A Real-time Nonlinear Model Predictive Controller for Yaw Motion Optimization of Distributed Drive Electric Vehicles

Ningyuan Guo, *Student Member, IEEE*, Basilio Lenzo, *Member, IEEE*, Xudong Zhang, *Member, IEEE*, Yuan Zou, *Senior Member, IEEE*, Ruiqing Zhai, Tao Zhang, *Student Member, IEEE*

***Abstract*— This paper proposes a real-time nonlinear model predictive control (NMPC) strategy for direct yaw moment control (DYC) of distributed drive electric vehicles (DDEVs). The NMPC strategy is based on a control-oriented model built by integrating a single track vehicle model with the Magic Formula (MF) tire model. To mitigate the NMPC computational cost, the continuation/generalized minimal residual (C/GMRES) algorithm is employed and modified for real-time optimization. Since the traditional C/GMRES algorithm cannot directly solve the inequality constraint problem, the external penalty method is introduced to transform inequality constraints into an equivalently unconstrained optimization problem. Based on the Pontryagin's minimum principle (PMP), the existence and uniqueness for solution of the proposed C/GMRES algorithm are proven. Additionally, to achieve fast initialization in C/GMRES algorithm, the varying predictive duration is adopted so that the analytic expressions of optimally initial solutions in C/GMRES algorithm can be derived and gained. A Karush-Kuhn-Tucker (KKT) condition based control allocation method distributes the desired traction and yaw moment among four independent motors. Numerical simulations are carried out by combining CarSim and Matlab/Simulink to evaluate the effectiveness of the proposed strategy. Results demonstrate that the real-time NMPC strategy can achieve superior vehicle stability performance, guarantee the given safety constraints, and significantly reduce the computational efforts.**

***Index Terms*— continuation/generalized minimal residual algorithm; direct yaw moment control; distributed drive electric vehicle; nonlinear model predictive control.**

I. Introduction

DDEVs have recently attracted enormous attention and they have gradually become promising candidates for future transportation, owing to their advantages including high reliability, fast drive response, good flexibility [1]. By reasonably arranging the output torque of the in-wheel motors (IWM), an external yaw moment can be generated and applied for yaw motion control so as to improve the vehicle handling stability, especially in critical driving conditions [2]. This is commonly named as DYC. Nevertheless, the inherent nonlinearity and over-actuated feature in DYC of DDEVs also bring the great challenge of strategy design.

The literature presents several interesting contributions. In [3], the authors designed a linear quadratic regulator (LQR)-based DYC strategy, whose tire cornering stiffness information was estimated online. To improve the robustness of a classical LQR, Ref. [4] developed a robust LQR controller featuring a gain scheduling mechanism. With varying vehicle speed, this control strategy can adaptively change the state feedback gains so as to reduce the tracking error dynamics. To tackle uncertainties on vehicle parameters, a model-based feedforward and feedback controller in [5] was designed for correcting the errors between reference vehicle states and the real ones. In [6], the authors investigated a modified composite nonlinear feedback (CNF) strategy for path-following and DYC, including a nonlinear feedback contribution for accelerating the control response speed and eliminating overshoot. To further improve the robustness, integral sliding mode control (ISMC) was combined with CNF techniques as a combined control method in [7].

The aforementioned approaches come with two important limitations. First, the constraints on states and/or control actions are often not considered. In some cases they are ignored during the control design phase and then introduced through heuristics post-processing approaches (e.g. adding saturation limits), yet this may jeopardize the control optimality. Also, some robust control methods may be quite conservative, because their primary purpose is to effectively deal with model uncertainties or external disturbances. Under normal driving conditions, the vehicle lateral dynamics are nearly linear, thus the controller robustness may not be a significant benefit, whilst the controller performance could not be satisfactory [8].

Model predictive control (MPC) is a suitable approach to address the above gaps. At each sample time, the future system states are obtained by updating the control-oriented model and optimized through optimization algorithms in the predictive horizon, where the constraints can be easily put in explicit form [9, 10]. In addition, the high flexibility in formulating control

This work is supported by the National Natural Science Foundation of China (Grant No. 51805030), in part by the National Natural Science Foundation of China (Grant No. 51775039), and in part by Graduate Technological Innovation Project of Beijing Institute of Technology (Grant No. 2019CX20020).

Ningyuan Guo, Xudong Zhang, Yuan Zou, Ruiqing Zhai, and Tao Zhang are with the Beijing Collaborative and Innovative Center for Electric Vehicle and School of Mechanical Engineering, Beijing Institute of Technology, Beijing, China (email: gny123@foxmail.com; xudong.zhang@bit.edu.cn; zouyuanbit@vip.163.com; 602903705@qq.com; ztao1208@126.com, Corresponding author: Xudong Zhang and Yuan Zou).

Basilio Lenzo is with the Department of Engineering and Mathematics, Sheffield Hallam University, Sheffield S1 1WB, UK (email: basilio.lenzo@shu.ac.uk)

problems through MPC allows to easily tackle nonlinearities or time-varying features of the system. Specifically, NMPC (nonlinear MPC) is needed due to the nonlinearity of tire behavior. However, NMPC may entail significant computational cost. To mitigate that, several approaches were proposed. A linear time varying MPC (LTV-MPC) controller was presented in [11], in which the nonlinear model is linearized through a Taylor expansion. Simulation and experimental results proved the feasibility of the LTV-MPC strategy, although with poorer tracking performance than the nominal NMPC. In [12], the explicit MPC was proposed for DYC of DDEVs. As for real-time calculation, the control problem was optimized offline, thus generating appropriate look-up tables to be used online. Another example of an explicit MPC law can be found in [13], where the NMPC problem was solved offline by using nearest point approach. Here 105 grid points were applied to build the final control law, and the simulation results showed similar performance to a nominal NMPC. Even so, such performance was obtained with a large number of points used in the offline optimization, leading to high memory requirements that limited the method applicability. Ref. [14] developed an NMPC strategy for DDEVs, where hardware-in-the-loop (HIL) experiments were carried out based on particle swarm optimization (PSO) algorithm under field programmable gate array (FPGA) chip. Owing to the parallel calculation capacity of PSO and FPGA, real-time calculation was possible; however, such method is not suitable for large scale applications due to the FPGA high cost.

Modern vehicles are equipped with an electronic stability program (ESP) supervisor, which triggers a friction brake-based DYC function when it detects a potential incipient loss of vehicle stability [15]. In normal driving conditions, instead, the ESP is not active. During a generic vehicle journey, the ESP may switch "ON/OFF" a number of times, this has an important effect on the initialization of the MPC and therefore on its ability to converge. With an unsuitable initial solution, the optimality of the algorithm may deteriorate, thereby the algorithm may diverge hindering vehicle safety. Very few contributions in the literature focus on this issue. Refs. [16, 17] set the initial values as a zeros vector, but that is appropriate only if the controller starts when the vehicle is standstill. Other approaches include numerical iteration methods [18, 19] to work out an appropriate initial solution; however, they are not real-time applicable.

This paper proposes a novel NMPC approach for DYC of DDEVs, which overcomes the above drawbacks by introducing the following novelties:

1) The C/GMRES algorithm is used within the NMPC, to achieve superior control performance with reduced computational burden.

2) The external penalty method is applied for implementing relevant model constraints in the C/GMRES algorithm, and the existence and uniqueness of the solution are proved.

3) A varying predictive duration is adopted, leading to an analytical expression for an optimal initial solution of the NMPC.

The remainder of this paper is organized as follows. Section II introduces the control-oriented model. Section III deals with the formulation of the NMPC strategy. Section IV investigates numerical simulations assessing the performance of the proposed strategy. The main conclusions are in Section V.

II. Control-Oriented Model of Distributed Drive Electric Vehicle

The studied DDEV is a passenger car with two axles and four wheels, where each wheel is assembled with an IWM and only the front wheels can steer. In this paper a single track model is adopted, with nonlinear tires.

A. Single Track Vehicle Model

The single track vehicle model is adopted for controller design in this paper, and the schematic diagram of vehicle dynamic is shown in Fig. 1. The vehicle dynamics is modeled as [20]:

$$\begin{cases} m_v(\dot{v}_y + v_x\gamma) = \sum F_y = F_{yf} + F_{yr} \\ I_z\dot{\gamma} = \sum M_z = l_a F_{yf} - l_b F_{yr} \end{cases} \tag{1}$$

where m_v, F_y, l_a, l_b, M_z and I_z are, respectively, the total vehicle mass, the lateral force, the distance from front axle to the center of gravity (CG), the distance from rear axle to CG, the vehicle yaw moment and yaw moment inertia. v_x and v_y represent the longitudinal and lateral velocity of the vehicle CG, respectively. F_{yf} and F_{yr} denote the lateral forces of the front and rear tires, which are expressed as

$$\begin{cases} F_{yf} = F_{yfl} + F_{yfr} \\ F_{yr} = F_{yrl} + F_{yrr} \end{cases} \tag{2}$$

where the subscripts "fl", "fr", "rl", and "rr" express that the corresponding variables are related to front, rear, front-left, front-right, rear-left and rear-right wheels. The tire sideslip angles of front and rear wheels, i.e., α_f and α_r, can be calculated as

$$\begin{cases} \alpha_f = \delta - (v_y + l_a\gamma)/v_x = \delta - (\beta + l_a\gamma/v_x) \\ \alpha_r = (l_b\gamma - v_y)/v_x = l_b\gamma/v_x - \beta \end{cases} \tag{3}$$

where δ is the front wheel steering angle. β and γ are the vehicle sideslip angle and the yaw rate, respectively.

