VEHICLE SYSTEM DYNAMICS
2024, VOL. 62, NO. 6, 1517–1540
https://doi.org/10.1080/00423114.2023.2242533

An integrated path-tracking and control allocation method for autonomous racing electric vehicles

Boyuan Li[a,b], Chenhui Lin[a], Javad Ahmadi[a,c], Efstathios Siampis[a], Stefano Longo[a] and Efstathios Velenis[a]

[a]The Centre for Automotive Engineering, School of Aerospace, Transport and Manufacturing, Cranfield University, Cranfield, UK; [b]Research Centre for Intelligent Transportation, Zhejiang Lab, Hangzhou, People's Republic of China; [c]School of Computing and Engineering, University of Derby, Derby DE22 3AW United Kingdom

ABSTRACT

In recent years, path-tracking controllers for autonomous passenger vehicles and Control Allocation (CA) methods for handling and stability control have both received extensive discussion in the literature. However, the integration of the path-tracking control with CA methods for autonomous racing vehicles has not attracted much attention. In this study, we design an integrated path-tracking and CA method for a prototype autonomous racing electric vehicle with a particular focus on the maximising the turning speed in tight cornering. The proposed control strategy has a hierarchical structure to improve the computational efficiency: the high-level path-tracking Model Predictive Control (MPC) based on a rigid body model is designed to determine the virtual control forces according to the desired path and desired maximum velocity profile, while the low-level CA method uses a Quadratically Constrained Quadratic Programming (QCQP) formulation to distribute the individual control actuator according to the desired virtual control values. The proposed controller is validated in a high-fidelity simulation vehicle model with the computational time of the optimisation controller presented to demonstrate the real-time control performance.

ARTICLE HISTORY
Received 4 March 2022
Revised 6 July 2023
Accepted 21 July 2023

KEYWORDS
Autonomous vehicle; control allocation; MPC; path-tracking

1. Introduction

The autonomous vehicle has been considered as the next generation vehicle and has received increased attention in the recent literature. A vital part of the control architecture of an autonomous vehicle is around the path following control, which includes two parts: the high-level motion planner, which uses environmental sensing and perception techniques to determine a desired path and speed profile, and the low-level path-tracking controller which is designed to track the desired path with desired speed and maintain the vehicle dynamics stability. In this study, we focus on the low-level path-tracking controller.

Many of earlier studies about the path-tracking used a driver model to determine the steering control input [1–5], then the study on path-tracking for driver model has

CONTACT Boyuan Li ✉ 1072704859@qq.com

This article has been republished with minor changes. These changes do not impact the academic content of the article.

been switched to autonomous passenger vehicle application recently [6–11]. In order to track the desired path and minimise the lateral tracking error, Attia et al. utilised a Nonlinear Model Predictive Control (NMPC) to determine the control input of steering angle [6]. In another study, based on the lateral tracking error, the H-infinity technique was applied to design the lateral tracking controller [7]. Tagne et al. proposed a higher-order Sliding Mode Controller (SMC) for the reference path-tracking control, since the robustness of SMC can be beneficial to overcome the nonlinear terms of the vehicle model [9].

In recent years, the independent wheel control in an over-actuated vehicle has been widely discussed in the vehicle dynamics and control literature. The independent wheel control provides redundant control actuators and is becoming popular due to the possibility of enhancing the performance of existing controller through the carefully design of Control Allocation (CA) strategy [12]. It is also suggested that CA can improve the cornering response by shaping the under-steer characteristics and ensuring stability in limited-handing condition [13]. Thus, integrating the CA strategy with path-tracking controller can greatly improve path-tracking and vehicle dynamics performance of autonomous vehicle with in-wheel motor, which has become a very attracting research area.

CA was originally used to control and allocate traction or brake toque of individual tire for the vehicle handling and stability control of traditional human-driving passenger vehicles [14–18]. In recent years, quite a few approaches integrate path-tracking with CA based on Direct Yaw Control (DYC) for autonomous passenger vehicles [19,20]. Guo et al. proposed a Linear Time Varying (LTV) based MPC as the path-tracking controller to determine the steering angle and required yaw moment, then a low-level CA law is applied to distribute the individual wheel torque to achieve the required yaw moment [19]. Wang et al. developed a modified Composite Nonlinear Feedback (CNF) controller to track the desired yaw rate and lateral velocity and then the required yaw moment control value is distributed to longitudinal tire force of individual wheel [20]. In [13], a hierarchical structure and a holistic structure of path-tracking controller is suggested and integrated with torque vectoring controller. In the above studies, a distribution of the DYC command on the individual wheel torque (also called torque vectoring) is applied. However, the CA problem can be extended to the distribution of the steering angle, motor driving torque, motor regenerating brake torque and hydraulic brake torque on the individual wheel. In this way, the redundant vehicle control actuators can be fully utilised to achieve the best control performance, but CA problem becomes more complex and requires a carefully designed CA strategy. In our previous study [21], the hierarchical control structure is proposed by including high-level PID based path-tracking controller and low-level CA approach. The CA method can optimally distribute the individual wheel torque and front wheel steering angle. Nevertheless, the simple PID path-tracking controller can only track the desired path with fixed longitudinal speed, which cannot optimise the path-tracking with maximum cornering speed in racing scenario (without the help of MPC). Furthermore, the CA method still has not fully considered the control allocation of front and rear wheel steering angle, motor driving torque, motor regenerating brake torque and hydraulic brake torque of individual wheel.

MPC has been widely applied in the path-tracking control of autonomous passenger vehicles since the control performance can be greatly improved by MPC through the preview of the future road profile [19,22–27]. However, there are limited studies on the path-tracking control of the autonomous racing vehicles. Alcala et al. proposed a linear parameter varying (LPV) MPC for autonomous racing vehicles [28]. The validation of the controller is based on the scaled electric vehicle and the vehicle speed is relatively low. It is questionable if the proposed controller can achieve good control performance for the high-speed full-sized autonomous racing vehicle. Similarly, Liniger et al. proposed a linearised MPC for the path-tracking of autonomous racing vehicle and an integrated MPC contouring control to achieve the combined path planning and path-tracking control, where the nonlinear bicycle model considering the longitudinal and lateral friction forces is modified as the LTV model [29]. In order to improve the internal model accuracy of the MPC, the learning based MPC approaches are proposed in [30]. The internal MPC model includes a nominal simple vehicle model and a self-learning model to improve the vehicle model accuracy. The major issue of this data-driven self-learning approach is the uncertain stochastic internal model will render the predicted state a random variable. This uncertainty will propagate throughout the whole prediction horizon, and the prediction model system may be unstable with a longer prediction horizon. Furthermore, the Gaussian Process Regression has been applied to train the self-learning model and the large number of training data points will significantly increase the computational burden.

In summary, in the current literature, the path-tracking controller design for autonomous racing vehicles is mainly based on the linearised MPC approach with simple internal model to improve the computational efficiency and guarantee the controller reliability. Although the self-learning MPC approach has been tried to improve the internal model accuracy, the reliability of this approach is questionable. Although CA and MPC approaches have been extensively applied in the path-tracking control for passenger vehicles, the integration of the path-tracking control and maximum speed control in tight cornering for the racing vehicle is less focused. The optimisation of redundant steering and driving actuators can greatly enlarge the working envelope of racing vehicles and push the racing vehicles close to the friction limit.

The aim of our study is to design an integrated path-tracking and CA method for a prototype autonomous racing electric vehicle in limited handling scenario, with the proposed method being able to satisfy the requirement for real-time implementation. The prototype vehicle under consideration has independent front- and rear-wheel steering, two hub motors with driving/regenerating braking capabilities on the two rear wheels, and one motor on the front axle that through an open differential equally distributes the driving/regenerating torque among the two front wheels. In addition, there are two independent hydraulic brake channels, one on the front axle and one on the rear axle of the vehicle, which is shown in Figure 1. Motivated by the above shortcomings in current literature, a hierarchical control structure of integrated path-tracking and CA method is presented in this study. In the high-level controller, a path-tracking MPC is proposed based on a rigid body vehicle model including the virtual control inputs of the total longitudinal, lateral force and yaw moment on the vehicle, while individual tire friction force and tire model are considered in the low-level controller and decoupled from high-level to improve the computational efficiency. This way a modular strategy is achieved, which can be easily modified and applied to most of the existing actuator structures of an electric vehicle.

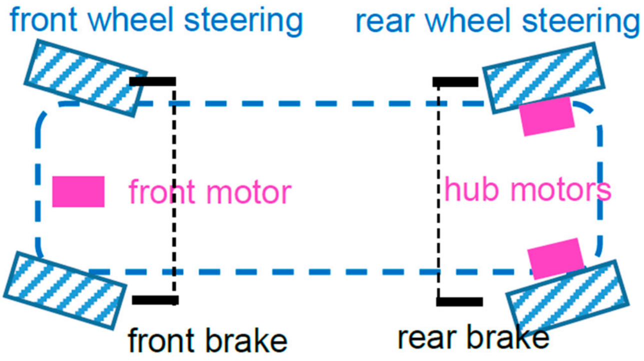

Figure 1. The structure diagram of the complete control system.

The major contribution of this study can be summarised as follows. First, the proposed hierarchical controller applies the virtual forces and moment to performance high-level path-tracking motion and effectively move the effect of the actual friction tire forces in the lower CA level which also accounts for the various actuators. Secondly, the path-tracking MPC with prediction horizon is proposed based on a rigid body vehicle dynamics model, and the tire friction circle and maximum yaw moment constraints are included in the high-level MPC to guarantee the feasibility of the computed commands to be passed on the low-level CA. Furthermore, a two-layer CA method is proposed to control and allocate the various steering and driving actuators: the upper layer CA optimally distributes the individual longitudinal tire force and steering angle according to the desired virtual control values from the path-tracking controller, formulated as a Quadratically Constrained Quadratic Programming (QCQP) problem to improve the computational efficiency, while the lower layer rules-based method is designed to distribute the required individual longitudinal tire force into the individual motor and hydraulic brake torques. Finally, the time-efficient solver generated by Forces Pro is used to solve the high-level path-tracking MPC and optimisation problem of CA in low-level in this study, which can satisfy the requirement for real-time path-tracking control [31]. It is noted that our study only focuses on the high-level path-tracking and low-level CA algorithm, and the detailed actuator model of the powertrain system is out of the scope and not considered in our study. The major novelty of our study can be suggested: (1) the path-tracking control and maximum speed control in tight cornering (through the offline pre-calculation of maximum cornering speed and the online speed control in MPC prediction horizon) are integrated together for the autonomous racing vehicle; (2) through more powerful in-wheel motor system and the potent racing tire (tire-road friction coefficient > 1), the proposed controller can fully utilise the potential of the racing vehicle and reach the acceleration limit; (3) in the high-level MPC path-tracking controller, the virtual total desired tyre force and yaw moment (yaw moment constraint is not well addressed in the literature) are constrained so that the low-level CA can successfully allocate individual actuators.

