and Fig. 9 reports the tracking variables, including the lateral vehicle position, the yaw angle and the yaw rate. In Figs 10 and 11 the slip ratios and wheel torques at the four wheels are shown. The simulation reflects a good tracking, but the delay of steering operation is noticed in Fig. 8, which has to be properly traded off with the precision of tracking planned trajectories. It is observed that the controller is able to well coordinate wheel torque control and the steering manoeuver in Figs 8 and 11. Between 0 and 2.5 s, the vehicle steers to the left to generate a positive yaw moment. Meanwhile, braking torques at both left-hand side wheels and traction torques at both right-hand side wheels are generated, which produces a positive yaw moment and helps the steering action. An opposite behavior can be observed for the rest of simulation.

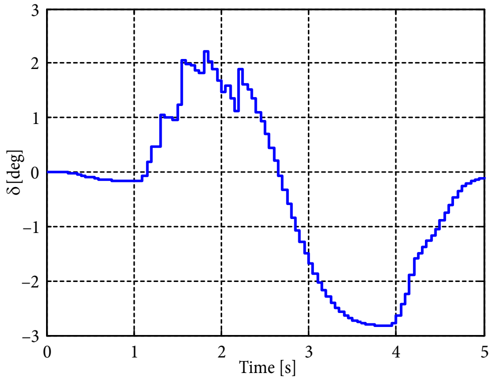

Fig. 8. Steering angle (lane-change case)

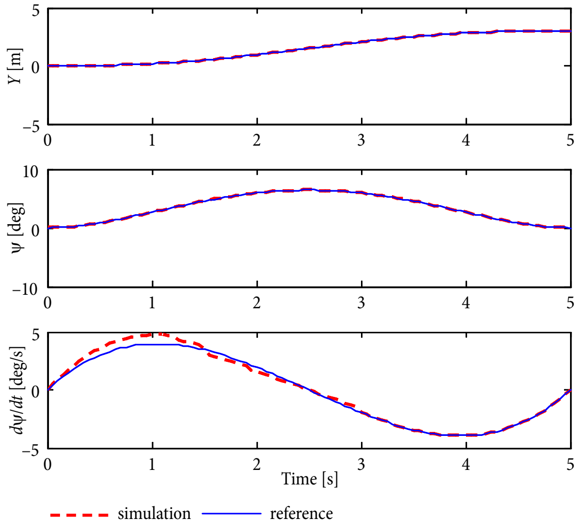

Fig. 9. Tracking trajectory comparison (lane-change case)

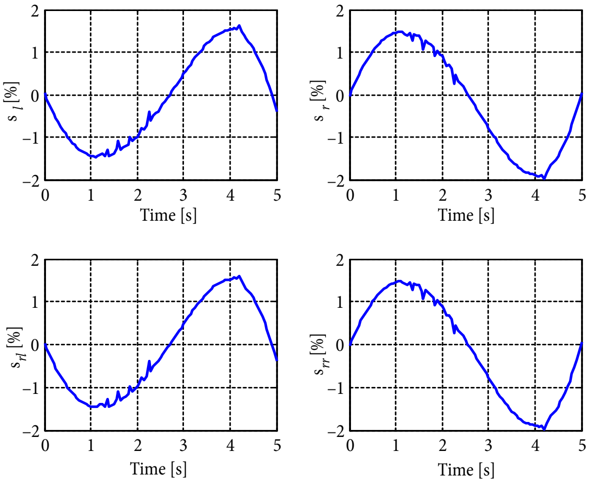

Fig. 10. Wheel slip ratios (lane-change case)

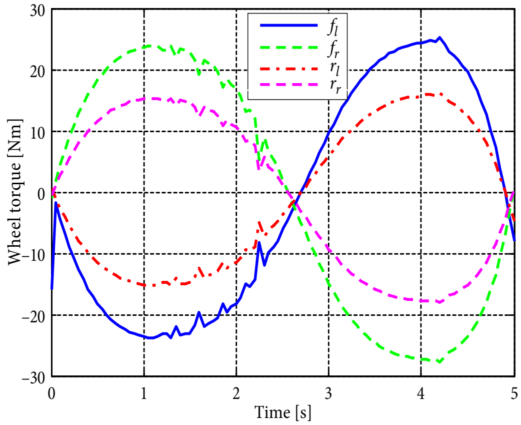

Fig. 11. Wheel torques (lane-change case)

Figs 12–15 present the simulation results in the scenario of right-angle turn. The following control parameters have been used:

– sampling time:

T = 0.05 s;

