

Received February 4, 2021, accepted February 23, 2021, date of publication March 4, 2021, date of current version March 16, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3063795

Collision Imminent Steering at High Speeds on Curved Roads Using One-Level Nonlinear Model Predictive Control

JOHN WURTS, JEFFREY L. STEIN, AND TULGA ERSAL 

Department of Mechanical Engineering, University of Michigan, Ann Arbor, MI 48109, USA

Corresponding author: Tulga Ersal (tersal@umich.edu)

Toyota Research Institute (“TRI”) provided funds to assist the authors with their research, but this article solely reflects the opinions and conclusions of the authors and not TRI or any other Toyota entity.

ABSTRACT Collision imminent steering is an automotive active safety feature designed to perform an evasive lane change maneuver when a forward collision cannot be avoided by braking alone. Previous work developed a nonlinear one-level model predictive controller to perform such a maneuver in a straight highway environment and at high speeds. In this article, a new formulation is presented that extends this capability to curved roads. The formulation includes a drivable tube concept, within which the maneuver is allowed to push the vehicle to its limits of handling, but only if the controller deems it necessary. Numerical simulation results showcase aggressive lane change maneuvers on curved roads at high speeds, allowing for both inside and outside lane changes, as well as single and double lane change maneuvers in a shorter distance than braking.

INDEX TERMS Advanced driver assistance systems, intelligent vehicles, autonomous vehicles, collision avoidance, nonlinear control systems.

I. INTRODUCTION


Collision imminent steering (CIS) is an automotive active safety feature designed to perform an evasive lane change maneuver in the event that the host vehicle detects a forward collision cannot be avoided by braking alone. Performing a CIS maneuver can be a challenging task for a controller especially at high speeds, because such a maneuver may require pushing the vehicle to its limits of handling, often a nonlinear vehicle dynamics regime [1].

Model predictive control (MPC) provides a formalism to take such limits into account explicitly while simultaneously optimizing the trajectory and control commands for CIS [1]–[6]. Such simultaneous treatment of trajectory planning and tracking is referred to as a one-level architecture [7]. This architecture is in contrast to the two-level architecture, where the trajectory planning and tracking problems are solved separately [8], [9]. Although various methods exist to establish a reference trajectory for a two-level architecture to handle curved roads, such as using the center line [10], variable lateral deviation limits defined by splines [11], the drivable tube [8] or safe corridor [9] concepts, and others [12]–[15], such pre-defined trajectories

do not guarantee safety and drivability at the limits of handling in a CIS maneuver. Therefore, a one-level architecture is preferred in this work due to its established advantages of not requiring a reference trajectory, and providing a control command consistent with the planned trajectory, both critical in the CIS context for safety and drivability.

One-level MPC formulations are available in the literature for lane change maneuvers. For example, in [4], a one-level linear MPC controller enforces a lane change in straight roads by restricting the allowable lateral position in the road at different longitudinal positions. The resulting controller then finds an optimal state trajectory and corresponding control trajectory that minimizes a threat assessment. By using a different optimality condition compared to path following, the controller is not reliant on a state trajectory provided *a priori*, and can incorporate online sensor information to generate a safer trajectory. However, this lane change formulation is dependent on a low speed application using a linearized dynamics model, exploiting the fact that for straight roads the lane change criteria are natively known *a priori*.

In the context of a CIS maneuver, various nonlinear one-level MPC controllers are designed to change lanes while operating at the limits of handling. In [2], the controller is designed to minimize the longitudinal distance traveled when departing the host lane in a straight road scenario. This

The associate editor coordinating the review of this manuscript and approving it for publication was Mauro Gaggero .

distance represents a threshold, beyond which no safe lane change can occur. In [1], a similar controller is developed based on the same lane change constraints, but seeks to minimize the peak tire slip for a fixed obstacle distance. While both controllers can incorporate the nonlinear system response, the controller formulation again exploits characteristics of a straight road, and thus cannot be natively extended to curved roads.

For curved road environments, the work in [16] designs a nonlinear one-level MPC controller to mimic human input while maintaining lateral deviation limits in curved roads. This controller is able to relax a path following optimality condition in favor of mirroring human input as best as safely possible. However, the lateral deviation information is separate from obstacle information, requiring multiple constraints for each obstacle encountered, growing the optimization problem. Additionally, the controller is designed for a vehicle kinematics model, which does not ensure drivability.

Thus, a gap is identified in the state of the art that this article aims to fill. The intended CIS application must obey both obstacle avoidance and lane boundary limits on curved roads. The drivable tube concept efficiently contains this information, but existing controllers that employ this concept are two-level and thus require a reference trajectory, which is not preferred for CIS at high speeds due to aforementioned reasons. Instead, it is desirable to employ a one-level architecture for CIS at high speeds, and there does not currently exist a formulation to natively incorporate the drivable tube to handle curved roads. With regards to the state of the art discussed, the salient contributions of this article are as follows.

- 1) Development of a nonlinear one-level MPC formulation for CIS at high speeds on curved roads that directly incorporates the online environmental and obstacle information without the need of an intermediate reference trajectory. Resulting CIS trajectory solutions ensure optimality and drivability.
- 2) Demonstration of the robustness of the controller formulation to plant-prediction model mismatch due to lower model fidelity in MPC.
- 3) Demonstration of the flexibility of the controller formulation to different types of CIS maneuvers, specifically, inside and outside lane changes, and single and double lane changes.

A preliminary version of this controller was presented in a conference [17], which did not consider the plant-prediction model mismatch, and only evaluated a single outside lane change. This article extends these preliminary results to close the loop with a full order plant model for evaluating robustness to model mismatch, analyzes two additional scenarios beyond the single outside lane change, and compares the controller to a two-level path following approach. A curved road study is presented in [18] focusing on adaptation to address uncertainty in the coefficient of friction, but the details of the problem formulation are not given. The present paper presents and discusses the underlying optimization problem structure in detail.

The remaining sections of the article proceed as follows. Sec. II describes a simulated highway environment, a drivable tube concept for collision avoidance, and the plant model used in numerical simulation. Sec. III describes the prediction model used in the MPC formulation, as well as generates the objective function and constraints used in the underlying optimization problem. Sec. IV catalogs numerical simulations for an outside single lane change, inside single lane change, outside double lane change, and comparison to a two-level path following controller. Sec. V highlights key design aspects of the controller, simulated performance metrics, as well as limitations.

II. HIGHWAY ENVIRONMENT AND HOST VEHICLE MODEL

A CIS maneuver is intended to take place if the vehicle detects it cannot avoid collision by braking alone. As such, a CIS maneuver is most likely to take place at high speed when the braking distance is long. The maximum highway speed in the USA reaches 85 MPH in Texas, though most states peak at 70 MPH. Hence, the simulated environment considers a host vehicle traveling at 35 m/s, representing about 78 MPH or 126 KPH.

As an expansion on existing CIS algorithms for straight road cases, a curved highway section is considered. At 35 m/s, the tightest radius turn allowed by U.S. highway building codes is 1,500 m [19]; however, a $r_{\text{turn}} = 500$ m curve is considered to emphasize the effects of road curvature and to generate a more extreme maneuver for consideration.

The simulated highway is a three lane highway section with the center lane following a right hand curve of constant 500 m curvature. Per highway construction code, the lanes are set at the minimum allowable lane width of $w_{\text{lane}} = 3.7$ m [20], again to generate a challenging scenario for the host vehicle. The host vehicle starts in the center lane with a state trajectory that follows the steady state road curvature.