This paper is organised as follows: the high-level path-tracking MPC is presented in section II, then low-level CA strategy is introduced in section III, and finally simulation examples are used to validate the effectiveness of the proposed integrated control method in section IV.

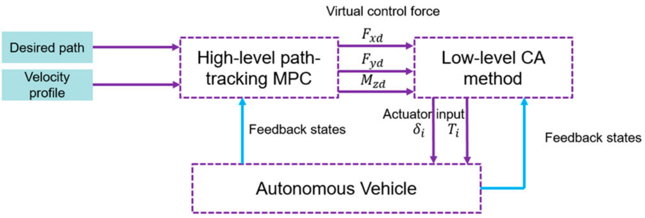

Figure 2. The structure diagram of the complete control system.

2. High-level path-tracking MPC design

The proposed hierarchical controller includes the high-level path-tracking controller and low-level CA method. The high-level path-tracking MPC method tracks the desired path and velocity profile and sends the required virtual control forces into the low-level CA to distribute the individual control actuator value to the vehicle. Vehicle measurements are also used to feed back the necessary vehicle state values to the high-level and low-level controllers. The detailed structure of the complete control system is presented in Figure 2.

2.1. Vehicle dynamics model for high-level path-tracking MPC

In this section, a three degrees-of-freedom (DOF) rigid-body nonlinear dynamics model considering the combined longitudinal, lateral and yaw dynamics is used to design the path-tracking MPC. The virtual control inputs of total longitudinal tire force, lateral tire force and yaw moment are utilised to simplify the vehicle dynamics model. In addition, the yaw angle error and lateral tracking error are also included in the vehicle model to present the path-tracking performance [20]. The 3 DOF vehicle model is presented by the following equations:

$$\dot{v}_x = v_y r + \frac{F_{xd}}{m} \tag{1a}$$

$$\dot{v}_y = -v_x r + \frac{F_{yd}}{m} \tag{1b}$$

$$\dot{r} = \frac{M_{zd}}{I_z} \tag{1c}$$

$$\dot{\psi}_e = r - \kappa v_x \tag{1d}$$

$$\dot{Y}_e = v_x \sin\psi_e + v_y \cos\psi_e \tag{1e}$$

where v_x, v_y, r represent the vehicle longitudinal velocity, lateral velocity, and yaw rate, respectively, I_z and m are the moment of vehicle inertia in terms of yaw axis and vehicle mass, F_{xd}, F_{yd}, M_{zd} are total longitudinal tire force, lateral tire force and yaw moment, Y_e and ψ_e are the vehicle lateral deviation and yaw angle error from the reference path and κ is the curvature of desired path.

2.2. The design of high-level path-tracking MPC

The general formulation of a nonlinear MPC can be presented as following equations by selecting the best control input to satisfy the cost function J:

$$\min_{x,u} \sum_{k=0}^{N-1} J(x_k, u_k, p_k)$$

Subject to: $x_{k+1} = f(x_k, u_k, p_k)$

$$k = 0, 1, \ldots N-1 \tag{2a}$$

$$x_k \in Xk = 1, \ldots N-1 \tag{2b}$$

$$x_0 = x(t) \tag{2c}$$

where N is the horizontal number of predicted time steps, x_k is the predicted vehicle state in the prediction horizon and u_k is the predicted control input, $f(x_k, u_k, p)$ in equation (2a) is the nonlinear system prediction model, which is the discrete formulation of nonlinear dynamics system (1) and is formulated in (4) and p is the external changing parameters determined outside of MPC. The boundary constraints of vehicle states are presented in (2b) while the initial values x_0 can be determined by the measured feedback state values at current time t.

1) External parameters p

The external parameters of MPC include the curvature of desired path $p_1 = \kappa$ and desired longitudinal velocity of vehicle in the body-fixed coordinate system $p_2 = v_{xd}$. Since this study only focuses on the path-tracking controller design, it can be simply assumed the desired path and the curvature of desired path are already known. The desired longitudinal velocity is determined by an optimal velocity profile which can be pre-calculated offline or online as part of the motion planning algorithm. Note that the optimal velocity profile calculated offline and applied in this study can make the autonomous vehicle travel along the desired path with maximum acceleration and minimum time [32].

The MPC design has N steps prediction horizon, determined as:

$$\kappa_k = \kappa(X_d(t) + v_x k T_s) k = 0, 1, \ldots N-1 \tag{3}$$

where $X_d(t)$ is the vehicle current time (t) travelling distance along the desired path. v_x is the feedback vehicle velocity and T_s is the sampling time of discrete-time vehicle model (2a).

2) Nonlinear prediction model

The nonlinear prediction model (2a) is determined by converting the nonlinear continuous dynamics model (1) into discrete form:

$$v_{x,k+1} = v_{x,k} + T_s \left(v_{y,k} r_k + \frac{F_{xd,k}}{m} \right) \tag{4a}$$

$$v_{y,k+1} = v_{y,k} + T_s\left(-v_{x,k}r_k + \frac{F_{yd,k}}{m}\right) \tag{4b}$$

$$r_{k+1} = r_k + T_s\frac{M_{zd,k}}{I_z} \tag{4c}$$

$$F_{xd,k+1} = F_{xd,k} + T_s\dot{F}_{xd} \tag{4d}$$

$$F_{yd,k+1} = F_{yd,k} + T_s\dot{F}_{yd} \tag{4e}$$

$$M_{zd,k+1} = M_{zd,k} + T_s\dot{M}_{zd} \tag{4f}$$

$$\psi_{e,k+1} = \psi_{e,k} + T_s(r_k - p_{1,k}v_{x,k}) \tag{4g}$$

$$Y_{e,k+1} = Y_{e,k} + T_s(v_{x,k}\sin\psi_{e,k} + v_{y,k}\cos\psi_{e,k}) \tag{4h}$$

where the model states $x_k = [v_{x,k}; v_{y,k}; r_k; F_{xd,k}; F_{yd,k}; M_{zd,k}; \psi_{e,k}; Y_{e,k}]^T$, and the control inputs $u_k = [\dot{F}_{xd,k}; \dot{F}_{yd,k}; \dot{M}_{zd,k}]^k$. It should be noted that in (4) the control inputs have been revised as the change rate of total desired longitudinal tire force, lateral force and yaw moment to reduce the oscillation of control inputs and improve the stability of control performance.

3) Cost function

The control targets of proposed MPC include the minimisation of yaw angle error and lateral deviation from desired path and achieving the desired target speed. The detailed mathematical presentation of the cost function can be presented as following:

$$J = \sum_{k=0}^{N-1} Q_v(v_{xd,k} - v_{x,k})^2 + Q_\psi(\psi_{e,k})^2 + Q_Y(Y_{e,k})^2 + {u_k}^T Q_u u_k + Q_{s1}s_1^2 + Q_{s2}s_2^2 \tag{5}$$

where Q_v, Q_ψ, Q_Y are scaling factors corresponding to the cost function terms of the longitudinal velocity tracking error, yaw angle tracking error and lateral deviation tracking error. Q_u is a 3-by-3 diagonal matrix which include the relative penalties on the control effort. Q_{s1} and Q_{s2} are penalties on the soft constrains s_1 and s_2. In order to achieve good control performance, all the scaling factors in cost function are normalised and written in discrete-time form:

$$Q_v = \frac{q_v T_s}{(v_{xm})^2} \tag{6a}$$

$$Q_\psi = \frac{q_\psi T_s}{(\psi_{em})^2} \tag{6b}$$

$$Q_Y = \frac{q_Y T_s}{(Y_{em})^2} \tag{6c}$$

$$Q_u = \begin{bmatrix} \frac{q_{u1}}{(dF_{xm})^2} & 0 & 0 \\ 0 & \frac{q_{u2}}{(dF_{ym})^2} & 0 \\ 0 & 0 & \frac{q_{u3}}{(dM_{zm})^2} \end{bmatrix} \tag{6d}$$

$$Q_{s1} = \frac{q_{s1} T_s}{(s_{1m})^2} \tag{6e}$$

$$Q_{s2} = \frac{q_{s2}T_s}{(s_{2m})^2} \tag{6f}$$

where v_{xm}, ψ_{em} and Y_{em} are nominal values of longitudinal velocity, yaw angle error and lateral deviation, respectively. $dF_{xm}, dF_{ym}, dM_{zm}$ are nominal values of change rate of desired longitudinal force, desired lateral force and yaw moment. s_{1m} and s_{2m} are nominal values of soft constraints. $q_v, q_\psi, q_Y, q_{u1-u3}, q_{s1-s2}$ are scaling values of each term, which need to be tuned as the following procedure. First the weighting of penalties terms q_{u1-u3}, q_{s1-s2} are tuned and fixed. Then q_v, q_ψ, q_Y related to velocity error, yaw angle error and lateral deviation can be tuned individually based on the relative importance of individual term.

4) Constraints

Equations (2b) suggests that states should satisfy certain constraints. Specifically, the total desired longitudinal and lateral tire force should satisfy the friction circle limit:

$$F_{xd,k}^2 + F_{yd,k}^2 - s_1^2 \le (\mu mg)^2 \tag{7}$$

where μ is the tire-road friction coefficient and g is the constant of gravitational acceleration. The tyre-road friction coefficient can be assumed to be known through friction coefficient estimators [33].

