

Friction-Robust Autonomous Racing Using Trajectory Optimization Over Multiple Models

RAJAN K. AGGARWAL (Member, IEEE) AND J. CHRISTIAN GERDES (Member, IEEE)

(Safe Motion Planning and Control for Autonomous Driving Under Multi-Source Uncertainty)

Department of Mechanical Engineering, Stanford University, Stanford, CA 94305 USA

CORRESPONDING AUTHOR: RAJAN K. AGGARWAL (e-mail: raggarw@stanford.edu).

The code supporting this paper is publicly available at <https://github.com/dynamicdesignlab/multimodel-trajectory-optimization>.

ABSTRACT Autonomous vehicle control in low-friction environments should be capable of using all of the available traction at the road to accomplish maneuvering objectives. In these environments, however, the limit of traction is difficult to estimate, which challenges standard motion planning techniques. In this paper, we introduce a trajectory optimization framework that natively incorporates the complex and nonlinear effects of friction uncertainty into the planning process to improve both the performance and robustness of maneuvering at high accelerations. The core approach of the method is to explicitly consider a range of possible dynamics models, *inclusive* of their closed-loop behavior, simultaneously in the optimization. We illustrate this method through a racing example, where the minimum-time objective facilitates intuitive performance and robustness metrics (lap time and tracking error limits), all while necessitating vehicle maneuvering through nonlinear and friction-sensitive regions of the state space. Experiments on an autonomous VW Golf GTI on a challenging winter ice track demonstrate the efficacy of this approach.

INDEX TERMS Autonomous vehicles, extreme maneuvering, friction uncertainty, robust trajectory optimization.

I. INTRODUCTION

Autonomous vehicles have the potential to significantly improve road safety by coordinating vehicle inputs in emergency scenarios to avoid collisions. To realize broad deployment of this technology, autonomous controllers will need to cope with a wide variety of difficult real-world weather conditions. Winter road conditions are of special interest, as approximately 70% of U.S. roadways receive 5 or more inches of snow per year [1]. Despite this relevance, reliable autonomous control in these environments remains an open challenge. Snow and ice significantly reduce the available traction at the road, which limits vehicle maneuverability. Even low-speed driving on low-friction surfaces can exceed the tires' friction potential and result in a loss of control. Furthermore, the significant variability in the road conditions makes it challenging to predictably maneuver, particularly in situations that demand high accelerations. In pursuit of expanding autonomous vehicles' operational domain, in this paper we present a controller that can manage large friction uncertainty in extreme

handling scenarios while commanding precise control over the vehicle's motion.

A. RELATED WORK

Managing model uncertainty is well-studied in the control systems literature, with origins dating to early developments in the field [2], [3]. Traditionally, when model uncertainty becomes too large for deterministic control, it is addressed with either adaptive or robust techniques.

Adaptive methods attempt to estimate the uncertain model parameters in real time, treat the resulting model as deterministic, and then adapt the control law accordingly [4], [5]. While there have been some studies that demonstrate the basic feasibility of this approach to autonomous vehicle control [6], [7], [8], one fundamental challenge in adaptive control is adapting quickly to rapid changes in the underlying dynamics model [9]. When driving on low-friction surfaces, sudden variations between snow and ice are possible; therefore, knowledge of the friction parameter at a given instant



FIGURE 1. The test vehicle, named Norbert, that helped validate the multi-model controller described in this paper. Norbert is an autonomous 2018 VW Golf GTI, shown here in the challenging winter test environment detailed in Section VII.

may not confidently predict its value in the immediate future. In order to successfully navigate these types of roads, autonomous control must be able to reason with this underlying friction variation.

Robust methods, in contrast, incorporate an uncertainty range *directly* into the motion planning process [10], [11], [12], [13]. The main benefit of robust control is that it achieves consistent performance despite different realizations of the uncertainty. Due to the inherent variability of winter road conditions, we focus on this class of method in this paper.

For application to trajectory optimization, one popular type of robust control is *tube motion planning*. The basic idea in tube motion planning is to plan for a trajectory assuming that a predefined feedback action will help counteract model mismatches and disturbances as they occur in the closed-loop. The set of possible trajectories may thus be bounded by a tube, which is then steered to remain within the acceptable space [14], [15]. While these tubes are sometimes amenable to direct computation, as in the case of linear systems, their computation for systems with nonlinear dynamics and uncertainty remains challenging. Continued research has thus aimed to extend the tube planning technique to more general applications [16].

The recent work by Gao et al. perhaps best represents the state of the art in applying tube motion planning to extreme vehicle handling scenarios [17]. In their paper, the authors focus on an emergency double-lane change maneuver on a challenging winter road. They incorporate the effect of friction uncertainty to compute an uncertainty tube around the nominal trajectory and ensure that this tube remains within the allowable drivable region. They successfully demonstrate their approach on a real test vehicle at a winter proving ground.

In order to overcome some of the challenges with tube planning for nonlinear systems, the authors in Gao et al. made a few key assumptions. These included linearizing aspects of the vehicle model and using empirical data to approximate the effects of the friction uncertainty as a linear disturbance. As the authors acknowledge, these approximations worked well for their test case, but limit the applicability of their method

to more general extreme handling scenarios. Indeed, for these types of scenarios, a study by Weber et al. [18] found that professional drivers consistently *relied on* the vehicle's nonlinear dynamics—in particular, the nonlinear force coupling in the tire behavior—to precisely maneuver on low-friction surfaces. As discussed in the recent survey paper on autonomous racing by Betz et al., vehicle motion planning methods that can inherently incorporate these nonlinear dynamics and uncertainty models remains an open challenge for the field [19].

In pursuit of this goal, we can look to investigate methods distinct from tube motion planning. One promising alternative, proposed by the robotics community, is the concept of *multiple-model* trajectory optimization. In this type of optimization, one plans simultaneously for a fixed number of discrete trajectories, where each one assumes feedback to a nominal trajectory under a different realization of the model uncertainty [20]. A significant benefit of multi-model motion planning over tube motion planning is that it can easily incorporate general nonlinear dynamics and parametric uncertainty. The results by Atkinson [21], Mordatch et al. [22] and Howell et al. [23] demonstrate the basic utility of this approach on different nonlinear robotic systems, holding promise for application to low-friction vehicle maneuvering.

B. CONTRIBUTIONS

The versatility of the multiple-model method and its intuitive construction makes it well-suited to our overall objective. We present three contributions to demonstrate the practical feasibility of extending the multi-model method to vehicle control on uncertain terrain. These contributions are related to the design, analysis, and testing of the multi-model framework for this application:

- 1) We extend the concept of multiple-model motion planning to the domain of vehicle dynamics in extreme maneuvering scenarios with friction uncertainty. We present a motion planner that achieves both performance and robustness by jointly optimizing over multiple discrete nonlinear models, advancing beyond the linearity approximations used in previous work [17].
- 2) We provide a physically-motivated analysis of the resulting robust maneuvering *strategy*, highlighting that the solution trajectories are selectively aggressive at different parts of the trajectory. Prior presentation in multiple-model research has focused primarily on final metrics of the controller (e.g., quantifying the performance and robustness); in this paper, we complement this analysis by discussing how it may be achieving these results.
- 3) Critically, we demonstrate the effectiveness of this approach through real experiments on an autonomous VW Golf GTI in a challenging snow and ice environment. Despite significantly varying friction coefficients—up to 20% variation within a given trial—our method exploits the higher friction when available to advance the performance objective, while maintaining tight tracking performance, even when the track traction

degrades. We empirically highlight significant improvements over deterministic control schemes, schemes that have been previously demonstrated to achieve lap times comparable to a champion amateur race car driver [24].

To support future research and reproducibility, the code supporting this paper is publicly available at <https://github.com/dynamicdesignlab/multimodel-trajectory-optimization>.

C. PAPER ORGANIZATION

The rest of this paper is structured as follows. We first introduce the general concept of multiple-model optimization in Section II, highlighting the inclusion of feedback action to counteract the effects of model mismatch. Section III describes the vehicle dynamics model used in this paper and discusses the friction coefficient as a significant source of model uncertainty. Next, we motivate our specific case study in Section IV, minimum-time maneuvering with friction uncertainty, and formulate the multiple model method to this problem. Trajectory optimization solutions are then investigated in Section V, where we observe that the resulting multi-model trajectory appears to be selectively aggressive at different parts of the track. Section VI evaluates how the multi-model controller behaves in simulation with various friction coefficients and abrupt friction changes, illustrating a basic robustness to these situations despite not modeling them explicitly a priori. Section VII then presents the experimental results showing the controller's ability to achieve both performance and robustness despite significant friction variation on a winter racetrack. Lastly, we discuss directions for future work and conclude in Section VIII.

II. CONCEPT OF MULTIPLE-MODEL OPTIMIZATION

The optimization considers general nonlinear dynamics of the form

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}; \boldsymbol{\theta}),$$

where $\mathbf{x} \in \mathbb{R}^{n_x}$ is the state vector and $\mathbf{u} \in \mathbb{R}^{n_u}$ is the control input vector. The uncertain parameter vector is given by $\boldsymbol{\theta} \in \mathbb{R}^{n_\theta}$, where the nominal values are represented by $\boldsymbol{\theta}^0$. In this paper, we target our concept on systems with smooth dynamics that can generally be optimized over using established trajectory optimization techniques.