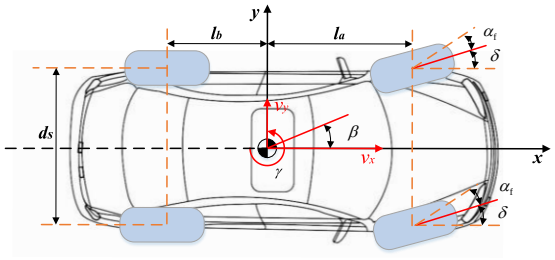


Fig. 1. Schematic diagram of vehicle dynamic.

B. Magic Formula Tire Model

An empirical tire model [21], MF developed by Pacejka, is employed to deal with the strong nonlinearity of tire lateral force under pure-slip condition:

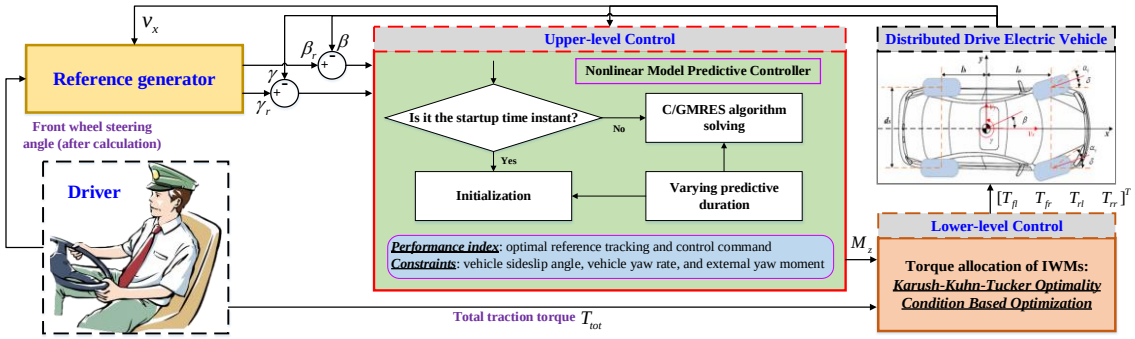


Fig. 2. Control framework illustration.

$$\begin{aligned} F_y &= M(\alpha,\mu,F_z) \\ &= D_o \sin\left\{C_o \arctan\left[B_o\alpha - E_o\left(B_o\alpha - \arctan(B_o\alpha)\right)\right]\right\} \end{aligned} \tag{4}$$

where $B_o = C_{F\alpha}/(C_o D_o)$ is the stiffness factor, and $D_o = \mu F_z$ is the peak factor. μ and F_z denote the road adhesion coefficient and the vertical load of the tire, respectively. $C_{F\alpha} = B_o C_o D_o = c_1 \sin\left(2\arctan(F_z/c_2)\right)$ is the cornering stiffness. The shape factors C_o, E_o, and the parameters c_1 and c_2 are determined through least-squares approximation [22].

C. Control-oriented Model for Controller Design

According to Eqs. (2) and (4), the lateral forces F_{yf} and F_{yr} in the single track vehicle model can be presented as

$$\begin{cases} F_{yf} = 2\times M(\alpha_f,\mu,F_{zf}) \\ F_{yr} = 2\times M(\alpha_r,\mu,F_{zr}) \end{cases} \tag{5}$$

where F_{zf} and F_{zr} denote the vertical loads of front and rear axles, respectively. By adding Eq. (5) into Eq. (1) with an external yaw moment $\tilde{M}_z$, the updates of β and γ can be yielded:

$$\begin{cases} \dot{\beta} = \dfrac{2\left(M(\alpha_f,\mu,F_{zf}) + M(\alpha_r,\mu,F_{zr})\right)}{m_v v_x} - \gamma \\ \dot{\gamma} = \dfrac{2\left(l_a M(\alpha_f,\mu,F_{zf}) - l_b M(\alpha_r,\mu,F_{zr})\right)}{I_z} + \dfrac{\tilde{M}_z}{I_z} \end{cases} \tag{6}$$

To sum up, based on Eqs. (3), (4), and (6), the control-oriented model of the DDEV can be formulated as,

$$\begin{cases} \dot{x} = f(x,u,w) \\ y = Cx \end{cases} \tag{7}$$

where $x = [\beta \quad \gamma]^T$ and $u = \tilde{M}_z$. $w = \delta$ represents the external disturbance, which is the steering angle of front wheels from driver. C is a row vector with all the elements of one. Now the formulation of control-oriented model is completed, and the proposed DYC strategy will be illustrated in the following.

III. Real-time Nonlinear Model Predictive Controller

A. Control Framework

The control framework of the proposed strategy is hierarchical, as shown in Fig. 2. Firstly, at each time instant, an embedded driver model from CarSim® provides the front wheel steering angle and the total traction torque demand. A reference generator produces the reference values of sideslip angle and yaw rate. Based on the feedback longitudinal vehicle velocity, vehicle sideslip angle error, and vehicle yaw rate error, a NMPC controller with C/GMRES algorithm is used to determine the desired external yaw moment. Focusing on the fast initialization in the C/GMRES algorithm, the varying predictive duration and an initialization approach are proposed, as described in detail below. Then, a KKT optimality condition based torque allocation method is employed in the lower-level control for optimal torque distribution of IWMs. Finally, the optimal torque command is sent to a DDEV model implemented in CarSim®.

B. Reference Generator

The reference generator produces the desired vehicle sideslip angle and yaw rate for achieving desired vehicle maneuverability and stability targets. In this paper, the control reference of β_r and γ_r are given as below.

$$\begin{cases} \beta_r = 0 \\ \gamma_r = \min\left(\max\left(\dfrac{v_x\delta}{(l_a+l_b)+\kappa v_x^{\,2}}, -\mu g/v_x\right), \mu g/v_x\right) \end{cases} \tag{8}$$

where κ represents the understeer gradient of vehicle. $\pm\mu g/v_x$ is a boundary of yaw rate for vehicle stability and derived from the certainly lateral acceleration $a_y \le |\mu g/v_x|$ [23].

C. Upper-level Control: Nonlinear Model Predictive Controller

1) Control Problem Construction

Here, the NMPC control problem of the yaw motion control can be written as

$$\begin{aligned} \min \quad & J_{mpc} = g(x(t_o+N_p),u(t_o+N_p)) + \int_{t_o}^{t_o+N_p-1} l(x(\tau),u(\tau))d\tau \\ \text{s.t.} \quad & \dot{x}(\tau) = f(x(\tau),u(\tau)) \\ & x_o = x(t_o) \\ & h(x(\tau),u(\tau)) \le 0 \end{aligned} \tag{9}$$

where $g(x(t_o+N_p),u(t_o+N_p))$ denotes the terminal cost of $\left(x(N_p)-x_r\right)^T W\left(x(N_p)-x_r\right)$, where W is the weight matrix of $\mathrm{diag}\{w_1 \quad w_2\}$. x_o is the initial system state. N_p represents the steps of predictive horizon. t_o is the current sample instant

of controller. $l(x(\tau),u(\tau))$ is the performance cost at each time step:

$$\begin{aligned} l(x(\tau),u(\tau)) = &\left(x(\tau)-x_r\right)^T Q\left(x(\tau)-x_r\right) \\ &+\left(u(\tau)-u(t_o-\Delta\tau)\right)^T R\left(u(\tau)-u(t_o-\Delta\tau)\right) \end{aligned} \tag{10}$$

where $Q=\mathrm{diag}\{q_1 \quad q_2\}$ and R express the weight factors corresponding to state error and control variable, respectively. $x_r=[\beta_r \quad \gamma_r]^T$ is the reference state for tracking, and $\Delta\tau$ is the time step in predictive horizon. $u(t_o-\Delta\tau)$ indexes the control command at the last sample instant, and the quadratic penalty item of $u(\tau)-u(t_o-\Delta\tau)$ in Eq. (10) is to avoid the optimized command chattering. The inequality constraints $h(x(\tau),u(\tau))$ are set as:

$$\begin{cases} \beta_{\min} \le \beta \le \beta_{\max} \\ \gamma_{\min} \le \gamma \le \gamma_{\max} \\ \tilde{M}_{z\min} \le \tilde{M}_z \le \tilde{M}_{z\max} \end{cases} \tag{11}$$

where the subscripts of "max" and "min" mean the allowable maximum and minimum related variables, respectively. $\tilde{M}_{z\min}=-\tilde{M}_{z\max}$ is imposed here for simplicity, and the boundaries of β and γ in Eq. (11) are defined as $\begin{cases} \beta_{\max}=-\beta_{\min}=\arctan(0.02\mu g) \\ \gamma_{\max}=-\gamma_{\min}=\mu g/v_x \end{cases}$ [20], where g is the gravitational acceleration.