The desired yaw moment is also constrained by its upper and lower limit:

$$M_{zd,k} - M_{zmax} - s_2^2 \le 0 \tag{8a}$$

$$-M_{zd,k} - M_{zmin} - s_2^2 \le 0 \tag{8b}$$

The total desired yaw moment includes the yaw moment induced by the longitudinal force M_{xz} and lateral force M_{yz}. Since longitudinal and lateral vehicle forces are coupled by friction limit, it is quite hard to determine the analytical solution of total yaw moment limit. Thus, in this study, a look-up table is built to explore the whole working envelope and determine the yaw moment limit, which is presented in Table 1. In Table 1, δ_{sf} and δ_{sr} are the searching increments of front-wheel and rear-wheel steering angle. It is also noticed the soft constraints s_1 and s_2 are applied in constraints (7, 8) to prevent the optimisation solver reaching an infeasible solution.

In order to improve the computational efficiency and satisfy the needs of real-time optimisation, the above NMPC problem can be linearised and solved using the fast NLP solver developed by Forces Pro [34], the detailed solving method is explained in Appendix B.

2.3. The high-level feedback path-tracking controller

In order to have the comparative study with our proposed MPC strategy, a high-level feedback path-tracking controller is presented here to calculate the desired total tire force and yaw moment [35]. The desired total longitudinal tire force can be calculated as:

$$F_{xd} = m(-rv_y + \dot{v}_{xd} - k_1 e_1) \tag{9a}$$

where $e_1 = v_x - v_{xd}$, $k_1 > 0$. $\dot{v}_{xd}$ is the desired longitudinal acceleration and can be assumed as zero. Since the specific longitudinal acceleration target is not set in our study,

Table 1. Algorithm to calculate the yaw moment limit.

```
Assume feedback values $v_x, v_y, r, a_x, a_y, \mu, F_{zi}$ (For i = fl, fr, rl, rr) are already available
For $\delta_f(i) = [\delta_{fmin} : \delta_{sf} : \delta_{fmax}]$
    For $\delta_r(j) = [\delta_{rmin} : \delta_{sr} : \delta_{rmax}]$
        If $v_x = 0$
            $F_{yfl} = F_{yfr} = F_{yrl} = F_{yrr} = 0$
        Else
            Calculate the side-slip angle of individual wheel $s_{yi}$
            Calculate the lateral tire force of individual wheel $F_{yi}$
        End
        $M_{yz} = l_f(F_{yfl} + F_{yfr}) - l_r(F_{yrl} + F_{yrr})$
        Calculate the maximum available individual longitudinal force
            $F_{xi}^{max} = \sqrt{(\mu F_{zi})^2 - F_{yi}{}^2}$
        Calculate $M_{xz}^{min}$ and $M_{xz}^{max}$ according to $F_{xi}^{max}$
        $M_z^{min}(i)(j) = M_{xz}^{min} + M_{yz}$, $M_z^{max}(i)(j) = M_{xz}^{max} + M_{yz}$
    End
End
$M_z^{min} = min[M_z^{min}(i)(j)]$, $M_z^{max} = max[M_z^{max}(i)(j)]$
```

we just set the desired longitudinal acceleration as zero to minimise the velocity change and improve the comfort during the desired speed tracking. The desired total lateral tire force can be calculated as:

$$F_{yd} = \frac{m}{\cos\psi_e}[-(\dot{v}_x \sin\psi_e + (r - \kappa_d v_x)(v_x \cos\psi_e - v_y \sin\psi_e)) + v_x r \cos\psi_e - k_2(v_x \sin\psi_e + v_y \cos\psi_e) - k_3 Y_e] \tag{9b}$$

where $k_2 > 0$ and $k_3 > 0$. The desired yaw moment can be calculated as:

$$M_{zd} = [-k_4(r - \kappa_d v_x) - k_5\psi_e + \dot{r}_d]I_z \tag{9c}$$

where $\dot{r}_d$ is the desired yaw acceleration, which is assumed as 0 to improve the yaw stability of the vehicle. $k_4 > 0$ and $k_5 > 0$.

3. Low-level control allocation design

After the virtual control input $u = [F_{xd}; F_{yd}; M_{zd}]$ has been determined by the upper-level MPC, the low-level CA optimisation algorithm is designed to control and allocate the individual actuator to achieve the desired virtual control force. In this study, the low-level CA algorithm is designed assuming a steer-by-wire and brake-by-wire architecture. The powertrain system includes a motor with an open differential on the front axle and two hub motors attached to the rear wheels. Two independent hydraulic brake channels are also present on the front and rear axle, respectively. Finally, the steering system can control the front- and rear-wheel steering independently.

3.1. Vehicle dynamics model for low-level CA

In this study, the design of low-level CA method are based on a two-track vehicle dynamics model. This model is more complex than the rigid-body dynamics model (1) and considers the individual wheel tire forces [36]. Note that the suspension and roll and pitch dynamics are neglected, a reasonable assumption for the on-road autonomous vehicle used in this paper.

The longitudinal motion is defined by:

$$m\dot{v}_x = mv_y r + \sum_{\substack{i=f,r\\ j=l,r}} F_{xij} \tag{10a}$$

the lateral motion:

$$m\dot{v}_y = -mv_x r + \sum_{\substack{i=f,r\\ j=l,r}} F_{yij} \tag{10b}$$

and the yaw motion:

$$I_z\dot{r} = l_f(F_{yfl} + F_{yfr}) - l_r(F_{yrl} + F_{yrr}) + \frac{b_f}{2}(F_{xfl} - F_{xfr}) + \frac{b_r}{2}(F_{xrl} - F_{xrr}) \tag{10c}$$

where F_{xij} and F_{yij} are longitudinal tire force and lateral tire force of individual wheel in the body-fixed coordinate system, $ij = fl, fr, rl, rr$ represent the front left, front right, rear left and rear right wheels respectively, l_f and l_r are the front and rear wheel base lengths, while b_f and b_r are the front and rear track widths.

The longitudinal and lateral tire force of individual wheel can be determined by the following equations:

$$F_{xij} = F_{tij}\cos\delta_{ij} - F_{sij}\sin\delta_{ij} \tag{11a}$$

$$F_{yij} = F_{tij}\sin\delta_{ij} + F_{sij}\cos\delta_{ij} \tag{11b}$$

where F_{tij} and F_{sij} are tire traction/brake force and side force of individual wheel in the wheel-centred coordinate system. For the front and rear wheel independent steering vehicle, $\delta_{fl} = \delta_{fr} = \delta_f$ and $\delta_{rl} = \delta_{rr} = \delta_r$. The individual lateral tire force can be calculated by the following equations:

$$F_{sfl} = C_{\alpha f}\alpha_{fl} = C_{\alpha f}\left[\delta_f - \tan^{-1}\left(\frac{l_f r + v_y}{v_x - 0.5b_f r}\right)\right] \tag{12a}$$

$$F_{sfr} = C_{\alpha f}\alpha_{fr} = C_{\alpha f}\left[\delta_f - \tan^{-1}\left(\frac{l_f r + v_y}{v_x + 0.5b_f r}\right)\right] \tag{12b}$$

$$F_{srl} = C_{\alpha r}\alpha_{rl} = C_{\alpha r}\left[\delta_r + \tan^{-1}\left(\frac{l_r r - v_y}{v_x - 0.5b_r r}\right)\right] \tag{12c}$$

$$F_{srr} = C_{\alpha r}\alpha_{rr} = C_{\alpha r}\left[\delta_r + \tan^{-1}\left(\frac{l_r r - v_y}{v_x + 0.5b_r r}\right)\right] \tag{12d}$$

where $C_{\alpha f}$ and $C_{\alpha r}$ are cornering stiffness of front tire and rear tire, respectively. The individual longitudinal tire force can be determined by the wheel dynamics equation, assuming the wheel acceleration is zero:

$$F_{tij} = \frac{T_{ij}}{R_\omega} \tag{13}$$

where R_ω is the wheel radius and T_{ij} is the traction or brake torque of each wheel.

3.2. Low-level CA design

The proposed low-level CA includes two layers: in the first layer, the individual longitudinal force and steering angle are optimally distributed according to desired virtual control input; in the second layer, the distributed individual longitudinal tire forces are mapped to individual motor torque and hydraulic brake torque. When the vehicle is braking, in terms of the energy saving, we would like to try to use the negative motor torque as much as possible. After the negative motor torque reaches the maximum value, the friction brake is used to compensate the remaining required negative brake torque. This mapping strategy in the second layer can be easily achieved by a simple rule-based method. We designed this hierarchical structure to separate the first layer QCQP optimisation and the second layer rule-based method to improve the computational efficiency.

1) The first layer of CA method

According to the total virtual control force and based on the vehicle dynamics model (10)-(13), the CA equation can be presented as followings:

$$\sum_{i=1,2,3} u_i = F_{xd} \tag{14a}$$

$$C_{\alpha f}(2u_4 - \alpha_{fld} - \alpha_{frd}) + C_{\alpha r}(2u_5 + \alpha_{rld} + \alpha_{rrd}) = F_{yd} \tag{14b}$$

$$-\frac{1}{2}b_r u_2 + \frac{1}{2}b_r u_3 + l_f C_{\alpha f}(2u_4 - \alpha_{fld} - \alpha_{frd}) - l_r C_{\alpha r}(2u_5 + \alpha_{rld} + \alpha_{rrd}) = M_{zd} \tag{14c}$$

where the allocated control actuator input $u_1 = F_{tf}$, $u_2 = F_{trl}$, $u_3 = F_{trr}$, $u_4 = \delta_f$ and $u_5 = \delta_r$. It can be noticed from Equation (13) that the steering angle in tire force mapping Equation (10) is considered as very small value, so $F_{xi} \approx F_{ti}$ and $F_{yi} \approx F_{si}$. $\alpha_{fld}, \alpha_{frd}, \alpha_{rld}, \alpha_{rrd}$ can be determined from (11):

$$\alpha_{id} = f(\delta_f, \delta_r, v_{xd}, v_{yd}, r_d) \tag{15}$$

where v_{yd} is the desired lateral velocity and is assumed zero to minimise the undesired side-slip angle and v_{xd} and r_d are the desired longitudinal and yaw rate, as determined in the high-level MPC.