To motivate a CIS maneuver, there is a stopped vehicle blocking the center lane at some fixed distance ahead. Identifying how far ahead the obstacle resides is left to other systems, but for simulation purposes the obstacle is 47 m ahead, as Sec. IV shows this is too short for braking, but within the window of opportunity for CIS. To avoid collision, the host vehicle must leave the starting center lane and change into the left or right lane for an outside or inside lane change, respectively. A CIS maneuver is considered safe if it completely leaves the starting lane before passing the stopped obstacle and remains inside the new lane's boundaries thereafter.

This high level description of a safe CIS maneuver is captured through a drivable tube concept for the MPC controller. The drivable tube represents the topographical area the host vehicle can reside within without causing a collision [21]. For the described scenario, as long as the perimeter of the host vehicle remains inside the lane boundaries, the maneuver is considered collision-free and safe. However, the MPC controller discussed later uses a reduced 3 degrees-of-freedom (3 DoF) bicycle model and only considers the

vehicle’s center of gravity (CG). Thus, a constricted drivable tube is generated with considerations to the vehicle’s CG.

The optimality of a safety critical maneuver is not necessarily measured by path following accuracy, but can depend on some other metric. To allow the controller the most flexibility, the drivable space is allowed to expand as large as safely possible, leaving it up to the controller to find the best path. Consider first an outside single lane change in a right turning curve (Fig. 1). The drivable space is the starting center lane and the left lane prior to passing the obstacle, then constricted exclusively to the left lane beyond the obstacle.

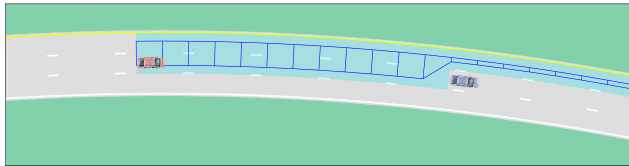


FIGURE 1. The host vehicle, red, encounters the stationary vehicle, blue. For a CIS to the outside lane, the safe area for the plant model is shown in teal, and drivable tube for the 3 DoF prediction model is shown in the blue parallelograms. The vehicle is allowed to use the entire center and left lane prior to passing the obstacle, then is restricted to left lane after passing.

In practice, the drivable tube would be generated online from sensor fusion, potentially leveraging vision based cameras, LiDAR, radar, ultrasonic sensors, and/or high definition maps. For the simulated highway environment, the drivable tube is generated geometrically.

The drivable tube is a slight constriction from the lane boundaries. It is constricted by the vehicle half width $\frac{w_v}{2}$ and safety buffer σ . The safety buffer represents an additional conservative constriction, because the vehicle rotation is not considered when evaluating the lane boundary limits. In practice, a safety buffer of $\sigma = 0.5$ m avoids corner clipping when passing the obstacle [2]. However, alternative formulations to address corner clipping exist, such as co-linear circles along the longitudinal length [16], or explicitly monitoring each corner [15].

For an outside lane change, the right tube boundary follows a radius of $r_{\text{turn}} - \frac{w_{\text{lane}}}{2} + \frac{w_v}{2} + \sigma$ prior to crossing the obstacle, then $r_{\text{turn}} + \frac{w_{\text{lane}}}{2} + \frac{w_v}{2} + \sigma$ thereafter. The left tube boundary follows $r_{\text{turn}} + \frac{3w_{\text{lane}}}{2} - \frac{w_v}{2} - \sigma$ throughout.

In contrast, for an inside lane change, the left tube boundary follows a radius of $r_{\text{turn}} + \frac{w_{\text{lane}}}{2} - \frac{w_v}{2} - \sigma$ before passing the obstacle, and $r_{\text{turn}} - \frac{w_{\text{lane}}}{2} - \frac{w_v}{2} - \sigma$ thereafter. The right tube boundary follows a constant $r_{\text{turn}} - \frac{3w_{\text{lane}}}{2} + \frac{w_v}{2} + \sigma$ throughout.

The drivable tube is stored as a sequence of matched pairs, where each pair consists of a left and right x - y position of the tube edge. Consecutive pairs in the sequence march along the road to form the drivable tube. The four points contained in a matched pair and the consecutive matched pair in the sequence form a parallelogram. The fore edge of one parallelogram is the same as the aft edge of the next sequential parallelogram. The MPC controller expects the drivable tube to be stored as this sequence of connected parallelograms and will solve the optimization problem to keep the prediction

model’s CG inside the parallelogram’s left and right edges. A key benefit of leveraging the drivable tube in this formulation is the flexibility on concavity; the drivable tube makes no assumption on strictly concave or convex representation, only that it is not self-intersecting.

The lengths of the parallelograms are flexible depending on the external system used to establish them. Parallelograms’ lengths do not need to be consistent, but parallelograms of zero length should be avoided. Shorter parallelogram lengths allow for a more accurate road description, but have a higher memory requirement for MPC solving. The scenario herein uses matched pairs spaced approximately 5 m down the road. Immediately before the obstacle, the parallelogram is smoothed by blending the consecutive parallelograms, preventing a zero length parallelogram that can lead to numerical difficulty. An example drivable tube constructed from the parallelogram, as well as the safe drivable space for the host vehicle, can be seen in Fig. 1.

The host vehicle is modeled as a luxury sedan, as this class of vehicles is more likely to feature latest technologies. Besides CIS, one such feature is active rear-wheel steering, which allows the rear wheels to be steered by an on-board computer independent of driver input and front wheel manipulation. Hence, the MPC controller is designed to leverage active rear wheel steering, if available, as leveraging rear wheel steering has been shown to improve vehicle performance [1].

The host vehicle model is a 14 DoF dynamics model that is validated against CarSim’s F-class sedan, a high fidelity driving simulator [22]. The underlying equations are omitted for brevity, but parameters used in the 14 DoF model are detailed in Table 1. While some parameters are published by the vehicle manufacturer, others are estimated by normalized trends [23].

TABLE 1. 14 DoF Plant Parameters.

Parameter	Symbol	Value
Vehicle sprung mass	m_{sprung}	1820 kg
Unsprung mass per wheel	m_{unsprung}	50 kg
Weight distribution	-	51.4/48.6 F/R
Wheel base	-	3.2 m
Vehicle width	w_v	1.9 m
Vehicle track width	l_c	1.6 m
Roll moment of inertia	I_{xx}	1023.8 kg m ²
Pitch moment of inertia	I_{yy}	3567.2 kg m ²
Yaw moment of inertia	I_{zz}	4095.0 kg m ²
Strut height	z_{strut}	0.590 m
Strut stiffness	k_{strut}	83000 N/m
Strut damping	d_{strut}	1896.1 Ns/m
Wheel stiffness	k_{wheel}	278000 N/m
Wheel radius	r_{wheel}	0.353 m

The host vehicle is modeled after a 2018 BMW 740i. While some parameters are published by the manufacturer, others are estimated based on normalized ratios for sedans.

The plant model uses the nonlinear Pacejka tire force model in (1), with parameters and variables defined in Table 2. Eq. (1) uses the lateral tire slip ratio s , which must be calculated at each of the tire contact points. These tire parameters are calculated to provide a peak tire force of

TABLE 2. Tire Properties.

Tire Parameter	Symbol	Value
Coefficient of friction	μ	0.8
Tire property	B	13
Tire property	C	1.285
Tire longitudinal velocity	V_x	
Tire lateral velocity	V_y	

0.8g at approximately 12° slip and 10% force relaxation at high slip angles, where g is the gravitational acceleration.

$$\begin{aligned}
 F_y &= \mu F_z \sigma_y \\
 \sigma_y &= -\sin(C \arctan(Bs)) \\
 s &= \frac{V_y}{V_x}
 \end{aligned} \tag{1}$$

With the drivable tube defined and host vehicle model established, the CIS controller is developed in the next section.