2) Continuation/ Generalized Minimal Residual Algorithm

Since the control problem (9) is highly nonlinear and with inequality constraints, the C/GMRES algorithm is employed for efficient solving. The calculation process is explicit so that the number of mathematical operations to perform in NMPC at each sample time is fixed, which ensures a finite computational time for online solving compared with numerical iteration algorithms [24]. Furthermore, the optimization quality can be guaranteed because of the use of the global optimality conditions of PMP.

Because the traditional C/GMRES algorithm cannot handle the inequality constraints, the external penalty method is adopted to construct an equivalently unconstrained optimization problem by transforming the inequality constraints to the penalty cost items in the performance index. The expression of external penalty cost can be written as:

$$\vartheta_j(x(\tau),u(\tau)) = \begin{cases} 0 & , h_j(x(\tau),u(\tau)) \le 0 \\ \rho_j \times h_j(x(\tau),u(\tau))^2 & , h_j(x(\tau),u(\tau)) > 0 \end{cases} \tag{12}$$

where $\vartheta_j(x(\tau),u(\tau))$ and ρ_j are the external penalty cost and the weight coefficient for the j th inequality constraint $h_j(x(\tau),u(\tau))$. In this paper, $j=1,2,3$, and $h_j(x(\tau),u(\tau))$ is:

$$\begin{cases} h_1(x(\tau),u(\tau)) = \beta^2 - {\beta_{\max}}^2 \\ h_2(x(\tau),u(\tau)) = \gamma^2 - {\gamma_{\max}}^2 \\ h_3(x(\tau),u(\tau)) = {\tilde{M}_z}^2 - {\tilde{M}_{z\max}}^2 \end{cases} \tag{13}$$

Note that only the maximum boundary of each variable is adopted in Eq. (13) since their maximum equals to the related minimum multiplied by -1. Then, the original control problem can be reformulated as below, where the inequality constraints are transformed in external penalty cost:

$$\begin{aligned} \min \quad & J_{mpc} = g(x(t_o+N_p),u(t_o+N_p)) \\ & +\int_{t_o}^{t_o+N_p-1} l(x(\tau),u(\tau)) + \sum_j^3 \vartheta_j(x(\tau),u(\tau))d\tau \\ \text{s.t.} \quad & \dot{x}(\tau) = f(x(\tau),u(\tau)) \\ & x_o = x(t_o) \end{aligned} \tag{14}$$

where ρ_1, ρ_2, and ρ_3 are the weight coefficients of external penalty cost regarding vehicle sideslip angle, yaw rate and external yaw moment, respectively. Now the control problem is available for the application of C/GMRES algorithm. Based on the PMP [25], the Hamiltonian function of problem (14) can be written:

$$\begin{aligned} H(x(\tau),u(\tau),\lambda^T(\tau)) = &\, l(x(\tau),u(\tau)) + \lambda^T(\tau)\times f(x(\tau),u(\tau)) \\ &+\sum_j^3 \vartheta_j(x(\tau),u(\tau)) \end{aligned} \tag{15}$$

where λ^T represents the co-state vector, and the optimality condition of PMP describes that if the optimal control sequence $\{u^*(\tau)\}_{\tau=t_o}^{t_o+N_p-1}$ exists, there must have the corresponding $\{\lambda^*(\tau)\}_{\tau=t_o}^{t_o+N_p-1}$ making the following relationships true:

$$u^*(\tau) = \arg\min[H(x^*(\tau),u(\tau),\lambda^{*T}(\tau))] \tag{16}$$

$$x^*(\tau+1) = f(x^*(\tau),u^*(\tau))\Delta\tau + x^*(\tau) \tag{17}$$

$$\lambda^*(\tau) = \lambda^*(\tau+1) + \frac{\partial H}{\partial x}(x^*(\tau+1),u(\tau+1),\lambda^{*T}(\tau+1))\Delta\tau \tag{18}$$

$$x_o(t) = x(t_o) \tag{19}$$

$$\lambda(t_o+N_p-1) = \frac{\partial g}{\partial x}(x^*(t_o+N_p-1)) \tag{20}$$

For simplicity, define the optimized vector as $U(t_o)=\underbrace{[u(t_o) \quad \cdots \quad u(\tau) \quad \cdots \quad u(t_o+N_p-1)]^T}_{N_p}$. By recursive calculations according to Eqs. (15) to (20), the optimization problem can be reformulated as:

$$\begin{aligned} F(U(t_o),x_o(t)) = &\begin{bmatrix} \frac{\partial H}{\partial u}(x^*(t_o),u^*(t_o),\lambda^{*T}(t_o)) \\ \vdots \\ \frac{\partial H}{\partial u}(x^*(\tau),u^*(\tau),\lambda^{*T}(\tau)) \\ \vdots \\ \frac{\partial H}{\partial u}(x^*(t_o+N_p-1),u^*(t_o+N_p-1),\lambda^{*T}(t_o+N_p-1)) \end{bmatrix} \\ &= 0 \end{aligned} \tag{21}$$

Ideally, Eq. (21) can be solved by numerical iteration algorithms [26], such as trust-region-dogleg (TRD) and interior point (IP) methods, yet entailing significant computational cost. Instead, the C/GMRES algorithm can be adopted for the above problem with acceptable computational efficacy, which avoids the calculations of Jacobian matrix, Hessian matrix and inverse. Based on continuation method [27], $F(U(t_o),x_o(t),t)$ can be transformed as a linear dynamic system,

$\dot{F}(U(t_o),x(t),t) = -\psi_s F(U(t_o),x(t),t)$, where ψ_s is the stability matrix for stabilizing $F(U(t_o),x_o(t),t)$ at original. If $\frac{\partial F}{\partial U}(U(t_o),x(t_o),t)$ is nonsingular, the solution $\dot{U}(t_o)$ is determined by,

$$\begin{aligned}&\dot{U}(t_o)\\&=[\frac{\partial F}{\partial U}(U(t_o),x(t),t)]^{-1}[-\psi_s F(U(t_o),x(t),t)\\&-\frac{\partial F}{\partial x}(U(t_o),x(t),t)\times\dot{x}(t)]\end{aligned} \tag{22}$$

In Eq. (22), a significant computational burden is caused from the Jacobians of F with respect to x and u and its inversion of $\frac{\partial F}{\partial U}$. To handle it, the products of Jacobians and vectors are estimated by forward difference approximation, as below,

$$\begin{aligned}\dot{F}(U,x,t) &= \frac{\partial F}{\partial U}\dot{U}+\frac{\partial F}{\partial x}\dot{x}+\frac{\partial F}{\partial t}\\&\approx \frac{F(U+\dot{U}h,x+\dot{x}h,t+h)-F(U,x,t)}{h}\\&:= D_h F(U,x,t:\dot{U},\dot{x},1)\end{aligned} \tag{23}$$

where h is a positive real value. From Eq. (23), Eq. (22) can be rewritten as,

$$\begin{aligned}&\frac{\partial F}{\partial U}(U(t_o),x(t),t)\dot{U}(t_o)\\&=-\psi_s F(U(t_o),x(t),t)-\frac{\partial F}{\partial x}(U(t_o),x(t),t)\times\dot{x}(t)\\&\Leftrightarrow D_h F(U(t_o),x(t)+\dot{x}h,t+h:\dot{U},0,0)\\&=-\psi_s F(U(t_o),x(t),t)-D_h F(U(t_o),x(t),t:0,\dot{x},1)\end{aligned} \tag{24}$$

TABLE I.
FDGMRES ALGORITHM ILLUSTRATION.

FDGMRES algorithm: $\dot{U}^* = \text{FDGMRES}(U,\dot{U},x,\dot{x},F,e_{tol},k_{\max},h)$

Input:

$\hat{r} := -\psi_s F(U(t_o),x(t),t) - D_h F(U(t_o),x(t),t:0,\dot{x},1) - D_h F(U(t_o),x(t),t:\dot{U},0,0)$, $v_1 := \hat{r}/\|\hat{r}\|$, $\varphi := \|\hat{r}\|$, $\varpi := \varphi$, $k := 0$

Output: $\dot{U}$

while $k < k_{\max}$ or $\varphi > e_{tol}$ do

  $k = k+1$, and $v_{k+1} := D_h F(U(t_o),x(t)+\dot{x}h,t+h:v_k,0,0)$

  for $j = 1,\ldots,k$ do

    $h_{jk} := v_{k+1}{}^T v_j$, and $v_{k+1} := v_{k+1} - h_{jk}v_j$

  end

  $h_{k+1,k} := \|v_{k+1}\|$, and $v_{k+1} := v_{k+1}/\|v_{k+1}\|$

  $e_1 = [1 \;\; 0 \;\; \cdots \;\; 0]^T \in \mathbb{R}^{k+1}$, $H_k = \{h_{ij}\} \in ¡^{(k+1)\times k}$ (if $i > j+1$ then $h_{ij} = 0$)

  Minimize $\|\varpi e_1 - H_k y^k\|$ to determine $y^k \in ¡^k$.