In addition, the control actuation input u_{1-5} should also satisfy the actuation constraints. First, the individual control actuator has its physical constraint, which is mathematically presented as follows:

$$u_{1-3} \le \frac{T_{dmax}}{R_\omega} \tag{16a}$$

$$\theta_{min} \le u_4 \le \theta_{max} \tag{16b}$$

$$\theta_{min} \le u_5 \le \theta_{max} \tag{16c}$$

where T_{dmax} represents the maximum driving toque of the individual motor, and θ_{min}, θ_{max} are the minimum and maximum steering angle respectively. Then the friction circle constraint should also be considered to show the maximum available friction force of each tire:

$$0.25u_1^2 + C_{\alpha f}^2(u_4 - \alpha_{fld})^2 \le (\mu F_{zfl})^2 \tag{17a}$$

$$0.25u_1^2 + C_{\alpha f}^2(u_4 - \alpha_{frd})^2 \le (\mu F_{zfr})^2 \tag{17b}$$

$$u_2^2 + C_{\alpha r}^2(u_5 + \alpha_{rld})^2 \le (\mu F_{zrl})^2 \tag{17c}$$

$$u_3^2 + C_{\alpha r}^2(u_5 + \alpha_{rrd})^2 \le (\mu F_{zrr})^2 \tag{17d}$$

where F_{zfl}, F_{zfr}, F_{zrl}, F_{zrr} are the vertical loads on the wheels. Considering load transfer effects, the vertical load of each wheel can be estimated by the following equations [37]:

$$F_{zfl} = \frac{m}{l_f + l_r}\left(\frac{1}{2}gl_r - \frac{1}{2}(\dot{v}_x - v_y r)h - \frac{l_r}{b_f}(\dot{v}_y + v_x r)h\right) \tag{18a}$$

$$F_{zfr} = \frac{m}{l_f + l_r}\left(\frac{1}{2}gl_r - \frac{1}{2}(\dot{v}_x - v_y r)h + \frac{l_r}{b_f}(\dot{v}_y + v_x r)h\right) \tag{18b}$$

$$F_{zrl} = \frac{m}{l_f + l_r}\left(\frac{1}{2}gl_f + \frac{1}{2}(\dot{v}_x - v_y r)h - \frac{l_f}{b_r}(\dot{v}_y + v_x r)h\right) \tag{18c}$$

$$F_{zrr} = \frac{m}{l_f + l_r}\left(\frac{1}{2}gl_f + \frac{1}{2}(\dot{v}_x - v_y r)h + \frac{l_f}{b_r}(\dot{v}_y + v_x r)h\right) \tag{18d}$$

where h is the distance of the vehicle Centre of Gravity (CG) from the ground. It is noted that the accelerations $\dot{v}_x$ and $\dot{v}_y$ and yaw rate r can be obtained from an Inertial Measurement Unit (IMU), and the measurement signals of v_x and v_y are available through the Global Positioning System (GPS) unit.

In order to solve the CA problem (14-18), the optimisation problem can be designed to minimise the virtual input error and the cost of using the control actuators:

$$\min_{u_i \text{ (for } i=1,2,3,4,5)} J = (\tau_{RHS} - Bu(k+j))^T Q_1(\tau_{RHS} - Bu(k+j)) + u(k+j)^T Q_2 u(k+j) \tag{19a}$$

The cost function (19a) can be re-written in the following QCQP form:

$$\min_{u_i \text{ (for } i=1,2,3,4,5)} J \approx u(k+j)^T(B^T Q_1 B + Q_2)u(k+j) + (-2BQ_1\tau_{RHS})u(k+j) \tag{19b}$$

s.t.

$$\begin{bmatrix} u_1 \\ u_4 \end{bmatrix}^T \begin{bmatrix} 0.25 & 0 \\ 0 & C_{\alpha f}^2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_4 \end{bmatrix} + \begin{bmatrix} 0 & -2\alpha_{fld}C_{\alpha f}^2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_4 \end{bmatrix} \le (\mu F_{zfl})^2 - (C_{\alpha f}\alpha_{fld})^2 \tag{19c}$$

$$\begin{bmatrix} u_1 \\ u_4 \end{bmatrix}^T \begin{bmatrix} 0.25 & 0 \\ 0 & C_{\alpha f}^2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_4 \end{bmatrix} + \begin{bmatrix} 0 & -2\alpha_{frd}C_{\alpha f}^2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_4 \end{bmatrix} \le (\mu F_{zfr})^2 - (C_{\alpha f}\alpha_{frd})^2 \tag{19d}$$

$$\begin{bmatrix} u_2 \\ u_5 \end{bmatrix}^T \begin{bmatrix} 1 & 0 \\ 0 & C_{\alpha r}^2 \end{bmatrix} \begin{bmatrix} u_2 \\ u_5 \end{bmatrix} + \begin{bmatrix} 0 & 2\alpha_{rld}C_{\alpha r}^2 \end{bmatrix} \begin{bmatrix} u_2 \\ u_5 \end{bmatrix} \le (\mu F_{zrl})^2 - (C_{\alpha r}\alpha_{rld})^2 \tag{19e}$$

$$\begin{bmatrix} u_3 \\ u_5 \end{bmatrix}^T \begin{bmatrix} 1 & 0 \\ 0 & C_{\alpha r}^2 \end{bmatrix} \begin{bmatrix} u_3 \\ u_5 \end{bmatrix} + \begin{bmatrix} 0 & 2\alpha_{rrd}C_{\alpha r}^2 \end{bmatrix} \begin{bmatrix} u_3 \\ u_5 \end{bmatrix} \le (\mu F_{zrr})^2 - (C_{\alpha r}\alpha_{rrd})^2 \tag{19f}$$

where $u^T = \begin{bmatrix} u_1 & u_2 & u_3 & u_4 & u_5 \end{bmatrix}, B = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 2C_{\alpha f} & 2C_{\alpha r} \\ 0 & -0.5b_r & 0.5b_r & 2l_f C_{\alpha f} & 2l_r C_{\alpha r} \end{bmatrix}$

$$\tau_{RHS} = \begin{bmatrix} F_x \\ F_y + C_{\alpha f}(\alpha_{fld} + \alpha_{frd}) - C_{\alpha r}(\alpha_{rld} + \alpha_{rrd}) \\ M_z + l_f C_{\alpha f}(\alpha_{fld} + \alpha_{frd}) + l_r C_{\alpha r}(\alpha_{rld} + \alpha_{rrd}) \end{bmatrix},$$

Note also that the boundary constraint (16) should be satisfied. Q_1 and Q_2 are the diagonal weighting matrices on the virtual control error and control effort respectively. $Q_1 = \text{diag}[q_{11}; q_{12}; q_{13}]$, $q_{11} > 0, q_{12} > 0, q_{13} > 0$, where q_{11} is the weighting factor for achieving the desired total longitudinal tire force, q_{12} is the weighting factor for achieving the desired total lateral tire force, q_{13} is the weighting factor for achieving the desired yaw moment; $Q_2 = \text{diag}[q_{21}; q_{22}; q_{23}; q_{24}; q_{25}]$, $q_{21} > 0$, $q_{22} > 0$, $q_{23} > 0$, $q_{24} > 0$, $q_{25} > 0$, where q_{21} is the weighting factor for penalising the allocated total front tire longitudinal force, q_{22} is the weighting factor for penalising the allocated rear left tire longitudinal force, q_{23} is the weighting factor for penalising the allocated rear right tire longitudinal force, q_{24} is the weighting factor for penalising the allocate front wheel steering angle, q_{25} is the weighting factor for penalising the allocate rear wheel steering angle. In order to achieve the good control performance, the weightings of q_{11}, q_{12}, q_{13} are set as much bigger than the weightings of $q_{21}, q_{22}, q_{23}, q_{24}, q_{25}$.

In order to improve the computational efficiency and satisfy the needs for real-time CA optimisation, the above QCQP problem can be solved by the time-efficient solver based on Primal Dual Interior Point (PDIP) method in Forces Pro [38] in appendix A.

2) The second layer of CA method

In the second layer of CA, the allocated individual longitudinal force from first layer is mapped into the individual motor torque or hydraulic brake torque. The mapping strategies of front axle and rear axle are presented in Figure 3(a,b), respectively. The main idea of the mapping strategy is trying to use the electric motor as much as possible (as electric motor can generate both traction torque and regenerated brake torque). If the electric motor cannot generate enough brake torque, the hydraulic brake will compensate the brake torque. In Figure 3(a), T_{mf} and T_{bf} represent the electric motor torque and hydraulic brake torque on the front axle respectively. T_{mf}^{min} and T_{mf}^{max} are the motor torque limit. In Figure 3(b), T_{mrl} and T_{mrr} are electric motors of rear left and rear right wheel and T_{br} is the hydraulic brake torque on the rear axle. T_{mrl}^{min} ,T_{mrl}^{max}, T_{mrr}^{min} and T_{mrr}^{max} are the motor torque limits. It is noted that constraints (16a) in first layer of CA has guaranteed $u_1 \leq \frac{T_{mf}^{max}}{R_\omega}$, $u_2 \leq \frac{T_{mrl}^{max}}{R_\omega}$ and $u_3 \leq \frac{T_{mrr}^{max}}{R_\omega}$, which is the precondition in Figure 3.