III. PREDICTION MODEL AND OPTIMAL CONTROL PROBLEM FORMULATION

MPC controllers leverage a model of the plant system to predict the future system response for a candidate control trajectory. To that end, the 3 DoF bicycle model has been shown to be sufficiently accurate in high speed obstacle avoidance applications, provided the obstacle is not excessively wide [24]. Thus, a 3 DoF bicycle model is used in this work, as well.

The 3 DoF prediction model states, control inputs, and equations of motion are given as follows.

$$x = \begin{bmatrix} \text{global } x \text{ position [m]} \\ \text{global } y \text{ position [m]} \\ \text{vehicle yaw [rad]} \\ \text{longitudinal velocity [m/s]} \\ \text{lateral velocity [m/s]} \\ \text{yaw rate [rad/s]} \\ \text{front steering angle [rad]} \\ \text{rear steering angle [rad]} \end{bmatrix} = \begin{bmatrix} x \\ y \\ \psi \\ u \\ v \\ \omega \\ \delta_f \\ \delta_r \end{bmatrix} \tag{2}$$

$$u = \begin{bmatrix} \text{front steering rate [rad/s]} \\ \text{rear steering rate [rad/s]} \end{bmatrix} = \begin{bmatrix} \dot{\delta}_f \\ \dot{\delta}_r \end{bmatrix} \tag{3}$$

$$\frac{dx}{dt} = \begin{bmatrix} u \cos(\psi) - v \sin(\psi) \\ u \sin(\psi) + v \cos(\psi) \\ \omega \\ 0 \\ -u \omega + \frac{F_{y,f} \cos(\delta_f) + F_{y,r} \cos(\delta_r)}{I_{zz}} \\ \dot{\delta}_f \\ \dot{\delta}_r \end{bmatrix} \tag{4}$$

In (4), the time derivative of longitudinal velocity is set to zero. This is because both the prediction and plant models lock the longitudinal velocity. Per US highway testing standards, the double lane change maneuver, also known as “moose avoidance maneuver,” does not allow acceleration or braking [25]. In simulations, longitudinal acceleration due to yawing under non-zero lateral velocity has been negligible.

The prediction model uses the same Pacejka tire force model as the plant and with the same coefficients. However, the prediction model is a single track bicycle model, thus the tire slip is only evaluated at the front and rear axle locations. The parameters used in the prediction model are listed in Table 3.

TABLE 3. 3 DoF Model Parameters.

Parameter	Symbol	Value
Vehicle mass	m	2020 kg
Weight distribution	-	51.4/48.6 F/R
Wheel base	-	3.2 m
Vehicle width	w_v	1.9 m
Front wheel to CG distance	l_f	1.56 m
Rear wheel to CG distance	l_r	1.64 m
Front wheel vertical force	$F_{z,f}$	10182.8 N
Rear wheel vertical force	$F_{z,r}$	9633.4 N
Yaw moment of inertia	I_{zz}	4095.0 kg m ²

The upper table contains values published by the manufacturer. The lower table contains values derived as needed for the 3 DoF prediction model.

The system starts at an initial state x_0 and is numerically simulated forward using Runge-Kutta fourth order (RK4) integration with a zero-order control hold. The zero-order control hold represents a constant control rate as input for a set block of time and is chosen to simulate loop timing control architectures typically found on distributed automotive systems. For this problem, an MPC time horizon of 3.2 s has been sufficiently long for the vehicle to steadily change lanes without excessive computational complexity. The time horizon is broken into constant control intervals of 50 ms, corresponding to the optimization problem of solving 64 consecutive front and rear control rates to obey safety criteria. At each closed-loop iteration, two sequential constant control rates are applied to the plant. The prediction model is integrated at 10 ms time steps, as smaller time steps show negligible accuracy in the state prediction.

For a candidate control trajectory of 64 control inputs of 50 ms duration integrated at 10 ms time steps, there are $n = 320$ discrete integration states in the prediction trajectory. Each of the states are evaluated against the following safety criteria, which form the constraints of the optimization problem.

For the numerical simulations herein, the initial state x_0 is the steady state solution in the starting lane. This can be calculated by evaluating the steady state conditions in (4) for a given u_0 and curvature. Solving (4) for front and rear steering architectures is achieved by setting the rear steering angle to $\delta_r = 0$, as this avoids a continuum of steady state conditions and scales to the front-only architectures directly.

In practice, the initial state is taken from on-board sensor information.

A. BOUNDARY CONSTRAINTS

In Sec. II, a drivable tube is generated, which represents the left and right edges between which the CG of the 3 DoF prediction model can safely reside. Using vector algebra, the relevant parallelogram is identified efficiently, and

corresponding left and right boundary constraints are evaluated. Consider Fig. 2 showing an integration state located inside a parallelogram.

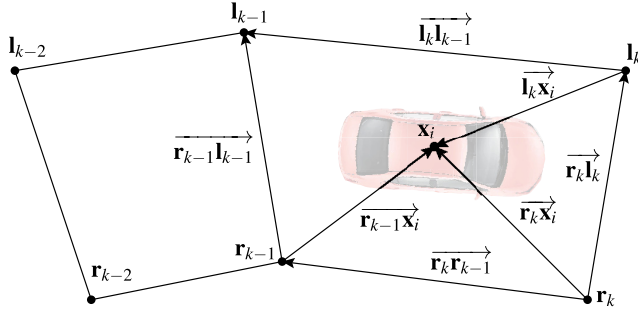


FIGURE 2. An integration state x_i is shown as residing inside a parallelogram. Using various vector cross products, the algorithm identifies when an integration state has left one parallelogram and traversed into the next. Once the active parallelogram is identified, the right and left boundary constraints are evaluated again with vector logic.

As the MPC controller integrates the state forward in time, the algorithm must identify which parallelograms in the drivable tube are applicable to that state. To find the active parallelogram, the vector cross product of the vehicle position with the fore parallelogram boundary is calculated.

Consider the matched pair (r_k, l_k) , representing the $(k+1)^{th}$ matched pair in the drivable tube sequence and the fore parallelogram boundary of the k^{th} parallelogram. For the i^{th} integration state in the prediction horizon, the vehicle's position vector relative to the right boundary in the matched pair is given by $\vec{r}_k x_i$. To find the active k^{th} parallelogram, k is increased until the direction of the cross product of the relative position vector with the fore boundary vector changes to negative, which is tested numerically by checking if $(\vec{r}_{k-1} x_i \times \vec{r}_{k-1} l_{k-1})(\vec{r}_k x_i \times \vec{r}_k l_k) < 0$ is satisfied. This fast vector algebra allows efficient scaling to the GPU hardware used to simulate the vehicle and evaluate the fitness and feasibility criteria.

With the active parallelogram k identified, the left and right lane boundaries are evaluated. The left lane boundary constraint is evaluated by the cross product of the relative position vector with left edge vector, $\vec{l}_k x_i \times \vec{l}_k l_{k-1}$. This constraint is formed such that if the cross product is less than zero, the vehicle integration state is to the right of the left lane boundary, which is valid. It is also important to note the validity of the constraint is based on the sign. This allows the valid limits of the constraint, being in reference to 0, to be established at the start of the MPC problem and avoids having to first seed a trajectory.