The overall goal of the trajectory optimization is to maximize performance while ensuring constraint satisfaction for discrete realizations of the parameter vector $\boldsymbol{\theta}$ ($\boldsymbol{\theta}^0$ and $\boldsymbol{\theta}^1, \boldsymbol{\theta}^2, \dots, \boldsymbol{\theta}^n$). We assume that, in the closed-loop, a lower-level controller attempts to track the nominal trajectory $\mathbf{x}^0(t)$. The inclusion of feedback action is critical in curtailing the spread, or uncertainty propagation, of the state evolution under parametric mismatch. The optimization explicitly considers this feedback and jointly optimizes over the $(1+n)$ trajectories. The resulting continuous-time optimal control

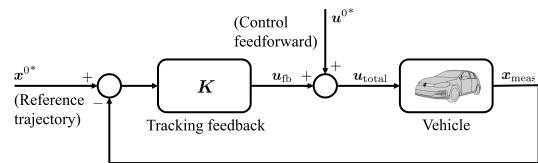


FIGURE 2. Block diagram depicting the overall control architecture. Offline, the multi-model trajectory optimization finds a reference trajectory x^0* and associated set of feedforward controls u^0* . Online, the controller attempts to track the reference trajectory by using simple linear feedback K written to the tracking error $(x^0* - x_{\text{meas}})$. This feedback u_{fb} augments the feedforward to form the total control effort to the vehicle, u_{total} . The n auxiliary solution trajectories from Opt. 1 are not directly used during runtime in the vehicle; instead, their influence is felt indirectly through the design of the nominal trajectory (x^0*, u^0*) .

problem is as follows:

$$\begin{aligned}
 & \underset{(x^0, u^0), (x^{(1:n)}, u^{(1:n)})}{\text{minimize}} \quad J^0 + J^1 + J^2 + \dots + J^n \\
 & \text{subject to} \\
 & \dot{\mathbf{x}}^0 = f(\mathbf{x}^0, \mathbf{u}^0; \boldsymbol{\theta}^0) \\
 & \dot{\mathbf{x}}^{(1:n)} = f(\mathbf{x}^{(1:n)}, \mathbf{u}^{(1:n)}; \boldsymbol{\theta}^{(1:n)}) \\
 & \mathbf{u}^{(1:n)} = \mathbf{u}^0 - \mathbf{K}(\mathbf{x}^0, \mathbf{x}^{(1:n)}) \\
 & \mathbf{x}^0, \mathbf{x}^{(1:n)} \in \mathcal{X} \\
 & \mathbf{u}^0, \mathbf{u}^{(1:n)} \in \mathcal{U}.
 \end{aligned} \tag{1}$$

The nominal state trajectory and control efforts are given by $(\mathbf{x}^0, \mathbf{u}^0)$ with a nominal parameter vector $\boldsymbol{\theta}^0$. The n rolled-out tracking trajectories are given by $(\mathbf{x}^1, \mathbf{u}^1), \dots, (\mathbf{x}^n, \mathbf{u}^n)$ for each of $\boldsymbol{\theta}^1, \dots, \boldsymbol{\theta}^n$. The total cost is the sum of costs on each trajectory, and the function f is the dynamics evolution function. The sets \mathcal{X} and \mathcal{U} define the acceptable state and control spaces, respectively. The feedback action is represented by the matrix \mathbf{K} , which maps the reference and measured states to a feedback control input. (In this formulation, we assume the feedback policy is fixed ahead of time, but an interesting area for future work is to explore co-optimizing this feedback with the trajectories themselves.) Lastly, we refer to solutions of this optimization using asterisks, e.g. $(\mathbf{x}^{0*}, \mathbf{u}^{0*})$.

We remark that, when solutions to Optimization 1 are found, nonlinear constraint satisfaction is guaranteed for all modeled parameter values. This robust constraint satisfaction is achieved due to the construction of the optimization, without needing to explicitly compute state tubes or tighten constraints beforehand. Fig. 2 depicts a block diagram showing how the multiple-model optimization (termed *multi-model* for brevity) fits into the overall vehicle control structure. While the multi-model optimization plans for $(1+n)$ trajectories, only the nominal trajectory solution \mathbf{x}^{0*} is used as the tracking objective of the control loop; the other solution trajectories influence the design of this main trajectory but are not used at runtime.

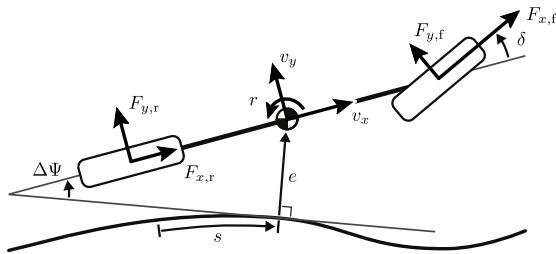


FIGURE 3. The vehicle diagram, its states, and tire force vectors. On each axle, this model groups the left and right tires together, resulting in the single-track abstraction.

III. VEHICLE DYNAMICS MODEL

This section introduces our application of extreme vehicle maneuvering with model uncertainty. We review our vehicle dynamics model and motivate the tire-road friction coefficient as an influential source of parametric uncertainty.

A. SINGLE-TRACK VEHICLE MODEL

We describe the vehicle's motion using a single-track model, chosen for its combination of descriptive ability and computational simplicity. This approach groups the left and right tire on each axle into a single tire, as depicted in Fig. 3, and models how these single tires produce forces to effect vehicle motion. This approximation has been demonstrated to capture actual vehicle motion well, in a wide variety of handling regimes (including emergency collision avoidance [25], racing [26], and drifting [27]). The corresponding vehicle state is defined as

$$\mathbf{x} := (v_x, v_y, r, s, e, \Delta\psi, \Delta F_z)^T,$$

where v_x is the vehicle's longitudinal velocity, v_y is the lateral velocity, r is the yaw rate, s is vehicle's track progress coordinate, e is the lateral position relative to the track centerline, $\Delta\psi$ is the vehicle heading angle relative to the track centerline, and ΔF_z is the longitudinal weight transfer (defined as positive for weight transfer rearward).

The control input vector is defined as

$$\mathbf{u} := (\delta, F_x)^T,$$

where δ is the front axle steering angle and F_x is the total longitudinal force command. The F_x command is further partitioned into front and rear axle force commands, $F_{x,f}$ and $F_{x,r}$, according to the drivetrain and braking layout of the vehicle. In this paper, our test vehicle is front wheel drive, so we direct positive F_x commands fully to the front axle. For negative F_x commands, we partition the braking command across both the front and rear axles according to a fixed distribution. This overall distribution logic is encapsulated in a smooth partitioning function; we use the function tested by Laurence and Gerdes [28] with parameters as listed in Appendix A.

Given this state and control input formulation, the equations of motion for the single-track model follow from rigid body

dynamics. The resulting equations are as follows:

$$\left\{ \begin{array}{l} m\dot{v}_x = -F_{y,f} \sin \delta + F_{x,f} \cos \delta + F_{x,r} - F_d + mrv_y \\ m\dot{v}_y = F_{y,f} \cos \delta + F_{x,f} \sin \delta + F_{y,r} - mr\dot{v}_x \\ I_z \dot{r} = a(F_{y,f} \cos \delta + F_{x,f} \sin \delta) - bF_{y,r} + M_{z,b} \\ \dot{s} = (v_x \cos \Delta\psi - v_y \sin \Delta\psi)/(1 - \kappa e) \\ \dot{e} = v_x \sin \Delta\psi + v_y \cos \Delta\psi \\ \dot{\Delta\psi} = r - \kappa \dot{s}. \end{array} \right.$$

Here, m is the mass of the vehicle, I_z is the polar inertia about the body z -axis, κ is the local track centerline curvature, a is the distance from the center of mass (c.o.m.) to the front track, b is the distance from the c.o.m. to the rear track, and $M_{z,b}$ is the brake yaw moment due to a lower-level differential brake controller (discussed further in Section III-C1). A drag term F_d is included which sums the effects of rolling resistance $C_{d,0}$ and aerodynamic resistance $C_{d,2}v_x^2$. The variables $F_{[x,y],[f,r]}$ represent the tire forces in the tire frame x - and y -directions, for the front and rear axles, respectively. The sign convention of these forces is included in Fig. 3 for clarity.

Lastly, the axle normal loads affect the maximum forces each tire can react and may significantly change during extreme maneuvering. To capture this effect, we append longitudinal weight transfer dynamics to the single-track model, using first-order dynamics of the form

$$\Delta\dot{F}_z = \frac{1}{\tau_{\text{long}}} \left(\frac{ma_x h_{\text{com}}}{L} - \Delta F_z \right). \quad (2)$$

As proposed in Subosits and Gerdes [29], this model approximates the behavior of weight transfer in a well-damped suspension. The variable τ_{long} is the corresponding time constant, L is the wheelbase ($L := a + b$), and a_x is the vehicle's inertial acceleration in the body x -direction. In terms of the vehicle's state \mathbf{x} and its derivative $\dot{\mathbf{x}}$, a_x can be represented via the kinematic relation $a_x = \dot{v}_x - rv_y$. With this incorporation of weight transfer, the front and rear axle normal loads are given by

$$F_{z,f} = \frac{b}{L}mg - \Delta F_z \text{ and } F_{z,r} = \frac{a}{L}mg + \Delta F_z,$$

with the gravitational acceleration constant g taken as 9.81 m/s^2 .

B. TIRE FORCE MODEL

The lateral tire forces $F_{y,f}$ and $F_{y,r}$ are functions of the lateral slip angles of each axle. The slip angle represents the difference in heading between the wheel's pointed direction and its velocity. They are defined for the front and rear axles as

$$\alpha_f := \tan^{-1} \left(\frac{v_y + ar}{v_x} \right) - \delta \text{ and } \alpha_r := \tan^{-1} \left(\frac{v_y - br}{v_x} \right).$$

We model the resulting tire forces with an F_x de-rated Fiala brush tire model, as described in Pacejka [30]. This tire model captures the dominant tire behavior for the handling regime of

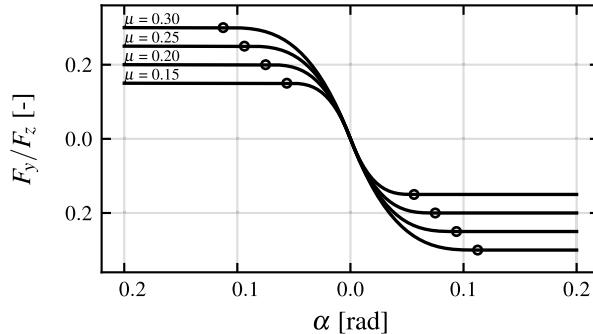


FIGURE 4. The lateral tire force model from (3) plotted for different friction coefficients in the snow and ice regime. As the slip angle α is increased, the tire produces more lateral force up until it reaches the slip limit α_{sl} . At this point, denoted by the \circ marks above, the lateral force is saturated and equivalent to μF_z . Extreme vehicle maneuvering often operates tires in their high-slip region, where the force sensitivity to friction is particularly high: this underscores the importance of friction robustness. (Full parameter set for the above plot: $F_x = 0$ kN, $F_z = 10$ kN, $C'_\alpha = 8$ rad $^{-1}$.)

interest and is amenable to trajectory optimization:

$$F_y = \begin{cases} -C_\alpha \tan \alpha + \frac{C_\alpha^2}{3F_y^{\max}} |\tan \alpha| \tan \alpha - \frac{C_\alpha^3}{27(F_y^{\max})^2} \tan^3 \alpha & \text{if } |\alpha| \leq \alpha_{sl}, \\ -F_y^{\max} \operatorname{sgn} \alpha & \text{otherwise.} \end{cases} \quad (3)$$

In this model, the inputs are the longitudinal force command F_x and slip angle α ; the parameters are the cornering stiffness C_α and friction coefficient μ ; and the output is the lateral force F_y . Fig. 4 depicts this function for varying values of the friction parameter μ .

The limit slip angle α_{sl} is further defined as

$$\alpha_{sl} := \tan^{-1}(3F_y^{\max}/C_\alpha),$$

and F_y^{\max} is computed using a friction circle limit:

$$F_y^{\max} := \sqrt{(\mu F_z)^2 - F_x^2}.$$

Lastly, to improve the model's compatibility with the gradient-based solvers used later in Section V, we augment the tire model with two features: a slight nonzero slope in the saturation regime (to avoid a zero-gradient), and a near-unity scalar to multiply F_x in the F_y^{\max} calculation (to avoid an undefined gradient). These features were initially proposed in Laurence and Gerdes [28]; the parameters we use for these adjustments are included in Table 1.

