

Real-Time Parallel Trajectory Optimization with Spatiotemporal Safety Constraints for Autonomous Driving in Congested Traffic

Lei Zheng, Rui Yang, Zengqi Peng, Haichao Liu, Michael Yu Wang, *Fellow, IEEE*, and Jun Ma

Abstract—Multi-modal behaviors exhibited by surrounding vehicles (SVs) can typically lead to traffic congestion and reduce the travel efficiency of autonomous vehicles (AVs) in dense traffic. This paper proposes a real-time parallel trajectory optimization method for the AV to achieve high travel efficiency in dynamic and congested environments. A spatiotemporal safety module is developed to facilitate the safe interaction between the AV and SVs in the presence of trajectory prediction errors resulting from the multi-modal behaviors of the SVs. By leveraging multiple shooting and constraint transcription, we transform the trajectory optimization problem into a nonlinear programming problem, which allows for the use of optimization solvers and parallel computing techniques to generate multiple feasible trajectories in parallel. Subsequently, these spatiotemporal trajectories are fed into a multi-objective evaluation module considering both safety and efficiency objectives, such that the optimal feasible trajectory corresponding to the optimal target lane can be selected. The proposed framework is validated through simulations in a dense and congested driving scenario with multiple uncertain SVs. The results demonstrate that our method enables the AV to safely navigate through a dense and congested traffic scenario while achieving high travel efficiency and task accuracy in real time.

I. INTRODUCTION

Autonomous vehicles (AVs) have the potential to significantly enhance driving efficiency. For instance, a study by the Texas A&M Transportation Institute found that the average commuters in the United States spent 54 hours stuck in traffic, costing them \$1,008 in wasted time and fuel in 2019 [1]. AVs could reduce these costs and improve the overall efficiency of the transportation system. Decades of research and industry development in the field of autonomous driving have led to the applications of AVs in various aspects of our lives, including urban transportation [2], delivery services [3], and mining [4], etc. Despite the advancements in autonomous driving technologies, safety remains a critical concern, particularly in dynamic urban traffic environments [5]. One of the key underlying factors to this concern is the multi-modal behaviors of surrounding vehicles (SVs), such as

This work was supported by the Project of Hetao Shenzhen-Hong Kong Science and Technology Innovation Cooperation Zone under Grant HZQB-KCZYB-2020083.

Lei Zheng, Rui Yang, Zengqi Peng, and Haichao Liu are with the Robotics and Autonomous Systems Thrust, The Hong Kong University of Science and Technology (Guangzhou), Guangzhou, China.

Michael Yu Wang is with the Department of Mechanical and Aerospace Engineering, Monash University, Clayton, Australia.

Jun Ma is with the Robotics and Autonomous Systems Thrust, The Hong Kong University of Science and Technology (Guangzhou), Guangzhou, China, also with the Division of Emerging Interdisciplinary Areas, The Hong Kong University of Science and Technology, Hong Kong SAR, China, and also with the HKUST Shenzhen-Hong Kong Collaborative Innovation Research Institute, Futian, Shenzhen, China.

All correspondence should be sent to Jun Ma (e-mail: jun.ma@ust.hk).

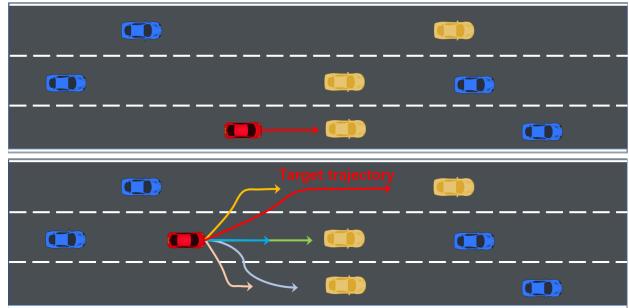


Fig. 1. Depiction of a three-lane driving scenario showing the red EV, yellow perceived SVs, and unperceived blue SVs. In the top subfigure, the EV is impeded by its front vehicles, while the bottom subfigure shows that with multiple parallel optimized trajectories, the EV can proactively change lanes to escape the congested scenario.

sudden deceleration and lane changes. These behaviors are difficult to predict, which significantly influence the motion of ego vehicles (EVs) and pose a safety threat to the EV. Besides, the multi-modal behaviors of SVs can lead to traffic congestion in dense traffic scenarios, which compromises the driving efficiency of the EV.

To overcome these challenges, the EV must be capable of rapidly replanning in response to the uncertain motion of SVs, ensuring split-second reaction to potential threats [6]. This requires computationally efficient trajectory optimization methods that account for the motion uncertainties of SVs. Additionally, the EV must proactively change lanes to avoid being impeded and reduce travel time, thereby improving driving efficiency, as illustrated in Fig. 1.

To enable efficient replanning in traffic flow, several works have focused on decoupling longitudinal and lateral motion to generate a smooth and feasible trajectory for the EV [7], [8], [9]. Although these works transform the motion of the EV into two independent one-dimensional movements to reduce planning complexity, this decoupling structure may result in a lack of coordination between the longitudinal and lateral movements of the EV, potentially leading to unsafe driving behaviors in sudden lane changes or abrupt braking [10]. To address this issue, researchers have developed hierarchical path-speed decomposition motion planning methods that enable EVs to interact safely with other vehicles in dynamic environments [11], [12], [13]. These methods involve planning a path to avoid static obstacles and optimizing a speed profile to avoid dynamic obstacles [14]. For example, in [13], an optimal path is selected among multiple candidate polynomial paths obtained from state sampling

methods, and an optimal acceleration profile is generated to obtain a velocity curve based on the responsibility sensitive safety model [15]. However, the path generated by the sampling module may not accurately reflect spatiotemporal information as it is not optimized for spatiotemporal safety constraints. This can impact the quality of the solution for the subsequent speed optimization module. Additionally, these methods cannot fully exploit the actuator potential of the nonlinear EV with nonholonomic constraints, resulting in suboptimal control policies.

