Real-time Motion Planning Based on MPC With Obstacle Constraint Convexification For Autonomous Ground Vehicles

1st Zhengzheng Dong
College of Intelligence Science and Technology
National University of Defense Technology
Changsha, China
dongzhengzheng@nudt.edu.cn

2nd Xin Xu
College of Intelligence Science and Technology
National University of Defense Technology
Changsha, China
xuxin_mail@263.net

3rd Xinglong Zhang
College of Intelligence Science and Technology
National University of Defense Technology
Changsha, China
zhangxinglong18@nudt.edu.cn

4th Xing Zhou*
College of Intelligence Science and Technology
National University of Defense Technology
Changsha, China
zhouxing@nudt.edu.cn

5th Xiaohui Li
College of Intelligence Science and Technology
National University of Defense Technology
Changsha, China

6th Xueqing Liu
College of Intelligence Science and Technology
National University of Defense Technology
Changsha, China

***Abstract*—This paper proposes a real-time local motion planning algorithm for autonomous vehicles, which is based on model predictive control (MPC) subject to static and dynamic obstacles. In the proposed planner, the constraints of the road boundaries and obstacles are approximated by low-dimensional linear convex hulls. The proposed MPC optimizes the planning and control goals within one optimization problem using a dynamical model. The first element of the computed optimal decision sequence at any time instant can be applied directly to the vehicle, and an additional tracking controller is not required. The performance of the proposed approach is evaluated in both static and dynamic obstacle scenarios within the high-fidelity CarSim simulation environment. The results show that the proposed algorithm can compute the control actions in real-time, and the vehicle can drive safely and smoothly amid static and dynamic obstacles. Moreover, the proposed approach outperforms the potential field method in terms of computational complexity and planning cost, and the MPC-based planner with kinematics in the high driving speed cases.**

***Index Terms*—motion planning, model predictive control, obstacle avoidance, autonomous ground vehicles**

I. Introduction

Autonomous driving technology has received extensive attention in recent years. The motion planning of autonomous vehicles, which has been the focus of research in academia and industry, is one of the largest challenges in autonomous driving technology. An autonomous vehicle contains several modules, including motion planning module and control module. In respect of motion planning, how the vehicle can reach the destination safely and quickly have been addressed in [1] with collision avoidance. Because most traditional motion planning methods did not simultaneously consider the dynamics characteristics of the autonomous vehicle, the planned trajectory might not be directly executed by the control module, due to the possible inconsistency of planning and control. Moreover, as noted in [2], for the obstacle avoidance in motion planning, a good representation of feasible regions is vital. Our paper borrows the idea from [3] to convexify the feasible region with low-dimensional polyhedral constraints. Indeed, the obstacle region can be transformed into a series of convex constraints of the system variable. The existing methods often express the constraints of control quantity as hard constraints and the constraints of state quantity as soft constraints, see [4], [5]. However, soft constraints cannot ensure that vehicles will not collide with obstacles. In our study, the transformed obstacle constraints are hard convex ones that is included in the model predictive control (MPC) formulation. With linearization of the original dynamics, the motion planning problem can be established as a quadratic programming problem, which can be computed efficiently in real-time.

The main contributions of this paper are summarized as follows.

1) We propose a new obstacle constraint treatment under

* Corresponding author. The work is supported by the National Natural Science Foundation of China under Grant 61751311, 61825305, the National Key R&D Program of China 2018YFB1305105, and the Natural Science Foundation of Hunan Province Of China under Grant 2019JJ50738.

978-1-7281-8025-0/20/$31.00 ©2020 IEEE

the framework of model predictive control. The nonconvex feasible regions shaped by obstacles are convexified with the intersections of a number of linear constraints, being included in the model predictive control framework. The resulting optimization problem is easy to be solved with mature solvers, such as QuadProg.

2) We integrate the planning and control module into a single-layer optimization problem. Different from classic treatment, where the motion planning and control modules are considered in different layers, we integrate the planning and control into one optimization problem to solve. In this way, the inconsistency of the planning and control, which is recognized as a crucial problem in the two-layer structure, can be avoided in the proposed approach.

The rest of this paper is organized as follows. Related works are primarily reviewed in Section II. Section III proposes the local trajectory planning algorithm based on constrained MPC. Simulation experiments and results are depicted in Section IV, while Section V provides the conclusion and possible future directions.