4. Simulation results

In this section, the proposed hierarchical path-tracking controller is validated using a high-fidelity vehicle dynamics model in CarMaker. The vehicle model is built based on an autonomous test platform developed by Delta Motorsport Limited. Specifically, the

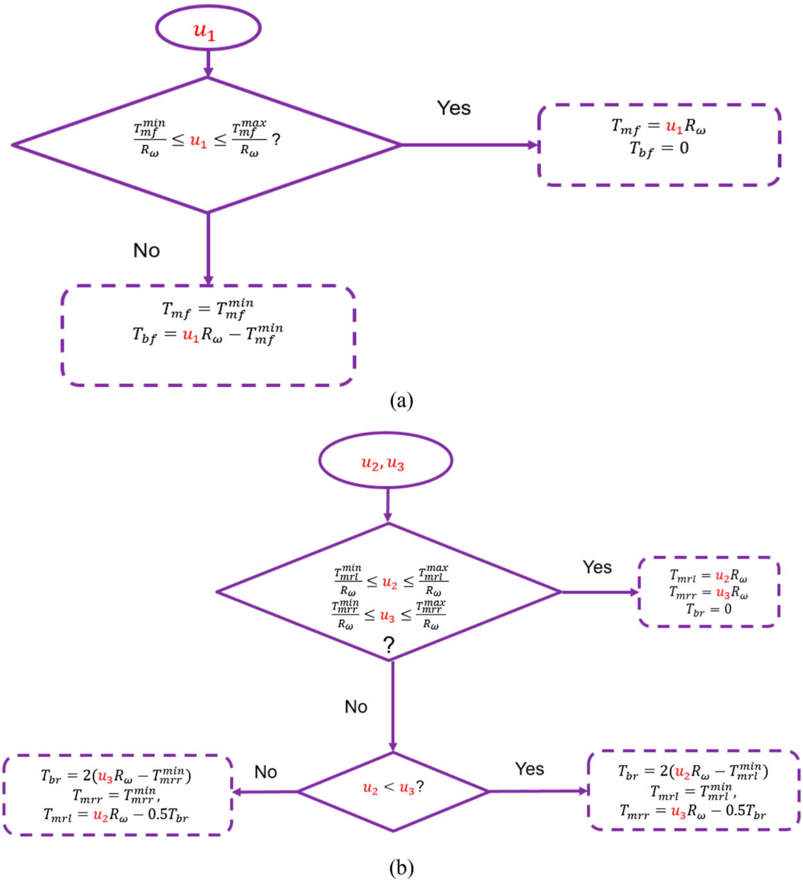
u_1
$\frac{T_{mf}^{min}}{R_\omega} \le u_1 \le \frac{T_{mf}^{max}}{R_\omega}$?
Yes
$T_{mf} = u_1 R_\omega$
$T_{bf} = 0$
No
$T_{mf} = T_{mf}^{min}$
$T_{bf} = u_1 R_\omega - T_{mf}^{min}$
(a)
u_2, u_3
$\frac{T_{mrl}^{min}}{R_\omega} \le u_2 \le \frac{T_{mrl}^{max}}{R_\omega}$
$\frac{T_{mrr}^{min}}{R_\omega} \le u_3 \le \frac{T_{mrr}^{max}}{R_\omega}$
?
Yes
$T_{mrl} = u_2 R_\omega$
$T_{mrr} = u_3 R_\omega$
$T_{br} = 0$
No
$u_2 < u_3$?
No
$T_{br} = 2(u_3 R_\omega - T_{mrr}^{min})$
$T_{mrr} = T_{mrr}^{min}$,
$T_{mrl} = u_2 R_\omega - 0.5 T_{br}$
Yes
$T_{br} = 2(u_2 R_\omega - T_{mrl}^{min})$
$T_{mrl} = T_{mrl}^{min}$,
$T_{mrr} = u_3 R_\omega - 0.5 T_{br}$
(b)

Figure 3. The allocation of motor torque and hydraulic brake toque (a) front axle (b) rear axle.

software simulation environment to test the optimisation control strategy is the jointed simulation of Matlab Simulink, Forces Pro optimisation solver and IPG CarMaker. The vehicle plant model and simulation environment (road and surrounding environment) are implemented in IPG CarMaker, and the high-level MPC controller and low-level control allocation method are implemented based on Matlab Simulink and Forces Pro optimisation solver (a Simulink optimisation code package). The IPG CarMaker based vehicle plant model and Simulink based optimisation strategy are integrated together through the CarMaker-Simulink co-simulation. The computer hardware to run the simulation is a IBM Thinkpad laptop (Intel(R) Core(TM) i7-5500U CPU @ 2.40 GHz and 8 GB RAM).

The simulation scenario chosen on the Silverstone racetrack, and the autonomous vehicle is trying to track the road center-line. The velocity profile with minimum-time solution along the whole racetrack is pre-calculated offline. Both the desired path and velocity profile are inputted into the hierarchical controller, and the controller distributes the steering and driving control inputs into the CarMaker vehicle model. The CarMaker model also provides the necessary information on the state values which can be then fed back to the hierarchical controller. The main vehicle parameters are presented in Table 2.

Table 2. Main parameters of Carmaker model.

Symbol	Meaning	Values
m	Vehicle total mass	700.28 kg
I_z	Moment inertial around yaw axle	1597.717 kg.m^2
l_f	Front wheel base	0.999 m
l_r	Rear wheel base	0.996 m
b_f	Front track width	1.52 m
b_r	Rear track width	1.52 m
R_ω	Wheel radius	0.32 m
C_α	Cornering stiffness for front and rear tyres	29220 N/rad
μ	Tire-road friction coefficient	1
k_{1-5}	Control gains of high-level feedback controller	[3 20 5 20 200]

A particular s-curve section of Silverstone track is selected to present the control performance of proposed controller. In order to find out the best sampling time and prediction horizon of the proposed high-level path-tracking MPC, the sensitivity analysis on different sampling time and prediction horizons (sampling time = 0.05, 0.075 s; prediction time = 2s, 4s, 6s) is carried out by comparing the path-tracking performance and velocity-tracking performance. When sampling time and prediction time are 0.05 and 2 s. respective, the best control performance can be achieved. Thus, the simulation results below are presented by using sampling and prediction time of 0.05 and 2 s. The control performance of the proposed CA ('MPC + CA') in the low-level is compared with two other, simpler vehicle configurations: the CA method without rear-wheel steering is simply presented as 'MPC + CA without RS' and the CA method without torque vectoring is presented as 'MPC + CA without TV'. It is noted that for the 'MPC + CA without RS', the weighting value related to rear-wheel steering angle in Q_2 is adjusted to disable the rear-wheel steering in (19). For the 'MPC + CA without TV', the B matrix is adjusted to assume the motor torque is equally distributed among two rear wheels in (19). In order to show the advantage of proposed high-level MPC, the simulation results of a high-level feedback path-tracking controller in (8) and proposed low-level CA method are also presented as 'FB + CA' in following figures. Figure 4(a) shows the desired path. Figure 4(b) presents the path-tracking performance and Figure 4(c) shows velocity tracking performance of different methods. According to Figure 4(b), 'MPC + CA' has the smallest lateral tracking error compared with 'MPC + CA without TV', 'MPC + CA without RS' and 'FB + CA'. Figure 4(c) demonstrates that 'MPC + CA' and 'FB + CA' can fast track the desired longitudinal velocity profile though the online optimisation and achieve the maximum cornering speed, while other two methods both show a certain level of delay. Figure 5 shows the total normalised acceleration, which is calculated by the following equation:

$$\bar{a} = \frac{\sqrt{a_x^2 + a_y^2}}{\mu g} \tag{20}$$

where a_x and a_y are actual longitudinal and lateral acceleration, respectively. It can be observed from Figure 5 that the 'MPC + CA' and the 'MPC + CA without TV' have larger normalised acceleration than 'MPC + CA without RS', which means that CA and CA without TV can take greater usage of tire force to achieve the limit-handling performance. The normalised acceleration of 'FB + CA' shows big spike, which causes the sudden increase

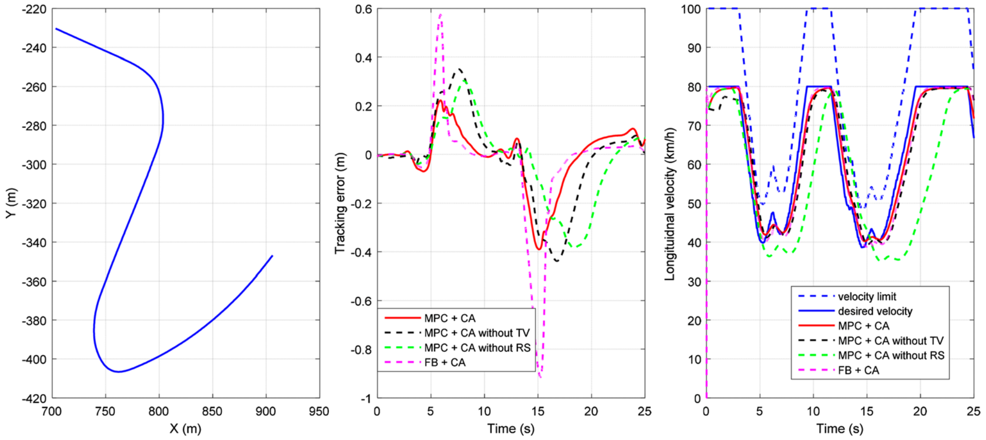

Figure 4. The tracking performance in the simulation (left) desired path (middle) lateral path-tracking error (right) velocity-tracking performance.

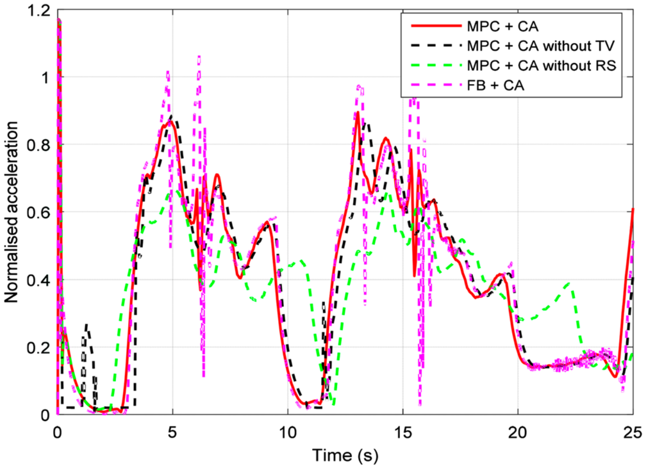

Figure 5. The normalised acceleration in the simulation.

of tracking error in Figure 4(b) and suggests the less stability of high-level feedback controller compared with MPC. It should be noted that the normalised acceleration limit is larger than 1 in this study because a more 'potent' tire model is assumed.