On the other hand, optimal control methods, such as Model Predictive Control (MPC) [16], have the ability to incorporate system models and constraints while anticipating future states for autonomous driving [17], [18], [19]. In [20], the safety term is encoded into the objective function of MPC as a distance term to obstacles to avoid a potential collision and realize aggressive lane change maneuvers. To proactively avoid collisions, the control barrier function [21] has been utilized to enforce the safety of the EV when overtaking the front vehicle in the MPC framework [17]. Despite the advantages of optimal control methods in handling constraints, the computationally burdensome optimization process, particularly the inversion of the Hessian matrix, remains a limiting factor for real-time planning within typical planning horizons of 5 to 10 seconds in autonomous driving [22], [23]. To improve the computational efficiency towards trajectory optimization problems for autonomous driving, researchers have turned attention to the alternating direction method of multipliers (ADMM) [24] to reduce the computational burden of optimization processes [25], [26]. The ADMM splits the optimization problem into several subproblems that can be efficiently solved within the optimal control framework. While existing works often rely on a predefined motion model without uncertainties for SVs [25], [26], these approaches can lead to safety concerns in dynamic and congested traffic flow due to model mismatches and uncertain multi-modal behaviors of SVs. Considering uncertain behaviors and computation efficiency, an efficient multi-modal MPC approach based on the alternating minimization method has been developed for safe and efficient interaction with uncertain SVs for the EV [27]. It has shown promise in parallel optimization of multiple trajectories in a traffic flow with multi-modal behaviors exhibited by the SVs, but it does not explicitly address driving consistency when evaluating candidate trajectories. This could lead to frequent lane changes in dense and congested traffic, which may compromise driving comfort.

In this paper, an efficient trajectory optimization framework is proposed such that the EV is able to navigate through dense and congested traffic safely and efficiently, as illustrated in Fig. 2. To facilitate efficient driving, multiple nominal trajectories are optimized simultaneously using multi-threading and direct multiple-shooting, with each trajectory corresponding to a unique target lane. Then, the optimal trajectory is evaluated and selected for the EV to execute.

The main contributions of this work are as follows: First, we propose a computationally efficient approach that

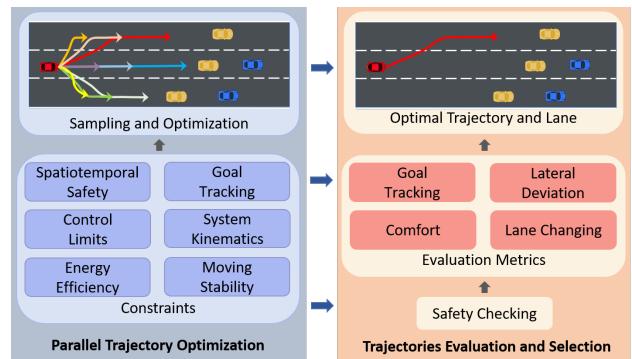


Fig. 2. The proposed optimization framework. The left module generates multiple candidate trajectories in parallel, with each one corresponding to a driving lane, using CPU multi-threading systems in real time. The right module evaluates the generated candidate trajectories and selects the optimal trajectory and lane based on various metrics.

leverages spatiotemporal information between the EV and SVs and employs a multi-threading and multiple-shooting technique for parallel trajectory optimization. This allows for the generation of multiple safe and feasible energy-efficient candidate trajectories in real time while striking a balance between performance and safety. Second, our approach prioritizes driving consistency as a crucial factor in trajectory evaluation, resulting in a smooth driving behavior that enhances driving continuity and stability, and this reduces the likelihood of abrupt lane changes. Finally, the proposed fully parallel implementation based on C++ and ROS2 achieves real-time performance in a dense and congested traffic flow.

The rest of this paper is organized as follows. The problem statement is presented in Section II. The proposed methodology is described in Section III. The numerical simulation of the proposed algorithm on an autonomous vehicle system is shown in Section IV. Finally, the conclusion is drawn in Section V.

II. PROBLEM STATEMENT

In this section, we consider a three-lane dense and congested drive scenario, as illustrated in Fig. 3. The EV is represented by a nonlinear kinematic bicycle model [28]. The state vector of the ego vehicle is defined as follows:

$$\mathbf{x} = [p_x, p_y, \theta, v, \omega]^T \in \mathcal{X}, \quad (1)$$

where p_x and p_y denote the longitudinal and lateral positions in the global coordinate, respectively; θ , v , and ω denote the heading angle, velocity, and yaw rate, respectively. The control input vector to the EV is defined as $\mathbf{u} = [a, \dot{\omega}]^T \in \mathcal{U}$, where $\omega = \frac{v \tan(\delta)}{L}$, L is the wheelbase, a is the acceleration, and δ is the steering angle of the front wheels. The nonlinear kinematic bicycle model can be formulated as follows:

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{p}_x \\ \dot{p}_y \\ \dot{\theta} \\ \dot{v} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} v \cos(\theta) \\ v \sin(\theta) \\ \omega \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} a \\ \dot{\omega} \end{bmatrix}. \quad (2)$$

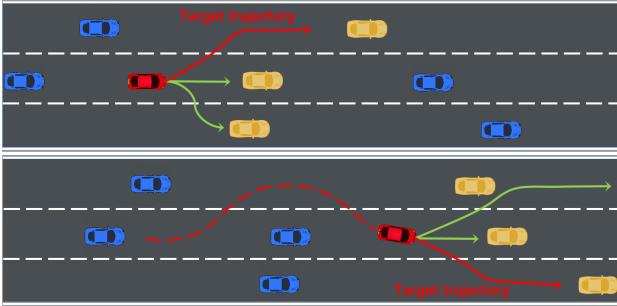


Fig. 3. Illustration of the motion of a red EV in a dynamic congested scenario with three lanes. Yellow and blue vehicles represent perceived and unperceived vehicles, respectively. The dashed red line shows the EV's executed trajectory, while the solid red line represents the target trajectory. Other lines with an arrow denote unselected candidate trajectories of the EV.

In this dense and congested driving scenario, the EV may be impeded by its front vehicles with uncertain behaviors (e.g., velocity keeping, acceleration, deceleration), resulting in poor driving efficiency and compromised safety. Besides, frequent lane changes in such scenarios may also destabilize the driving experience and threaten the EV's passengers' safety. To make this problem tractable, we make the following assumptions:

Assumption 1: (Safety Responsibility) [5] When the two vehicles are driving in the same direction, if the rear vehicle c_r hits the front vehicle c_f from behind, then the rear vehicle c_r is responsible for the accident.

Assumption 2: (Driving Maneuver) [5] A vehicle will respond appropriately to dangerous situations and not hit another vehicle from behind as long as the front vehicle does not brake stronger than the maximum deceleration of the vehicle behind.

Assumption 3: (Perception Ability) An EV can obtain information about the position and speed of the three nearest SVs.

To tackle these challenges, we aim to design an efficient trajectory planning framework that enables the EV to safely interact with SVs while keeping high travel efficiency. The framework generates optimized trajectories in parallel, which closely follow the centerline of different target lanes. Each trajectory $\tau(t) \in \mathcal{T}$ corresponds to a specific target lane $\xi \in \Xi$, given as follows:

$$\mathcal{T} := \left\{ \mathbf{u}^{(j)}(t), \mathbf{x}^{(j)}(t) \right\}_{j=1}^{N_t}, \quad 0 \leq t \leq T, \quad (3)$$

$$\Xi := \{lane1, lane2, lane3\}, \quad (4)$$

where T and N_t represent the optimization horizon and the number of trajectories, respectively; Ξ denote the set of driving lanes, where each target lane represents either a lane-keeping maneuver, a left-lane-change, or a right-lane-change maneuver for the EV, as depicted in Fig. 3.