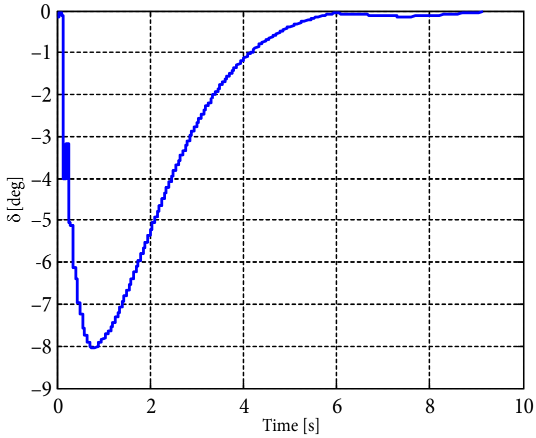

Fig. 12. Steering angle (right-angle turn case)

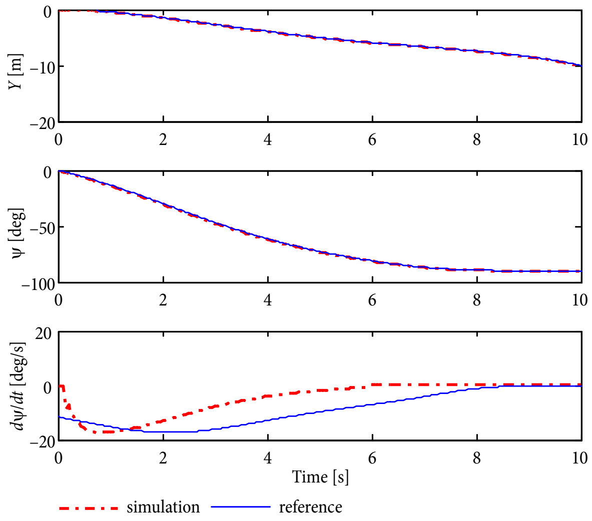

Fig. 13. Tracking trajectory comparison (right-angle turn case)

– horizons:

$H_p = 20$, $H_c = 6$;

– bounds:

$-10 \le \delta \le 10$ deg;

$-1 \le \Delta\delta \le 1$ deg;

$-3 \le s \le 3\%$;

– weights on tracking errors:

$Q_Y = 20$;

$Q_\psi = 100$;

$Q_{\dot\psi} = 1$;

– weights on input rates:

$R_\delta = 10^3$;

$R_{s_{f,l}} = R_{s_{f,r}} = R_{s_{r,l}} = R_{s_{r,r}} = 10^{-3}$;

$R_{ij} = 0$ for $i \neq j$;

– weights on inputs:

$S_\delta = 10^{-7}$;

$S_{s_{f,l}} = S_{s_{f,r}} = S_{s_{r,l}} = S_{s_{r,r}} = 10^{-3}$;

$S_{ij} = 0$ for $i \neq j$;

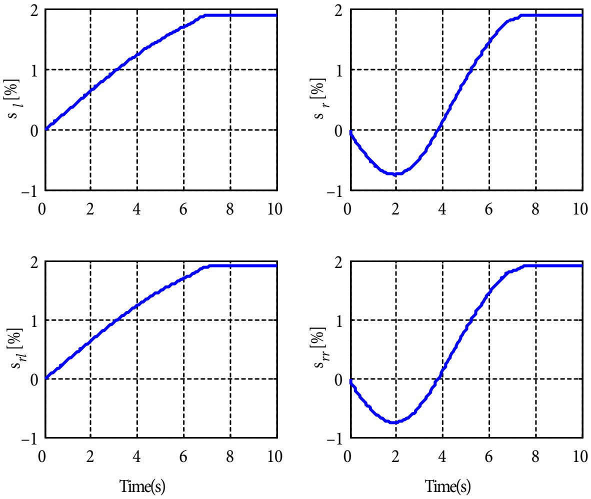

Fig. 14. Wheel slip ratios (right-angle turn case)

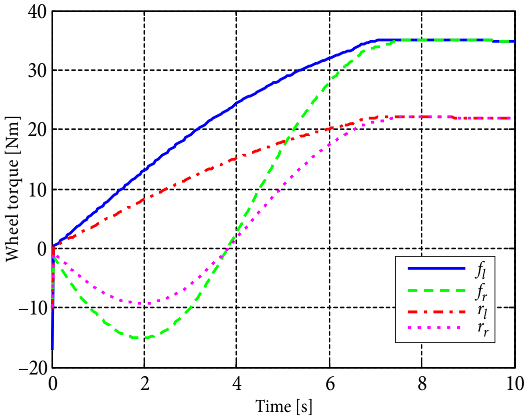

Fig. 15.Wheel torques (right-angle turn case)