C. SECONDARY EFFECTS OF WEIGHT TRANSFER

The primary motivation of modeling weight transfer in Section III-A is to capture the varying axle normal loads during vehicle maneuvering, which in turn affects the maximum force the tires can generate. However, this weight transfer may also result in secondary effects on the dynamics. In this paper, we include two of these additional effects below: a differential braking yaw moment and a nonlinear cornering

TABLE 1. Nonlinear Program Parameters

Parameter	Variable	Value	Units
Time cost	W_t	1.0	-
Steering slew rate cost	$W_{\dot{\delta}}$	5.0	-
Long. force slew rate cost	$W_{\dot{F}_x}$	5.0	-
Steering gain, lat. pos.	K_e	0.18	rad/m
Steering gain, heading	$K_{\Delta\psi}$	1.5	rad/rad
Long. force gain, speed	K_{v_x}	2.0	kN/(m/s)
Tire sat. slope param. [28]	ξ	0.95	-
Tire F_y^{\max} param. [28]	ρ	0.99	-

stiffness dependence. We have observed empirically that these effects are important to model in situations with high weight transfer, as is characteristic in extreme maneuvering.

1) YAW MOMENT DUE TO DOUBLE-TRACK EFFECTS

The longitudinal force command F_x is partitioned into front and rear axle force commands via a partitioning function, as described in Section III-A. While this is fully prescriptive for the single-track model, these forces must nonetheless be expanded into left and right force commands *within* each axle when running experiments on the (four-wheeled) test vehicle. This expansion may result in additional body yaw moments that should be accounted for in the planning stage and may depend on weight transfer. We assess yaw moments for both the tractive ($F_x \geq 0$) and braking ($F_x < 0$) cases as follows.

For tractive force commands, our test vehicle transmits the engine torque through a front axle differential before powering the wheels. This differential is nominally open, resulting in an even torque split. We therefore model no yaw moment in the tractive case.

For braking force commands, we must command brake force references for each wheel. During cornering, weight transfer occurs laterally, which affects the normal load distribution on the wheels. In order to prevent wheel lockup and extract more braking capability, the brake commands are partitioned within each axle depending on the degree of lateral weight transfer. In this paper, we simply partition each axle's braking force command according to this weight distribution. This partitioning results in a net *brake yaw moment*, modeled as

$$M_{z,b} = \left(\frac{\gamma(h_{com} + gh_l R_\phi) a_y L}{gb - a_x h_{com}} \right) F_{x,f} + \left(\frac{(1-\gamma)(h_{com} + gh_l R_\phi) a_y L}{ga + a_x h_{com}} \right) F_{x,r}.$$

Derivation and parameter details are provided in Appendix B. We note that this moment is only a function of the vehicle's acceleration and brake commands, and hence is simple to append to the dynamics model; the other variables are constant parameters of the vehicle.

2) CORNERING STIFFNESS VARIATION WITH NORMAL LOAD

Tire testing results indicate that the cornering stiffness is not constant as the normal load varies, and that a linear relation

can capture this effect to first order [30], [31]. To include this effect, we model the cornering stiffness C_α as linear with respect to the normal load, $C_\alpha := C'_\alpha F_z$, where C'_α denotes the stiffness sensitivity.

D. DYNAMICS SOURCE OF UNCERTAINTY

In this paper, we focus on the tire-road friction coefficient as the primary source of uncertainty in the dynamics. This coefficient is important to consider for several reasons: i) it dictates the maximum force a tire can react, which defines the limit of vehicle maneuverability; ii) its uncertainty results directly in tire force uncertainty, which in turn integrates into velocity and position prediction uncertainties, and iii) it defines the slip limit of the tire, after which point there is low steering controllability over the vehicle's lateral motion. Despite the friction coefficient's importance for vehicle control at the limits of handling, it remains difficult to estimate online. We therefore focus on addressing this uncertainty due to its simultaneous importance and estimation difficulty.

The uncertain parameter vector is defined as $\theta := (\mu_f, \mu_r)^T \in \mathbb{R}_{>0}^2$, where μ_f and μ_r are the front and rear friction coefficients. With this definition, the single-track (st) dynamics from Eqns (2)-(3) are compactly represented by the function $f_{st}: \dot{x} = f_{st}(x, u; \theta)$. Importantly, the single-track model selected here is not able to natively model the effects of independent friction coefficients on the left and right sides of the vehicle (known as split- μ conditions). The experiments at a winter testing facility in Section VII suggest that the resulting multi-model controller can capture enough of the four-wheel friction uncertainty on the track considered to achieve acceptable path tracking performance, but an important area of future work will be to extend this method to double-track dynamics incorporating split- μ uncertainty models.

IV. MULTIPLE-MODEL TRAJECTORY OPTIMIZATION FOR FRICTION-ROBUST RACING

In this section, we apply the multiple-model optimization to the minimum-time maneuvering control problem. The objective of minimum time is effective at forcing the vehicle to the limit of its grip, where the controller's robustness to friction uncertainty and ability to manage the nonlinear vehicle dynamics can be clearly assessed. Racing in this capacity can thus serve as a simple proxy problem for a multitude of extreme vehicle handling applications.

A. NONLINEAR PROGRAM FORMULATION

The optimization from Section II is now transcribed into a Nonlinear Program (NLP). We assume that the true friction value is unknown but lies between a lower and upper bound. In the NLP, we set the nominal parameter vector θ^0 to the upper friction estimate, as we desire optimistic and high performance out of the controller. The worst-case parameter vector θ^1 is set to the lower friction estimate. All other variables with a superscript of 0 refer to the high-friction rollout, and those with a superscript of 1 refer to the low-friction rollout. The

resulting NLP is as follows:

$$\underset{\substack{(x_{0:N}^0, u_{0:N}^0), \\ (x_{0:N}^1, u_{0:N}^1)}}{\text{minimize}} \quad \frac{1}{2} (J^0 + J^1)$$

subject to

$$\begin{aligned} h_{\text{trap}}(x_j^0, u_j^0, s_j, x_{j+1}^0, u_{j+1}^0; \theta^0) &= \mathbf{0} \\ h_{\text{trap}}(x_j^1, u_j^1, s_j, x_{j+1}^1, u_{j+1}^1; \theta^1) &= \mathbf{0} \\ u_i^1 - (u_i^0 - K(x_i^1 - x_i^0)) &= \mathbf{0} \\ -\frac{w_i}{2} \leq e_i^0, e_i^1 &\leq \frac{w_i}{2} \\ -\delta^{\max} \leq \delta_i^0, \delta_i^1 &\leq \delta^{\max} \\ F_{x_i}^0 v_{x,f_i}^0, F_{x_i}^1 v_{x,f_i}^1 &\leq P_{\text{eng}}^{\max} \\ -\mu_f^0 F_{z,f_i}^0 \cos \alpha_{f_i}^0 &\leq F_{x,f_i}^0 \leq \mu_f^0 F_{z,f_i}^0 \cos \alpha_{f_i}^0. \end{aligned} \tag{4}$$

The subscripts $i = 0, \dots, N$ and $j = 0, \dots, (N-1)$ of a given variable index its location in the N -step horizon. The track width at index i is given by w_i , and the maximum front axle steering angle and engine power values are given by δ^{\max} and P_{eng}^{\max} .

1) NLP OBJECTIVE

The primary objective is to minimize maneuvering time. Two additional terms are added to penalize actuator slew rates, to prevent the NLP from exploiting unmodeled actuator dynamics. The template for cost J is

$$\begin{aligned} J &= W_t \left(\frac{t_N}{t_{\text{norm}}} \right)^2 \\ &+ W_{\dot{\delta}} \sum_{j=0}^{N-1} \frac{1}{N} \left(\frac{1}{\dot{\delta}_{\max}} \frac{\delta_{j+1} - \delta_j}{t_{j+1} - t_j} \right)^2 \\ &+ W_{\dot{F}_x} \sum_{j=0}^{N-1} \frac{1}{N} \left(\frac{1}{\dot{F}_{x,\max}} \frac{F_{x,j+1} - F_{x,j}}{t_{j+1} - t_j} \right)^2, \end{aligned} \tag{5}$$

where J^0 and J^1 are inherited by adding the appropriate superscripts to t , $\dot{\delta}$, and \dot{F}_x appropriately. The cost terms are normalized by reasonable physical values to improve solver performance and promote intuitive tuning with weights $W_{[t, \dot{\delta}, \dot{F}_x]}$. While the experimental results in Section VII suggest this simplified approach to actuator modeling works well for our application, more detailed models may be considered for better transient performance, models that account for effects such as brake pressure buildup, throttle response delays, or actuator slew limits.

2) NLP CONSTRAINTS

The first two constraints enforce dynamic continuity of the trajectories via the function h_{trap} . The dynamics are discretized using a trapezoidal scheme [32]:

$$\begin{aligned} h_{\text{trap}}(\mathbf{x}_j, \mathbf{u}_j, s_j, \mathbf{x}_{j+1}, \mathbf{u}_{j+1}, s_{j+1}; \boldsymbol{\theta}) := \\ \mathbf{x}_{j+1} - \mathbf{x}_j - (1/2)\Delta s (f_{\text{st}}(\mathbf{x}_j, \mathbf{u}_j, s_j; \boldsymbol{\theta}) \\ + f_{\text{st}}(\mathbf{x}_{j+1}, \mathbf{u}_{j+1}, s_{j+1}; \boldsymbol{\theta})). \end{aligned} \quad (6)$$

For this application, we make a minor modification to the dynamics from Section III, translating them from temporal to spatial indexing using the relation $d\mathbf{x}/ds := (1/\dot{s}) \cdot \dot{\mathbf{x}}$. The system state is then redefined to be $\mathbf{x} := (v_x, v_y, r, t, e, \Delta\psi, \Delta F_z)^T \in \mathbb{R}^7$, where t is now added to the state and s becomes the new independent variable. This conversion enables explicit costing of the final time in the NLP and obviates the need to re-interpolate the track curvature κ along the horizon as the NLP solver iterates. Further discussion on the relative benefits between temporal and spatial discretizations for vehicle control applications can be found in Frasch et al. [33] and Laurence et al. [28].

The third constraint employs linear feedback, coupling the trajectories with gain matrix \mathbf{K} . The matrix encodes decoupled lateral and longitudinal feedback via three fixed parameters, K_e , $K_{\Delta\psi}$ and K_{v_x} :

$$\mathbf{K} = \begin{pmatrix} 0 & 0 & 0 & 0 & K_e & K_{\Delta\psi} & 0 \\ K_{v_x} & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

A similar feedback structure was employed in a racing controller by Subosits and Gerdes [29] from which we base our gain selection on in Section V.