Likewise, the right lane boundary is generated by the cross product of the right edge vector with the relative position vector, $\vec{r}_k r_{k-1} \times \vec{r}_k x_i$. Here, the order is swapped to maintain a cross product of less than or equal to zero to be valid in convention with constraint formulation. Together, the left and right lane boundary constraints form (5) and (6).

$$d_l = \vec{l}_k x_i \times \vec{l}_k l_{k-1} \leq 0 \quad (5)$$

$$d_r = \vec{r}_k r_{k-1} \times \vec{r}_k x_i \leq 0 \quad (6)$$

Note, the constraint is formed as at least C^2 smooth within the active parallelogram. This helps with solving the numerical optimization problem using gradient based optimization, improving convergence time [26].

Further, the left and right boundary constraints are evaluated based on the instantaneous integration state in the environment at each iteration during the nonlinear optimization solve, in contrast to other methods that use an initial seeding or reference seeding for lateral displacement limits. This ensures the left and right constraints are relevant at the converged optimal trajectory, as well as all the intermediate trajectories, and not just the first candidate trajectory. Although the active parallelogram can change during the optimization, resulting in a discontinuity as the relevant left and right boundary limits change, this has not led to convergence difficulty or instability in testing.

Next, the vehicle stability criteria are developed.

B. VEHICLE STABILITY: TIRE SLIP

The Pacejka tire formula captures the nonlinear tire relaxation at high slip. This is a dangerous regime to operate in, because the tire force has low response to steering input, which can result in quickly losing control of the vehicle [27]. Additionally, this relaxation can be difficult for gradient based optimization, as the tire force response has a gradient inversion above the peak slip, causing local minima.

To ensure vehicle stability, the optimization problem restricts the peak allowable tire response. The stable tire region can be categorized by differentiating (1) with respect to the slip ratio, giving $\frac{dF_y}{ds} > 0$. However, evaluating this constraint, and appropriate sensitivities, is computationally expensive in practice. Instead, analysis has shown the vehicle stability can be maintained by directly restricting the slip angle to some α_{peak} , which depends on the tire properties, which might vary between front and rear [1]. For the tire properties used in this example, offline analysis of expected vehicle states shows setting α_{peak} to within 8° ensures stability while retaining approximately 98% of the available tire force.

This constraint is formulated on the front and rear wheels as follows and is enforced at every integration state in the prediction trajectory.

$$\alpha_f = \left| \delta_f - \arctan \left(\frac{v + \omega l_f}{u} \right) \right| \leq \alpha_{peak} \quad (7)$$

$$\alpha_r = \left| \delta_r - \arctan \left(\frac{v - \omega l_r}{u} \right) \right| \leq \alpha_{peak} \quad (8)$$

The tire slip only forms half of the vehicle stability consideration. Next, the terminal stability is developed.

C. VEHICLE STABILITY: TERMINAL STATE

The MPC controller is structured with terminal state constraints that leave the vehicle settled in the next lane at the end of the maneuver. These terminal state constraints are structured to avoid initiating a maneuver that the MPC controller

initially sees as viable, but becomes infeasible later as the prediction horizon recedes.

If the final state in the prediction trajectory is settled in the intended lane, then that candidate trajectory performs a complete CIS maneuver, and the controller is safe to begin taking action. This ensures recursive feasibility under ideal assumptions, because the terminal state at the end of the first closed-loop iteration ensures drivability thereafter. The terminal state, x_{stable} , is derived from road information.

For the sample case, the terminal state would be a right hand turn following a radius of $r_{\text{lane}} = r_{\text{turn}} + w_{\text{lane}}$ for the outside lane change, and $r_{\text{lane}} = r_{\text{turn}} - w_{\text{lane}}$ for the inside lane change. Similar to the initial state, the terminal stable state for the new radius is calculated by evaluating the steady state conditions from (4).

In practice, this approach would require some form of estimate for road curvature at the end of the maneuver, which supports a look-up table or alternative estimate for the stable states.

One challenging element of terminal state constraint is establishing *a priori* the location of final state in the prediction horizon. While it is easy to describe the desired position as centered in the next lane, it is difficult to state how far down the lane the final state is, and correspondingly, how much the vehicle needs to be rotated relative to its starting position. For this implementation, the terminal (x_t, y_t) position is set based on the radius from the curve center (x_c, y_c) , not explicitly road position. The terminal vehicle rotation angle is set based on (x_t, y_t) position, and calculated such that the vehicle velocity is tangent to the road center.

In practice, an online estimator can generate a localized curve to implement the radius and tangential constraints. Alternatively, the drivable space boundaries can be artificially contracted to mandate the vehicle is in the center of the lane far in the prediction horizon.

It is not essential that the terminal position constraints are as accurate as possible. The terminal position constraints are enforced far in the prediction horizon where the vehicle has the most control authority and ability to correct in closed-loop implementation.

Based on this discussion, the terminal position (x_t, y_t) and rotation ψ_t are structured as follows.

$$(x_t - x_c)^2 + (y_t - y_c)^2 = r_{\text{lane}}^2 \quad (9)$$

$$\psi_t = \arctan\left(\frac{y_t - y_c}{x_t - x_c}\right) - \arctan\left(\frac{v_t}{u_0}\right) \quad (10)$$

There are three unknowns in (9) and (10); this allows the solver to numerically satisfy the unknown distance traveled internally. The complete terminal state is structured as follows.

$$x_{\text{stable}} = [x_t \ y_t \ \psi_t \ u_0 \ v_t \ \omega_t \ \delta_{f,0} \ 0]^T \quad (11)$$

D. VEHICLE PHYSICAL LIMITS

The last set of constraints on the MPC problem is to capture the physical limits of the vehicle. Specifically, the front and rear wheels are both steering angle and steering rate limited.

For this vehicle, the front wheels have a simulated maximum steering angle limit of $\pm 35^\circ$ and a maximum rate limit of ± 70 deg/s. For rear wheel steering capable vehicles, the rear wheels typically do not have as strong control authority, and their steering angle range and rate are comparatively reduced. For this vehicle, a simulated limitation on the rear wheel angle is set at $\pm 10^\circ$ and rate limited to ± 35 deg/s. To model a front-only steering vehicle, the rear wheels' angle limit is set at 0° .

Like the other constraints, the steering rate limits are enforced at every integration point in the prediction horizon. Because the optimization problem is structured such that the design variables are the steering rates, the physical rate limitations are enforced as bounds on the design variables.

Additionally, the steering angle is a linear function of the design variables, and for certain numerical optimizers, it might be handled differently. While it does not change the structure of the numerical optimization problem presented here, this subtle difference should be considered for optimization convergence timing purposes.

Hitherto, all the constraints for the optimization problem are built. Any candidate trajectory that obeys all these constraints is considered feasible. Next, the objective function is formulated to determine the optimal trajectory.

E. OBJECTIVE FUNCTION: MINIMUM SLIP

Recall in Sec. I, many of the obstacle avoidance algorithms are designed as path following formulations. In the context of CIS, following a predefined reference path does not necessarily correlate to the safest trajectory. Instead, a different metric is referenced.

In [3], the safety metric is to perform the maneuver in the shortest amount of time. In [2], the safety metric is designed to leave the starting lane in the shortest distance. In [4] and [16], the safety metric is to minimize a threat metric and deviation from human input, respectively.

While leaving the starting lane in the shortest distance can provide an informative metric for how late to intervene, it might not be the best in practice. If the vehicle decides it will intervene, it can be beneficial to intervene immediately instead of delaying.

If the vehicle detects it cannot brake in time and must perform a lane change, one option is to still perform the minimum distance lane change, thus maximizing the available gap between where the vehicle is leaving the lane and the obstacle. However, this would be an overly aggressive maneuver, as the minimum distance lane change will always load the vehicle tires to the maximum allowable force [2].