II. Related works

There have been abundant contributions in local trajectory planning based on optimization method, see for instance [6]. In motion planning, the objective function of trajectory optimization includes safety, comfort, stability, and driving efficiency. The kinematic and dynamic constraints of the vehicle were embodied in the constraint function in [7] to ensure the executability of the trajectory. In [8], the motion planning problem of the intelligent vehicle was transformed into the optimization problem in the Frenet coordinate system. The method decoupled the problems in the horizontal and vertical directions by transforming the Cartesian coordinate system into the Frenet coordinate system and then computed the minimum jerk trajectory satisfying the constraints in the vertical and horizontal directions in real-time. In [9], a trajectory planning system based on optimal control theory, graph search, and topological analysis were proposed to generate the optimal trajectory in each planning time instant. In [10], based on the dynamic optimization, a unified framework was proposed. In [11], Baidu Apollo self-driving platform used the optimal control method for realizing motion planning. As a first step, the dynamic planning method found the initial rough path, and then it used the quadratic programming method to re-define the curve.

The trajectory planning based on MPC has also been widely studied. In [12], real-time obstacle avoidance based on MPC was realized in the static and dynamic environments, where vehicle shape was considered as a convex polygon with linear constraints and the obstacles were treated as points. In [4], a convex quadratic programming-based MPC was proposed, where potential field functions are used to avoid obstacles in a static and dynamic environment. In [13], two frameworks were proposed for the problem of tracking the reference trajectory with simultaneous obstacle avoidance. The first one was to directly calculate a nonlinear model predictive control. The other was a layered method, where the upper layer used a simple vehicle model for local trajectory planning, while the lower layer utilized a dynamic model for tracking control. In [14], a local trajectory planning and tracking control framework was proposed. First, the reference trajectory was smoothed by state space sampling, then several smooth and dynamic trajectories were generated based on the model prediction method. By designing the cost function, a safe and comfortable trajectory was selected for tracking control. The tracking controller consisted of feedback and feedforward terms for tracking the planned trajectory.

In the above-reviewed methods, the obstacle is treated by potential field function or by sampling. As a different perspective, a convex planning approach based on MPC was proposed in [3] where the shape of obstacles was considered to formulate the convex obstacle constraints, with which the problem of motion planning was transformed into a convex optimization problem. In [15] polygon constraints of obstacles were established with sensor data, resulting in linear convex polygon constraints to be solved in a convex optimization problem. The idea in [15] inspires our work. Specifically, we propose a real-time online motion planning method in the framework of model predictive control, with the obstacle constraints convexified by linear convex hulls. Different from [3], [15], the constraint convexification is based on the decomposition of the obstacle constraint into an intersection of many linear convex subsets.

III. Online motion planning algorithm based on MPC with constraint convexification

Considering the motion planning method of the vehicle dynamics model, we can plan a trajectory in line with vehicle motion. At the same time, the optimal control quantity can be used as the reference input of the vehicle. In this paper, firstly, the vehicle dynamics are modeled, and the planning and control methods based on the vehicle dynamics model are studied. In order to improve the calculation efficiency, the low dimension representation of convex polygon obstacles is used. The dynamic model can be expressed as follows:

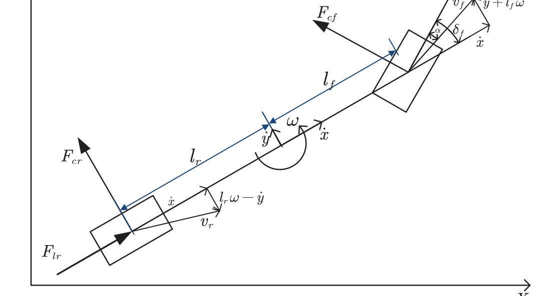

Fig. 1. Vehicle dynamic model.

$$
\begin{aligned}
&\ddot{x} = \dot{y}\omega + a_x \\
&m\ddot{y} = 2\left[C_{\alpha f}\left(\delta_f - \frac{\dot{y}+l_f\omega}{\dot{x}}\right) + C_{\alpha r}\frac{l_r\omega-\dot{y}}{\dot{x}}\right] - m\dot{x}\omega \\
&\dot{\varphi} = \omega \\
&I_z\dot{\omega} = 2\left[l_f C_{\alpha f}\left(\delta_f - \frac{\dot{y}+l_f\omega}{\dot{x}}\right) - l_r C_{\alpha r}\frac{l_r\omega-\dot{y}}{\dot{x}}\right] \\
&\dot{X} = \dot{x}\cos\varphi - \dot{y}\sin\varphi \\
&\dot{Y} = \dot{x}\sin\varphi + \dot{y}\cos\varphi
\end{aligned}
\tag{1}
$$