  $\varphi := \|\varpi e_1 - H_k y^k\|$, $V_k = [v_i] \in \mathbb{R}^{mN_p \times k}$, where m is the dimension sum of u , and mN_p equals to that of $\dot{U}$.

end

$\dot{U}^* := \dot{U} + V_k y^k$

This is an approximately linear equation regarding $\dot{U}$ and can be fast solved by GMRES algorithm. The GMRES algorithm is derived from Krylov-subspace method, which is designed to solve large sparse linear equations $Ax = b$ with nonsymmetrical matrix A for the minimization of residual $\|b - Ax\|$. The advantage of this algorithm is that, in principle, it can reduce the residual monotonically and it converges to the optimal solution within a number of iterations equaling to the dimension of the given equation. The GMRES algorithm with forward approximation is commonly called FDGMRES. For more details, the calculation process of FDGMRES algorithms are illustrated in Table I [28].

Remark 1: To handle the inequality constraints, common approaches are the external penalty method, the auxiliary variable method, and the barrier function method. In [29] and [30], the inequality constraints were hold by transforming them into a set of same-dimensional equality constraints, in which the dummy variables of ϑ_j was defined to construct the equalities as $h_j(x(t_o),u(\tau)) + \vartheta_j{}^2 = 0$. A small dummy penalty term was added to the cost function to avoid singularity at any $\vartheta_j = 0$, namely $\tilde{l}(x(\tau),u(\tau)) = l(x(\tau),u(\tau)) - \sum_j^m w_j \vartheta_j$, where $\tilde{l}(x(\tau),u(\tau))$ is the new cost function in predictive period, and w_j is the weight coefficient regarding the j th penalty item. The auxiliary variable method was proven to be difficult to stabilize and tune [31]. For the barrier function method, Ref. [32] mentioned that the inequality constraints can be processed as an additional cost item of the log function in the performance index. This method is potentially effective to handle the boundary optimization problem in the C/GMRES algorithm. However the method requires good accuracy of the control-oriented model, which is not deemed the case here, mainly due to the simplifying hypotheses coming with a single track vehicle model. So, the external penalty method in Eq. (12) is finally chosen to deal with the inequality constraints.

Remark 2: It should be noted that the optimal solution of C/GMRES algorithm is existing and exclusive only when $\frac{\partial F}{\partial U}$ is nonsingular [29]. From Eq. (21), $\frac{\partial F}{\partial U}$ is actually a Hessian matrix of the Hamiltonian function with respective to U . In addition, given that the optimized variable u is only related to the Hamiltonian function at one time instant of predictive horizon, $\frac{\partial F}{\partial U}$ is a $N_p \times N_p$ block diagonal matrix. Therefore, whether $\frac{\partial F}{\partial U}$ is nonsingular can be determined by verifying the invertibility of $\frac{\partial^2 H}{\partial u^2}$. The expression of $\frac{\partial^2 H}{\partial u^2}$ can be

calculated to be $\begin{cases} R & , h_3(x(\tau),u(\tau)) \le 0 \\ R+2\rho_3 & , h_3(x(\tau),u(\tau)) > 0 \end{cases}$. Since R and ρ_3 are positive, $\frac{\partial^2 H}{\partial u^2} > 0$ indicating that $\frac{\partial F}{\partial U}$ is nonsingular.

3) Varying Predictive Duration and Initialization

To adapt the frequent initialization issue of C/GMRES algorithm, a fast initialization approach is customized for C/GMRES algorithm under the considerations that the NMPC controller starts at nonzero states and/or nonzero external disturbance (e.g. the vehicle is assembled with an ESP as mentioned in Section I). Here a time-dependent duration of predictive horizon is imposed [30],

$$T(t) = T_f \times (1 - e^{-\varepsilon t}) \tag{25}$$

where $T(t)$ is the duration of predictive horizon; T_f is a given time duration constant; t is a time value that records the time duration of controller operation, which is reset to zero when the controller restarts; and ε is a coefficient that determines the increase rate of $T(t)$. With greater ε, $T(t)$ increases more rapidly and tends to T_f as $t \to \infty$. By applying Eq. (25) when the controller is starting, the time duration of predictive horizon is zero, and $u^*(0) = u^*(\tau) = u^*(N_p - 1)$, $x^*(0) = x^*(\tau) = x^*(N_p - 1)$, and $\lambda^*(0) = \lambda^*(\tau) = \lambda^*(N_p - 1) = \frac{\partial g}{\partial x}(x^*(N_p - 1))$. Hence, the initialization can be reduced to find only one variable $u(0)$. Since the Hamiltonian function of Eq. (15) is convex regarding $u(0)$, the optimal solution $u^*(0)$ can be solved according to $\frac{\partial H}{\partial u} = 0$ and the boundary of u.

The boundary of u, $\tilde{M}_{z\min\_new}$ and $\tilde{M}_{z\max\_new}$, can be determined by combining the constraints regarding states and controls. Based on the state update function (7), the transformed constraints of control variable can be worked out by reverse calculation:

$$\gamma_{\min} \le \gamma(t_o) + \left(\frac{2\left(l_a M(\alpha_f(t_o),\mu,F_{zf}) - l_b M(\alpha_r(t_o),\mu,F_{zr})\right)}{I_z} + \frac{\tilde{M}_z(0)}{I_z} \right) \Delta t \le \gamma_{\max}$$

$$\Leftrightarrow \begin{cases} \tilde{M}_z(0) \ge I_z \frac{\gamma_{\min} - \gamma(t_o)}{\Delta t} - 2\left(l_a M(\alpha_f(t_o),\mu,F_{zf}) - l_b M(\alpha_r(t_o),\mu,F_{zr})\right) \\ \tilde{M}_z(0) \le I_z \frac{\gamma_{\max} - \gamma(t_o)}{\Delta t} - 2\left(l_a M(\alpha_f(t_o),\mu,F_{zf}) - l_b M(\alpha_r(t_o),\mu,F_{zr})\right) \end{cases} \tag{26}$$

where Δt is the sampling step of controller. Note that since $\beta(t_o)$ is not determined by $\tilde{M}_z(0)$ during initialization, the state constraint of β is not considered here. Then, the boundary of u in initialization is obtained:

$$\begin{cases} \tilde{M}_{z\max\_new} = \min\left(I_z \frac{\gamma_{\max} - \gamma(t_o)}{\Delta t} - 2\left(l_a M(\alpha_f(t_o),\mu,F_{zf}) - l_b M(\alpha_r(t_o),\mu,F_{zr})\right), \tilde{M}_{z\max} \right) \\ \tilde{M}_{z\min\_new} = \max\left(I_z \frac{\gamma_{\min} - \gamma(t_o)}{\Delta t} - 2\left(l_a M(\alpha_f(t_o),\mu,F_{zf}) - l_b M(\alpha_r(t_o),\mu,F_{zr})\right), \tilde{M}_{z\min} \right) \end{cases} \tag{27}$$

Thus $u(0)^*$ can be calculated by:

$$u(0)^* = \min\left(\max\left(\frac{\tilde{M}_{z\max}\rho_3 + R \times u(t_o - \Delta t) - \frac{w_2(\gamma(t_o) - \gamma_r(t_o))}{I_z}}{(R + 2\rho_3)}, \tilde{M}_{z\min\_new} \right), \tilde{M}_{z\max\_new} \right) \tag{28}$$

Moreover, when initializing, $u(t_o - \Delta t)$ is zero, and $\rho_3 = 0$ since the limits $\tilde{M}_{z\min\_new}$ and $\tilde{M}_{z\max\_new}$ are imposed in the calculation of Eq. (28). Eq. (28) can be rewritten as:

$$u(0)^* = \min\left(\max\left(\frac{w_2(\gamma_r(t_o) - \gamma(t_o))}{R \times I_z}, \tilde{M}_{z\min\_new} \right), \tilde{M}_{z\max\_new} \right) \tag{29}$$

Then the optimal $U(0)$ can be set as $U(0) = \underbrace{[1 \ \cdots \ 1]}_{N_p}^T \times u(0)^*$. For the determination of $\dot{U}(0)$, the optimized control command is assumed to vary smoothly, owing to the penalty item $\left(u(\tau) - u(t_o - \Delta t)\right)^T R \left(u(\tau) - u(t_o - \Delta t)\right)$ in the performance index. Thus, $\dot{U}(0) = \underbrace{[0 \ \cdots \ 0]}_{N_p}^T$.

To sum up, the operation schematic of the proposed NMPC controller is illustrated in Fig. 3. It is noteworthy that this schematic is applicable both if DYC is always on, or if it is managed by an ESP supervisor. In the latter case, when the ESP supervisor triggers DYC, there are two possible scenarios: 1) at the controller startup instant, the controller running time is reset to zero, the initialization calculation proceeds, and the desirable yaw motion command is calculated; 2) at a generic instant which is not the startup instant of controller, the predictive time duration length is obtained through Eq. (25) and the desirable external yaw moment command is worked out by the C/GMRES algorithm. To accelerate the convergence rate of optimization, the warm start approach is arranged here to set the initial values in C/GMRES algorithm, which means the optimized solution $U^*(t_o - \Delta t)$ is imposed as the initial solution $U(t_o)$ at the zero-th iteration.