Alternatively, the minimum slip formulation is used to avoid pushing the vehicle to its limits of handling unnecessarily, while still being capable of it if need be [1]. The minimum slip formulation is intended to minimize the peak tire slip experienced in the maneuver, and hence maximize the additional tire force available. This allows the most additional control authority for mid-maneuver corrections in closed loop, and results in a minimally aggressive lane change.

Minimizing a maximum function can be difficult in gradient based optimization. Thus, the Kreisselmeier-Steinhauser (KS) constraint aggregation function is used to minimize the tire slip in a gradient smooth fashion. Details on the properties of the KS function and important considerations are discussed in [1].

Using the instantaneous tire slip at every integration point similar to (7) and (8), the KS aggregation is generated as follows.

$$KS(\alpha, \rho_{\text{slip}}) = \frac{1}{\rho} \ln \left(\sum_{i=1}^n e^{\rho_{\text{slip}} \alpha_i} \right) \quad (12)$$

Here, n is the number of integration points, and α is a vector containing the front and rear instantaneous tire slip angles at every integration state in the prediction horizon. In testing, a constraint aggregation parameter of $\rho_{\text{slip}} = 264$ has been shown to be stable [1].

With the objective and constraints defined, the numerical optimization problem can be formulated.

F. NUMERICAL OPTIMIZATION PROBLEM

Combining the objective functions and constraints described above, the numerical optimization problem is formulated as follows.

$$\begin{aligned} & \min_u KS(\alpha, \rho) \\ & \text{subject to } d_{l,i} \leq 0 \quad \forall i \in [1, n] \\ & \quad d_{r,i} \leq 0 \quad \forall i \in [1, n] \\ & \quad \alpha_{f,i} \leq 0 \quad \forall i \in [1, n] \\ & \quad \alpha_{r,i} \leq 0 \quad \forall i \in [1, n] \\ & \quad (x_t)_i - (x_{\text{stable}})_i = 0 \quad \forall i \in [4, 8] \\ & \quad (x_t - x_c)^2 + (y_t - y_c)^2 = r_{\text{lane}}^2 \\ & \quad \arctan \left(\frac{y_t - y_c}{x_t - x_c} \right) = \frac{\pi}{2} - \arctan \left(\frac{v_t}{u_0} \right) \\ & \quad |\delta_{\bullet}(t_i)| \leq \delta_{\bullet, \text{max}} \quad \forall t_i \\ & \quad |\dot{\delta}_{\bullet}(t_i)| \leq \dot{\delta}_{\bullet, \text{max}} \quad \forall t_i \end{aligned} \quad (13)$$

where \bullet in the subscripts in the last two inequalities is a placeholder for f for front and r for rear tires. Note, the system dynamics are not explicitly stated, as they are incorporated through the RK4 integration.

Solving the resulting numerical optimization problem gives a control trajectory over a fixed horizon that is predicted to perform the CIS maneuver. Any feasible trajectory is considered safe, and the optimal trajectory is considered safest.

Next, various highway scenarios are constructed to evaluate the performance of the CIS controller.

IV. NUMERICAL SIMULATIONS AND RESULTS

The optimal control problem (13) is solved using IPOPT [28] on a 2017 HP Omen desktop with an Intel i7-7700k CPU and NVidia GTX 1080 discrete GPU. Solutions converge in approximately 55 ms using a custom GPU-accelerated multiple shooting implementation. Eq. (13) is presented as a

single shooting trajectory simulation, but variations between single shooting, multiple shooting, and collocation based implementations have not resulted in variations in the converged solution, rather differences in solution time. While the examples chosen herein are compatible with real-time implementation, it is not possible to guarantee the full nonlinear formulation will converge in real-time for all cases.

The plant vehicle and environment simulation is modeled in Python, and relevant CIS controller information is passed into IPOPT. When parsing the plant vehicle and environment, the simulation passes the exact vehicle states to the controller, bypassing sensor noise and data acquisition, but still retaining plant deviation from the prediction model due to model fidelity mismatch. The model mismatch is sufficient for the plant model to violate constraints if control commands are optimized once at the beginning and provided in open-loop. However, the 100 ms closed loop timing shows to be sufficiently robust.

The first two simulations begin with an obstacle fixed 47 m down the road and the vehicle traveling at 35 m/s. As a baseline, the approximate braking distance is estimated by calculating the required lateral acceleration to remain in the center lane, then calculating the peak deceleration available so as to not exceed the 0.8g coefficient of friction limit. For this scenario, the simulated vehicle requires approximately 79 m for limit braking. In practice, vehicles are capable of decelerating faster due to aerodynamic drag; however, humans prefer to decelerate less aggressively at around 0.4g [19], requiring 169 m to brake.

A. OUTSIDE LANE CHANGE

Consider first an outside lane change, representing a change into the left lane for a right hand turn. Fig. 3 shows four concurrent plots highlighting different aspects of the maneuver. The first plot shows the $x - y$ trajectory through the lane corridor, second plot shows the front and rear wheel angles, third plot shows the front and rear slip angles, and fourth plot shows the front and rear steering rate commands.

This is a comparatively easy maneuver, because the vehicle begins by relaxing the tires and opening up the radius to the turn. Although the vehicle does turn to the left initially, this is a brief maneuver immediately followed by a long sweeping right turn. The vehicle holds the turn, just leaving the starting lane when passing the obstacle, and then just coming to the edge of the outer lane limit before stabilizing for the end of the maneuver.

This single lane change to the outside reaches a peak tire slip of approximately 4.6° , representing about 86% of the available tire force. Because the objective is structured to minimize the peak tire slip, there is no penalty for holding the vehicle at that peak slip for prolonged periods of time. As a result, the optimal solution turns left to maximum tire slip of $+4.5^\circ$, then begins the sweeping turn to the right, during which the vehicle is loaded to a tire slip of -4.6° .

There is some deviation from the peak slip later in the trajectory for two reasons. First, the controller acts in the

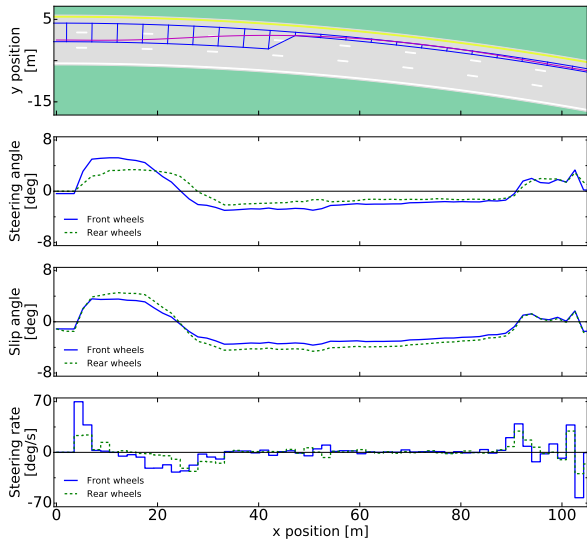


FIGURE 3. A CIS maneuver to the outside lane. The vehicle identifies an obstacle 47 m away, and begins a quick left turn followed by a long sweeping right turn into the next lane.

receding horizon, and minimizes the peak slip from that point forward. Even though the vehicle reaches a higher peak slip earlier in the maneuver, the closed-loop solution will only push to the peak slip if necessary. Second, the discrepancy between the 3 DoF prediction model and the 14 DoF plant model becomes apparent, causing small perturbations throughout the maneuver. The 14 DoF plant includes higher order dynamics, such as suspension response, which the controller cannot account for, but does correct for in the closed-loop through feedback.