where $\dot{x}$ and $\dot{y}$ are the longitudinal and lateral velocities of the vehicle. φ and ω are the yaw angle and yaw rate of the vehicle. X and Y represent the position of the vehicle in global coordinate system. a_x, δ_f are the vehicle longitudinal acceleration and the front steering angle. l_f and l_r denotes the distances from the front and rear axles to the vehicle center of gravity. $C_{\alpha f}$, $C_{\alpha r}$ indicate the cornering stiffnesses of the front and rear tires. I_z, m stand the inertia moment and mass of the vehicle.

Let $\xi = (\dot{x}, \dot{y}, \varphi, \omega, X, Y,)$, $u = (a_x, \delta_f)$, and $\eta = (X, Y, \varphi)$. Then the dynamic model (1) can be rewritten as:

$$
\begin{cases} \dot{\xi}(t) = f(\xi(t), u(t)) \\ \eta(t) = C\xi(t), \end{cases} \tag{2}
$$

where

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

Note that, (2) is nonlinear. Successive linearization is used in the following to formulate a linear MPC problem. To proceed, assume $\hat{\xi}$ is the state variable obtained by invariably applying the control quantity $u(0)$, then

$$
\dot{\hat{\xi}}(t) = f\left(\hat{\xi}(t), \hat{u}(t)\right) \tag{3}
$$

where $\hat{u}(t) = u(0)$ for $t \geq 0$.

The time-varying linear model can be obtained via first-order Taylor expansion at current state and input $\left(\hat{\xi}(t), \hat{u}(t)\right)$:

$$
\begin{aligned}
\dot{\xi}(t) - \dot{\hat{\xi}}(t) =& A_0(t)\left(\xi(t) - \hat{\xi}(t)\right) \\
&+ B_0(t)\left(u(t) - \hat{u}(t)\right)
\end{aligned}
\tag{4}
$$

where $A_0(t) = \frac{\partial f(\xi,u)}{\partial \xi}|_{\hat{\xi}(t),u(0)}$, $B_0(t) = \frac{\partial f(\xi,u)}{\partial u}|_{\hat{\xi}(t),u(0)}$.

Define $\tilde{\xi} = \xi - \hat{\xi}$, $\tilde{u} = u - \hat{u}$, then one has

$$
\dot{\tilde{\xi}}(t) = A_0(t)\tilde{\xi}(t) + B_0(t)\tilde{u}(t) \tag{5}
$$

The sampled version of (5) can be obtained for later use, i.e.,

$$
\tilde{\xi}(t+1) = A_d(t)\tilde{\xi}(t) + B_d(t)\tilde{u}(t) \tag{6}
$$

where $A_d(t) = I + \Delta T A_0(t)$, $B_d(t) = \Delta T B_0(t)$, where ΔT is the sampling interval.

Expand (6) and combine it with (2) yielding

$$
\begin{aligned}
\xi(t+1) &= A_d(t)\xi(t) + B_d(t)u(t) + d(t) \\
\eta(t) &= C_d\xi(t)
\end{aligned}
\tag{7}
$$

where $d(t) = \hat{\xi}(t+1) - A_d(t)\hat{\xi}(t) - B_d(t)\hat{u}(t)$.

A. Obstacle constraint convexification

The unmanned ground vehicle may encounter static and dynamic obstacles in the process of movement. The obstacles can be road boundaries, convex and concave objects. As concave objects can be approximated by convex ones, for simplicity, in this paper we only consider road boundaries and convex obstacles. The distance between the vehicle and the obstacles is expressed by Euclidean distance, that is

$$
d(s, \varPhi) = \min_{s,z\in\varPhi} d_C(s, z) \tag{8}
$$

where $s = (X, Y)$, $\varPhi$ is the obstacle space, d_C is the distance in Cartesian coordinate system. In order to avoid collision between vehicles and obstacles in motion planning, constraints need to be met, i.e.,

$$
d(s, \varPhi) \geq d_{\min} \tag{9}
$$

where $d_{\min}$ is a safety threshold value. Note that, constraint (9) is non-convex. Solving the planning problem with (9) is difficult and the feasibility of the optimization problem might not be guaranteed. For this reason, we convexify the obstacle constraint following the same line with [3].