The proposed parallel trajectory optimization framework operates over a planning horizon T , generating trajectory τ for each target lane $\xi \in \Xi$. The goal is to determine the

optimal lane ξ^* and trajectory τ^* in real time to achieve the following objectives:

- (O1) *Efficiency*: The proposed framework can parallel optimize multiple reference trajectories along the center lane in real time, enabling the EV to navigate through congested traffic with high travel efficiency.
- (O2) *Safety*: The proposed strategy can leverage spatiotemporal information to ensure safe navigation through dynamic congested traffics while maintaining a safe distance from other vehicles.
- (O3) *Consistency and Stability*: The proposed approach should ensure spatiotemporal consistency to avoid frequent lane changes, reducing the likelihood of abrupt maneuvers that may affect driving stability.

III. METHODOLOGY

In this section, we first describe the goal-oriented movement of the EV in Section III-A, which involves designing appropriate costs to achieve desired tasks. Following up on our previous work [29], we introduce the spatiotemporal safety module for collisions avoidance in Section III-B. Subsequently, we present the parallel sampling and trajectory optimization method in Section III-C. Lastly, the decision-making module that aims to select the optimal trajectory and the target lane is presented in Section III-D.

A. Goal-Oriented Movement

The goal-oriented movement module is responsible for generating trajectories aligned with the target lanes' centerline. To achieve this, we sample several points along each target lane at the end of the planning horizon and use them as constraints to generate candidate trajectories that align with the target lanes' centerline, as illustrated by the arrow of each trajectory in the left module of Fig. 2.

To ensure that each optimized trajectory remains close to the centerline of the lane, we enforce the final lateral position of each trajectory to match the lateral position of corresponding sampling points. This is accomplished by setting the terminal cost to ensure stability in the quasi-infinite horizon approach [30]. Likewise, we can set the EV's final target yaw angle and yaw rate state to tiny values to endow the EV with a stable driving mode. Therefore, we introduce a terminal cost to enforce the final state of each trajectory in the vicinity of the desired state as follows:

$$\begin{cases} C_T = \phi(\mathbf{x}(t+T)) = (\iota_T \mathbf{x}(t+T))^T \mathbf{Q}_T (\iota_T \mathbf{x}(t+T)), \\ C_T > \int_0^T \mathcal{L}(\mathbf{x}(t), \mathbf{u}(t)) dt, \end{cases} \quad (5)$$

where $\mathbf{x}(t+T)$ is the terminal state vector; \mathbf{Q}_T is a weighting matrix; $\iota_T = diag(0, 1, 1, 0, 1)$ extracts the lateral position, heading angle, and yaw rate from the state vector; $\mathcal{L}(\mathbf{x}(t), \mathbf{u}(t))$ represents the running cost at time t . \mathbf{Q}_T is set to be a large value to ensure the terminal cost is larger than the running cost [31].

To further enhance driving task performance, we introduce a goal-tracking cost term for each trajectory as follows:

$$C_m(t) = (\iota_m(\mathbf{x}(t) - \mathbf{x}_d(t)))^T \mathbf{Q}_m \iota_m(\mathbf{x}(t) - \mathbf{x}_d(t)), \quad (6)$$

where $\mathbf{x}_d(t)$ denotes the desired state vector at time t that the EV aims to reach to accomplish its task; ι_m extracts the particular state vector of the EV from the state vector \mathbf{x} . For instance, in a cruise task, we set $\iota_m = \text{diag}(0, 1, 0, 1, 0)$ to extract the lateral position and velocity from the state vector \mathbf{x} . This enables the EV to cruise at a desired speed and drive close to the centerline of the lane.

B. Spatiotemporal Safety Module

To realize safe interaction between the EV and the i th SV, we need to keep the EV's state within a safe region \mathcal{S} defined as follows:

$$\mathcal{S} := \{\mathbf{x}(t) \in \mathcal{X} | h(\mathbf{x}(t), \mathbf{O}_i(t)) \geq 0\}, \quad (7)$$

where $\mathbf{O}_i = [\mathbf{O}_{p,i}, \mathbf{O}_{v,i}]^T$; $\mathbf{O}_{p,i} = [o_{x,i}, o_{y,i}]$ and $\mathbf{O}_{v,i} = [o_{v_x,i}, o_{v_y,i}]$ denote the position and velocity vectors of the i th SV, which evolves according to an uncertain system model $f^{SV}(\cdot)$; t denotes current time; $h : \mathbb{R}^n \rightarrow \mathbb{R}$, is a continuously differentiable safety barrier function that encodes state constraints as follows:

$$h(\mathbf{x}(t), \mathbf{O}_i(t)) = \frac{(p_x(t) - o_{x,i})^2}{a^2} + \frac{(p_y(t) - o_{y,i})^2}{b^2} - 1, \quad (8)$$

where a and b represent the major and minor axis lengths of a safe ellipse encompassing the form of the EV, respectively.

Since the laws of motion govern the SVs' positions, we heuristically infer that their nominal trajectory over the planning horizon is based on a constant velocity model in a planning horizon as follows:

$$\bar{f}^{SV}(\mathbf{O}_i(t)) = \begin{bmatrix} \dot{o}_{x,i} \\ \dot{o}_{y,i} \end{bmatrix} = \begin{bmatrix} \dot{o}_{v_x,i} \\ \dot{o}_{v_y,i} \end{bmatrix}. \quad (9)$$

Nevertheless, the multi-modal uncertain behaviors of SV are hard to model accurately, especially with a simple constant velocity model, which may lead to more significant errors as the prediction horizon becomes longer. In this context, considering safety equally in each time step during the planning horizon can lead to overly conservative actions that compromise driving efficiency. To compensate for these errors and strike a balance between safety and performance in trajectory planning, we introduce a time-varying discount weight $w_i(t)$ in the safety constraint, given by:

$$w_i(t) = \lambda_i \exp\left(\frac{-t}{\gamma}\right), \quad (10)$$

where γ is a constant discount factor, and λ_i is a factor representing the safety weight regarding the i th SV.