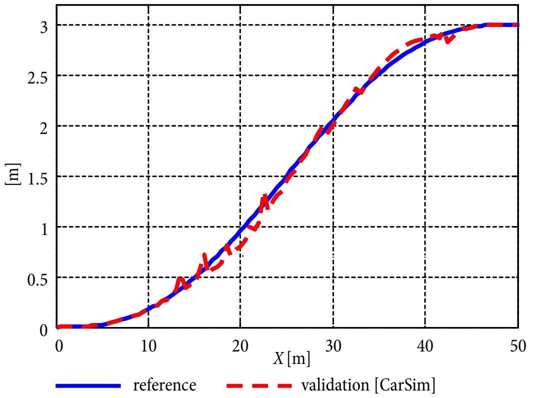

Fig. 16. Validation of controller compared with the reference path (lane-change case)

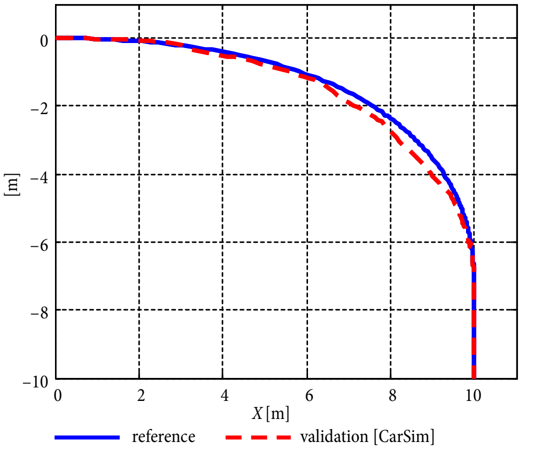

Fig. 17. Validation of controller compared with the reference path (right-angle turn case)

Fig. 13 shows a good tracking of the planned path in the right-angle turn scenario. In Figs 12 and 15, we also notice the cooperation of the steering and wheel torques in order to achieve the necessary yaw moment. When the controller executes a right steering in the first 6 seconds, the combination of wheel torques at the two sides achieves a negative yaw moment. After 7 s the vehicle nearly completes the right turn, the steering angle approaches to zero and wheel torques no longer generate any yaw moment.

The feasibility of proposed MPC-based controller on the actual electric vehicle is further evaluated with a high-fidelity full-vehicle model from *CarSim*, which replaces the mathematical vehicle model in Fig. 7. The parameters in the *CarSim* model are listed in Table and the comparisons between the calculated and referred trajectories are shown in Figs 16–17. The validation results prove that the adopted MPC-based controller can guarantee an overall good tracking of planned trajectories, although the tracking performance should be improved (probably through more tuning work) in order to eliminate oscillations and partial deviations.

Table. Main parameters of vehicle model in *CarSim*

Parameter	Symbol	Value
Sprung mass	m	1125 kg
Yaw moment of inertia	I	1519 kg·m^2
Distance of c.g to front axle	a	1.1 m
Distance of c.g to rear axle	b	1.3 m
Half track of front and rear axle	c	(0.7, 0.7) m
Effective wheel rolling radius	r	0.3 m
Wheel rotational inertia	I_ω	1.28 kg·m^2

Conclusions

In this paper a motion-planning system and an MPC-based controller are developed for autonomous electric vehicles.

The motion-planning program determines desired tajectories for specified scenarios, and then the MPC controller computes the steering angle and torques at each wheel.

The simulation results and validation in *CarSim* show that the proposed planner algorithm and controller are feasible and can achieve requirements of autonomous driving in normal scenarios.

Future research will aim to improve the motion-planning algorithm and deal with a more complex driving environment considering possible obstacles.

Acknowledgements

The work is supported by NSFC (No. 51575103, 51375086), Foundation of State Key Laboratory of Mechanical Transmission (SKLMT-KFKT-201206) and Foundation of State Key Laboratory of Automotive Safety and Energy (KF14192). Six talent Peaks Project in Jiangsu Province (No. 2014-JXQC-001) and Qing Lan Project.