Let assume that any obstacle avoidance constraint $\varPhi$ can be decomposed into the intersection of several subsets $\varPhi = \cap_i \varPhi_i$, each subset can be expressed as $\varPhi_i := \{s : \phi_i(s) \geq 0\}$, where $\phi_i(s)$ is a smooth convex function about s. This means that there is a positive definite matrix H_i, which makes $\phi_i(s+v) + 2\phi_i(s) + \phi_i(s-v) \geq -v^T H_i^* v$, see [3]. With this conclusion, we can convexify almost all forms of obstacles. The convex form of the constraint is defined as: $\varPhi = \cap_i \psi_i$, where $\psi_i = \phi_i$, if ϕ_i is a convex set. In addition, if the complement of ϕ_i is a convex set, in this case $\phi_i(s)$ can be designed as a convex function. Then $\phi_i(s) \geq \phi_i(s_r) + \nabla\phi_i(s_r)(s - s_r)$, where s_r represents the reference point of the plan. Therefore, the convex constraint can be expressed as

$$
\psi_i := \{s : \phi_i(s_r) + \nabla\phi_i(s_r)(s - s_r) \geq 0\} \tag{10}
$$

With the approximating procedure described above, the original constraint can be transformed into convex ones readily being included in the motion planning optimization problems.

B. Motion planning based on constrained MPC

In the motion planning problem, the objective is to minimize the error between vehicle and reference trajectory and the increment of control amount without collision avoidance. At any discrete-time t, an online optimization problem is to be solved:

$$
\begin{aligned}
\min_{\Delta u} J =& \sum_{i=1}^{N_p} \|\eta(t+i) - \eta_{ref}(t+i)\|_Q^2 \\
&+ \sum_{i=0}^{N_c-1} \|\Delta u(t+i)\|_R^2
\end{aligned}
\tag{11a}
$$

$$
\begin{aligned}
s.t. \quad \eta(t+i+1) =& C_d A_d(t+i)\xi(t+i) \\
&+ C_d B_d(t+i)u(t+i) \\
&+ C_d d(t+i),\ i = 0, \cdots, N_p - 1
\end{aligned}
\tag{11b}
$$

$$u_{\min} \leq u\left(t+i\right) \leq u_{\max},\ i=0,\cdots,N_c-1 \tag{11c}$$

$$\varDelta u_{\min} \leq \varDelta u\left(t+i\right) \leq \varDelta u_{\max},\ i=0,\cdots,N_c-1 \tag{11d}$$

$$s(t+i) \in\ \varPhi(t+i),\ i=1,\cdots,N_p \tag{11e}$$

where $\Delta u(t+1) = u(t+1) - u(t)$, (11a) is the objective function, (11c) and (11d) are the constraint are on the control quantity and its increment, N_p and N_c are prediction horizon and control horizon, respectively. (11e) is the time-varying state constraint for collision avoidance. The main implementation steps of the online motion planning for autonomous driving based on MPC with constraint convexification are described in Algorithm 1.

Algorithm 1 Implementation steps of the proposed algorithm.

```
Require: set parameters Q, R, N_p, N_c, d_0 > 0; the reference points (X_r, Y_r, φ_r)_{i=1}^L;
for t = 1, ··· do
    for i = t, ··· , t + N_p do
        for each obstacle Φ do
            if the distance d(s(t), Φ) < d_0 then
                Convexify the constraint with low-dimension convex hulls with (10) and save;
            end if
        end for
    end for
    Compute Δu(t : t + N_c|t) via solving the planning problem (11);
    Apply u(t|t) = Δu(t|t) + u(t − 1) to the vehicle and update the system state.
end for
```

IV. SIMULATION RESULTS

In this section, the proposed algorithm is first validated in the Matlab environment, then tested in a joint Matlab-CarSim platforms test environment. The comparison with the potential field method is made. In the Matlab environment simulation, static obstacle avoidance on straight and curve road conditions, and dynamic obstacle avoidance are considered, while in the joint simulation tests, static obstacle avoidance on a straight road is tested. It is highlighted that, in the joint tests, all the calculations are done in Matlab and directly applied to the high-fidelity vehicle in CarSim. The solver in the CarSim solves the differential equations and updates the state to be used in the planning optimization algorithm. The optimization problem (11a) is solved with the QuadProg solver in the MATLAB 2019 environment. The simulation is performed in a Desktop with Intel i7-8700K CPU @ 3.7GHz and 16GB RAM, and Windows 10 operating system.