Additionally, the importance of the computational speed becomes apparent from this maneuver. When the host vehicle identifies the obstacle at some distance into the future, it does not know if a safe CIS maneuver exists, thus does not immediately take action. Instead, the CIS controller begins solving for the maneuver, seeding the trajectory as where it thinks the host vehicle will be 100 ms into the future. As a result, there is no control intervention over the first 100 ms, effectively bringing the obstacle 3.5 m closer.

B. INSIDE LANE CHANGE

Next, consider an inside lane change, where the vehicle changes into the right lane for a right hand turn. Fig. 4 shows the same four concurrent plots as the previous example.

This is a comparatively difficult maneuver, because the vehicle must make a more aggressive turn-in than the starting trajectory. The inside CIS maneuver begins with a stronger turn to the right to begin the lane change, holds that tighter turn for a stretch, then makes a left turn to countersteer. Similar to the outside lane change, the vehicle just changes lanes when passing the obstacle to the inside, and travels to the edge limit of the inside lane. The maneuver finishes with the vehicle settled in the inside lane at the new steady state terminal trajectory.

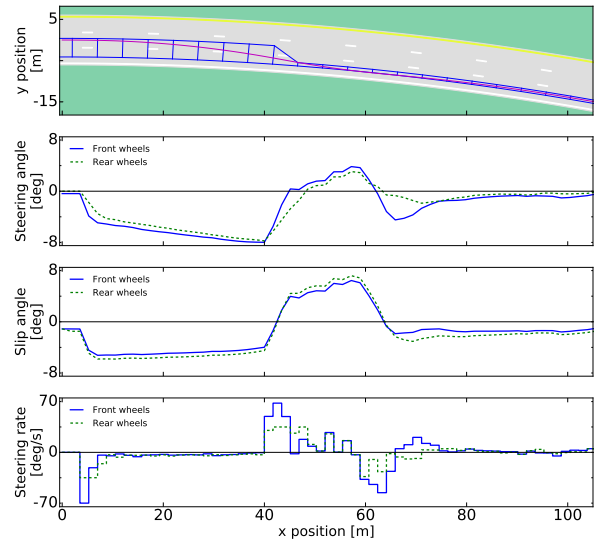


FIGURE 4. For a CIS maneuver to the inside lane, the vehicle must quickly further load the tires to a tighter turn, which is comparatively more aggressive. Additionally, just prior to passing the obstacle, the host begins an equally aggressive countersteer to avoid turning too far into the inside lane.

One difference with the inside lane change is the countersteering takes place just before crossing the obstacle. This is because the MPC controller can see that the right lane boundary constraint will be active, and must avoid turning too far to the right. This can be difficult for a non-professional human driver, as identifying the perfect moment to initiate the countersteer is not trivial. However, the MPC accounts for the future states, and makes the appropriate corrections in a timely manner.

Additionally, the nature of the necessary countersteer emphasizes the need for a one-level MPC architecture. It would be difficult to identify an optimal drivable $x - y$ reference trajectory *a priori* requiring such a countersteer, and would be challenging to develop a controller to safely follow said reference trajectory as it drives at the edges of the drivable tube and requires operating at the limits of handling.

This inside lane change requires a peak tire slip of approximately 7.2° , corresponding to approximately 95% of the available tire force. While the CIS controller is able to find a maneuver to perform the inside lane change for the obstacle at 47 m away, this is approaching the minimum distance for which an inside CIS maneuver can be feasible.

While the outside lane change is less aggressive than the inside lane change, potential obstacle occlusion is not considered in this work, which might prevent ensuring the outside lane is safe for a lane change. Hence, a double lane change maneuver may be necessary, which is investigated next.

C. OUTSIDE DOUBLE LANE CHANGE

In the previous simulations, the vehicle performs a single lane change, where the terminal position is in either the outer or inner lane. Alternatively, a double lane change maneuver can be conceived, where the vehicle must return to the starting lane after passing the obstacle. In this scenario, the center lane

is blocked between 57 m and 67 m, and the outside lane is blocked at 97 m. Thus, the host vehicle must leave the starting lane within 57 m, return to the center lane before 97 m but after 67 m, and must settle in the original lane at the end of the trajectory. The resulting four concurrent plots can be seen in Fig. 5.

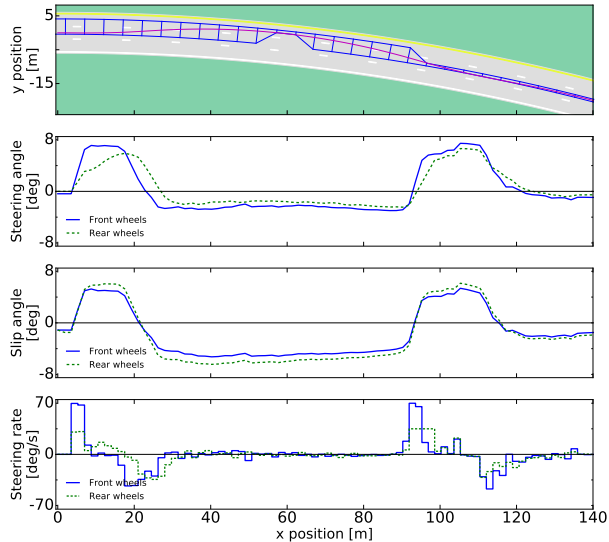


FIGURE 5. An outside double lane change CIS maneuver. The vehicle identifies an obstacle 57 m away, but must find a trajectory that returns to the initial lane on completion.

Similar to the outside single lane change, the outside double lane change begins with a quick turn to the left and then a sweeping turn to the right. However, based on the obstacle distance and outer lane restriction at 97 m, the vehicle countersteers comparatively sooner. In contrast to the single lane change, the double lane change does not reach the outer lane boundary, but rather must begin re-entering the starting lane as soon as it passes the obstacle.

The nonlinear MPC is able to properly account for the outer lane restriction, and identifies that the least aggressive maneuver requires an earlier countersteer to allow the vehicle to settle in the starting lane in time. Further, it is difficult to generate an optimal reference trajectory *a priori* that minimizes peak tire excitation, highlighting the importance of a one-level architecture.

For this example, the outer lane is artificially restricted at 97 m, mandating a double lane change. If this restriction were not in place, and either the outer lane or starting lane are equally acceptable terminal lanes, then a higher level controller would have to evaluate the target terminal lane. This can be challenging for the controller, because based on the obstacle distance and prediction horizon, it cannot be guaranteed there is sufficient travel distance after the obstacle for the host to return into the starting lane. This can be problematic, as the MPC cannot identify sufficient space to return, and can only conclude it is an infeasible problem. Thus, the decision coming from a higher level controller to perform a single lane change versus double lane change must be conscious of this corner case. In general, for maneuvers

that require the vehicle to operate up to the limits of handling, the CIS controller provides an alternative option to collision. If the controller fails to converge on a feasible safe trajectory, the maneuver is not initiated, as the controller cannot guarantee that its actions would avoid the imminent collision safely.

Based on these evaluations, it is concluded that the CIS controller developed with regards to a drivable tube can adapt to various types of CIS maneuvers. Thus, this controller can expand to variations in the vehicle models, road curvatures, speeds, and highway settings.

D. COMPARISON TO TWO-LEVEL PATH FOLLOWING

The controller developed in Sec. III incorporates specific considerations for curved highways within a one-level architecture. Due to these specific considerations, it is difficult to make a direct comparison to state-of-the-art controllers, as differences in factors such as intended speed, application environment, and obstacle distance prevent a fair comparison. Instead, to capture a common approach in the literature, the developed one-level controller is modified to create a two-level path following CIS maneuver as a benchmark.