With this discount weight $w_i(t)$ and the barrier function (8), we introduce a spatiotemporal safety cost $C_s(t)$ as follows:

$$C_s(t) = \sum_{i=1}^M w_i(t) H(\mathbf{x}(t), \mathbf{O}_i(t)), \quad (11)$$

where M denotes the number of the perceived SVs, and $H(\mathbf{x}(t), \mathbf{O}_i(t))$ is a safety measurement function based on

the barrier function (8), as follows:

$$H(t) =$$

$$\frac{1}{\eta + h(\mathbf{x}(t), \mathbf{O}_i(t))} \left(1 - \frac{h(\mathbf{x}(t), \mathbf{O}_i(t)) - c}{\varepsilon + \sqrt{(h(\mathbf{x}(t), \mathbf{O}_i(t)) - c)^2}} \right), \quad (12)$$

where $\eta \in \mathbb{R}^+$ denotes a scale factor, and $\varepsilon \in \mathbb{R}^+$ is a small regularization constant facilitating numerical stability.

Remark 1: Although the spatiotemporal cost terms C_s are technically soft, they act like hard constraints as the penalty is obtained immediately after the constraint boundary is violated. However, unlike hard constraints, they have the advantage that the importance of different safety requirements regarding different time steps in a horizon can be adjusted by setting different weighting matrices $w_i(t)$.

C. Parallel Trajectory Optimization Framework

We design an optimal control framework to model the parallel trajectory optimization problem. Then, we transform the parallel optimal control problem into a parallel nonlinear programming (NLP) problem to optimize the nominal sampled trajectories based on the direct multiple shooting method [32]. Specifically, the optimal control for the j th trajectory takes the following form:

$$\min_{\mathbf{u}^{(j)}(t), \mathbf{x}^{(j)}(t)} \int_0^T \mathcal{L}^{(j)}(\mathbf{x}(t), \mathbf{u}(t)) dt + \phi^{(j)}(\mathbf{x}(T)) \quad (13a)$$

$$\text{s.t. } \mathbf{x}(0) = \mathbf{x}_0, \mathbf{O}_i(0) = \mathbf{O}_{i,0}, \quad (13b)$$

$$\dot{\mathbf{x}}(t) = f^{EV}(\mathbf{x}, \mathbf{u}), \quad (13c)$$

$$\dot{\mathbf{O}}_i(t) = \bar{f}^{SV}(\mathbf{O}_i), \quad (13d)$$

$$\mathbf{x}_{min} \preceq \mathbf{x}^{(j)}(t) \preceq \mathbf{x}_{max}, \quad (13e)$$

$$\mathbf{u}_{min} \preceq \mathbf{u}^{(j)}(t) \preceq \mathbf{u}_{max}, \quad (13f)$$

$$(\mathbf{u}^{(j)}(t), \mathbf{x}^{(j)}(t)) \in \mathcal{U} \times \mathcal{X}, \quad (13g)$$

$$\forall t \in [0, T],$$

where \mathbf{x}_0 and $\mathbf{O}_{i,0}$ denote the initial state of the EV and the i th SV, respectively; (13c) is the nonlinear motion model of the EV defined in (2); \mathbf{x}_{min} and \mathbf{x}_{max} denote the minimum and maximum state constraints of the EV, respectively; \mathbf{u}_{min} and \mathbf{u}_{max} denote the minimum and maximum control inputs of the EV, respectively; $\phi^{(j)}(\mathbf{x}(T)) = C_T^{(j)}$ represents the terminal cost for the j th trajectory. The running cost of the j th trajectory at time t is denoted by $\mathcal{L}^{(j)}$ in the form:

$$\mathcal{L}^{(j)}(\mathbf{x}(t), \mathbf{u}(t)) = C_m^{(j)}(t) + C_s^{(j)}(t) + \|\mathbf{u}^{(j)}(t)\|_{\mathbf{R}}^2, \quad (14)$$

where $C_m^{(j)}(t)$, $C_s^{(j)}(t)$, and $\|\mathbf{u}^{(j)}(t)\|_{\mathbf{R}}^2$ denote the goal-tracking cost, spatiotemporal safety cost, and energy consumption cost for the j th trajectory, respectively; \mathbf{R} is a positive semi-definite diagonal matrix determining energy efficiency.

We further reformulate the initial optimal control problem (13a)-(13g) into a multiple shooting-based constrained NLP problem to efficiently and accurately handle nonlinear EV's kinematics (2) to achieve efficient numerical solutions for parallel trajectory optimization [33]. The fundamental idea

of direct multiple shooting is to break the trajectory into N shorter segments. By doing this, the overall trajectory optimization problem can be transformed into N smaller shooting intervals that span the optimization horizon T [32]. Then, continuity constraints between two shooting intervals are enforced as follows:

$$\bar{\mathbf{x}}_{k+1} := f(\mathbf{x}_k, \mathbf{u}_k) \ominus \mathbf{x}_{k+1} = \mathbf{0}, \quad (15)$$

where \ominus denotes the difference operator of the state manifold that is needed to optimize over manifolds [34]; $f(\mathbf{x}_k, \mathbf{u}_k)$ denotes the simulation of the nonlinear kinematics (2) over one interval, starting from state \mathbf{x}_k with control input \mathbf{u}_k . In this study, we utilize 4th-order Runge-Kutta integration with a sampling time of $T_s = 100$ ms to discretize the original system (2). The resulting NLP problem, which enables parallel trajectory optimization based on multi-threading computation, can be formulated as follows:

$$\min_{u_k^{(j)}, x_{k+1}^{(j)}} C_T^{(j)} + \sum_{k=0}^{N-1} \left(C_{m,k}^{(j)} + \sum_{i=1}^M C_{s,k}^{(j)} + \|\mathbf{u}_k^{(j)}\|_{\mathbf{R}}^2 \right) \quad (16a)$$

$$\text{s.t. } \mathbf{x}(0) = \mathbf{x}_0, \mathbf{O}_i(0) = \mathbf{O}_{i,0}, \quad (16b)$$

$$\bar{\mathbf{x}}_{k+1} := f(\mathbf{x}_k, \mathbf{u}_k) \ominus \mathbf{x}_{k+1} = \mathbf{0}, \quad (16c)$$

$$O_{k+1} = \hat{f}^{SV}(\mathbf{O}_k), \quad (16d)$$

$$\mathbf{x}_{min} \preceq \mathbf{x}_k^{(j)} \preceq \mathbf{x}_{max}, \quad (16e)$$

$$\mathbf{u}_{min} \preceq \mathbf{u}_k^{(j)} \preceq \mathbf{u}_{max}, \quad (16f)$$

$$(\mathbf{u}_k^{(j)}, \mathbf{x}_{k+1}^{(j)}) \in \mathcal{U} \times \mathcal{X}, \quad (16g)$$

$$\forall k \in \mathcal{I}_0^{N-1},$$

where \hat{f}^{SV} is a discrete form of the predictive model of SVs (9) based on an Euler method.