A. Obstacle avoidance on a straight road

We first conduct the static obstacle avoidance tests on a straight road in the Matlab environment. The reference trajectory used in the tests is collected from the GPS data sets recorded by a real vehicle Hongqi-EV5. In the simulation experiment, the reference trajectory is transformed from the GPS data sets under the global coordinate frame into the local coordinate system. Also, we consider static vehicles as obstacles to be avoided on the reference trajectory. The simulation results are shown in Fig. 2. The results show that the proposed approach can generate smooth trajectories and can realize collision avoidance.

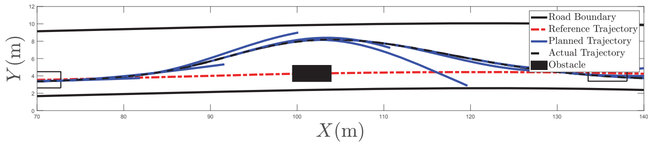

Fig. 2. The trajectory of the straight road: the black line represents the road boundaries, the red line represents the reference track, the blue line is the local track planned in real-time during the vehicle movement, and the dashed black line is the real trajectory of the vehicle.

To further observe the effectiveness of the proposed approach, the front wheel steering of the vehicle and computational time are displayed in Fig. 3. It can be seen that the computed control actions are smooth enough, and the calculation time is acceptable for real-time applications.

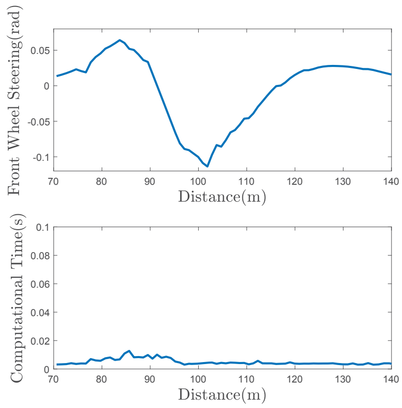

Fig. 3. The results on the straight road.

B. Obstacle avoidance on a curved road

There are two obstacles in this case, which are distributed in different positions on the curved road sections. The reference trajectory is also collected from GPS data sets. The experimental results obtained in this scenario are shown in Fig. 4-5. Fig. 5 is a partially enlarged view of the obstacle avoidance in Fig. 4, from which one can see that the vehicle can successfully avoid the collision in the presence of obstacles on the curved road.

Fig. 6 depicts the curve of the front wheel steering of the vehicle and the calculation time in this scene. The results show that the obtained control actions are smooth. Also, the average computational time is acceptable for real-time uses.

C. Obstacle avoidance with dynamic obstacles

Consider the case that an obstacle moves along the reference trajectory with a constant speed. The ego vehicle behind it has to overtake it. The initial position of the dynamic polygonal obstacle is (102.18, 4.28), and the speed of the obstacle is set

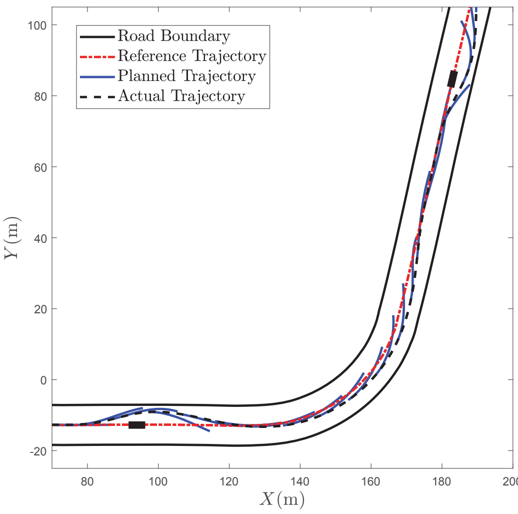

Fig. 4. The planned results of the vehicle on the curved road.

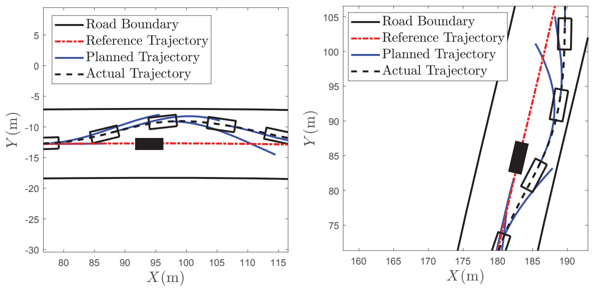

Fig. 5. Partial results on collision avoidance of the vehicle on the curved road.