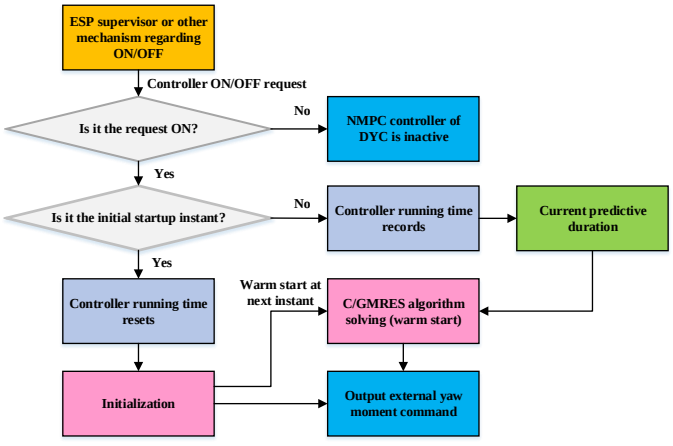


Fig. 3. Operation schematic of proposed NMPC controller.

TABLE II.
TWO-STEPS METHOD FOR TORQUE ALLOCATION OPTIMIZATION [33].

Two-steps method based on KKT global optimality condition

Define $\xi=[T_{fl}\quad T_{fr}\quad T_{rl}\quad T_{rr}]^T=[\xi_{fl}\quad \xi_{fr}\quad \xi_{rl}\quad \xi_{rr}]^T$, $\sigma_1=[\sigma_{1fl}\quad \sigma_{1fr}\quad \sigma_{1rl}\quad \sigma_{1rr}]^T$, $\sigma_2=[\sigma_{2fl}\quad \sigma_{2fr}\quad \sigma_{2rl}\quad \sigma_{2rr}]^T$, and $ij=fl,fr,rl,rr$.

1) Preliminary optimization:

Assume the cost function vector ξ belongs to the designed boundary.

Initialization: Set the Lagrange multiplier vector $\sigma_1=\sigma_2=0$

Solve: Quasi-optimal solution $\tilde{\xi}^*$ using $\dfrac{\partial\left(\xi^T\Omega\xi+(\mathrm{K}\xi-v)^T\Xi(\mathrm{K}\xi-v)\right)}{\partial\xi}=0$

If There exists $\tilde{\xi}_{ij}^*$ in $\tilde{\xi}^*$ exceeding its lower or upper boundary, namely $T_{\max}$ or $T_{\min}$

Move to the secondary optimization

Else

Return $\tilde{\xi}^*$ as the globally optimal solution ξ^* for the cost function (31)

End

2) Secondary optimization:

Initialization: For $ij=fl,fr,rl,rr$

If $\tilde{\xi}_{ij}^*$ are greater than $T_{\max}$, **Set** σ_{1ij} equals to zero and $\tilde{\xi}_{ij}^*$ equals to $T_{\max}$

If $\tilde{\xi}_{ij}^*$ are less than $T_{\min}$, **Set** σ_{2ij} equals to zero and $\tilde{\xi}_{ij}^*$ equals to $T_{\min}$

End

Set the other variables as independent variables.

Solve: Quasi-optimal solution $\tilde{\xi}^*$ using $\dfrac{\partial\left(\xi^T\Omega\xi+(\mathrm{K}\xi-v)^T\Xi(\mathrm{K}\xi-v)+\sigma_1^T(\xi-\xi_{\max})+\sigma_2^T(\xi_{\min}-\xi)\right)}{\partial\xi}=0$ of Eq. (32)

If There exists $\tilde{\xi}_{ij}^*$ in $\tilde{\xi}^*$ exceeding the lower or upper boundary, namely $T_{\max}$ or $T_{\min}$

Repeat the secondary optimization

Else

Return $\tilde{\xi}^*$ as the globally optimal solution ξ^* for the cost function (31)

End

D. Lower-level Control: Optimal Torque Allocation Based on Karush-Kuhn-Tucker Optimality Condition

Owing to the actuation redundancy of DDEV, the torque output of IWMs should be appropriately distributed. The relationship between $\xi=[T_{fl}\quad T_{fr}\quad T_{rl}\quad T_{rr}]^T$ as well as the given $\tilde{M}_z$ and T_{tot} is presented as:

$$v=\mathrm{K}\xi \tag{30}$$

where $v=[\tilde{M}_z\quad T_{tot}]^T$ and $\mathrm{K}=\begin{bmatrix} d_s/(2r_w)\times[-1\quad 1\quad -1\quad 1] \\ [1\quad 1\quad 1\quad 1] \end{bmatrix}$. d_s and r_w denote the wheel track and the effective radius of wheel, respectively. The cost function can be formulated as:

$$\begin{aligned} &J=\xi^T\Omega\xi+(\mathrm{K}\xi-v)^T\Xi(\mathrm{K}\xi-v) \\ &\text{s.t.}\quad \xi_{\min}\le\xi\le\xi_{\max} \end{aligned} \tag{31}$$

where $\Xi=\mathrm{diag}\{\Xi_1,\Xi_2\}$ is a penalty weight matrix for satisfying the equality conditions, and Ξ_1 and Ξ_2 are the weight coefficients regarding $\tilde{M}_z$ and T_{tot}, respectively. Here $\xi_{\max}$ and $\xi_{\min}$ are defined as $[1\quad 1\quad 1\quad 1]^T\times T_{\max}$ and $[1\quad 1\quad 1\quad 1]^T\times T_{\min}$, respectively. The item $\xi^T\Omega\xi$ in Eq. (31) represents the tire workload usage, where the weight matrix Ω is expressed as $\mathrm{diag}\{\dfrac{1}{(r_w\mu F_{zfl})^2},\dfrac{1}{(r_w\mu F_{zfr})^2},\dfrac{1}{(r_w\mu F_{zrl})^2},\dfrac{1}{(r_w\mu F_{zrr})^2}\}$. Based on no equality constraint and $\xi_{\min}\le\xi\le\xi_{\max}$ in Eq. (31), the linearly independent constraint qualification (LICQ) [34] is hold so that KKT optimality condition is applicable. The problem of Eq. (31) can be rewritten as an equivalent one which holds the following expressions:

$$\begin{cases} \dfrac{\partial\left(\xi^T\Omega\xi+(\mathrm{K}\xi-v)^T\Xi(\mathrm{K}\xi-v)+\sigma_1^T(\xi-\xi_{\max})+\sigma_2^T(\xi_{\min}-\xi)\right)}{\partial\xi}=0 \\ \sigma_1\ge 0,\sigma_2\ge 0 \\ \xi-\xi_{\max}\le 0,\xi_{\min}-\xi\le 0 \\ \sigma_1^T\times(\xi-\xi_{\max})=0,\sigma_2^T\times(\xi_{\min}-\xi)=0 \end{cases} \tag{32}$$

where σ_1 and σ_2 are the Lagrange multiplier vectors. To fast find the optimal solution ξ^* of Eq. (32), a two-steps method can be applied consisting of the preliminary and the secondary optimization, which has been published in our previous research [33] and is concisely described in Table II.

IV. NUMERICAL SIMULATION VALIDATION

In this paper, the powerful and high-fidelity vehicle simulation software CarSim® is adopted to effectively validate the proposed strategy [35]. The parameters regarding the vehicle and the proposed strategy are listed in Table III. The control performance is validated in scenarios without and with an ESP supervisor. In the latter case, the proposed initialization method is assessed. Two additional NMPC algorithms (i.e., the active set (AS) and IP ones) are also tested and compared with the proposed C/GMRES algorithm.

Commonly, the vehicle might lose its stability when the lateral acceleration is greater than 0.4 g. Here the switching "ON/OFF" mechanism of ESP supervisor in this paper is set, as shown in Fig. 4. When the lateral acceleration is greater than 0.3 g, the DYC controller is active, which is represented as number "1". If the DYC controller is already active, it will be arranged to be "OFF" until the vehicle lateral acceleration is less than 0.15 g. The selected threshold 0.3 g aims to pose a margin of 0.1 g (namely from 0.3 g to 0.4 g) for vehicle states adjustment of DYC controller, and the hysteresis here is to avoid the frequent switching of controller.

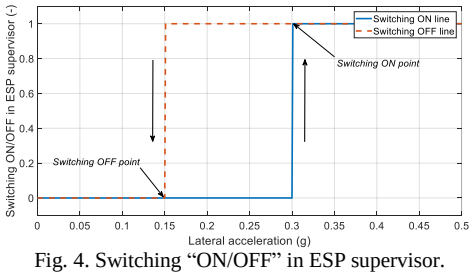


Fig. 4. Switching "ON/OFF" in ESP supervisor.