Despite the underlying nonlinear dynamics of the vehicle, we found that this simple linear feedback policy worked well for both the simulated and experimental friction cases presented later. We posit that this may be due to a general “linearizability” of the vehicle dynamics for different friction values. Talvala et al. provide additional insights on the effectiveness of linear feedback on lateral vehicle control, demonstrating that linear feedback can stabilize the dynamics even as tires operate in their nonlinear regions [34].

The next three constraints ensure the vehicle stays within the road edges, the steering command is within the actuator limits, and the longitudinal force command is within the engine’s power limit.

The final constraint limits the front axle longitudinal force as a function of $\cos \alpha_f$. This is inspired by analysis presented in Pacejka, which concludes that a pre-generated slip angle limits the amount of longitudinal force a tire can react when the longitudinal slip is limited [30]. Our test vehicle runs slip limiting controllers—in the form of anti-lock braking (ABS) and traction control (TCS) systems—and so we use this model as a reasonable approximation of the $F_{x,f}$ limit.

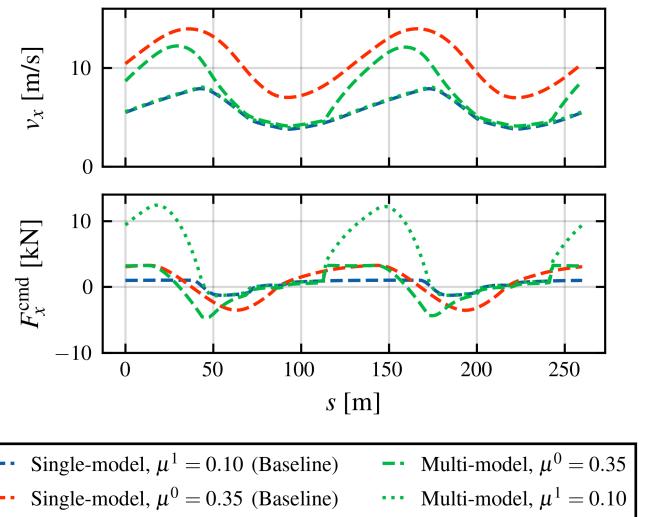


FIGURE 5. Longitudinal speed profile (top) and force commands (bottom) resulting from the single- and multi-model trajectory optimization solutions. Two different single-model solutions are shown, corresponding to the lower friction bound μ^1 and upper friction bound μ^0 . The multi-model optimization plans for the optimistic friction value μ^0 (solid green), but internally maintains a rollout corresponding to experiencing lower friction μ^1 in the closed loop (dashed green). The multi-model velocity profile is selectively aggressive, accelerating significantly faster than the conservative single-model baseline during corner exit but roughly matching its velocity profile through corner entry and apex.

B. ENABLING OPPORTUNISTIC BEHAVIOR

The last constraint, which limits $F_{x,f}^0$, is written only to the upper friction bound. This results in a nominal trajectory solution \mathbf{x}^0 which is optimistic and inclines the closed-loop system to opportunistic behavior. For the low-friction trajectory rollout in the NLP, if longitudinal forces *above* the low-friction limit are probed by the optimization routine, the dynamics function f_{st} first saturates the force command to the appropriate longitudinal force limit before calculating state derivatives. This models the hardware-saturation effect that the slip-limiting controllers will have on the tire’s force. The effects of this design choice on trajectory solutions are discussed in Section V and can be seen in Fig. 5.

C. SELECTING THE NUMBER OF MODELS

The multiple-model optimization presented in Section II can accommodate a theoretically unlimited number of parameter realizations, where the primary trade-off is between descriptive ability and computational effort. In this paper, we build the NLP considering only two realizations of the parameter vector, the lower and upper friction bounds. This decision simplifies the implementation and is guided by an intuition of trajectory *boundedness*. If state and control constraints are satisfied for both the low- and high-friction rollouts, then we anticipate they will be satisfied for any realization of the friction between those two limits. This intuition is proved for the case of locally linear dynamics in Appendix C.

While the analysis provided does not address the full nonlinear dynamics, we validate the method’s performance in

both simulation and experiment on the nonlinear system with varying road friction in Sections VI and VII. Future work should consider investigating nonlinear trajectory boundedness on the theoretical side, perhaps by leveraging additional tools such as using Taylor series remainders to bound uncertainty propagation [35] or developing specialized Lyapunov functions to prove trajectory convergence [36].

V. TRAJECTORY OPTIMIZATION SOLUTIONS

Before simulation and experimental evaluation however, we first study the multi-model optimization solution on a low-friction oval track to better understand its behavior. For comparison, we assess the multi-model solution relative to similar trajectory optimizations that assume only a single friction value (termed *single-model* optimizations here). In particular, we compare the multi-model solution to two single-model solutions: one corresponding to the low-friction bound μ^1 and one to the upper μ^0 . We observe that the multi-model optimization develops its own maneuvering strategy, distinct from the single-model class of solution. It appears that this strategy is able to achieve both high-performance and friction robustness by being selectively aggressive, pushing the vehicle's dynamics only at certain parts of the track.

A. OPTIMIZATION SETUP

The trajectory optimization is applied to a low-friction oval racetrack which is 260 meters long, 6 meters wide, and has a minimum radius of curvature of 18 meters. The vehicle parameters are derived from a 2018 VW Golf GTI and listed in Appendix A. The selected surface friction coefficient range is $\mu^1 = 0.10$ to $\mu^0 = 0.35$, representing a realistic range of uncertainty for nonstudded tires on a winter road. The NLP is configured with $N = 260$ knot points at equally spaced spatial intervals ($\Delta s = 1$ m) and is setup to solve for a convergent lap (the first state must coincide with the last). Lastly, the NLP is solved with IPOPT, an open-source interior-point solver [37], using CASADI for algorithmic differentiation [38] and the MA57 library for matrix operations [39]. We use the default settings of IPOPT and run the optimization on a desktop computer with an Intel i7-12700KF CPU. Table 1 summarizes additional parameters of the NLP.

B. LONGITUDINAL BEHAVIOR COMPARISON

Fig. 5 compares the longitudinal speed profiles and force commands between the single- and multi-model optimization solutions. The single-model optimizations assume a single friction coefficient, either the lower value μ^1 or upper value μ^0 . The multi-model optimization, in contrast, assumes the full range explicitly during motion planning, returning both an optimistic trajectory corresponding to μ^0 and a contingency closed-loop rollout corresponding to μ^1 . As the friction value is increased for the single-model optimization, the solution's speed profile shifts faster and the force commands increase in amplitude. The general shape and pattern of the profiles remains similar.

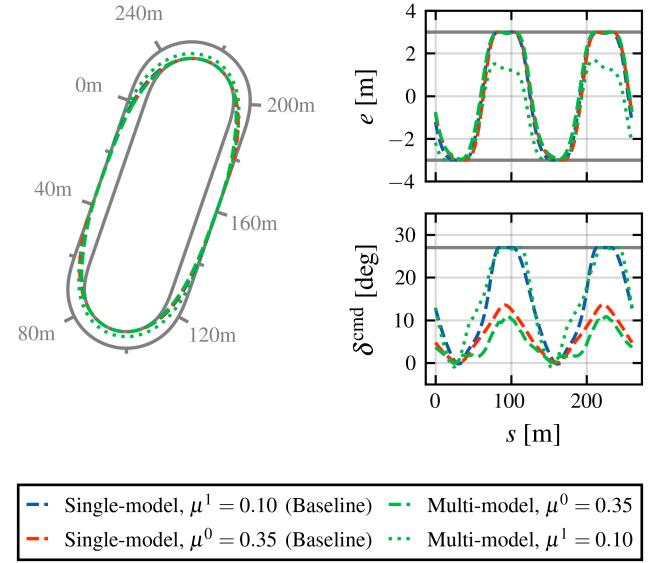


FIGURE 6. A comparison of the lateral behavior between the single- and multi-model optimization solutions. Similar to Fig. 5, the solutions from two different single-model optimizations are shown (low friction μ^1 in blue and high friction μ^0 in red), alongside the multi-model solutions (green). The lateral behavior is characterized here by the racing trajectory (left), lateral position reference (top right), and steering angle command (bottom right). While all three optimizations converge on similar reference racing lines (shown by the dashed lines), the multi-model optimization predicts and plans for sliding through the corner in the case of experiencing lower friction (shown by the dotted green line).

In contrast, the multi-model solution takes on a noticeably different form, not resembling any one single-model solution. On corner entry ($s = 40\text{-}70$ and $s = 205\text{-}235$ m), the multi-model solution targets a conservative speed, closer to the single-model's solution with μ^1 . In the middle of the corner, the multi-model solution targets a speed similar to the conservative single-model solution's. On corner exit, the multi-model solution plans to be aggressive, aiming for the speed resembling single-model solution with μ^0 . At the end of the corner, it also applies a significant F_x command, larger than any in the single-model solution set.

C. LATERAL BEHAVIOR COMPARISON

Next, we study the optimization solutions' lateral behavior by comparing their racing lines and steering references. As shown in Fig. 6, the single-model racing line does not vary appreciably as the friction value is increased. The multi-model solution takes a slightly tighter line on corner exit than the single-model trajectories, and anticipates almost a 2 m deviation from the reference line in the low-friction, closed-loop rollout. The multi-model optimization's preference for a tighter racing line on corner exit may be to build in a buffer to the outer road in order to account for this sliding a priori. In effect, we interpret the low-friction rollout in Optimization 4 as a spatially-varying constraint tightener, building in buffers where necessary along the track to ensure state and control constraint satisfaction for both realizations of the dynamics.

TABLE 2. Trajectory Optimization Solve Metrics for the Three Controllers Compared in Section V.

	Single-model (0.10)	Single-model (0.35)	Multi-model (0.35, 0.10)
Solve time [s]	4.66 ± 0.03	0.45 ± 0.00	27.52 ± 1.32
Lap time(s) [s]	45.5	24.5	(36.3, 46.5)
N_{iter} [-]	249	38	481
Time per it. [s]	0.02 ± 0.00	0.01 ± 0.00	0.06 ± 0.00

D. FEASIBILITY FOR ONLINE CONTROL

Table 2 collects the solve metrics of the optimization routines for the three solutions identified. For each trajectory’s solve time, the average and standard deviation from ten distinct trials are reported using the default parameter settings of IPOPT. In all cases, the solve times were below the total lap times of the solutions, indicating the optimization formulations are practical for online implementation.

VI. SIMULATION STUDY

With these motion plans, we now study how the controllers do in the closed-loop when simulated against different friction profiles. This section considers two main types of friction profiles: abruptly-changing profiles representing ice patches, and constant profiles with intermediate values between the modeled limits. The simulations in this section indicate the multi-model controller tracks its reference better than the single-model baselines in all cases, and it generally satisfies its own lateral position tube, despite not modeling abruptly-changing or intermediate friction profiles explicitly.