First, an input reference path is generated from the drivable tube. The drivable tube is constructed in a topographical x - y space, thus the reference trajectory and controller fitness are based on x - y information. As the drivable tube is assumed to be readily available, the reference trajectory is obtained as a sequence of line segments connecting the middle of the tube by averaging the vertices $[r_k, l_k]$, resulting in $\bar{r}_k = p_k$. For the example, in Sec. IV-A, the reference trajectory splits the middle and outside lanes prior to passing the obstacle, then transitions to the middle of the outside lane thereafter.

To generate the path following controller, the fitness metric in (13) is modified to minimize the squared lateral displacement along the reference. Similar to how the lateral displacement constraints are generated in (5), (6), this fitness formulation avoids having to index the reference position to each discrete integration point in the prediction horizon. The path following objective function is calculated as follows, leveraging the active quadrangle k and associated nomenclature same as (5), (6).

$$J(x) = \sum_{i=1}^n \left(\frac{(\overrightarrow{p_k x_i})_y (\overrightarrow{p_k p_{k-1}})_x - (\overrightarrow{p_k x_i})_x (\overrightarrow{p_k p_{k-1}})_y}{|\overrightarrow{p_k p_{k-1}}|} \right)^2 \quad (14)$$

This leverages the scalar vector rejection of the discrete integration position on the reference trajectory, where $(\cdot)_x$ and $(\cdot)_y$ represent the x and y components of a given vector (\cdot) , respectively.

This objective function strictly tries to reduce the squared orthogonal displacement of the state trajectory on the reference trajectory without consideration of the relative ψ orientation or other vehicle states. This updated objective function is substituted into (13), and the resulting controller is applied to the outside single lane change. Fig. 6 shows a path following CIS maneuver for an obstacle at 55 m away.

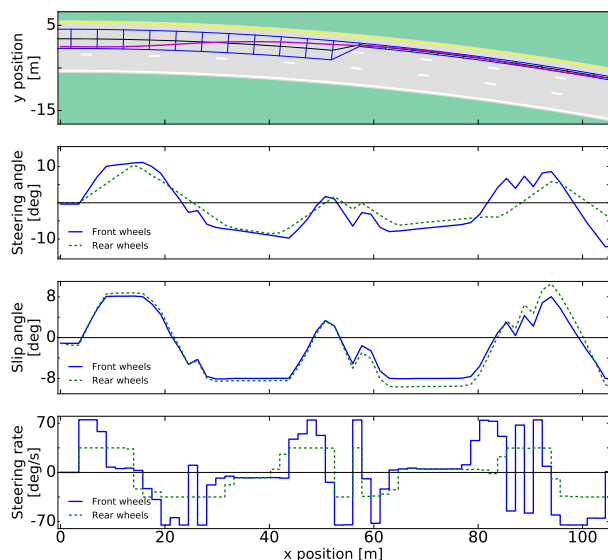


FIGURE 6. Two-level path following CIS maneuver results. In this simulation, the vehicle is allowed to use both front steering and independent active rear steering. The reference trajectory splits the drivable tube in the middle, as seen by the black line in the first subplot. The controller is able to perform the lane change maneuver, but performs a double left steer-right steer maneuver prior to passing the obstacle to better track the transition quadrangle.

While the controller still obeys the strict lateral lane boundary constraints, the controller is allowed to and actively operates at the dynamic handling limits of the vehicle. This can be seen in the third subplot, where the front and rear slip angles approach the maximum allowable limit multiple times. However, the shape of the tire slip plot is critically different from the slip minimization controller, as seen in the slip sequence prior to passing the obstacle. In the path following controller, the minimum deviation solution requires the turning to the left then the right, and again to the left and to the right as passing the obstacle. This is to best track the middle of the drivable tube throughout the entire maneuver, still subject to the drivable tube constraints. In the first left-right sequence, the vehicle is pushed to the middle of the drivable space, then the second left-right sequence sets the vehicle up to track the transition quadrangle as well as complete the lane change maneuver when passing the obstacle.

This is a critical distinction compared to the minimum slip controller. With the path following objective, the controller will always push the prediction model to operate at the limits of handling, as this condition will better track the reference trajectory. While the predicted maneuver is considered safe by the same CIS hard constraints, this is an undesirable operating condition for closed-loop control due to the discrepancies between the prediction model and the plant. While both the minimum slip controller and path following controller are subject to the same model discrepancies, because the latter controller pushes the vehicle to its limits of handling, it has no buffer to avoid accidentally pushing the plant beyond the peak allowable tire slip. Indeed, this has occasionally occurred in simulation, where in mid-maneuver iterations the prediction model is initialized at the measured x_0 with a tire

slip exceeding the tire slip constraint, thus resulting in an infeasible trajectory and optimization convergence failure.

Overall, the path following controller formulation encourages the vehicle to operate at the peak limits of handling, which is an undesirable operating condition that should only be entered if absolutely necessary for safety. While there are many options to modifying the path following controller such as additional states in the reference trajectory, modifications to generating the reference trajectory, and controller action penalty, these are all tuning parameters and cannot universally address all CIS maneuver conditions. Thus, it is advisable to avoid the path following controller formulation in favor of the minimum slip formulation.

V. CONCLUSION

In this work, a new one-level nonlinear MPC controller is developed and shown to solve the CIS maneuver planning and control problem on curved roads. The controller is provided a drivable tube, within which the prediction model is considered safe. The MPC controller then attempts to find a steering sequence that keeps the vehicle entirely within the drivable tube as a hard safety constraint while obeying vehicle dynamics and stability constraints.

A curved highway section is constructed to evaluate different CIS options. By adjusting the drivable tube as input to the controller, the controller is capable of performing both inside and outside lane changes, with double and single lane change options. Numeric simulations show both inside and outside CIS scenarios can change lanes in less distance than emergency braking. Based on the expandability of the drivable tube formulation, it is reasonable to expect that scenarios with different highway speeds, road curvature, and lane sizing from the example case shown herein can be directly incorporated into the existing framework. Further, various driving scenarios such as straight-curve-straight sections, variable curvature, and inflecting curvature can be captured by the drivable tube concept and natively used in the CIS controller.

One simplification made for simulation purposes is between the interaction of the prediction model with the plant model and environment. The simulated plant model and environment are known exactly and the relevant information directly transcribed into the form the prediction model is expecting. As a result, difficult-to-measure parameters such as coefficient of friction, lateral velocity, and lane boundaries are known exactly and passed into the prediction model and controller.

To address this simplification, future directions should evaluate the performance of the CIS controller under sensor noise and parameter uncertainty. The results herein, which show a CIS maneuver can safely avoid collision when braking alone is not an option, are based on accurate knowledge of the plant vehicle and environment. For physical vehicle implementation, it is important to validate the closed-loop iterations of the CIS controller retain the safety metrics, and, if need be, make adjustments to the formulation to account for uncertainty.