We can optimize the NLP problem (16a)-(16g) for each nominal trajectory in a parallel manner using multi-threading computation. Hence, we can obtain optimized candidate trajectories $\{\tau_j\}_{j=1}^{N_t}$, where $\tau_j = \{\mathbf{u}_k^{(j)}, \mathbf{x}_{k+1}^{(j)}\}_{k=0}^{N-1}$; each trajectory τ_j corresponds to a target lane ξ_j enforced by the terminal constraint $C_T^{(j)}$.

To achieve efficient trajectory optimization to realize fast replanning that enables the EV to respond quickly and robustly to dynamic environments and further mitigate the effects of trajectory prediction errors of the SVs, we implement a receding horizon approach to optimize the NLP problem (16a)-(16g) based on sequential quadratic programming (SQP).

At each time step, the EV will execute only the first control input \mathbf{u}_0^* in the optimized sequence $\{\mathbf{u}_k^*\}_{k=0}^{N-1}$, which is obtained from the optimal trajectory among N_t candidate trajectories evaluated in the subsequent Section III-D. The remaining control sequence $\{\mathbf{u}_k^*\}_{k=1}^{N-1}$ is subsequently utilized as an initial solution for the warm start of the NLP problem (16a)-(16g), thereby enhancing the convergence speed.

Remark 2: The quadratic structure of the objective function (16a) allows for efficient optimization using the Hessian matrix information. However, computing the exact inverse

Hessian matrix is often computationally burdensome in real-time applications. Instead, we utilize Gauss-Newton methods [35] to approximate the inverse Hessian matrix from gradient information, facilitating more efficient optimization.

D. Decision-Making with Consistency

After checking the safety of the next position of each candidate trajectory based on the barrier function (8), we aim to determine the optimal decision behavior ξ^* and trajectory τ^* for the EV. Specifically, we design an evaluation algorithm that considers driving efficiency and stability for the optimized candidate trajectories obtained in (16a)-(16g) as follows:

$$\{\xi^*, \tau^*\} = \underset{\xi_j \in \Xi, \tau_j \in \mathcal{T}}{\operatorname{argmin}} s(\xi_j, \tau_j, \mathbf{w}), \quad (17)$$

where the overall cost function $s(\xi, \tau, \mathbf{w})$ is defined as:

$$s(\xi_j, \tau_j, \mathbf{w}) = \mathbf{w}^T \mathbf{f}(\xi_j, \tau_j), \quad (18)$$

where the weight vector $\mathbf{w} = [w_g, w_l, w_c, w_m]^T$ determines the relative importance of each sub-cost for a trajectory; $\mathbf{f}(\xi_j, \tau_j)$ denotes a vector of sub-costs that captures various aspects of the j th trajectory's performance as follows:

$$\mathbf{f}(\xi_j, \tau_j) = [F_g, F_l, F_c, F_m]^T, \quad (19)$$

where F_g , F_l , F_c , and F_m represent the goal-tracking, lateral deviation, comfort, and consistency costs, respectively.

1) *Goal-tracking metric:* The goal-tracking cost evaluates the ability of the optimized candidate trajectories to accomplish a driving task. For a cruise driving task, this cost aims to minimize the difference between the actual velocity of the EV and the target cruise velocity v_g . To account for the motion uncertainty of SVs over a long horizon, we apply an exponentially decreased weight with a discount factor γ_g for the longer $N - N_c$ time steps. Thus, we can compute an initial cost for this term using the following equation:

$$C_g = \sum_{i=1}^{N_c-1} \|v_i - v_g\|^2 + \sum_{i=N_c}^N \exp\left(\frac{-(i-N_c)}{\gamma_g}\right) \|v_i - v_g\|^2, \quad (20)$$

where v_i denotes the actual velocity of the EV at time step i ; N_c represents the number of initial time steps with relatively reliable position prediction of SVs at the beginning of the optimization horizon, which has a total of N time steps (e.g., $N_c = 10$ when $N = 50$).

To compare the goal-tracking cost among different sub-costs, we normalize C_g for each trajectory. This eliminates the influence of different units on performance metrics, giving us the goal-tracking cost F_g as follows:

$$F_g = \frac{C_g - C_{g,min}}{C_{g,max} - C_{g,min}}, \quad (21)$$

where $C_{g,min}$ and $C_{g,max}$ are the minimum and maximum values of C_g across all candidate trajectories, respectively.

2) *Lateral deviation metric*: The lateral deviation cost penalizes the distance of the optimized candidate trajectories to the center of each lane. The goal of this cost is to keep the EV close to the centerline, thereby enabling the EV to drive more stably and reduce the risk of collision with other vehicles. It can be represented by a function C_l that depends on the lateral position of the EV relative to the centerline of each target lane as follows,

$$C_l = \sum_{i=1}^{N_c-1} \|p_{y,i} - y_c\|^2 + \sum_{i=N_c}^N \exp\left(\frac{-(i-N_c)}{\gamma_l}\right) \|y_i - y_c\|^2, \quad (22)$$

where $p_{y,i}$ is the lateral position of the EV at time step i ; y_c is the lateral position of the centerline of the lane; γ_l is a discount factor. Similar to the goal-tracking cost, we normalize C_l for each trajectory using the following lateral deviation cost:

$$F_l = \frac{C_l - C_{l,min}}{C_{l,max} - C_{l,min}}, \quad (23)$$

where $C_{l,min}$ and $C_{l,max}$ are the minimum and maximum values of C_l among all candidate trajectories, respectively.

3) *Comfort metric*: The comfort cost penalizes the discomfort experienced by passengers due to the change in acceleration. The goal of this cost is to minimize the acceleration change. It can be represented by a function C_c that depends on the jerk of the EV as follows,

$$C_c = \sum_{i=1}^{N_c-1} \|j_i\|^2 + \sum_{i=N_c}^N \exp\left(\frac{-(i-N_c)}{\gamma_c}\right) \|j_i\|^2, \quad (24)$$

where $j_i = \frac{a_{i+1}-a_i}{T_s}$ is the jerk of the EV at time step i ; γ_c is a discount factor. We normalize C_c for each trajectory to get the comfort cost as follows:

$$F_c = \frac{C_c - C_{c,min}}{C_{c,max} - C_{c,min}}, \quad (25)$$

where $C_{c,min}$ and $C_{c,max}$ are the minimum and maximum values of C_c across all candidate trajectories, respectively.