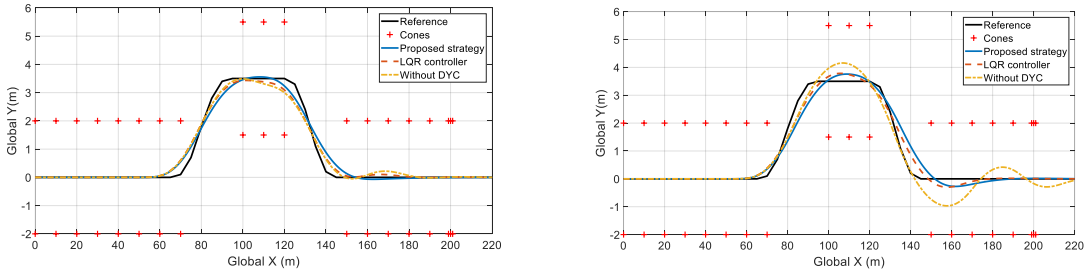
A. Control Performance

In this section, two double-lane-change (DLC) drive tests with different velocity and road adhesion coefficient are adopted, which are named "Case 1" and "Case 2". In Case 1, the target longitudinal velocity and road adhesion coefficient are 100 km/h and 0.85, respectively. In Case 2, the initial and target longitudinal velocity and the road adhesion coefficient are 80 km/h and 0.4, respectively. To comprehensively assess the performance of the proposed controller, two additional strategies are implemented and compared with the proposed one: i) a LQR-based DYC strategy; ii) the baseline vehicle, i.e. no DYC control.

Fig. 5 shows the path tracking results of Case 1 and Case 2. The path tracking effects by the proposed strategy and the LQR controller are similar. Without DYC controller, under Case 2, there is a greater swing adjusting operations of driver, indicating the importance of DYC in improving vehicle stability. Fig. 6 illustrates the vehicle sideslip angle results. The proposed strategy can implement the smaller yaw sideslip angle compared with that by LQR controller, illustrating more predominant vehicle stabilization capacity.

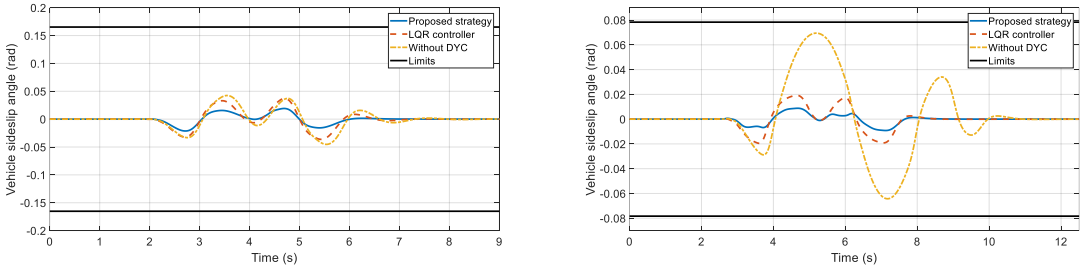
Table III.
PARAMETERS REGARDING VEHICLE AND PROPOSED STRATEGY.

Parameter	Value	Unit
Vehicle mass m_v	1412	kg
Distance from CG to front axle l_a	1.015	m
Distance from CG to rear axle l_b	1.895	m
Coefficient of MF model c_1	2.664×10^5	-
Coefficient of MF model c_2	3.334×10^4	-
Wheel track d_s	1.675	m
Wheel radius r_w	0.308	m
Vehicle moment of inertia around Z axis I_z	1536.7	kg m^2
Predictive horizon steps N_p	8	-
Sample cycle of controller Δt	0.02	s
Time step in predictive horizon $\Delta\tau$	0.02	s
Weight matrix of output state Q	diag{10, 7×10^5}	-
Weight matrix of control increment R	diag{10^{-2}}	-
Weight matrix of external penalty items W	diag{10^2, 10^5, 10^{-3}}	
Boundary of external yaw moment $\tilde{M}_z$	[-4000,4000]	N m
Time duration constant T_f	0.2	-
Coefficient regarding varying rate ε	10	-
Stability matrix in C/GMRES ψ_s	50	-
Max. iteration number in C/GMRES $k_{\max}$	4	-
Terminal residual norm in C/GMRES e_{tol}	0.001	-
Weight matrix of torque allocation K	diag{5, 20}	-
Boundary of IWM torque output $[T_{\min}, T_{\max}]$	[-600,600]	N m



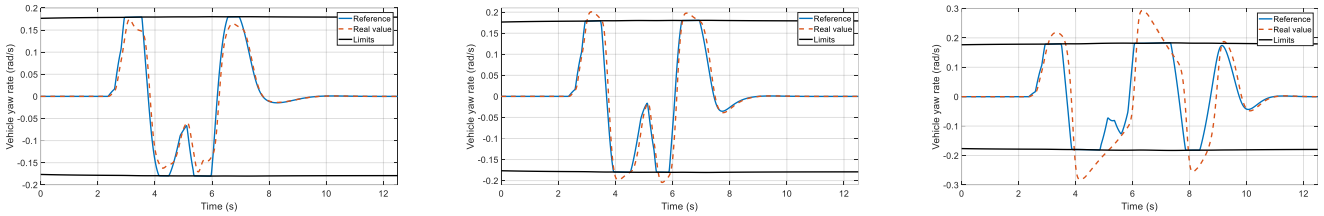
(a) (b)

Fig. 5. Path tracking results. (a) Case 1; (b) Case 2.



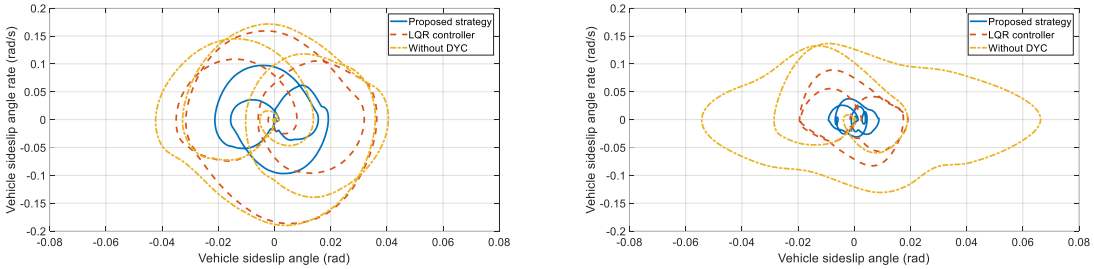
(a) (b)

Fig. 6. Vehicle sideslip angle results. (a) Case 1; (b) Case 2.



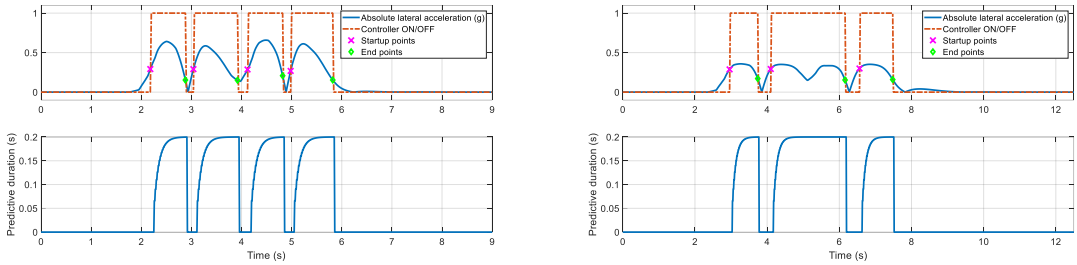
(a) (b) (c)

Fig. 7. Vehicle yaw rate results under Case 2. (a) proposed strategy; (b) LQR controller; (c) without DYC.



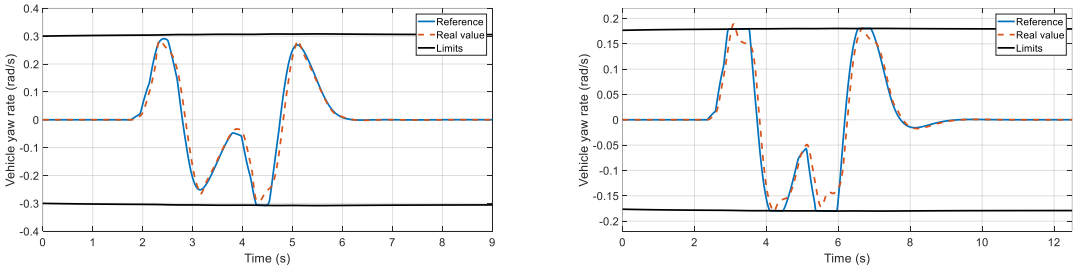
(a) (b)

Fig. 8. Vehicle sideslip angle-sideslip angle rate phase plane. (a) Case 1; (b) Case 2.



(a) (b)

Fig. 9. ESP supervisor “ON/OFF”, vehicle lateral acceleration and predictive duration results. (a) Case 1; (b) Case 2.



(a) (b)

Fig. 10. Vehicle yaw rate results by proposed strategy. (a) Case 1; (b) Case 2.