References

Borrelli, F.; Falcone, P.; Keviczky, T.; Asgari, J.; Hrovat, D. 2005. MPC-based approach to active steering for autonomous vehicle systems, *International Journal of Vehicle Autonomous Systems* 3(2–4): 265–291. http://dx.doi.org/10.1504/IJVAS.2005.008237

Chang, S.; Gordon, T. J. 2007. Model-based predictive control of vehicle dynamics, *International Journal of Vehicle Autonomous Systems* 5(1–2): 3–27. http://dx.doi.org/10.1504/IJVAS.2007.014945

Durali, M.; Javid, G. A.; Kasaiezadeh, A. 2006. Collision avoidance maneuver for an autonomous vehicle, in *9th IEEE International Workshop on Advanced Motion Control*, 27–29 March 2006, Istanbul, Turkey, 249–254. http://dx.doi.org/10.1109/AMC.2006.1631666

Falcone, P.; Borrelli, F.; Asgari, J.; Tseng, H. E.; Hrovat, D. 2007. Predictive active steering control for autonomous vehicle systems, *IEEE Transactions on Control Systems Technology* 15(3): 566–580. http://dx.doi.org/10.1109/TCST.2007.894653

Falcone, P.; Tseng, H. E.; Borrelli, F.; Asgari, J.; Hrovat, D. 2008. MPC-based yaw and lateral stabilisation via active front steering and braking, *Vehicle System Dynamics: International Journal of Vehicle Mechanics and Mobility* 46(Supplement): 611–628. http://dx.doi.org/10.1080/00423110802018297

Kim, B.; Necsulescu, D.; Sasiadek, J. 2004. Autonomous mobile robot model predictive control, *International Journal of Control* 77(16): 1438–1445. http://dx.doi.org/10.1080/00207170412331317738

Koenig, S.; Likhachev, M. 2002. Improved fast replanning for robot navigation in unknown terrain, in *Proceedings: ICRA'02: IEEE International Conference on Robotics and Automation*, 11–15 May 2002, Washington, DC, 1: 968–975. http://dx.doi.org/10.1109/ROBOT.2002.1013481

Kuwata, Y.; Karaman, S.; Teo, J.; Frazzoli, E.; How, J. P.; Fiore, G. 2009. Real-time motion planning with applications to autonomous urban driving, *IEEE Transactions on Control Systems Technology* 17(5): 1105–1118. http://dx.doi.org/10.1109/TCST.2008.2012116

Lafta, H. A.; Hassan, Z. F. 2013. A hybrid system geno-fuzzified neural network for mobile robot control, *British Journal of Mathematics & Computer Science* 3(4): 724–739. http://dx.doi.org/10.9734/BJMCS/2013/4587

Levinson, J.; Askeland, J.; Becker, J., *et al.* 2011. Towards fully autonomous driving: systems and algorithms, in *Proceedings of the 2011 IEEE Intelligent Vehicles Symposium (IV)*, 5–9 June 2011, Baden-Baden, Germany, 163–168. http://dx.doi.org/10.1109/IVS.2011.5940562

Lin, C.-C.; Chuang, J.-H. 2010. A potential-based path planning algorithm for hyper-redundant manipulators, *Journal of the Chinese Institute of Engineers* 33(3): 415–427. http://dx.doi.org/10.1080/02533839.2010.9671630

Naranjo, J. E.; González, C.; García, R.; De Pedro, T.; Sotelo, M. A. 2007. Using fuzzy logic in automated vehicle control, *IEEE Intelligent Systems* 22(1): 36–45. http://dx.doi.org/10.1109/MIS.2007.18

Pacejka, H. B.; Bakker, E. 1992. The magic formula tyre model, *Vehicle System Dynamics: International Journal of Vehicle Mechanics and Mobility* 21(Supplement): 1–18. http://dx.doi.org/10.1080/00423119208969994

Park, J.-M.; Kim, D.-W.; Yoon, Y.-S.; Kim, H.-J.; Yi, K.-S. 2009. Obstacle avoidance of autonomous vehicles based on model predictive control, *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering* 223(12): 1499–1516. http://dx.doi.org/10.1243/09544070JAUTO1149

Phinni, M. J.; Sudheer, A. P.; Rama Krishna, M.; Jemshid, K. K. 2008. Obstacle avoidance of a wheeled mobile robot: a genetic-neuro-fuzzy approach, in *IISc Centenary – International Conference on Advances in Mechanical Engineering (IC-ICAME)*, 2–4 July 2008, Bangalore, India, 1–3.

Shim, T.; Adireddy, G.; Yuan, H. 2012. Autonomous vehicle collision avoidance system using path planning and model-predictive-control-based active front steering and wheel torque control, *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering* 226(6): 767–778. http://dx.doi.org/10.1177/0954407011430275

Urmson, C.; Anhalt, J.; Bagnell, D., *et al.* 2008. Autonomous driving in urban environments: boss and the urban challenge, *Journal of Field Robotics* 25(8): 425–466. http://dx.doi.org/10.1002/rob.20255

Yershov, D. S.; LaValle, S. M. 2011. Simplicial Dijkstra and A* algorithms for optimal feedback planning, in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 25–30 September 2011, San Francisco, CA, 3862–3867. http://dx.doi.org/10.1109/IROS.2011.6095032