A. SIMULATION SETUP

The three controllers (single-model assuming μ^0 , single-model assuming μ^1 , and multi-model assuming the range (μ^0, μ^1)) are simulated in closed-loop for one full lap using the feedback gains specified in Table 1 and an explicit Runge-Kutta scheme (RK45) for numerical integration.

B. FRICTION PULSE (ICE PATCH) RESPONSES

We first look at the case of abrupt changes in friction due to encountering an ice patch. The ice patches are configured to be 10 m long, as this correlates to approximately the length of time necessary for an onboard state estimator to detect a change in road conditions and adapt a motion plan accordingly [40]. The friction coefficients on each axle are interpolated from the road friction profile as a function of each axle’s s -coordinate, capturing the relative “phasing” of driving over an ice patch due to the wheelbase of the vehicle. Fig. 7 plots the behavior of the the upper-bound single-model and multi-model controllers when faced with ice patches at three different locations throughout the corner.

For ice patch locations during corner entry (Case 1), both controllers recover to complete the full lap, albeit the single-model baseline exhibits much larger tracking errors. When the ice patch is placed near the corner apex (Case 2), the single-model baseline plows off the track but the multi-model

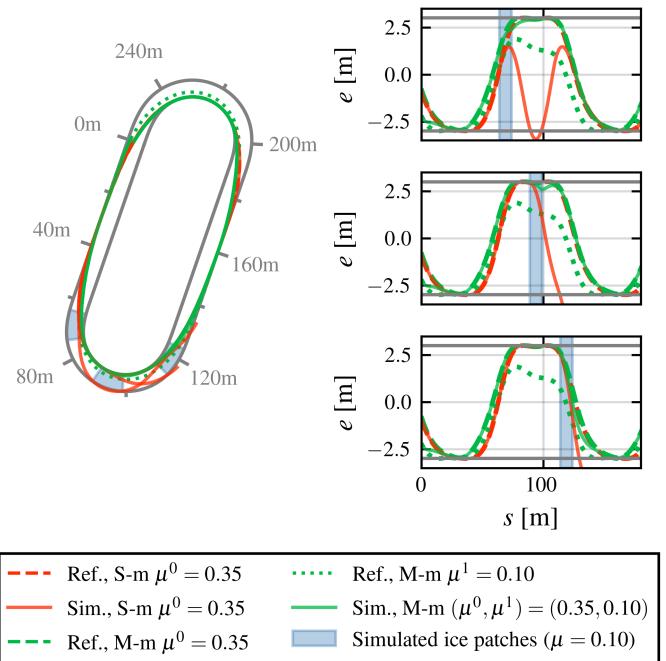


FIGURE 7. Simulation results of the upper-bound single-model and multi-model controllers when faced with ice patches at three different locations throughout the corner. The road friction coefficient is 0.10 inside the ice patch and 0.35 elsewhere. Left: Track overhead view, showing locations of the ice patches (blue) and the resulting single-model (red) and multi-model (green) trajectories. Right: Lateral position traces of the two controllers in the first corner for the three different patch locations. The single-model controller is only able to complete the full lap for one of the three cases, whereas the multi-model controller completes the full lap for all three cases.

controller successfully remains on. Lastly, for the case of ice on the exit (Case 3), the single-model controller understeers wide while the multi-model controller stays on the track to finish the lap.

One explanation for this observed behavior in the single-model case is that the lateral dynamics are most sensitive to the friction coefficient at high lateral slip angles, which occur near the corner apex. For an ice patch in this region, the lateral tire force quickly saturates and the vehicle understeers as a result. The multi-model controller is subject to the same underlying phenomena of course, but it may be deriving improved robustness to ice patches due to modeling a range of friction coefficients in its original motion plan.

C. INTERMEDIATE FRICTION VALUE SIMULATIONS

We next consider the case where the controllers experience intermediate friction values between the two modeled limits. The three controllers from Section V are simulated for one full lap with road friction coefficients ranging from 0.10 to 0.35 in 0.0025 increments. Fig. 8 plots the lateral position references and simulated traces of the three controllers.

For the single-model controllers, Fig. 8 conveys they could not complete full laps once the friction coefficient deviated a certain amount from the modeled value. In contrast, the

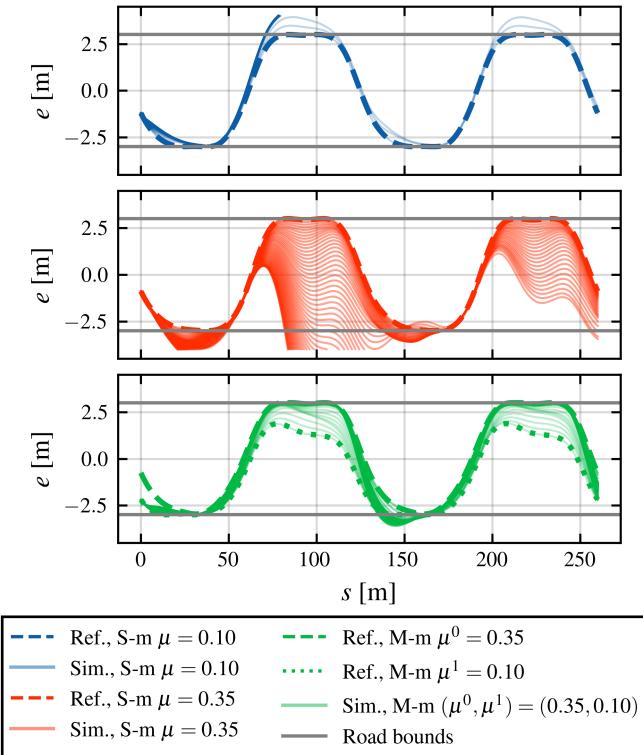


FIGURE 8. Simulation results of the lower-bound single-model (top), upper-bound single-model (center), and multi-model (bottom) controllers from Section V when simulated against different friction coefficients in the 0.10 to 0.35 range. In contrast to the multi-model controller, the single-model controllers could not complete full laps for all modeled friction coefficients. Simulation progress for these single-model simulations was stopped if the lateral error exceeded 1 m off track.

multi-model controller navigated the full lap for all intermediate friction values. While the corresponding lateral position traces largely stayed within the multi-model tube, we did observe brief tube exceedances for intermediate friction values near $s = 140$ m. This result suggests that intermediate parameter values may result in tube exceedance for the nonlinear system if they are not modeled explicitly in the multi-model optimization, but future studies will have investigate this effect further to make a more conclusive statement.

VII. EXPERIMENTAL DESIGN AND RESULTS

This section now evaluates our method's performance experimentally in a challenging snow and ice environment. In this type of environment, autonomous vehicle control is difficult as the friction uncertainty can constitute a large fraction of the total friction coefficient. Furthermore, these environments exhibit traction characteristics that vary both as a function of spatial position as well as a function of lap number throughout a specific test due to changing road conditions. This setting underscores the importance of friction robustness in vehicle control and serves as a stress test for our method.

We test the multi-model framework on the same oval racetrack as studied in the prior section. We compare the



FIGURE 9. Characteristic track surface conditions at the beginning (a) and end (b) of each experiment. During testing, the initial surface of snow unevenly wears away, revealing the underlying ice from the frozen lake. We observed that the amount of surface wear at each s -coordinate is nonuniform and can vary between the left and right tires over the course of an experiment.

controller's performance to two single-model trajectory plans: a low-friction (conservative) plan, and a high-friction (optimistic) plan. Over the course of multiple laps, the surface snow unevenly wears away to expose the underlying ice, resulting in localized areas of reduced traction. These track condition characteristics are illustrated in Fig. 9. During the experiments we observed portions of the track with left-to-right friction variation, which are not explicitly captured by the single-track model; hence, these conditions serve to further challenge the modeling assumptions in our proposed control scheme.

A. EXPERIMENTAL PLATFORM

The test vehicle is an automated 2018 Volkswagen Golf GTI, shown in Fig. 1. The vehicle is equipped with a dSpace MicroAutoBox II real-time computer which interfaces with the low-level controllers of the vehicle and can command the steering angle, throttle pedal percent, and individual brake caliper pressures. An Oxford Technical Solutions RT4003 provides vehicle position, velocity, and acceleration estimates by combining inertial measurements and global positioning system measurements. This state estimation system directly provides the signals necessary to both interpolate the reference trajectory (via the measured track s coordinate) and compute the control feedback (via the measured lateral position e , heading offset $\Delta\psi$, and longitudinal velocity v_x). The vehicle state estimates and control efforts are logged at a frequency of 200 Hz.

B. EXPERIMENTAL DESIGN

As in the prior section, we compare the multi-model method to two baselines: a single-model trajectory solution assuming the low friction bound and a single-model trajectory solution assuming the high friction bound. For the specific road and tire combination of the testing setup, we found that a friction coefficient range of 0.25–0.35 was representative of the observed conditions; we hence use these limits here for the lower- and upper-bounds of the single-model baselines and multi-model optimization.

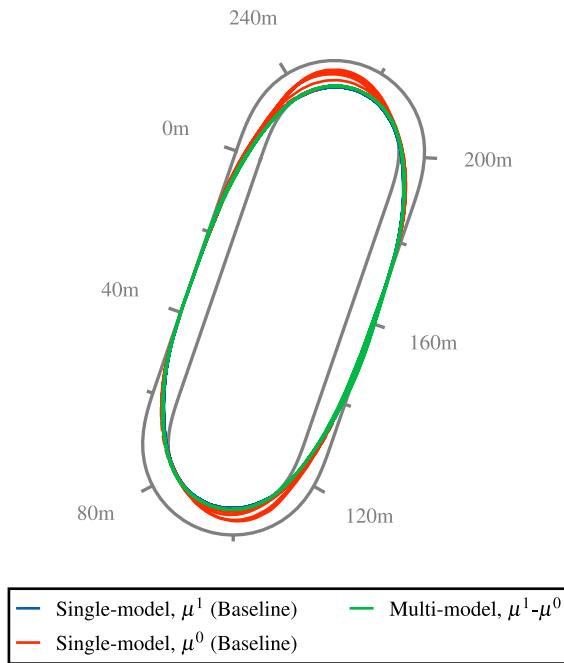


FIGURE 10. Overhead view of the recorded vehicle position for the three experiments: single-model assuming μ^1 (blue); single-model assuming μ^0 (red); and multi-model assuming the range $\mu^1-\mu^0$ (green). The dashed line on each graph is the reference trajectory and the solid lines are the observed motion. To ensure each experiment started with the same surface conditions, the vehicle was moved to a different region of the skidpad before each run and the coordinates were rezeroed. The multi-model and low-friction single-model trajectories exhibited close tracking, and thus appear on top of each other in the graph above. The controller struggled to track the high-friction single-model trajectory though, at times accruing a peak tracking error of 3 meters.