In Figure 6, the simulation results of robustness analysis on proposed 'MPC + CA' controller are presented. First, it is assumed that the friction coefficient can not be estimated accurately. Figure 6 compares the path-tracking error and longitudinal velocity tracking error when the estimated friction coefficient is 1 and 0.7, respectively (the actual value is 1). When the friction coefficient is wrongly estimated as 0.7, the path-tracking error is relatively larger. This is because the actual total acceleration is larger than the wrongly estimated friction limit in low-level CA and the control allocation performance is compromised. Nevertheless, the path-tracking error is still acceptable as it is smaller than 0.5 m. Secondly, the measurement white noise of the longitudinal acceleration (variance is 0.03 m/s^2), lateral acceleration (variance is 0.06 m/s^2) and yaw rate (variance is

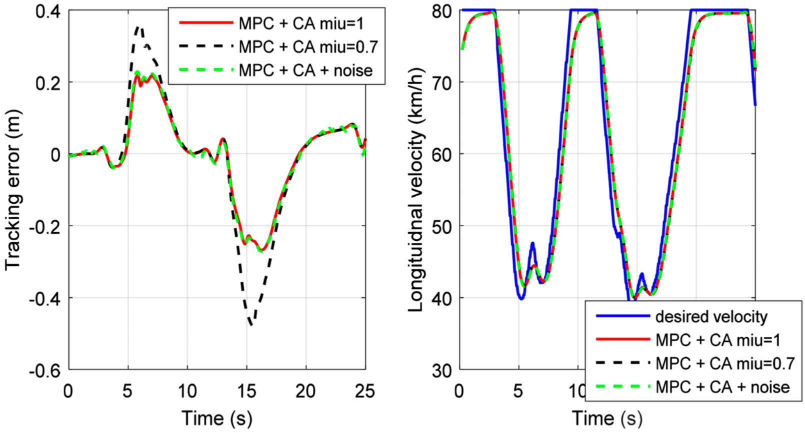

Figure 6. The robustness analysis of the proposed control strategy

0.0003 rad/s) are considered in the simulation and the control performance is similar as the results when the measurement noise is not considered. In general, Figure 6 suggests that the proposed ‘MPC + CA’ controller has good robustness performance.

Figure 7 shows the tire forces and yaw moment performance for different controllers. It can be seen in Figure 7 that the desired longitudinal force and yaw moment are tracked quite well. The desired lateral force is much smaller than actual values. This is because the desired lateral tire force is calculated based on the lateral position tracking error, the small desired lateral force suggests that the vehicle has good lateral tracking performance. Figure 7(a) suggests that the total longitudinal tire forces of ‘MPC + CA without RS’ are much smaller than ‘MPC + CA’ and ‘MPC + CA without TV’, which can be the reason why ‘MPC + CA without RS’ have worse longitudinal velocity tracking performance compared to the other two strategies. According to Figure 7(b), the ‘MPC + CA without RS’ shows smaller total lateral tire force since the vehicle can only achieve front-wheel steering and the total available lateral tire force that can be generated is reduced. Figure 7 (d) shows that ‘MPC + CA without TV’ cannot generate additional torque-vectoring yaw moment. Figures 7(a,b,d) demonstrate that the total forces and the additional torque-vectoring yaw moment in the case of the ‘MPC + CA’ are larger than the other methods and consequently the overall path-tracking and velocity-tracking performance is the best amongst the controllers. The CA without TV disables the torque vectoring function and cannot generate additional controlled yaw moment when the vehicle is making the sharp turning as shown in Figure 7(d), so the tracking performance is compromised as shown in Figure 4. The CA without RS disables the rear-wheel steering function and limits the available total lateral tire force so the tracking performance is also compromised (as shown also in Figure 4). Finally, Figure 7(a-d) suggest that tire forces and yaw moment of ‘FB + CA’ method shows big spike during the sharp turning, which cause the compromised path-tracking performance.

Figure 8 compares the torque outputs of the three motors. The torques from two rear hub motors of ‘MPC + CA’ and ‘FB + CA’ in Figures 8(a,d) and ‘MPC + CA without RS’ in Figure 8(c) show obvious torque vectoring performance. Figure 9 suggests that the steering angles from the ‘MPC + CA’ and ‘MPC + CA without TV’ are larger than ‘MPC + CA

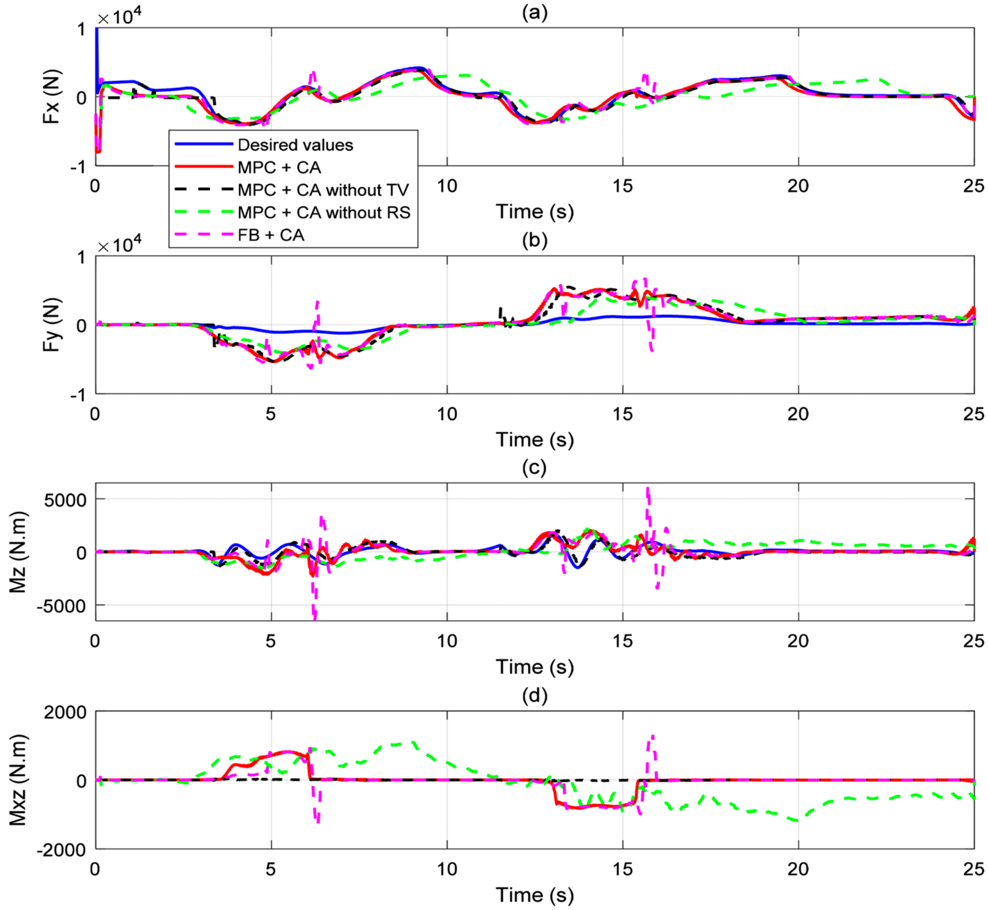

Figure 7. The actual tire force and yaw moment in the simulation (a) total longitudinal force (b) total lateral force (c) total yaw moment (d) torque vectoring yaw moment.

without RS', which is consistent with the results for the total lateral force in Figure 7(b). In Figure 9(c), the rear wheel steering angle of 'MPC + CA without RS' is zero. Figure 9(d) shows the steering angles of 'FB + CA' is much bigger than other methods, which impairs the stability of the vehicle.

In order to validate the real-time performance of the proposed hierarchical controller, the solving time of high-level path-tracking MPC and low-level CA method are presented in Figure 10. Since the sampling time of the proposed controller is 0.05s, the solving time in Figure 10 is smaller than 0.05 s and demonstrates that the real-time computational requirement can be satisfied. It is noted that the high-level MPC solver takes longer time to find a solution around 2, 5.5, 17, 19 and 21.5 s. The reason behind this is more iterations are required for the solver to find the optimal solution, or even the optimal solution cannot be found within the max iteration and only the sub-optimal solution is found. It is also noted that unlike the high-level MPC controller which required to solve the MPC optimisation problem in prediction steps, the low-level CA controller only need to solve the standard

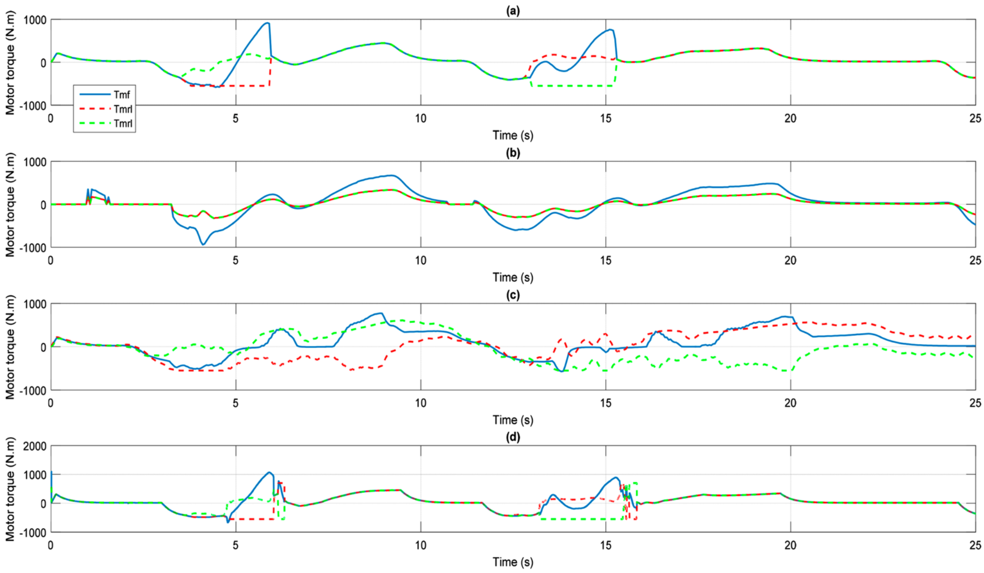

Figure 8. The motor torque in the simulation (a) MPC + CA (b) MPC + CA without TV (c) MPC + CA without RS (d) FB + CA.