REFERENCES

- [1] J. Wurts, J. L. Stein, and T. Ersal, "Collision imminent steering at high speed using nonlinear model predictive control," *IEEE Trans. Veh. Technol.*, vol. 69, no. 8, pp. 8278–8289, Aug. 2020.
- [2] J. Wurts, J. L. Stein, and T. E. Ersal, "Collision imminent steering using nonlinear model predictive control," in *Proc. Amer. Control Conf.*, Jun. 2018, pp. 4772–4777.
- [3] I. Chakraborty, P. Tsiotras, and R. Sanz Diaz, "Time-optimal vehicle posture control to mitigate unavoidable collisions using conventional control inputs," in *Proc. Amer. Control Conf.*, Jun. 2013, pp. 2165–2170.
- [4] S. J. Anderson, S. C. Peters, T. E. Pilutti, and K. Iagnemma, "An optimal-control-based framework for trajectory planning, threat assessment, and semi-autonomous control of passenger vehicles in hazard avoidance scenarios," *Int. J. Vehicle Auto. Syst.*, vol. 8, nos. 2–4, pp. 190–216, 2010.
- [5] J. Liu, P. Jayakumar, J. L. Stein, and T. E. Ersal, "Combined speed and steering control in high-speed autonomous ground vehicles for obstacle avoidance using model predictive control," *IEEE Trans. Veh. Technol.*, vol. 66, no. 10, pp. 8746–8763, Oct. 2017.
- [6] C. E. Beal and J. C. Gerdes, "Model predictive control for vehicle stabilization at the limits of handling," *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 4, pp. 1258–1269, Jul. 2013.
- [7] J. T. Betts, "Survey of numerical methods for trajectory optimization," *J. Guid., Control, Dyn.*, vol. 21, no. 2, pp. 193–207, Mar. 1998.
- [8] M. Brown, J. Funke, S. Erlien, and J. C. Gerdes, "Safe driving envelopes for path tracking in autonomous vehicles," *Control Eng. Pract.*, vol. 61, pp. 307–316, Apr. 2017.
- [9] D. Madas, M. Nosratinia, M. Keshavarz, P. Sundstrom, R. Philippsen, A. Eidehall, and K.-M. Dahlen, "On path planning methods for automotive collision avoidance," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2013, pp. 931–937.
- [10] S. D. Cairano, U. V. Kalabic, and K. Berntorp, "Vehicle tracking control on piecewise-clothoidal trajectories by MPC with guaranteed error bounds," in *Proc. IEEE 55th Conf. Decis. Control*, Dec. 2016, pp. 709–714.
- [11] W. Schwarting, J. Alonso-Mora, L. Paull, S. Karaman, and D. Rus, "Safe nonlinear trajectory generation for parallel autonomy with a dynamic vehicle model," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 9, pp. 2994–3008, Sep. 2018.
- [12] N. R. Kapania, J. Subosits, and J. C. Gerdes, "A sequential two-step algorithm for fast generation of vehicle racing trajectories," *J. Dyn. Syst., Meas., Control*, vol. 138, no. 9, Sep. 2016, Art. no. 091005.
- [13] P. A. Theodosis and J. C. Gerdes, "Generating a racing line for an autonomous racecar using professional driving techniques," in *Proc. ASME Dyn. Syst. Control Conf. Bath/ASME Symp. Fluid Power Motion Control*, Jan. 2011, pp. 853–860.
- [14] M. Gerdtz, S. Karrenberg, B. Müller-Bөbler, and G. Stock, "Generating locally optimal trajectories for an automatically driven car," *Optim. Eng.*, vol. 10, no. 4, p. 439, 2009.
- [15] Y. Yoon, J. Mann Park, H. J. Kim, and S. Sastry, "Utilizing parallax information for collision avoidance in dynamic environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2008, p. 4186.
- [16] W. Schwarting, J. Alonso-Mora, L. Pauli, S. Karaman, and D. Rus, "Parallel autonomy in automated vehicles: Safe motion generation with minimal intervention," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2017, pp. 1928–1935.
- [17] J. Wurts, J. L. Stein, and T. E. Ersal, "Minimum slip collision imminent steering in curved roads using nonlinear model predictive control," in *Proc. Amer. Control Conf.*, Jul. 2019, pp. 3975–3980.
- [18] J. Wurts, J. Dallas, J. L. Stein, and T. E. Ersal, "Adaptive nonlinear model predictive control for collision imminent steering with uncertain coefficient of friction," in *Proc. Amer. Control Conf.*, Jul. 2020, pp. 4856–4861.
- [19] *Speed Concepts: Informational Guide*. Federal Highway Admin., Washington, DC, USA, 2009.
- [20] *A Policy on Geometric Design of Highways and Streets-2001*. American Assoc. State Highway and Transp. Officials, Washington, DC, USA, 2001.
- [21] S. M. Erlien, S. Fujita, and J. C. Gerdes, "Safe driving envelopes for shared control of ground vehicles," *IFAC Proc. Volumes*, vol. 46, no. 21, pp. 831–836, 2013.
- [22] T. Shim and C. Ghike, "Understanding the limitations of different vehicle models for roll dynamics studies," *Vehicle Syst. Dyn.*, vol. 45, no. 3, pp. 191–216, Mar. 2007.
- [23] G. J. Heydinger, R. A. Bixel, W. R. Garrott, M. Pyne, J. G. Howe, and D. A. Guenther, "Measured vehicle inertial parameters-NHTSA's data through November 1998," SAE Tech. Paper 1999-01-1336, 1999.
- [24] J. Liu, P. Jayakumar, J. L. Stein, and T. E. Ersal, "A study on model fidelity for model predictive control-based obstacle avoidance in high-speed autonomous ground vehicles," *Vehicle Syst. Dyn.*, vol. 54, no. 11, pp. 1629–1650, Nov. 2016.
- [25] Y. Peng and X. Yang, "Comparison of various double-lane change manoeuvre specifications," *Vehicle Syst. Dyn.*, vol. 50, no. 7, pp. 1157–1171, Jul. 2012.
- [26] J. Wurts, J. L. Stein, and T. E. Ersal, "Increasing computational speed of nonlinear model predictive control using analytic gradients of the explicit integration scheme with application to collision imminent steering," in *Proc. IEEE Conf. Control Technol. Appl.*, Aug. 2018, pp. 1026–1031.
- [27] J. Liu, P. Jayakumar, J. L. Stein, and T. E. Ersal, "A nonlinear model predictive control formulation for obstacle avoidance in high-speed autonomous ground vehicles in unstructured environments," *Vehicle Syst. Dyn.*, vol. 56, no. 6, pp. 853–882, Jun. 2018.
- [28] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Math. Program.*, vol. 106, no. 1, pp. 25–57, Mar. 2006.



JOHN WURTS received the B.S. degree in mechanical engineering from Case Western Reserve University, Cleveland, OH, in 2013, and the M.S. and Ph.D. degrees in mechanical engineering from the University of Michigan, Ann Arbor, MI, in 2018 and 2020, respectively. Prior to his graduate studies, he has worked in the automotive industry with a concentration on vehicle safety through passive crash safety design and active crash features. His research interests include

optimal design of mechanical systems and controllers, with applications to automotive and aerospace industries.



JEFFREY L. STEIN received the B.S. degree in premedical studies from the University of Massachusetts, Amherst, MA, in 1973, and the S.B., S.M., and Ph.D. degrees in mechanical engineering from the Massachusetts Institute of Technology, Cambridge, MA, in 1976, 1976, and 1983, respectively. Since 1983, he has been with the University of Michigan, Ann Arbor, MI, where he is currently a Professor of mechanical engineering. His research interests include computer-based

modeling and simulation tools for system design and control, with applications to vehicle-to-grid integration, vehicle electrification, conventional vehicles, machine tools, and lower leg prosthetics. He has particular interest in algorithms for automating the development of proper dynamic mathematical models, i.e., minimum yet sufficient complexity models with physical parameters.



TULGA ERSAL received the B.S.E. degree from Istanbul Technical University, Istanbul, Turkey, in 2001, and the M.S. and Ph.D. degrees from the University of Michigan, Ann Arbor, MI, USA, in 2003 and 2007, respectively, all in mechanical engineering. He is currently the Associate Research Scientist of the Department of Mechanical Engineering, University of Michigan. His research interests include modeling, simulation, and control of dynamic systems, with applications

to vehicle and energy systems. He is a member of ASME.

• • •