The single-model trajectory optimizations are structured identically to the multi-model formulation in Optimization 4, except there is only one dynamics model. The cost weights, state constraints, and control constraints are kept identical across all optimizations for comparison purposes. Each optimization returns a solution trajectory $(\mathbf{x}^*, \mathbf{u}^*)$ that is then tracked in real-time with the same linear feedback K , as diagrammed in Fig. 2.

C. EXPERIMENTAL RESULTS

For each experiment, the controller operated for multiple laps until either the performance equilibrated or the tracking error grew unacceptably large. Fig. 10 shows the overhead view of the three different experiments, and Fig. 11 charts the lap time and lateral tracking error as a function of lap number.

Fig. 11 indicates that while the low-friction single-model trajectory performs reliably, it records slow lap times and does not utilize all of the available traction to advance its objective. The high-friction single-model trajectory initially performs well, but later accrues unacceptable tracking error as the friction coefficient degrades. In contrast, the multi-model controller achieves the lowest tracking error of all controllers and yet remains fast for all laps. These results demonstrate that the

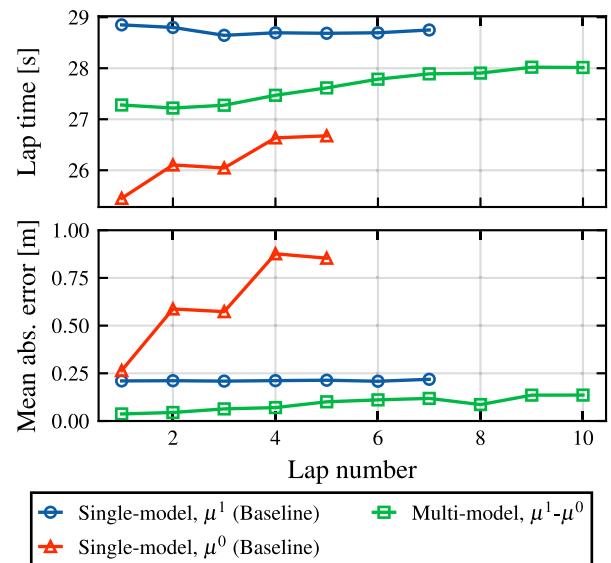


FIGURE 11. Lap time and average absolute value of lateral tracking error for each trajectory over multiple laps. As the laps progressed, the track's friction coefficient degraded, creating a natural model mismatch for the controllers. We stopped the trials once the performance equilibrated or the vehicle accrued too much lateral tracking error. The multi-model controller performed consistently throughout this surface degradation and achieved the best tracking errors of all controllers with an average lateral error of less than 15 cm.

multi-model strategy is able to take advantage of the higher friction early in the test to reduce lap time, while maintaining tight tracking even as the friction coefficient changes. These results were achieved despite the fact that the multi-model optimization did not explicitly consider friction variation.

In terms of tracking error, it is interesting to note that the multi-model controller achieved even better performance than the conservative single-model controller, which assumed the low friction bound. This (perhaps counterintuitive) result indicates that despite selecting the worst-case friction bound in motion planning, significant tracking errors may still result. Fundamentally, friction misestimates in either direction are still model mismatches, and must be addressed by reactive feedback. In contrast, the multi-model trajectory is *proactive* about uncertainty, reasoning about this friction range explicitly when solving for a trajectory. This proactivity may be the reason behind the remarkably low tracking errors observed in the challenging snow and ice conditions.

Fig. 12 provides another perspective on these results by plotting the peak front slip angle, peak rear slip angle, and root-mean-square acceleration of each controller across all experimental laps. As observed with the lap time results, the low-friction single-model controller achieves consistent metrics throughout the laps but uses low magnitudes of tire slip and hence traverses slowly around the track. The high-friction single-model controller is much more aggressive, which results in early advantages in terms of lap time, but by the fifth lap is saturating both the front and rear tires resulting in the observed tracking error. The multi-model controller

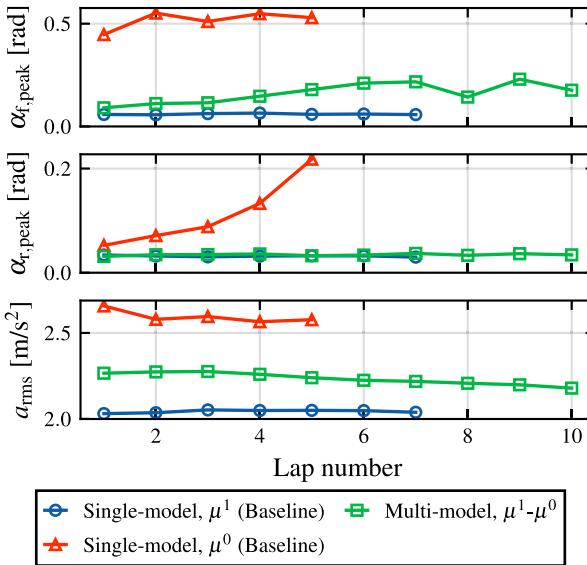


FIGURE 12. The peak front slip angle, peak rear slip angle, and root-mean-square acceleration of each controller for each lap of the experiments.

operates between these two extremes, able to use the tires in their nonlinear regions to achieve competitive lap times while maintaining low lateral tracking errors.

With reference to our modeling assumptions in Section III, we observe that the multi-model controller handled the vehicle's double-track dynamics and left-to-right friction variation well in experiment. As we look towards developing production systems from here, we highlight that dedicated double-track modeling and split- μ testing is an important part of system validation. Important next steps for future work include expanding the multi-model method to incorporate these double-track dynamics, as well as including more sophisticated friction profiles in optimization and experimentally validating the controller with a wider variety of vehicle and track types.

VIII. CONCLUSION

In this paper, we described a practical vehicle control framework that can maneuver aggressively through conditions with significant road friction uncertainty. The primary insight in this controller was to explicitly consider the entire range of the uncertain parameter during the motion planning process, leveraging optimization of the nonlinear closed-loop behavior to find an uncertainty-aware trajectory. After studying the optimization's robust maneuvering strategy for a racing application, we demonstrated the efficacy of this approach in simulation and then on an experimental platform in a challenging snow and ice environment. The resulting racing controller exploited higher friction when available to advance its performance objective, while still maintaining low tracking errors even when the friction coefficient significantly degraded. Future work includes extending this method to

TABLE 3. Vehicle Parameters

Parameter	Variable	Value	Units
Vehicle mass	m	1868	kg
Vehicle polar inertia	I_z	3049	$\text{kg}\cdot\text{m}^2$
Dist. from c.o.m. to front axle	a	1.19	m
Dist. from c.o.m. to rear axle	b	1.44	m
Vehicle track width	t	1.50	m
Height of c.o.m.	h_{com}	0.55	m
Dist. from c.o.m. to roll axis	h_l	0.46	m
Long. wt. transf. time const.	τ_{long}	0.10	sec
Vehicle roll rate	R_ϕ	4.4	deg/g
Lat. load transf. dist., front frac.	γ	0.64	-
Rolling resistance	$C_{d,0}$	218	N
Aerodynamic drag coeff.	$C_{d,2}$	0.42	$\text{N}/(\text{m}/\text{s})^2$
Max. steering angle	δ^{\max}	± 27	deg
Max. steering slew rate	$\dot{\delta}^{\max}$	$\pm 20.$	deg/s
Max. engine power	P_{eng}^{\max}	172	kW
Max. long. force slew rate	\dot{F}_x^{\max}	$\pm 10.$	kN/s

TABLE 4. Tire and Longitudinal Force Parameters for Experiments

Parameter	Variable	Value	Units
Front/Rear cornering stiffness coefficient	$C'_{\alpha,f} / C'_{\alpha,r}$	8 / 13	1/rad
Front lower/upper bound friction coefficient	μ_f^1 / μ_f^0	0.25 / 0.35	-
Rear lower/upper bound friction coefficient	μ_r^1 / μ_r^0	0.25 / 0.35	-
Front/Rear drive force distribution	$\chi_{f,\text{dr}} / \chi_{r,\text{dr}}$	1 / 0	-
Front/Rear brake force distribution	$\chi_{f,\text{br}} / \chi_{r,\text{br}}$	0.60 / 0.40	-

receding horizon implementations, as well as expanding its application to additional driving scenarios such as emergency collision avoidance.

APPENDIX A VEHICLE AND TIRE PARAMETERS

Vehicle parameters for the test platform are included in Table 3. Table 4 lists the test vehicle's tire parameters and drive and brake force distribution.

APPENDIX B BRAKING YAW MOMENT DUE TO DIFFERENTIAL BRAKING

To prevent unloaded wheel lockup during deceleration, we partition the braking force command for each axle across the individual wheels depending on their relative normal loads. This partitioning necessarily introduces a brake yaw moment, denoted by $M_{z,b}$, whose magnitude can significantly influence the yaw dynamics. In this appendix, we approximate the effect of this moment assuming steady-state load transfer and present a formulation that is amenable to higher-level trajectory optimizations.

We first model the steady-state longitudinal load transfer as

$$\Delta F_{z,\text{long,ss}} = \frac{ma_x h_{\text{com}}}{L} \quad (7)$$

and the steady-state lateral load transfer as

$$\Delta F_{z,\text{lat,ss}} = \frac{ma_y h_{\text{com}}}{t} + \frac{mgh_l \phi}{t}. \quad (8)$$

Here, a_x and a_y are the inertial body accelerations, h_l is the vertical distance between the c.o.m. and the roll axis, t is the track width, and ϕ is the body roll angle (positive along the body x -axis). The first term in (8) represents lateral load transfer due to inertial acceleration, while the second term represents load transfer due to the migration (roll) of the center of mass. The suspension behavior is approximated as linear with small angles, with roll angle $\phi := R_\phi a_y$ where R_ϕ is the vehicle's roll rate.

The resulting wheel normal loads for the front-left, front-right, rear-left and rear-right wheels are given by

$$\begin{aligned} F_{z,\text{fl}} &= \frac{1}{2} \left(\frac{mgb}{L} - \Delta F_{z,\text{long}} \right) - \gamma \Delta F_{z,\text{lat}} \\ F_{z,\text{fr}} &= \frac{1}{2} \left(\frac{mgb}{L} - \Delta F_{z,\text{long}} \right) + \gamma \Delta F_{z,\text{lat}} \\ F_{z,\text{rl}} &= \frac{1}{2} \left(\frac{mga}{L} + \Delta F_{z,\text{long}} \right) - (1 - \gamma) \Delta F_{z,\text{lat}} \\ F_{z,\text{rr}} &= \frac{1}{2} \left(\frac{mga}{L} + \Delta F_{z,\text{long}} \right) + (1 - \gamma) \Delta F_{z,\text{lat}}. \end{aligned}$$

The parameter γ is the lateral load transfer distribution, which captures the relative roll stiffnesses of the front and rear axles.