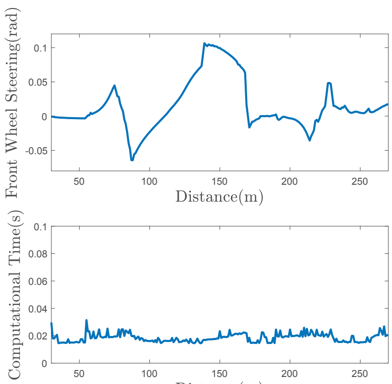

Fig. 6. The results on the curved road.

as 3m/s. The vehicle speed is 10m/s. The simulation results are shown in Fig. 7-8.

Fig. 7 shows the entire process of the overtaking. It can be seen that, when the obstacle enters the observation range of the vehicle, which corresponds to the position $l = 90$ m, the constraint of the obstacle is calculated and included to the real-time planner, and the overtaking action is performed. The vehicle can safely accomplish the overtaking action at the

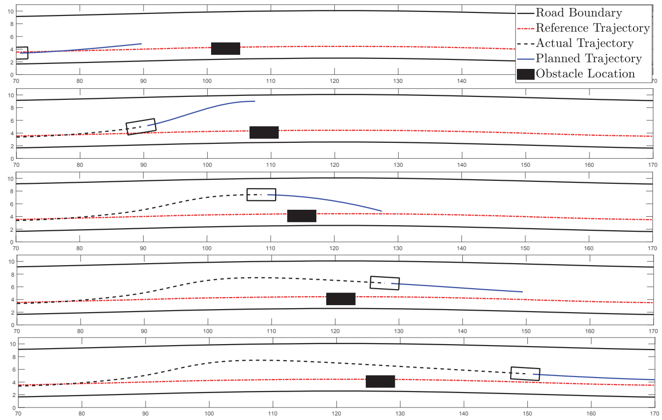

Fig. 7. The planned results of the vehicle in dynamic environment

position $l = 150$ m. The vehicle successfully recovers back to the trajectory track after the overtaking procedure.

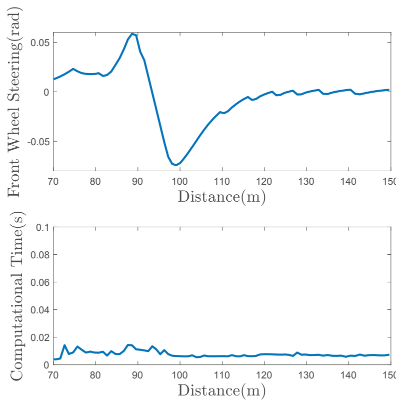

Fig. 8. The results under dynamic obstacles

D. Simulation experiments in the CarSim environment

With the proposed approach, we have carried out simulation experiments on the CarSim platform and compared the approach with the potential field method and the MPC method using kinematics. The experimental scene of CarSim is shown in Fig. 9-10, where an overtaking task is considered.

1) Comparison with the potential field method: In the potential field algorithm, the cost function and the parameters included are set the same as the proposed approach, while the obstacle constraints are treated with the potential field function.

The prediction horizon was set as $N_p = 20$ in our method. For a fair comparison, a same value of the prediction horizon is suggested for the potential field method.

First, as it can be seen in Fig. 10, when the obstacle is detected, the ego vehicle with the proposed algorithm implemented can realize collision avoidance and overtaking in time. To show the comparison results, the real trajectory generated by the vehicle, control actions and the computational

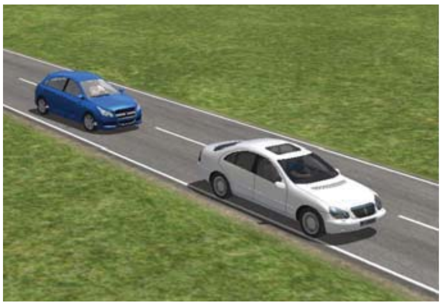

Fig. 9. An overtaking scene in the CarSim environment: the blue vehicle is the ego vehicle, while the white one is the obstacle.

Fig. 10. Obstacle avoidance in CarSim platform.

time of the both methods are collected and shown in Fig. 11-12.