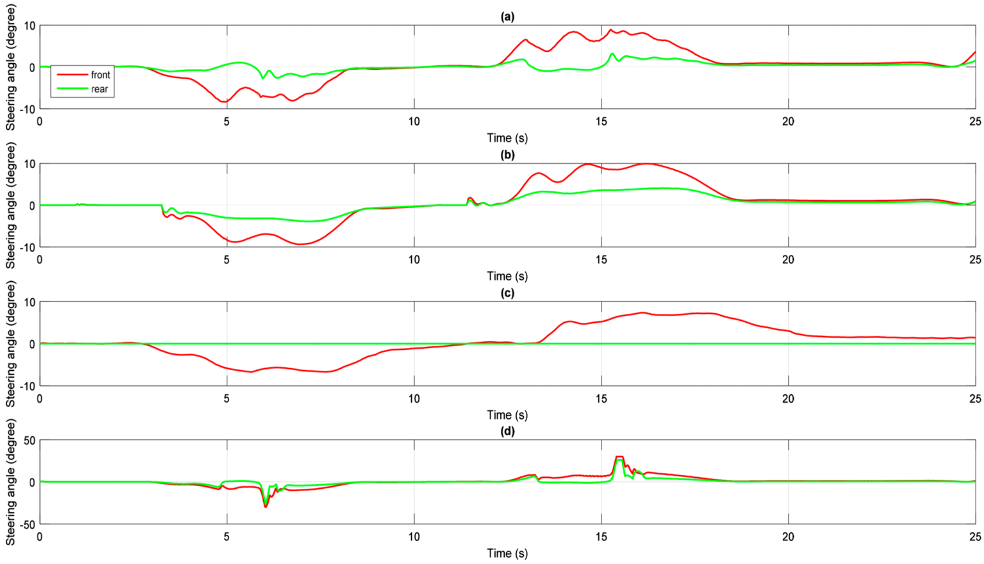

Figure 9. The wheel steering angle in the simulation (a) MPC + CA (b) MPC + CA without TV (c) MPC + CA without RS (d) FB + MPC.

QCQP optimisation problem without the prediction model. So the optimisation solving time of low-level controller is much smaller than the high-level controller.

In the literature, the execution time of MPC path-tracking controller [39] based on vehicle lateral dynamics is 100 ms with the prediction steps of 20. Wang et al. suggests a classic Linear-Time-Varying (LPV) MPC based on a 3 DOF nonlinear vehicle dynamics model

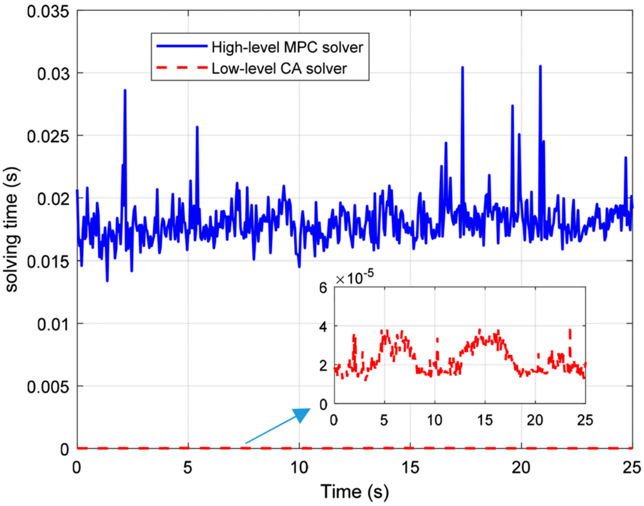

Figure 10. The solving time of CA in the simulation.

including the Brush tyre model and the analysis on the computational efficiency is carried out [40]. They set the sampling time as 50 ms (The execution time is close to 50 ms) with the time steps of 14. For our designed two-layer MPC, the execution time is around 20 ms with time steps of 40, which shows better computational efficiency compared with above two holistic MPC approaches.

5. Conclusion

In this study, a hierarchical two-level integrated path-tracking and CA controller is designed. The high-level path-tracking controller is composed by on a nonlinear rigid body dynamics model and low-level CA is designed using a QCQP formulation. The major findings can be summarised as follows:

(1) By selecting the best sampling time of 0.05 s and prediction time of 2 s, and reduce the lateral tracking error within 0.5 m with maximum cornering speed during limit handling scenario in the suggested simulation platform, and shows better performance than a benchmark feedback path-tracking controller.
(2) The proposed low-level CA method fully utilizes torque vectoring of two rear motors and front and rear-wheel independent steering, and the path-tracking and velocity-tracking performance are improved when compared with a controller without torque vectoring or a controller without rear-wheel steering.
(3) The total solving time of high-level MPC and low-level CA is smaller than the sampling time for the whole strategy, which shows that the proposed method can be implemented in real-time.

In future work, the real-time performance will be tested on a real-time target machine (such as dSpace or Speedgoat) before deploying the final solution on the autonomous test platform.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This work was supported by Innovate UK through AID-CAV project.

References

[1] Markkula G, Benderius O, Wolff K, et al. A review of near collision driver behavior models. Hum Factors. 2012;54(6):1117–1143. doi:10.1177/0018720812448474

[2] Edelmann J, Pl¨ochl M, Reinalter W, et al. A passenger car driver model for higher lateral accelerations. Veh Syst Dyn. 2007;45(12):1117–1129. doi:10.1080/00423110701203644

[3] Chatzikomis C, Spentzas KN. A path-following driver model with longitudinal and lateral control of vehicle's motion. Eng Res. 2009;73(4):257–266. doi:10.1007/s10010-009-0112-5

[4] Plochl M, Edelmann J. Driver models in automobile dynamics application. Veh Syst Dyn. 2007;45(7–8):699–741. doi:10.1080/00423110701432482

[5] Sharp RS, Casanova D, Symonds P. A mathematical model for driver steering control, with design, tuning and performance results. Veh Syst Dyn. 2000;33:289–326. doi:10.1076/0042-3114(200005)33:5;1-Q;FT289

[6] Attia R, Orjuela R, Basset M. Combined longitudinal and lateral control for automated vehicle guidance. Veh Syst Dyn. 2014;52(2):261–279. doi:10.1080/00423114.2013.874563

[7] Roselli F, Corno M, Savaresi SM, et al. H-infinity control with look-ahead for lane keeping in autonomous vehicles. Proc. IEEE Conf. Control Technol. Appl, 2017, pp. 2220–2225.

[8] Falcone P, Borrelli F, Asgari J, et al. Predictive active steering control for autonomous vehicle systems. IEEE Trans Control Syst Technol. May 2007;15(3):566–580. doi:10.1109/TCST.2007.894653

[9] Tagne G, Talj R, Charara A. Higher-order sliding mode control for lateral dynamics of autonomous vehicles, with experimental validation. Proc. IEEE Intell. Veh. Symp, 2013, pp. 678–683.

[10] Gao Y, Cao DP, Shen YH. Path-following control by dynamic virtual terrain field for articulated steer vehicles. Veh Syst Dyn. 2019;58:1–25. doi:10.1080/00423114.2019.1648837

[11] Guo HY, Liu J, Cao DP, et al. Dual-envelop-oriented moving horizon path tracking control for fully automated vehicles. Mechatronics. 2018;50:422–433. doi:10.1016/j.mechatronics.2017.02.001

[12] De Castro R, Tanelli M, Araújo RE, et al. Minimum-time path-following for highly redundant electric vehicles. IEEE Trans Control Syst Technol. 2015;24(2):487–501. doi:10.1109/TCST.2015.2458773

[13] Chatzikomis C, Sorniotti A, Gruber P, et al. Comparison of path tracking and torque-vectoring controllers for autonomous electric vehicles. IEEE Trans Intell Veh. 2018;3(4):559–570. doi:10.1109/TIV.2018.2874529

[14] Fujimoto H, Maeda K. Optimal yaw-rate control for electric vehicles with active front-rear steering and four-wheel driving-braking force distribution. Proc. 39th Annu. Conf. IEEE Ind. Electron. Soc., 2013, pp. 6514–6519.

[15] Tjønnås J, Johansen TA. Stabilization of automotive vehicle using active steering and adaptive brake control allocation. IEEE Trans Control Syst Technol. 2010;18(3):545–558. doi:10.1109/TCST.2009.2023981

[16] Wang J, Longoria RG. Coordinated and reconfigurable vehicle dynamics control. IEEE Trans Control Syst Technol. 2009;17(3):723–732. doi:10.1109/TCST.2008.2002264

[17] Chen Y, Wang J. Adaptive energy-efficient control allocation for planar motion control of over-actuated electric ground vehicles. IEEE Trans Control Syst Technol. 2014;22(4):1362–1373. doi:10.1109/TCST.2013.2287560

[18] Li B, Du H, Li W, et al. Integrated dynamics control and energy efficient optimization for overactuated electric vehicles. Asian J Control. 2018;20(5):1952–1966. doi:10.1002/asjc.1686

[19] Guo J, Luo Y, Li K, et al. Coordinated path-following and direct yaw-moment control of autonomous electric vehicles with sideslip angle estimation. Mech Syst Signal Process. 2018;105:183–199. doi:10.1016/j.ymssp.2017.12.018

[20] Wang R, Hu C, Yan F, et al. Composite nonlinear feedback control for path following of four-wheel independently actuated autonomous ground vehicles. IEEE Trans Intell Transp Syst. 2016;17(7):2063–2074. doi:10.1109/TITS.2015.2498172

[21] Li B, Siampis E, Lin C, et al. A time-efficient integrated path-tracking and control allocation method for autonomous electric vehicle. 2019 IEEE 58th Conference on Decision and Control (CDC), pp.6700-6705, 2019.