4) *Consistency metric*: This cost penalizes frequent lane-changing behaviors, which may result in erratic and inconsistent driving. It can be represented by a function C_m that depends on the last target lateral position $y_{g,k-1}$ of last target lane ξ_{k-1} and new target lateral position $y_{g,k}$ of goal lane ξ_k as follows:

$$C_m = \|y_{g,k} - y_{g,k-1}\|^2, \quad (26)$$

where k denotes the current time step of driving. Similarly, we normalize C_c for each trajectory using the following equation:

$$F_m = \frac{C_m - C_{m,min}}{C_{m,max} - C_{m,min}}, \quad (27)$$

where $C_{m,min}$ and $C_{m,max}$ are the minimum and maximum values of C_m among all candidate trajectories, respectively.

IV. SIMULATION STUDY

In this section, we evaluate the effectiveness of our proposed parallel trajectory optimization framework in an adaptive cruise driving task under a dense and congested traffic flow.

TABLE I
PARAMETERS OF VEHICLE MODEL

	v_{max}	24 m/s	v_{min}	0 m/s
θ_{max}	0.227 rad	θ_{min}	-0.227 rad	
ω_{max}	5 rad/s	ω_{max}	-5 rad/s	
a_{max}	3 m/s ²	a_{min}	-1.5 m/s ²	
$\dot{\omega}_{max}$	2 rad/s ²	$\dot{\omega}_{min}$	-2 rad/s ²	

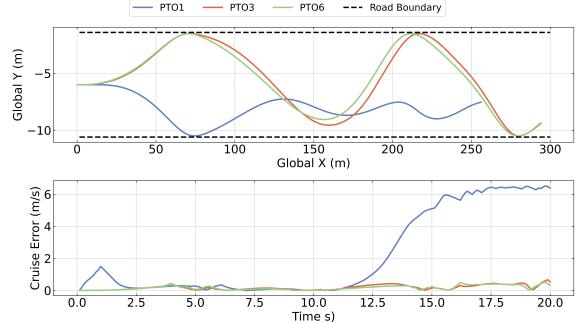


Fig. 4. Cruise performance of the EV with a target cruise speed 15 m/s and optimization horizon $T = 5$ s during 20 s simulation.

A. Simulation Setup

We conduct the simulation experiments using ROS2 on an Ubuntu 22.04 LTS system environment with an AMD Ryzen 5 5600G CPU with 6 cores and 12 threads running at a clock speed of 3.26 GHz and 16 GB of RAM. The control and communication frequencies between the EV and SVs are set as 10 Hz. The simulation time and optimization horizon T are set as 20 s and 5 s, respectively. We utilize the state-of-the-art optimization solver Acado [36] as the SQP solver for the NLP problem (16a)-(16g) for each nominal trajectory in a parallel manner using multi-threading computation in C++.

The EV aims to navigate through a dense and congested traffic flow in a one-direction three-lane road while avoiding the other nine SVs and keeping a target cruise speed $v_g = 15$ m/s. The SVs follow the intelligent driver model adopted from [27] and move parallel to the centerline while adapting their cruise velocity based on the distance to the cars in front. The initial states and target speeds of SVs are set as follows:

- Initial States: $\mathbf{O}_0 = [-10, -10, 9.5, 0]^T$, $\mathbf{O}_1 = [25, -10, 8.5, 0]^T$, $\mathbf{O}_2 = [60, -10, 9.0, 0]^T$, $\mathbf{O}_3 = [70, -6, 8, 0]^T$, $\mathbf{O}_4 = [85, -6, 8.5, 0]^T$, $\mathbf{O}_5 = [100, -6, 9.2, 0]^T$, $\mathbf{O}_6 = [130, -2, 10, 0]^T$, $\mathbf{O}_7 = [110, -2, 8, 0]^T$, $\mathbf{O}_8 = [160, -2, 12, 0]^T$.
- The target longitudinal speed is set as 10 m/s, 8 m/s, 12 m/s, 9.5 m/s, 8.5 m/s, 9.0 m/s, 8.0 m/s, 8.5 m/s, and 9.2 m/s, respectively.

The pertinent parameters of the nonlinear EV are shown in Table I. The initial state vector and control input of the EV are set as $x_0 = [0, -6, 0, 15, 0]^T$ and $u_0 = [0, 0]^T$, respectively. The following parameters are used: $N_c = 10$, $N = 50$, $\mathbf{Q}_m = diag(0, 10^3, 0, 10^5, 0)$, $w_1 = w_2 = w_3 = 5 \times e^{-\frac{t}{50}}$, $\mathbf{Q}_T = diag(0, 10^9, 10^9, 0, 10^6)$, $\mathbf{R} = diag(2 \times 10^4, 1 \times 10^6)$, $c = 8$, $\eta = 1$, $\varepsilon = 10^{-5}$, $\gamma = 50$, $\gamma_g = \gamma_c = \gamma_l = 40$, $a = 3$ m, $b = 2$ m, $p_{y,min} = -10.5$ m, $p_{y,max} = -1.5$ m, $M = 3$, and $\mathbf{w} = [2500, 150, 100, 100]^T$.

We evaluate the following set of Algorithms:

TABLE II
DRIVING PERFORMANCE COMPARISON

Algorithm	Cruise error		Safety		Efficiency		Consistency	
	e_{mean} (m/s)	e_{max} (m/s)	S_{min} (m)	\mathcal{P}_{safe} (%)	\mathcal{A}_{mean} (m/s ²)	\mathcal{T}_{solve} (ms)	\mathcal{L}_{long} (m)	\mathcal{P}_{LC} (%)
PTO1	2.136	6.527	1.635	100	0.116	23.096	256.159	-
PTO3	0.182	0.668	2.760	100	0.148	24.553	293.705	100
PTO6	0.166	0.5184	3.195	100	0.150	28.690	294.407	100

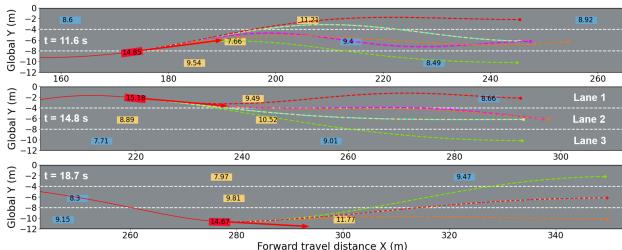


Fig. 5. Snapshots of the EV’s trajectory with the optimization horizon $T = 5$ s in dense and congested traffic. Red, yellow, and blue rectangles denote the EV, perceived, and unperceived SVs, respectively. Each dashed line represents a lane-oriented trajectory, and the red ones denote the selected target trajectory. The red arrow denotes the current velocity vector of the AV. The number in each rectangle denotes the current velocity of each vehicle. Simulation video accessible at <https://youtu.be/86oQ83jVPYg>.