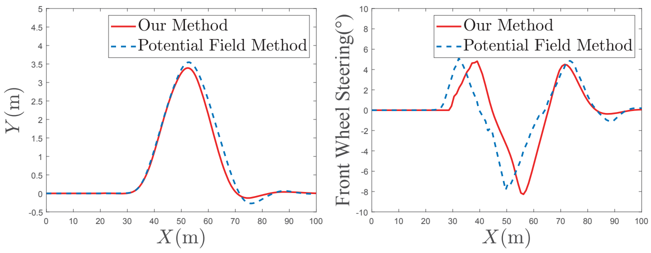

Fig. 11. The resulting trajectory and control obtained with the proposed and potential field approaches in the CarSim environment.

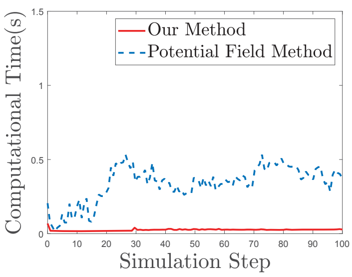

Fig. 12. Computational time along the simulation step with the proposed and potential field approaches.

Fig. 11 indicates that both methods can realize obstacle avoidance. It is worth noting that, the potential field method needs to adjust the penalty matrix of potential field function, while in our method, the safety distance between the vehicle and the obstacle is set as the hard constraint in order to ensure that there is no collision with the obstacle. Compared with adjusting the weights of obstacle avoidance, adjusting the threshold of the safety distance is more intuitive.

From Fig. 12, it can be seen that our method has better real-time performance. The detailed statistical comparison of the computational time and cost of the two methods are shown in Table I, where the average cost J is $J = \sum_{i=1}^{N_{sim}} \left[(x_i - x_i^r)^T Q (x_i - x_i^r) + \Delta u_i^T R \Delta u_i \right]$ and the overall simulation length is $N_{sim} = 100$. It straight-forwardly illustrates that the average cost obtained with our approach is smaller than that with the potential field approach.

TABLE I
COMPARISON WITH POTENTIAL FIELD

method	average computational time	average cost
Potential field ($N_p = 40$)	4.4831s	428.7996
Potential field ($N_p = 30$)	1.6890s	425.4505
Potential field ($N_p = 20$)	0.3375s	420.4226
Ours($N_p = 40$)	0.0246s	248.4568
Ours($N_p = 30$)	0.0188s	287.0313
Ours($N_p = 20$)	0.0127s	241.1388

2) Comparison with the MPC method using kinematics: The selection of model can directly affect the effect of a model predictive controller. When the vehicle speed is relatively low, the MPC with kinematics has satisfying effects. However, Unmanned Ground Vehicles often run in the traffic environment at a relatively high speed. Therefore, it is necessary to consider the vehicle dynamic model.

We compared the proposed approach with the MPC method using kinematics at the speed of $20m/s$. In the MPC with kinematics method, the cost function and the parameters are the same with the proposed approach. The kinematic model is as follows:

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \cos\varphi \\ \sin\varphi \\ \tan\delta_f / l \end{bmatrix} v \tag{12}$$

where X and Y represent the position of the vehicle in global coordinate system. φ is the vehicle's yaw angle. v, δ_f are the vehicle's speed and the front steering angle. l is the distance from the front axle center to the rear axle center.

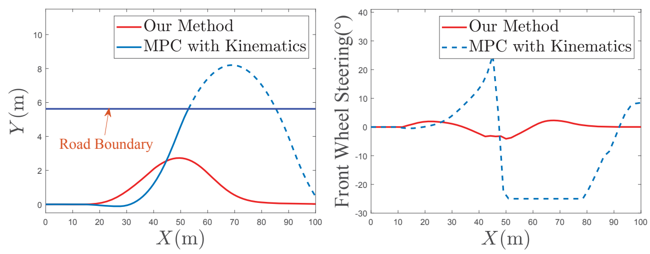

Fig. 13. The resulting trajectory and control obtained with the proposed approach and the MPC method using kinematics in the CarSim environment.

We also linearized the kinematic model to formulate a linear MPC problem and the treatment of obstacle constraints is the same as our method. The simulation results are shown in Fig. 13.

From Fig. 13, it can be seen that the proposed approach can complete the overtaking, but the MPC with kinematics method runs beyond the road boundaries. In the process of motion planning and control, the control quantity based on the MPC with kinematics method are not smooth enough and hard to be applied directly to the vehicle.