Next, we partition the brake force commands $F_{x,\text{f}}$ and $F_{x,\text{r}}$ within each axle according to this normal force distribution. The brake yaw moment is then calculated as

$$\begin{aligned} M_{z,\text{b}} &= - \left(\frac{F_{z,\text{fl}}}{F_{z,\text{fl}} + F_{z,\text{fr}}} F_{x,\text{f}} \right) \frac{t}{2} + \left(\frac{F_{z,\text{fr}}}{F_{z,\text{fl}} + F_{z,\text{fr}}} F_{x,\text{f}} \right) \frac{t}{2} \\ &\quad - \left(\frac{F_{z,\text{rl}}}{F_{z,\text{rl}} + F_{z,\text{rr}}} F_{x,\text{r}} \right) \frac{t}{2} + \left(\frac{F_{z,\text{rr}}}{F_{z,\text{rl}} + F_{z,\text{rr}}} F_{x,\text{r}} \right) \frac{t}{2}, \end{aligned}$$

which, with the appropriate substitutions, simplifies to

$$\begin{aligned} M_{z,\text{b}} &= \left(\frac{\gamma(h_{\text{com}} + g h_l R_\phi) a_y L}{gb - a_x h_{\text{com}}} \right) F_{x,\text{f}} \\ &\quad + \left(\frac{(1 - \gamma)(h_{\text{com}} + g h_l R_\phi) a_y L}{ga + a_x h_{\text{com}}} \right) F_{x,\text{r}}. \end{aligned}$$

For the purposes of trajectory optimization, we note that this relation simple to incorporate, since it is only a function of the vehicle's acceleration and braking force commands; the other terms are fixed parameters of the vehicle.

APPENDIX C

PROOF OF TRAJECTORY BOUNDEDNESS IN THE CASE OF LINEAR DYNAMICS

As referenced in Section IV, in this appendix we show the trajectory boundedness property of multiple-model motion planning for the simplified case of linear dynamics.

Claim C.1: Subject to the following assumptions, if the realized parameter vector is bounded by the limit parameter

vectors as

$$\boldsymbol{\theta} = \lambda \boldsymbol{\theta}^1 + (1 - \lambda) \boldsymbol{\theta}^0, \quad (9)$$

then the realized state trajectory is bounded by the limit trajectories as

$$\mathbf{x}(s) = \lambda \mathbf{x}^1(s) + (1 - \lambda) \mathbf{x}^0(s) \quad (10)$$

for a fixed $\lambda \in [0, 1]$ and over all track coordinates s .

Assumption C.1.1: The vehicle dynamics in (2) are locally linear around the $\mathbf{x}^0(s)$ and $\mathbf{x}^1(s)$ trajectories. More specifically, when (9) is satisfied, we assume the dynamics are described by

$$\begin{aligned} \mathbf{x}'(s) &= f_{\text{st}}(\mathbf{x}^0(s), \mathbf{u}^0(s), s; \boldsymbol{\theta}^0) + \frac{\partial f_{\text{st}}}{\partial \mathbf{x}}(s) (\mathbf{x}(s) - \mathbf{x}^0(s)) \\ &\quad + \frac{\partial f_{\text{st}}}{\partial \mathbf{u}}(s) (\mathbf{u}(s) - \mathbf{u}^0(s)) + \frac{\partial f_{\text{st}}}{\partial \boldsymbol{\theta}}(s) (\boldsymbol{\theta} - \boldsymbol{\theta}^0). \end{aligned} \quad (11)$$

All of the derivative terms in Eqn (11) are functions of s , and evaluated using $\mathbf{x}^0(s)$, $\mathbf{u}^0(s)$ and $\boldsymbol{\theta}^0$.

Assumption C.1.2: The realized parameter vector $\boldsymbol{\theta}$ is unknown, but constant, with respect to s .

Assumption C.1.3: The initial vehicle state is the same regardless of the realized parameter vector. This is often the case in real-time applications (the first state must coincide with the measured state) and can be enforced for offline applications with a simple constraint.

Proof: Building upon Assumption C.1.1, we start by defining error states from the the trajectory $\mathbf{x}^0(s)$ as

$$\mathbf{e}^1(s) := \mathbf{x}^1(s) - \mathbf{x}^0(s) \quad (12)$$

$$\mathbf{e}(s) := \mathbf{x}(s) - \mathbf{x}^0(s), \quad (13)$$

which have associated linear dynamics

$$\mathbf{e}^{1'}(s) = (\mathbf{A}(s) - \mathbf{B}(s)\mathbf{K}) \mathbf{e}^1(s) + \mathbf{T}(s) (\boldsymbol{\theta}^1 - \boldsymbol{\theta}^0) \quad (14)$$

$$\mathbf{e}'(s) = (\mathbf{A}(s) - \mathbf{B}(s)\mathbf{K}) \mathbf{e}(s) + \mathbf{T}(s) (\boldsymbol{\theta} - \boldsymbol{\theta}^0) \quad (15)$$

via substitution with (11). To ease the notation, the partial derivatives $\partial f_{\text{st}}/\partial \mathbf{x}$, $\partial f_{\text{st}}/\partial \mathbf{u}$ and $\partial f_{\text{st}}/\partial \boldsymbol{\theta}$ are represented by \mathbf{A} , \mathbf{B} , and \mathbf{T} , respectively. We further simplify by representing $(\mathbf{A}(s) - \mathbf{B}(s)\mathbf{K})$ as the closed-loop matrix $\mathbf{A}_{\text{cl}}(s)$.

The systems in (14)–(15) are linear spatially-varying which have analytical solutions given by

$$\mathbf{e}^1(s) = \Phi(s, s_0) \mathbf{e}^1(s_0) + \int_{s_0}^s \Phi(s, \sigma) \mathbf{T}(\sigma) (\boldsymbol{\theta}^1 - \boldsymbol{\theta}^0) d\sigma \quad (16)$$

$$\mathbf{e}(s) = \Phi(s, s_0) \mathbf{e}(s_0) + \int_{s_0}^s \Phi(s, \sigma) \mathbf{T}(\sigma) (\boldsymbol{\theta} - \boldsymbol{\theta}^0) d\sigma, \quad (17)$$

where the state transition function $\Phi(s, s_0) : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}^{7 \times 7}$ is defined as the solution to

$$\frac{\partial}{\partial s} \Phi(s, s_0) = \mathbf{A}_{\text{cl}} \Phi(s, s_0) \quad (18)$$

with initial condition equal to the identity matrix [41]

$$\Phi(s_0, s_0) = \mathbf{I}_{7 \times 7}. \quad (19)$$

We observe that because A_{cl} is common to both error evolutions, so is the state transition matrix. As the initial error states were assumed to be zero, (16)–(17) simplify to:

$$\mathbf{e}^1(s) = \int_{s_0}^s \Phi(s, \sigma) \mathbf{T}(\sigma) (\boldsymbol{\theta}^1 - \boldsymbol{\theta}^0) d\sigma \quad (20)$$

$$\mathbf{e}(s) = \int_{s_0}^s \Phi(s, \sigma) \mathbf{T}(\sigma) (\boldsymbol{\theta} - \boldsymbol{\theta}^0) d\sigma. \quad (21)$$

Next, we substitute Eqn (9) into (21) and simplify with (20):

$$\begin{aligned} \mathbf{e}(s) &= \int_{s_0}^s \Phi(s, \sigma) \mathbf{T}(\sigma) (\boldsymbol{\theta} - \boldsymbol{\theta}^0) d\sigma \\ \mathbf{e}(s) &= \int_{s_0}^s \Phi(s, \sigma) \mathbf{T}(\sigma) (\lambda (\boldsymbol{\theta}^1 - \boldsymbol{\theta}^0)) d\sigma \\ &= \lambda \mathbf{e}^1(s). \end{aligned} \quad (22)$$

After expanding the error states with their definitions ((14)–(15)), we arrive at the original claim:

$$\begin{aligned} \mathbf{e}(s) &= \lambda \mathbf{e}^1(s) \\ \mathbf{x}(s) - \mathbf{x}^0(s) &= \lambda (\mathbf{x}^1(s) - \mathbf{x}^0(s)) \\ \mathbf{x}(s) &= \lambda \mathbf{x}^1(s) + (1 - \lambda) \mathbf{x}^0(s). \end{aligned} \quad \blacksquare$$

Corollary C.1.1: State boundedness. The convex combination in Claim C.1 applies individually to each element of \mathbf{x} . This implies that all box state constraints in Optimization 4 are also satisfied for the realized trajectory, regardless of the realized $\boldsymbol{\theta}$.

Corollary C.1.2: Control boundedness. The control efforts for the realized trajectory $\mathbf{u}(s)$ are bounded by the control efforts for the limit trajectories $\mathbf{u}^1(s)$ and $\mathbf{u}^0(s)$. We show this by first starting with the expression for $\mathbf{u}(s)$

$$\mathbf{u}(s) = \mathbf{u}^0(s) - \mathbf{K}(\mathbf{x}(s) - \mathbf{x}^0(s)),$$

and then expanding $\mathbf{x}(s)$ with (10),

$$\begin{aligned} \mathbf{u}(s) &= \mathbf{u}^0(s) - \mathbf{K}(\lambda \mathbf{x}^1(s) + (1 - \lambda) \mathbf{x}^0(s) - \mathbf{x}^0(s)) \\ &= \mathbf{u}^0(s) - \lambda \mathbf{K}(\mathbf{x}^1(s) - \mathbf{x}^0(s)), \end{aligned}$$

and finally substituting in the relation $\mathbf{u}^1(s) = \mathbf{u}^0(s) - \mathbf{K}(\mathbf{x}^1(s) - \mathbf{x}^0(s))$ from the multi-model optimization formulation:

$$\begin{aligned} \mathbf{u}(s) &= \mathbf{u}^0(s) - \lambda(\mathbf{u}^0(s) - \mathbf{u}^1(s)) \\ &= \lambda \mathbf{u}^1(s) + (1 - \lambda) \mathbf{u}^0(s). \end{aligned}$$

As with the prior corollary, this result applies individually to each element of $\mathbf{u}(s)$. Box constraints on individual control efforts in Optimization 4 are therefore satisfied regardless of the realized parameter vector $\boldsymbol{\theta}$.

ACKNOWLEDGMENT

The authors would like to thank the members of the Dynamic Design Lab for their helpful discussions during the development of this work. We thank Volkswagen Group

Innovation for their funding support of this research and Dr. Björn Mennenga and Konrad Weigmann of Volkswagen for their insightful assistance during experiments. The PACCAR Corporation, Bridgestone Corporation, and Thunderhill Raceway Park helped facilitate the early phases of this work. We are grateful to the anonymous reviewers of this paper for their feedback to improve both its content and clarity.