The vehicle yaw rate results of Case 2 are depicted in Fig. 7. The proposed strategy can effectively track the reference when the real yaw rate is relatively far from limits. The proposed strategy accounts for such limits in advance and it adjusts the control command in time for restricting the yaw rate within the constraints. However, the traditional LQR controller cannot explicitly impose the inequality constraints so that its maximum yaw rate is greater than the limit, namely $\mu g / v_x$. Without the DYC, the vehicle yaw rate is oscillating in Fig. 7 (c), which explains why the path tracking result is chattering at the turning maneuvers of Fig. 5 (b). More intuitively, Fig. 8 shows the vehicle sideslip angle-sideslip angle rate phase planes to illustrate the vehicle stability effect.

B. Effectiveness of Initialization Method in C/GMRES algorithm

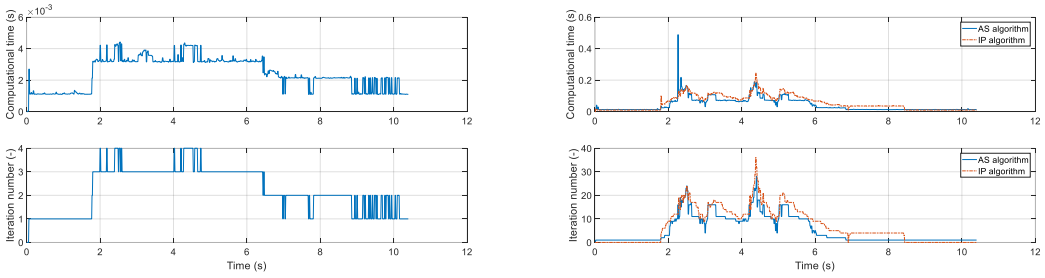
The results regarding ESP supervisor triggering and the varying predictive duration are illustrated in Fig. 9. The DYC controller is active under the turning maneuvers of DLC test cycle, since at those instants, the lateral acceleration is greater and the vehicle has the possibility of losing its stability. When the DYC controller is active, the predictive duration of NMPC extends and gradually reaches the given value of 0.2 s as increasing running time. The vehicle yaw rate results by the proposed strategy with ESP supervisor are shown in Fig. 10. With ESP supervisor, the real yaw rate by the proposed strategy can be effectively bounded, owing to its capacity of considering the yaw rate constraint in the predictive horizon. As illustrated in Fig. 10 (b), under Case 2, the real yaw rate is out of the limits at time $\approx$ 3 s for a short period and then rapidly decreases to meet the constraint. This can be explained as although the tire nonlinearity features are taken into account through the MF tire model, still some error is present due to the vehicle model,

affecting the NMPC controller. That said, under this situation, a greater penalty cost is added into the optimization of the NMPC controller, and the controller quickly adjusts the control command to restrict the yaw rate for minimizing the performance cost, as illustrated in Fig. 10 (b) between ≈ 3 and ≈ 3.5 s.

C. Comparison with Active Set and Interior Point Algorithms

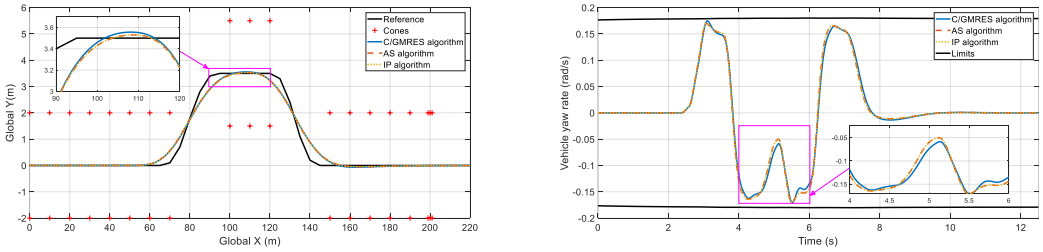
To illustrate the superiority of the C/GMRES algorithm, the AS and IP algorithms are employed for comparison. These two algorithms are implemented through the Matlab® function "*fmincon*", and their maximum toleration error are set to 0.01. All the simulations are implemented under Intel(R) Core(TM) i5-9400F CPU @ 2.9GHz desktop computer.

Fig. 11 shows the computational time and iteration number for the three algorithms under Case 1. Compared with AS and IP algorithms, the computational time of C/GMRES algorithm is significantly lower. Although the terminal residual norm is 0.001, most of the optimization by the C/GMRES algorithm are achieved at an iteration number lower than the preset maximum one $k_{\max}$ (namely 4), showing its outstanding calculation capacity. Table IV lists the three algorithms' computational times. Here, the effects of the ESP supervisor are not included. The C/GMRES algorithm has a distinct advantage in computational efficiency, with a computational burden that is one order of magnitude lower than AS and IP algorithms. Moreover, the maximum calculation time of C/GMRES algorithm is 110.98 to 53.42 times smaller than the other two. This is of significance since it is the critical index of whether the controller can be applied in real time. In light of the above, it is reasonable to deduce that the proposed strategy is suitable for real-time application under robot operating system (ROS) based hardware environment, like Raspberry Pi. For a more comprehensive comparison, Fig. 12 depicts the control performance results by three algorithms. Since the approximate errors of continuation method exist [19], as can be found in the zoomed figures, the path tracking and yaw rate results by C/GMRES algorithm are slightly different with those by AS and IP algorithms. However, it is acceptable focusing on the extreme reduction of computational time.



(a) (b)

Fig. 11. Computational time and iteration number under Case 1. (a) C/GMRES algorithm; (b) AS and IP algorithms.



(a) (b)

Fig. 12. Control performance comparison under Case 2. (a) path tracking; (b) vehicle yaw rate.

TABLE IV.
COMPUTATIONAL TIME COMPARISON.

Test cycle	Algorithm	Mean		RMSE		Maximum	
		Value (s)	Calculation burden ratio	Value (s)	Calculation burden ratio	Value (s)	Calculation burden ratio
Case 1	C/GMRES	0.0024	1	0.0010	1	0.0044	1
	AS	0.0425	17.71	0.0435	43.50	0.4883	110.98
	IP	0.0546	22.75	0.0458	45.80	0.2441	55.48
Case 2	C/GMRES	0.0024	1	0.0010	1	0.0053	1
	AS	0.0455	18.96	0.0480	48.00	0.3354	63.28
	IP	0.0599	24.96	0.0576	57.60	0.2831	53.42

V. CONCLUSION

In this paper, a DYC control strategy of DDEVs is proposed for improving the vehicle stability and handling. The control framework is arranged as two levels. In the upper-level control, the NMPC controller is adopted for DYC so as to produce the desirable external yaw moment. The C/GMRES algorithm is chosen to gain the optimal solution with fast computational efficiency, and external penalty method is employed to handle the inequality constraints. In addition, for fast initialization of C/GMRES algorithm, the varying predictive duration and the initialization method are also introduced in the proposed NMPC controller. In the lower-level control, the optimal torque

allocation is achieved by a two-steps method based on KKT optimality condition. The simulation results demonstrated that:

1) Compared with a LQR controller and the baseline vehicle, the proposed strategy can achieve considerably smaller vehicle sideslip angle with desirable yaw rate tracking effect, indicating its superior capacity in vehicle handling and stability.

2) Owing to the optimization of NMPC in the preview sight, the proposed strategy can simultaneously achieve the smooth transient performance and hold the inequality constraints. Instead, the LQR controller can only adjust the yaw motion of vehicle with greater overshoot for limiting the constraints.

3) Even when the initial tracking errors of states are nonzero, the proposed initialization method, namely the varying predictive duration and the analytic expression, allows to quickly obtain the optimally initial solution. This guarantees that the C/GMRES algorithm based NMPC can be applied for DDEVs with ESP supervisor. That is, the proposed strategy is applicable under the case that the status “ON/OFF” of a DYC switches multiple times during a generic vehicle journey.

4) Compared with AS and IP algorithm based NMPC controllers, the proposed one can implement similar control performance but at least one order of magnitude reduction on computational time. Moreover, the computational time of the proposed strategy per sample step in Windows operation system is around 0.003 s, indicating that it is applicable for real-world vehicle application under ROS hardware environment, such as Raspberry Pi.

Future works will focus on the experimental assessment of the proposed controller on a real-world DDEV.

REFERENCES

[1] B. Lenzo, F. Bucchi, A. Sorniotti, and F. Frendo, "On the handling performance of a vehicle with different front-to-rear wheel torque distributions," *Vehicle System Dynamics*, pp. 1-20, 2019.

[2] Y. Chen, J. K. Hedrick, and K. Guo, "A novel direct yaw moment controller for in-wheel motor electric vehicles," *Vehicle System Dynamics*, vol. 51, pp. 925-942, 2013.

[3] LuXiong, ZhuopingYu, YangWang, ChenYang, and YufengMeng, "Vehicle dynamics control of four in-wheel motor drive electric vehicle using gain scheduling based on tyre cornering stiffness estimation," *Vehicle System Dynamics*, vol. 50, p. 16, 2012.

[4] Z. Wang, U. Montanaro, S. Fallah, A. Sorniotti, and B. Lenzo, "A gain scheduled robust linear quadratic regulator for vehicle direct yaw moment Control," *Mechatronics*, vol. 51, pp. 31-45, 2018/05/01/ 2018.

[5] MasaoNAGAI, YutakaHIRANO, and SachikoYAMANAKA, "Integrated Robust Control of Active Rear Wheel Steering and Direct Yaw Moment Control," *Vehicle System Dynamics*, vol. 29, pp. 416-421, 1998.