V. CONCLUSIONS

In this paper, a real-time online local trajectory planning algorithm based on MPC with dynamical vehicular model has been proposed for unmanned grounded vehicles. Besides targeting at the global reference trajectory, the proposed approach also has to consider collision avoidance with static and dynamic obstacles. The main characteristics of the proposed approach are that: the nonconvex obstacle constraints have been approximated by a set of convex hulls and included into the online optimization problem; also, the motion planning and control have been integrated as one optimization problem to ensure consistency between trajectory planning and tracking. The simulation studies in different scenarios have been conducted in the high-fidelity CarSim environment. The results show that the proposed algorithm is effective for enforcing smooth trajectory planning and tracking, and can successfully avoid collisions amid static and dynamic obstacles. Compared with the potential field method and the MPC method using kinematics, the proposed approach has better real-time efficiency than the potential field method, and is more suitable for high-speed situations than the MPC with kinematics method.

REFERENCES

[1] S. Li, Z. Li, Z. Yu, B. Zhang, and N. Zhang, "Dynamic Trajectory Planning and Tracking for Autonomous Vehicle With Obstacle Avoidance Based on Model Predictive Control," IEEE Access, vol. 7, pp. 132074–132086, 2019.

[2] S. Lai, M. Lan, and B. M. Chen, "Model Predictive Local Motion Planning With Boundary State Constrained Primitives," IEEE Robot. Autom. Lett., vol. 4, no. 4, pp. 3577–3584, Oct. 2019.

[3] C. Liu, C.-Y. Lin, and M. Tomizuka, "The Convex Feasible Set Algorithm for Real Time Optimization in Motion Planning," SIAM J. Control Optim., vol. 56, no. 4, pp. 2712–2733, Jan. 2018.

[4] Houjie Jiang, Zhuping Wang, Qijun Chen, and Jin Zhu, "Obstacle avoidance of autonomous vehicles with CQP-based model predictive control," in 2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Budapest, Hungary, 2016, pp. 001668–001673.

[5] P. Falcone, F. Borrelli, H. E. Tseng, J. Asgari, and D. Hrovat, "A hierarchical Model Predictive Control framework for autonomous ground vehicles," in 2008 American Control Conference, Seattle, WA, 2008, pp. 3719–3724.

[6] T. M. Howard and A. Kelly, "Optimal Rough Terrain Trajectory Generation for Wheeled Mobile Robots," Int. J. Robot. Res., vol. 26, no. 2, pp. 141–166, Feb. 2007.

[7] X. Qian, I. Navarro, A. de La Fortelle, and F. Moutarde, "Motion planning for urban autonomous driving using Bézier curves and MPC," in 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), Rio de Janeiro, Brazil, 2016, pp. 826–833.

[8] M. Werling, S. Kammel, J. Ziegler, and L. Gröll, "Optimal trajectories for time-critical street scenarios using discretized terminal manifolds," Int. J. Robot. Res., vol. 31, no. 3, pp. 346–359, 2012.

[9] T. Gu, "Improved trajectory planning for on-road self-driving vehicles via combined graph search, optimization and topology analysis," PhD Thesis, Carnegie Mellon University, 2017.

[10] B. Li and Z. Shao, "A unified motion planning method for parking an autonomous vehicle in the presence of irregularly placed obstacles," Knowl.-Based Syst., vol. 86, pp. 11–20, Sep. 2015.

[11] H. Fan et al., "Baidu Apollo EM Motion Planner," ArXiv180708048 Cs, Jul. 2018.

[12] Z. Wang, G. Li, H. Jiang, Q. Chen, and H. Zhang, "Collision-Free Navigation of Autonomous Vehicles Using Convex Quadratic Programming-Based Model Predictive Control," IEEEASME Trans. Mechatron., vol. 23, no. 3, pp. 1103–1113.

[13] Y. Gao, T. Lin, F. Borrelli, E. Tseng, and D. Hrovat, "Predictive Control of Autonomous Ground Vehicles With Obstacle Avoidance on Slippery Roads," in ASME 2010 Dynamic Systems and Control Conference, Volume 1, Cambridge, Massachusetts, USA, 2010, pp. 265–272.

[14] X. Li, Z. Sun, D. Cao, D. Liu, and H. He, "Development of a new integrated local trajectory planning and tracking control framework for autonomous ground vehicles," Mech. Syst. Signal Process., vol. 87, pp. 118–137, Mar. 2017.

[15] J. Ziegler, P. Bender, T. Dang, and C. Stiller, "Trajectory planning for Bertha x2014; A local, continuous method," in 2014 IEEE Intelligent Vehicles Symposium Proceedings, MI, USA, 2014, pp. 450–457.