- **PTO1:** This is an ablation study of our approach, achieved by solving (16a)-(16g) with a single optimized trajectory that is consistent with the center lane.
- **PTO3:** Our proposed method, obtained by solving (16a)-(16g) and (17), which optimizes over three nominal trajectories. Each trajectory corresponds to a unique lane in the three-lane road.
- **PTO6:** Similar to PTO3, it optimizes over six nominal trajectories. Four correspond to the center lane, and the remaining two correspond to the two outermost lanes. This is illustrated in Fig. 5.

B. Results

This subsection evaluates the performance of the three algorithms in terms of task accuracy, safety, driving efficiency, and driving stability, as shown in Table II. The minimum barrier function value S_{min} of all three algorithms is positive with respect to the three nearest vehicles, ensuring safe interaction between the EV with uncertain SVs. Furthermore, PTO6 exhibits a significantly larger safety barrier value than PTO1 (3.195 m versus 1.365 m), with all candidate optimized trajectories satisfying the safety requirements with $\mathcal{P}_{safe} = 100\%$. These findings demonstrate that the spatiotemporal safety module enables the EV to safely interact with SVs.

Figure 4 depicts the evolution of cruise errors, providing an intuitive understanding of the cruise performance. Notably, the cruise error of PTO1 increases rapidly after 11 s, which is caused by the congested scenario in front of the EV. In contrast, PTO6 and PTO3 can enable the EV to escape from the congested driving scenario through multiple lane searching. To further elucidate the driving process, Fig. 5 presents the trajectories and velocities of the EV during its escape from the congested driving scenario based on PTO6. As a result, PTO6 achieves the minimum deviation e_{max} from the desired cruise speed v_g among the

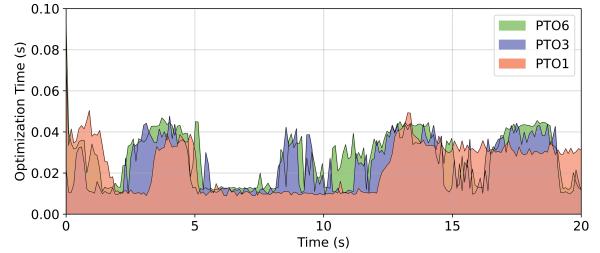


Fig. 6. The evolution of optimization time of three algorithms with planning horizon $T = 5$ s.

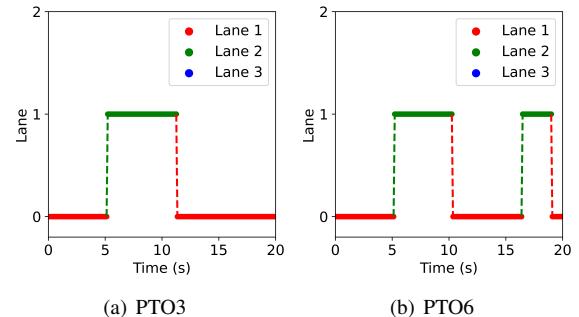


Fig. 7. The evolution of the EV’s target lane.

three algorithms and reduces the mean absolute cruise error e_{mean} by 92.223% and 8.791% compared to PTO3 and PTO1, respectively. These results highlight the advantage of using multiple parallel trajectory optimization methods in handling the uncertain behavior of SVs, which helps to reduce the impact of trajectory prediction errors resulting from uncertain SVs, leading to significant improvements in tracking accuracy.

In terms of driving efficiency, all three algorithms have an average optimization time \mathcal{T}_{solve} less than 100 ms, and the specific optimization time evolution can be observed in Fig. 6. These findings demonstrate that these algorithms facilitate real-time replanning of the EV in this congested traffic. Regarding travel efficiency, PTO3 and PTO6 exhibit a notably longer travel distance \mathcal{L}_{long} than PTO1 (294.407 m and 293.705 m versus 265.159 m). These results demonstrate that our framework significantly improves driving efficiency while maintaining a low average acceleration \mathcal{A}_{mean} and short solving time \mathcal{T}_{solve} through leveraging multiple searching trajectories. With regard to driving stability, Fig. 7 illustrates the evolution of the selected target lane during driving, revealing that there is no sudden lane change during each phase, resulting in good driving consistency. This finding is further supported by the consistency percentage of decision-making maneuvers $\mathcal{P}_{LC} = 100\%$, which indicates that there is no abrupt lane change during the driving task, further confirming the EV’s good driving consistency.

V. CONCLUSION

In this paper, we presented a parallel trajectory optimization method for fast replanning for the EV to achieve high travel efficiency in dense and congested traffic. Our approach efficiently generates feasible and safe candidate trajectories, while maintaining high computational efficiency based on the multiple-shooting and multi-threading techniques. Our simulation results demonstrated the effectiveness of our framework in improving safety, reducing travel time, and maintaining driving consistency for the EV in dense and congested traffic flow. In future work, we plan to extend our framework to address multi-agent navigation problems.