[22] Ji J, Khajepour A, Melek WW, et al. Path planning and tracking for vehicle collision avoidance based on model predictive control with multiconstraints. IEEE Trans Veh Technol. 2017;66(2):952–964. doi:10.1109/TVT.2016.2555853

[23] Raffo GV, Gomes GK, Normey-Rico JE, et al. A predictive controller for autonomous vehicle path tracking. IEEE Trans Intell Transport Syst. 2009;10(1):92–102. doi:10.1109/TITS.2008.2011697

[24] Kim W, Kim D, Yi K, et al. Development of a path-tracking control system based on model predictive control using infrastructure sensors. Veh Syst Dyn. 2012;50(6):1001–1023. doi:10.1080/00423114.2011.597864

[25] Kim E, Kim J, Sunwoo M. Model predictive control strategy for smooth path tracking of autonomous vehicles with steering actuator dynamics. Int J Automot Technol. Dec, 2014;15(7):1155–1164. doi:10.1007/s12239-014-0120-9

[26] Peng H, Wang W, An Q, et al. Path tracking and direct yaw moment coordinated control based on robust MPC with the finite time horizon for autonomous independent-drive vehicles. IEEE Trans Veh Technol. 2020;69(6):6053–6066. doi:10.1109/TVT.2020.2981619

[27] Zhang B, Zong C, Chen G, et al. Electrical vehicle path tracking based model predictive control with a Laguerre function and exponential weight. IEEE Access. 2019;7:17082–17097. doi:10.1109/ACCESS.2019.2892746

[28] Alcalá E, Puig V, Quevedo J, et al. Autonomous racing using linear parameter varying-model predictive control (lpv-mpc). Control Eng Pract. 2020;95:104270. doi:10.1016/j.conengprac.2019.104270

[29] Liniger A, Domahidi A, Morari M. Optimization-based autonomous racing of 1:43 scale RC cars. Optim Control Appl Meth. 2015;36:628–647. doi:10.1002/oca.2123

[30] Kabzan J, Hewing L, Liniger A, et al. Learning-Based model predictive control for autonomous racing. IEEE Robot Automat Lett. Oct. 2019;4(4):3363–3370. doi:10.1109/LRA.2019.2926677

[31] Domahidi A, Jerez J. FORCES Professional. embotech GmbH, Zürich, Switzerland, Tech. Rep., Jul. 2014. [Online]. Available: http:// embotech.com/FORCES-Pro.

[32] Velenis E, Tsiotras P. Minimum-Time travel for a vehicle with acceleration limits: theoretical analysis and receding-horizon implementation. J Optim Theory Appl. 2008;138:275–296. doi:10.1007/s10957-008-9381-7

[33] Li B, Du H, Li W. Comparative study of vehicle tyre–road friction coefficient estimation with a novel cost-effective method. Veh Syst Dyn. 2014;52(8):1066–1098. doi:10.1080/00423114.2014.920090

[34] Zanelli A, Domahidi A, Jerez JL, et al. Forces NLP: an efficient implementation of interior-point methods for multistage nonlinear nonconvex programs. Int J Control. 2017;93:1–17. doi:10.1080/00207179.2017.1316017

[35] Li B, Ahmadi J, Lin C, et al. Integrated path-tracking and control allocation controller for autonomous electric vehicle under limited handling condition. 2020 IEEE Intelligent Vehicle Symposium (IV), pp.547-552, 2020.

[36] Boada B, Boada M, Díaz V. Fuzzy-logic applied to yaw moment control for vehicle stability. Veh Syst Dyn. 2005;43:753–770. doi:10.1080/00423110500128984

[37] Zhao Y, Zhang J. Yaw stability control of a four-independent-wheel drive electric vehicle. Int J Electric Hybrid Veh. 2009;2(1):64–76. doi:10.1504/IJEHV.2009.027677

[38] Domahidi A, Zgraggen A, Zeilinger MN, et al. Efficient interior point methods for multistage problems arising in receding horizon control. Conference on Decision and Control (CDC), Maui, HI, USA, December 2012, pp.668–674.

[39] Xu S, Peng H. Design, analysis, and experiments of preview path tracking control for autonomous vehicles. IEEE Trans Intell Transp Syst. 2019;21(1):48–58. doi:10.1109/TITS.2019.2892926

[40] Wang Z, Bai Y, Wang J, et al. Vehicle path-tracking linear-time-varying model predictive control controller parameter selection considering central process unit computational load. J Dyn Syst Meas Contr. 2019;141(5):051004. doi:10.1115/1.4042196

[41] Bock H, Plitt K. A multiple shooting algorithm for direct solution of optimal control problems. Proceedings of the 9th IFAC World Congress (pp. 242–247); 1984. Budapest: Pergamon Press.

Appendices

Appendix A

The QCQP optimisation problem (19b) can be rewritten as:

$$\min_{u_i \text{ (for } i=1,2,3,4,5)} \quad J = l(u_i(k+j)) \tag{A1a}$$

The inequality constraints (19c–19f) can be rewritten as:

$$g_j(u_i(k+j)) \leq 0 \tag{A1b}$$

In order to solve the MPC optimisation problem (A1) in a time-efficient manner, a primal-dual interior point method is applied. Particularly, the computation of search direction is achieved by solving the Karush-Kuhn-Tucker (KKT) system and the details are given as followings:

a) The formulation of KKT system

The KKT optimal condition for problem (A1) is given as:

$$h(u_i) + J(u_i)^T \lambda = 0 \tag{A2a}$$

$$g_{j=1,2,3,4}(u_i) + s = 0 \tag{A2b}$$

$$\wedge S = 0 \tag{A2c}$$

where $h(u_i) = \nabla l(u_i(k+1))$, $g_j(u_i) = g_j(u_i(k+1))$,$J(u_i) = \text{blkdiag}[\nabla g_j(u_i(k+1))]$. λ is the Lagrange multiplier. s is the slack variable and $S = \text{diag}(s)$, $\wedge = \text{diag}(\lambda)$.

b) Search direction

The Newton system is determined by linearising KKT system:

$$\begin{bmatrix} H(u_i,\lambda) & J(u_i)^T & 0 \\ J(u_i)^T & 0 & I \\ 0 & S & \wedge \end{bmatrix} \begin{bmatrix} \Delta u_i \\ \Delta\lambda \\ \Delta s \end{bmatrix} = - \begin{bmatrix} r_C \\ r_I \\ r_s \end{bmatrix} \tag{A3}$$

Where $H(u_i,\lambda) = \nabla h(u_i) + \text{blkdiag}[\nabla^2 g_j(u_i)\lambda_k]$.

c) Block elimination

Block elimination can be applied to simplify the search direction computation. It can be assumed that the elements of $\wedge$ are strictly positive, Δs can be eliminated from (A3) by using $\Delta s =$

$\wedge^{-1}(r_s - S\Delta\lambda)$. The Lagrange multiplier can be rearranged as following:

$$\Delta\lambda = S^{-1} \wedge (r_I + J(u_i)\Delta u_i) + S^{-1} r_s \tag{A4}$$

Thus, the search direction equations (B3) can be simplified as:

$$\Phi \Delta u_i = -r_d \tag{A5}$$

where $\Phi = H(u_i, \lambda) + J(u_i)^T S^{-1} \wedge J(u_i), r_d = r_C + J(u_i)^T S^{-1} \wedge r_I + J(u_i)^T S^{-1} r_s$. Thus, the search direction of (A5) can be solved in a time-efficient manner.

Appendix B

The nonlinear optimisation problem (2-8) can be rewritten as the followings:

$$\min_{u_i, x_j} J = \sum_{k=1}^{N} l_k(z_{i+j}(k)) \tag{B1a}$$

subject to:

$$c_k(z_{i+j}(k+1), z_{i+j}(k)) = 0 \tag{B1b}$$

$$h_k(z_{i+j}(k)) \leq 0 \tag{B1c}$$

The nonlinear MPC problem (B1) can be rewritten as the following linearised KKT version:

$$\begin{bmatrix} H(z_{i+j}, y_c) & J_{eq}(z_{i+j})^T & J_{ineq}(z_{i+j})^T & 0 \\ J_{eq}(z_{i+j}) & 0 & 0 & 0 \\ J_{ineq}(z_{i+j}) & 0 & 0 & I \\ 0 & 0 & S & Y_D \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta y_c \\ \Delta y_d \\ \Delta s \end{bmatrix} = - \begin{bmatrix} r_S \\ r_C \\ r_D \\ r_N \end{bmatrix} \tag{B2}$$

where $H(z_{i+j}, y_c) = \nabla^2_{z_{i+j}} l_k(z_{i+j}(k)) + \sum_{m=1}^{r-1} y_{c,k-1[m]} \nabla^2_{z_{i+j}} c_{k-1[m]}(z_{i+j}(k), z_{i+j}(k-1)) + \sum_{m=1}^{r} y_{c,k[m]} \nabla^2_{z_i} c_{k[m]}(z_{i+j}(k+1), z_{i+j}(k))$. y_c is the Lagrange multiplier of the equality constraints, and y_d is the Lagrange multiplier of the inequality constraints. s is the slack variable and $S = \text{diag}(s)$. J_{ineq} is the block diagonal consisting of $\nabla h_k(z_{i+j})$, while J_{eq} is the block diagonal consisting of $\nabla c_k(z_{i+j})$. Y_D is the diagonal matrix with the elements of y_d on its diagonal.

When dealing with the nonlinear MPC problem instead of convex QP problem, H is a nonlinear function of z_{i+j} and y_c, while the Jacobians of equality constraints J_{eq} and inequality constraints J_{ineq} are a function of z_{i+j}. It is important to mention that it is general difficult to ensure the KKT formulation (B2) has the desired inertial as QP problem, so additional care must be taken to solve (B2).

First, the linear system is reduced by eliminating Δs and Δy_d :

$$\begin{bmatrix} \Phi & J_{eq}(z_{i+j})^T \\ J_{eq}(z_{i+j}) & 0 \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta y_c \end{bmatrix} = - \begin{bmatrix} r_d \\ r_C \end{bmatrix} \tag{B3}$$

where $\Phi = H(z_{i+j}, y_c) + J_{ineq}(z_{i+j})^T S^{-1} Y_D J_{ineq}(z_{i+j}), r_d = r_S + J_{ineq}(z_{i+j})^T S^{-1} Y_D r_D - J_{ineq}(z_{i+j})^T S^{-1} r_N$.

Secondly, a more compact and symmetric system can be obtained as following equation:

$$Y \Delta y_c = \beta \tag{B4}$$

where $Y = J_{eq}(z_{i+j}) \Phi^{-1} J_{eq}(z_{i+j})^T, \beta = r_C - J_{eq}(z_{i+j}) \Phi^{-1} r_d$.

Forces NLP provides linear solver to solve the symmetric form (B3) and normal form (B4). For the QP formulation, Φ and Y can be assumed as positive definite and the problem can be easily solved by the Cholesky decomposition. However, for the nonlinear MPC formulation, the Hessian of Lagrangian is potentially indefinite and Y cannot be factorised with the Cholesky decomposition. A number of methods are applied to approximate the Hessian matrix and the positive definite can be guaranteed [38,41].