[6] C. Hu, R. Wang, F. Yan, and M. Chadli, "Composite Nonlinear Feedback Control for Path Following of Four-Wheel Independently Actuated Autonomous Ground Vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, pp. 2063-2074, 2016.

[7] C. Hu, R. Wang, and F. Yan, "Integral Sliding Mode-based Composite Nonlinear Feedback Control for Path Following of Four-Wheel Independently Actuated Autonomous Electric Vehicles," *IEEE Transactions on Transportation Electrification*, vol. 2, pp. 221-230, 2016.

[8] X. Jin, Z. Yu, G. Yin, and J. Wang, "Improving Vehicle Handling Stability Based on Combined AFS and DYC System via Robust Takagi-Sugeno Fuzzy Control," *IEEE Transactions on Intelligent Transportation Systems*, vol. PP, pp. 1-12, 2017.

[9] J.M.Maciejowski, "Predictive Control with Constraints," *London, UK : Pearson Education*, 2002.

[10] G. C. Goodwin, D. S. Carrasco, and M. M. Seron, "Predictive control: a historical perspective," *International Journal of Robust and Nonlinear Control*, vol. 22, pp. 1296-1313, 2012.

[11] P. Falcone, F. Borrelli, J. Asgari, H. E. Tseng, and D. Hrovat, "Predictive Active Steering Control for Autonomous Vehicle Systems," *IEEE Transactions on Control Systems Technology*, vol. 15, pp. 566-580, 2007.

[12] M. Metzler, D. Tavernini, A. Sorniotti, and P. Gruber, "An Explicit Nonlinear MPC Approach to Vehicle Stability Control," in *Proceedings of The 14th International Symposium on Advanced Vehicle Control*, 2018.

[13] M. Canale and L. Fagiano, "Vehicle yaw control using a fast NMPC approach," in *2008 47th IEEE Conference on Decision and Control*, 2008, pp. 5360-5365.

[14] H. Guo, F. Liu, F. Xu, H. Chen, D. Cao, and Y. Ji, "Nonlinear Model Predictive Lateral Stability Control of Active Chassis for Intelligent Vehicles and Its FPGA Implementation," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, pp. 2-13, 2019.

[15] L. Zhai, T. Sun, and J. Wang, "Electronic Stability Control Based on Motor Driving and Braking Torque Distribution for a Four In-Wheel Motor Drive Electric Vehicle," *IEEE Transactions on Vehicular Technology*, vol. 65, pp. 4726-4739, 2016.

[16] H. Guo, D. Cao, H. Chen, Z. Sun, and Y. Hu, "Model predictive path following control for autonomous cars considering a measurable disturbance: Implementation, testing, and verification," *Mechanical Systems and Signal Processing*, vol. 118, pp. 41-60, 2019/03/01/ 2019.

[17] U. Rosolia, S. D. Bruyne, and A. G. Alleyne, "Autonomous Vehicle Control: A Nonconvex Approach for Obstacle Avoidance," *IEEE Transactions on Control Systems Technology*, vol. 25, pp. 469-484, 2017.

[18] J. Zhang and T. Shen, "Real-Time Fuel Economy Optimization With Nonlinear MPC for PHEVs," *IEEE Transactions on Control Systems Technology*, vol. 24, pp. 2167-2175, 2016.

[19] Z. Han, N. Xu, H. Chen, Y. Huang, and B. Zhao, "Energy-efficient control of electric vehicles based on linear quadratic regulator and phase plane analysis," *Applied Energy*, vol. 213, pp. 639-657, 2018.

[20] R. Rajamani, *Vehicle Dynamics and Control*: Springer Science, 2006.

[21] H. B. Pacejka, *Tire and Vehicle Dynamics (Third Edition)*. Oxford: Butterworth-Heinemann, 2012.

[22] C. Unterrieder, C. Zhang, M. Lunglmayr, R. Priewasser, S. Marsili, and M. Huemer, "Battery state-of-charge estimation using approximate least squares," *Journal of Power Sources*, vol. 278, pp. 274-286, 2015.

[23] X. Zhang and D. Göhlich, "Improvement in the vehicle stability of distributed-drive electric vehicles based on integrated model-matching control," *Proceedings of the Institution of Mechanical Engineers Part D Journal of Automobile Engineering*, vol. 232, p. 095440701770128, 2017.

[24] J. V. Frasch, A. J. Gray, M. Zanon, H. J. Ferreau, and M. Diehl, "An auto-generated nonlinear MPC algorithm for real-time obstacle avoidance of ground vehicles," in *Control Conference*, 2013.

[25] D. E. Kirk, "Optimal Control Theory: An Introduction," *Dover Publications*, 2004-04-30.

[26] A. Antoniou and W. S. Lu, *Practical Optimization: Algorithms and Engineering Applications*, 2010.

[27] E. L. Allgower and K. Georg, "Numerical Continuation Methods: An Introduction," *Mathematics of Computation*, vol. 13, pp. xxvi,388, 2003.

[28] C. T. Kelley, *Iterative methods for linear and nonlinear equations* vol. 16: Siam, 1995.

[29] T. Ohtsuka, "A continuation/GMRES method for fast computation of nonlinear receding horizon control," *Automatica*, vol. 40, pp. 563-574, 2004.

[30] Y. Hamada, T. Tsukamoto, and S. Ishimoto, "Receding horizon guidance of a small unmanned aerial vehicle for planar reference path following," *Aerospace Science & Technology*, vol. 77, pp. 129-137, 2018.

[31] M. Huang, H. Nakada, K. Butts, and I. Kolmanovsky, "Nonlinear Model Predictive Control of a Diesel Engine Air Path: A Comparison of Constraint Handling and Computational Strategies," *Ifac Papersonline*, vol. 48, pp. 372-379, 2015.

[32] S. Chao, B. Buckham, and S. Yang, "Modified C/GMRES Algorithm for Fast Nonlinear Model Predictive Tracking Control of AUVs," *IEEE Transactions on Control Systems Technology*, vol. PP, pp. 1-9, 2016.

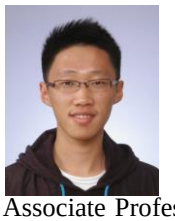
[33] X. Zhang, D. Göhlich, and W. Zheng, "Karush–Kuhn–Tuckert based global optimization algorithm design for solving stability torque allocation of distributed drive electric vehicles," *Journal of the Franklin Institute*, vol. 354, pp. 8134-8155, 2017.

[34] Boyd, Vandenberghe, and Faybusovich, "Convex Optimization," *IEEE Transactions on Automatic Control*, vol. 51, pp. 1859-1859, 2006.

[35] M. S. Corporation. (2019). *CarSim Information [Online]*. Available: https://www.carsim.com/products/carsim/index.php

Ningyuan Guo received the M.S. degree in mechanical engineering from Kunming University of Science and Technology, Kunming, China, in 2018. He is currently pursuing the Ph.D. degree in mechanical engineering at Beijing Institute of Technology, Beijing, China. His research interests include vehicle dynamics control, distributed drive electric vehicles, power management of hybrid electric vehicles.

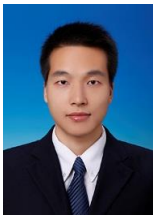
Basilio Lenzo (M'13) received the M.Sc. degree in mechanical engineering from the University of Pisa and the Scuola Superiore Sant'Anna, Pisa, Italy, in 2010, and the Ph.D. degree in robotics from the Scuola Superiore Sant'Anna in 2013. In 2015 and 2016 he was a Research Fellow with the University of Surrey, Guildford, U.K. He is a Senior Lecturer in automotive engineering with Sheffield Hallam University, Sheffield, U.K. His research interests include vehicle dynamics, simulation, control, and robotics.



Xudong Zhang received the M.S. degree in mechanical engineering from Beijing Institute of Technology, Beijing, China, in 2011 and the Ph.D. degree in mechanical engineering from Technical University of Berlin, Berlin, Germany, in 2017. Since 2017, he has been an Associate Professor with Beijing Institute of Technology. His main research interests include distributed drive electric vehicles, vehicle dynamics control, vehicle state estimation, torque allocation, and power management of hybrid electric vehicles.

Yuan Zou received his Ph.D. from Beijing Institute of Technology in 2005 and is currently a professor in School of Mechanical Engineering, Beijing Institute of Technology. He conducted research about ground vehicle propulsion modeling and optimal control at University of Michigan Ann Arbor and ETH Zurich. His research interests include modeling and control for electrified vehicle and transportation system.

Ruiqing Zhai received the B.S. degree from Central South University, Changsha, China, in 2017. He is currently working toward the master's degree with the Beijing Institute of Technology, Beijing, China. His current research focuses on power management, electrical system and vehicle dynamic control.



Tao Zhang received the B.S. degree from Yantai University, China, in 2012, and M.S. degree from Beijing University of Technology, China, in 2015. He is currently a Ph.D. candidate in Beijing Institute of Technology, China. His current research interests include the hardware design of vehicle controller and vehicle dynamics control.