REFERENCES

- [1] D. Schrank, B. Eisele, T. Lomax, *et al.*, “Urban mobility report 2019,” Texas A&M Transportation Institute, College Station, Technical Report, 2019.
- [2] S. Narayanan, E. Chaniotakis, and C. Antoniou, “Shared autonomous vehicle services: A comprehensive review,” *Transportation Research Part C: Emerging Technologies*, vol. 111, pp. 255–293, 2020.
- [3] T. Liu *et al.*, “The role of the hercules autonomous vehicle during the COVID-19 pandemic: An autonomous logistic vehicle for contactless goods transportation,” *IEEE Robotics & Automation Magazine*, vol. 28, no. 1, pp. 48–58, 2021.
- [4] F. Tian, R. Zhou, Z. Li, L. Li, Y. Gao, D. Cao, and L. Chen, “Trajectory planning for autonomous mining trucks considering terrain constraints,” *IEEE Transactions on Intelligent Vehicles*, vol. 6, no. 4, pp. 772–786, 2021.
- [5] S. Shalev-Shwartz, S. Shammah, and A. Shashua, “On a formal model of safe and scalable self-driving cars,” *arXiv preprint arXiv:1708.06374*, 2017.
- [6] K. Leung, E. Schmerling, M. Zhang, M. Chen, J. Talbot, J. C. Gerdes, and M. Pavone, “On infusing reachability-based safety assurance within planning frameworks for human–robot vehicle interactions,” *The International Journal of Robotics Research*, vol. 39, no. 10–11, pp. 1326–1345, 2020.
- [7] M. Werling, S. Kammel, J. Ziegler, and L. Gröll, “Optimal trajectories for time-critical street scenarios using discretized terminal manifolds,” *The International Journal of Robotics Research*, vol. 31, no. 3, pp. 346–359, 2012.
- [8] Y. Zhang, H. Sun, J. Zhou, J. Pan, J. Hu, and J. Miao, “Optimal vehicle path planning using quadratic optimization for Baidu Apollo open platform,” in *IEEE Intelligent Vehicles Symposium*. IEEE, 2020, pp. 978–984.
- [9] M. Sharath and N. R. Velaga, “Enhanced intelligent driver model for two-dimensional motion planning in mixed traffic,” *Transportation Research Part C: Emerging Technologies*, vol. 120, p. 102780, 2020.
- [10] C. Miller, C. Pek, and M. Althoff, “Efficient mixed-integer programming for longitudinal and lateral motion planning of autonomous vehicles,” in *IEEE Intelligent Vehicles Symposium*. IEEE, 2018, pp. 1954–1961.
- [11] C. Liu, W. Zhan, and M. Tomizuka, “Speed profile planning in dynamic environments via temporal optimization,” in *IEEE Intelligent Vehicles Symposium*. IEEE, 2017, pp. 154–159.
- [12] H. Fan, F. Zhu, C. Liu, L. Zhang, L. Zhuang, D. Li, W. Zhu, J. Hu, H. Li, and Q. Kong, “Baidu Apollo EM motion planner,” *arXiv preprint arXiv:1807.08048*, 2018.
- [13] Z. Jian, S. Chen, S. Zhang, Y. Chen, and N. Zheng, “Multi-modal-based local path planning methodology for autonomous driving: An integrated framework,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 5, pp. 4187–4200, 2020.
- [14] W. Xu, Q. Wang, and J. M. Dolan, “Autonomous vehicle motion planning via recurrent spline optimization,” in *IEEE International Conference on Robotics and Automation*. IEEE, 2021, pp. 7730–7736.
- [15] B. Gassmann, F. Oboril, C. Buerkle, S. Liu, S. Yan, M. S. Elli, I. Alvarez, N. Aerrabotu, S. Jaber, P. van Beek, D. Iyer, and J. Weast, “Towards standardization of av safety: C++ library for responsibility sensitive safety,” in *IEEE Intelligent Vehicles Symposium*, 2019, pp. 2265–2271.
- [16] D. Q. Mayne, “Model predictive control: Recent developments and future promise,” *Automatica*, vol. 50, no. 12, pp. 2967–2986, 2014.
- [17] J. Zeng, B. Zhang, and K. Sreenath, “Safety-critical model predictive control with discrete-time control barrier function,” in *American Control Conference*. IEEE, 2021, pp. 3882–3889.
- [18] L. Zheng, R. Yang, Z. Wu, J. Pan, and H. Cheng, “Safe learning-based gradient-free model predictive control based on cross-entropy method,” *Engineering Applications of Artificial Intelligence*, vol. 110, p. 104731, 2022.
- [19] J. Yin, Z. Zhang, E. Theodorou, and P. Tsotras, “Trajectory distribution control for model predictive path integral control using covariance steering,” in *IEEE International Conference on Robotics and Automation*, 2022, pp. 1478–1484.
- [20] J. Wurts, J. L. Stein, and T. Ersal, “Collision imminent steering using nonlinear model predictive control,” in *American Control Conference*. IEEE, 2018, pp. 4772–4777.
- [21] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, “Control barrier functions: Theory and applications,” in *European Control Conference*. IEEE, 2019, pp. 3420–3431.
- [22] A. Sadat, M. Ren, A. Pokrovsky, Y.-C. Lin, E. Yumer, and R. Urtasun, “Jointly learnable behavior and trajectory planning for self-driving vehicles,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2019, pp. 3949–3956.
- [23] J. Ma, Z. Cheng, X. Zhang, Z. Lin, F. L. Lewis, and T. H. Lee, “Local learning enabled iterative linear quadratic regulator for constrained trajectory planning,” *IEEE Transactions on Neural Networks and Learning Systems*, 2022.
- [24] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, *et al.*, “Distributed optimization and statistical learning via the alternating direction method of multipliers,” *Foundations and Trends® in Machine learning*, vol. 3, no. 1, pp. 1–122, 2011.
- [25] J. Ma, Z. Cheng, X. Zhang, M. Tomizuka, and T. H. Lee, “Alternating direction method of multipliers for constrained iterative LQR in autonomous driving,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 23 031–23 042, 2022.
- [26] R. Han, S. Wang, S. Wang, Z. Zhang, Q. Zhang, Y. C. Eldar, Q. Hao, and J. Pan, “RDA: An accelerated collision free motion planner for autonomous navigation in cluttered environments,” *IEEE Robotics and Automation Letters*, vol. 8, no. 3, pp. 1715–1722, 2023.
- [27] V. K. Adajania, A. Sharma, A. Gupta, H. Masnavi, K. M. Krishna, and A. K. Singh, “Multi-modal model predictive control through batch non-holonomic trajectory optimization: Application to highway driving,” *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 4220–4227, 2022.
- [28] J. Chen, W. Zhan, and M. Tomizuka, “Constrained iterative LQR for on-road autonomous driving motion planning,” in *IEEE International Conference on Intelligent Transportation Systems*, 2017, pp. 1–7.
- [29] L. Zheng, R. Yang, Z. Peng, M. Y. Wang, and J. Ma, “Spatiotemporal receding horizon control with proactive interaction towards safe and efficient autonomous driving in dense traffic,” *arXiv preprint arXiv:2308.05929*, 2023.
- [30] G. De Nicolao, L. Magni, and R. Scattolini, “Stabilizing receding-horizon control of nonlinear time-varying systems,” *IEEE Transactions on Automatic Control*, vol. 43, no. 7, pp. 1030–1036, 1998.
- [31] F. A. Fontes, “A general framework to design stabilizing nonlinear model predictive controllers,” *Systems & Control Letters*, vol. 42, no. 2, pp. 127–143, 2001.
- [32] J. T. Betts, *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming*, 2nd ed. Philadelphia, PA: SIAM, 2010.
- [33] F. Borrelli, A. Bemporad, and M. Morari, *Predictive Control for Linear and Hybrid Systems*. New York, NY, USA: Cambridge University Press, 2017.
- [34] C. Mastalli, W. Merkt, J. Martí-Saumell, H. Ferrolho, J. Solà, N. Mansard, and S. Vijayakumar, “A feasibility-driven approach to control-limited DDP,” *Autonomous Robots*, vol. 46, no. 8, pp. 985–1005, 2022.
- [35] S. Gratton, A. S. Lawless, and N. K. Nichols, “Approximate Gauss–Newton methods for nonlinear least squares problems,” *SIAM Journal on Optimization*, vol. 18, no. 1, pp. 106–132, 2007.
- [36] B. Houska, H. Ferreau, and M. Diehl, “ACADO Toolkit – An Open Source Framework for Automatic Control and Dynamic Optimization,” *Optimal Control Applications and Methods*, vol. 32, no. 3, pp. 298–312, 2011.