REFERENCES

- [1] “United states department of transportation federal highway administration,” Sep. 2024. [Online]. Available: https://ops.fhwa.dot.gov/weather/weather_events/snow_ice.htm
- [2] H. S. Black, “Stabilized feedback amplifiers,” *Bell Syst. Tech. J.*, vol. 13, no. 1, pp. 1–18, Jan. 1934. [Online]. Available: <https://ieeexplore.ieee.org/document/6767960>
- [3] K. J. Åström and R. M. Murray, *Feedback Systems: An Introduction for Scientists and Engineers*, 2 ed. Princeton, NJ, USA: Princeton Univ. Press, 2021.
- [4] K. J. Åström and B. Wittenmark, *Adaptive Control* (Dover Books on Engineering), 2nd ed. Mineola, NY, USA: Dover Publications, 2008.
- [5] J.-J. E. Slotine and W. Li, *Applied Nonlinear Control*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1991.
- [6] Y. Chen and J. Wang, “Adaptive vehicle speed control with input injections for longitudinal motion independent road frictional condition estimation,” *IEEE Trans. Veh. Technol.*, vol. 60, no. 3, pp. 839–848, Mar. 2011. [Online]. Available: <https://ieeexplore.ieee.org/document/5692858>
- [7] K. Bernertor, R. Quirynen, T. Uno, and S. D. Cairano, “Trajectory tracking for autonomous vehicles on varying road surfaces by friction-adaptive nonlinear model predictive control,” *Veh. Syst. Dyn.*, vol. 58, no. 5, pp. 705–725, May 2020. [Online]. Available: <https://www.tandfonline.com/doi/full/10.1080/00423114.2019.1697456>
- [8] Z. Qiu, X. Zhu, A.-T. Nguyen, and G. Yin, “Adaptive neural-network-based fault tolerant control for autonomous vehicles with dynamic output constraints,” *IEEE/ASME Trans. Mechatron.*, early access, Jun. 12, 2025, doi: [10.1109/TMECH.2025.3575035](https://doi.org/10.1109/TMECH.2025.3575035).
- [9] B. D. Anderson and A. Dehghani, “Challenges of adaptive control—past, permanent and future,” *Annu. Rev. Control*, vol. 32, no. 2, pp. 123–135, Dec. 2008. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S136757880800028X>
- [10] K. Zhou, J. C. Doyle, K. Glover, and J. C. Doyle, *Robust and Optimal Control*. Upper Saddle River, NJ, USA: Prentice-Hall, 1996.
- [11] P. A. Ioannou and J. Sun, *Robust Adaptive Control*. Mineola, NY, USA: Dover, 2012.
- [12] S. Bhattacharyya, “Robust control under parametric uncertainty: An overview and recent results,” *Annu. Rev. Control*, vol. 44, pp. 45–77, 2017. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S1367578816301389>
- [13] C. Hu, S. Ge, Y. Shi, W. Gao, W. Guo, and X. Zhang, “Composite nonlinear trajectory tracking control of co-driving vehicles using self-triggered adaptive dynamic programming,” *IEEE Trans. Consum. Electron.*, vol. 71, no. 2, pp. 3474–3485, May 2025. [Online]. Available: <https://ieeexplore.ieee.org/document/10918631/>
- [14] W. Langson, I. Chryssochoos, S. Raković, and D. Mayne, “Robust model predictive control using tubes,” *Automatica*, vol. 40, no. 1, pp. 125–133, Jan. 2004. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0005109803002838>
- [15] J. B. Rawlings, D. Q. Mayne, and M. Diehl, *Model Predictive Control: Theory, Computation, and Design*, 2nd ed. Santa Barbara, CA, USA: Nob Hill Publishing, 2020.
- [16] D. Q. Mayne, “Model predictive control: Recent developments and future promise,” *Automatica*, vol. 50, no. 12, pp. 2967–2986, Dec. 2014. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0005109814005160>
- [17] Y. Gao, A. Gray, H. E. Tseng, and F. Borrelli, “A tube-based robust nonlinear predictive control approach to semiautonomous ground vehicles,” *Veh. Syst. Dyn.*, vol. 52, no. 6, pp. 802–823, Jun. 2014. [Online]. Available: <https://www.tandfonline.com/doi/abs/10.1080/00423114.2014.902537>

- [18] T. P. Weber, R. K. Aggarwal, and J. C. Gerdes, "Human-inspired autonomous racing in low friction environments," *IEEE Trans. Intell. Veh.*, vol. 10, no. 5, pp. 3684–3696, Jun. 2025. [Online]. Available: <https://ieeexplore.ieee.org/document/10681660/>
- [19] J. Betz et al., "Autonomous vehicles on the edge: A survey on autonomous vehicle racing," *IEEE Open J. Intell. Transp. Syst.*, vol. 3, pp. 458–488, 2022.
- [20] M. McNaughton, "CASTRO: Robust nonlinear trajectory optimization using multiple models," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, San Diego, CA, USA, Oct. 2007, pp. 177–182. [Online]. Available: <https://ieeexplore.ieee.org/document/4399046/>
- [21] C. G. Atkeson, "Efficient robust policy optimization," in *Proc. Amer. Control Conf.*, Montreal, QC, USA, Jun. 2012, pp. 5220–5227. [Online]. Available: <https://ieeexplore.ieee.org/document/6315619/>
- [22] I. Mordatch, K. Lowrey, and E. Todorov, "Ensemble-CIO: Full-body dynamic motion planning that transfers to physical humanoids," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Hamburg, Germany, Sep. 2015, pp. 5307–5314. [Online]. Available: <https://ieeexplore.ieee.org/document/7354126/>
- [23] T. A. Howell, C. Fu, and Z. Manchester, "Direct policy optimization using deterministic sampling and collocation," *IEEE Robot. Automat. Lett.*, vol. 6, no. 3, pp. 5324–5331, Jul. 2021. [Online]. Available: <https://ieeexplore.ieee.org/document/9387078/>
- [24] J. Kegelman, "Learning from professional race car drivers to make automated vehicles safer," Ph.D. Dissertation, Stanford Univ., Stanford, CA, USA, 2018.
- [25] J. Funke, M. Brown, S. M. Erlien, and J. C. Gerdes, "Collision avoidance and stabilization for autonomous vehicles in emergency scenarios," *IEEE Trans. Control Syst. Technol.*, vol. 25, no. 4, pp. 1204–1216, Jul. 2017. [Online]. Available: <https://ieeexplore.ieee.org/document/7585053/>
- [26] N. R. Kapania and J. C. Gerdes, "Design of a feedback-feedforward steering controller for accurate path tracking and stability at the limits of handling," *Veh. Syst. Dyn.*, vol. 53, no. 12, pp. 1687–1704, Dec. 2015. [Online]. Available: <https://www.tandfonline.com/doi/full/10.1080/00423114.2015.1055279>
- [27] T. P. Weber and J. C. Gerdes, "Modeling and control for dynamic drifting trajectories," *IEEE Trans. Intell. Veh.*, vol. 9, no. 2, pp. 3731–3741, Feb. 2024. [Online]. Available: <https://ieeexplore.ieee.org/document/10349953/>
- [28] V. A. Laurence and J. C. Gerdes, "Long-horizon vehicle motion planning and control through serially cascaded model complexity," *IEEE Trans. Control Syst. Technol.*, vol. 30, no. 1, pp. 166–179, Jan. 2022. [Online]. Available: <https://ieeexplore.ieee.org/document/9366415/>
- [29] J. K. Subotsis and J. C. Gerdes, "Impacts of model fidelity on trajectory optimization for autonomous vehicles in extreme maneuvers," *IEEE Trans. Intell. Veh.*, vol. 6, no. 3, pp. 546–558, Sep. 2021. [Online]. Available: <https://ieeexplore.ieee.org/document/9325924/>
- [30] H. B. Pacejka and I. Besselink, *Tire and Vehicle Dynamics*, 3rd ed. Oxford, U.K.: Elsevier, 2012.
- [31] M. Abe, *Vehicle Handling Dynamics: Theory and Application*, 2nd ed. Oxford, U.K.: Elsevier, 2015.
- [32] J. T. Betts, *Practical Methods for Optimal Control Using Nonlinear Programming (Advances in Design and Control no. 36)*, 3rd ed. Philadelphia, PA, USA: SIAM, 2020.
- [33] J. V. Frasch et al., "An auto-generated nonlinear MPC algorithm for real-time obstacle avoidance of ground vehicles," in *Proc. Eur. Control Conf.*, Zurich, Switzerland, Jul. 2013, pp. 4136–4141. [Online]. Available: <https://ieeexplore.ieee.org/document/6669836/>
- [34] K. L. Talvala, K. Kritayakirana, and J. C. Gerdes, "Pushing the limits: From lanekeeping to autonomous racing," *Annu. Rev. Control*, vol. 35, no. 1, pp. 137–148, Apr. 2011. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S1367578811000101>
- [35] A. P. Leeman, J. Köhler, A. Zanelli, S. Bennani, and M. N. Zeilinger, "Robust nonlinear optimal control via system level synthesis," *IEEE Trans. Autom. Control*, vol. 70, no. 7, pp. 4780–4787, Jul. 2025.
- [36] X. Zhou and J. Wang, "Variable-exponent Lyapunov function-based attracting-manifold adaptive control design with application to ground vehicle lateral guidance," *IEEE Trans. Intell. Veh.*, early access, May 15, 2024, doi: [10.1109/TIV.2024.3401214](https://doi.org/10.1109/TIV.2024.3401214).
- [37] 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.
- [38] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "CasADI: A software framework for nonlinear optimization and optimal control," *Math. Program. Comput.*, vol. 11, no. 1, pp. 1–36, Mar. 2019. [Online]. Available: <https://link.springer.com/10.1007/s12532-018-0139-4>
- [39] HSL, "HSL: A collection of fortran codes for large scale scientific computation," 2024. [Online]. Available: <https://www.hsl.rl.ac.uk/index.html>
- [40] L. R. Ray, "Nonlinear tire force estimation and road friction identification: Simulation and experiments," *Automatica*, vol. 33, no. 10, pp. 1819–1833, Oct. 1997. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0005109897000939>
- [41] C.-T. Chen, *Linear System Theory and Design* (Oxford Series in Electrical and Computer Engineering), 3rd ed. New York, NY, USA: Oxford Univ. Press, 1999.



RAJAN K. AGGARWAL (Member, IEEE) received the B.S. degree in mechanical engineering from the University of Illinois in Urbana-Champaign, Champaign, IL, USA in 2019 and the M.S. degree in mechanical engineering from Stanford University, Stanford, CA, USA, in 2022. He is currently working toward the Ph.D. degree in mechanical engineering with Stanford University, Stanford, CA, where he studies motion planning and control for vehicles in extreme handling scenarios.



J. CHRISTIAN GERDES (Member, IEEE) received the Ph.D. degree from the University of California at Berkeley, Berkeley, CA, USA, in 1996. He is currently a Professor Emeritus in mechanical engineering with Stanford University, Stanford, CA, USA and Field Safety Architect with Waymo LLC. His laboratory studies how cars move, how humans drive cars, and how to design future cars that work cooperatively with the driver or drive themselves. When not teaching on campus, he can often be found at the racetrack with students trying out their latest prototypes for the future.