

Model Predictive Control for Aggressive Driving Over Uneven Terrain

Tyler Han*, Alex Liu, Anqi Li, Alexander Spitzer, Guanya Shi, Byron Boots

University of Washington

<https://sites.google.com/cs.washington.edu/off-road-mpc>

Abstract—Terrain traversability in unstructured off-road autonomy has traditionally relied on semantic classification, resource-intensive dynamics models, or purely geometry-based methods to predict vehicle-terrain interactions. While inconsequential at low speeds, uneven terrain subjects our full-scale system to safety-critical challenges at operating speeds of 7–10 m/s. This study focuses particularly on uneven terrain such as hills, banks, and ditches. These common high-risk geometries are capable of disabling the vehicle and causing severe passenger injuries if poorly traversed. We introduce a physics-based framework for identifying traversability constraints on terrain dynamics. Using this framework, we derive two fundamental constraints, each with a focus on mitigating rollover and ditch-crossing failures while being fully parallelizable in the sample-based Model Predictive Control (MPC) framework. In addition, we present the design of our planning and control system, which implements our parallelized constraints in MPC and utilizes a low-level controller to meet the demands of our aggressive driving without prior information about the environment and its dynamics. Through real-world experimentation and traversal of hills and ditches, we demonstrate that our approach captures fundamental elements of safe and aggressive autonomy over uneven terrain. Our approach improves upon geometry-based methods by completing comprehensive off-road courses up to 22% faster while maintaining safe operation.

I. INTRODUCTION

Autonomous vehicles have the potential to be applied to agriculture, transportation in underdeveloped regions, and search and rescue [28, 33, 2]. To succeed in these applications, autonomous vehicles must be able to operate without a priori knowledge in *unstructured* and *off-road* environments, which often weaken or entirely violate assumptions commonly made in on-road autonomous vehicle research [29, 34]. In particular, unlike the on-road setting, successful off-road driving depends critically on safe traversal of *uneven terrain* such as hills, ditches, banks, and bumps. To demonstrate realistic autonomy in unstructured environments, terrain mapping and control optimization are performed onboard and in real-time. No prior information is provided to the system. In this work, we aim to address two significant failure risks encountered by a full-scale off-road autonomous vehicle (Polaris RZR Turbo S4) traveling at speeds of 7–10 m/s over uneven terrain. Either rollovers or ditches are easily capable of disabling the vehicle and causing severe bodily harm to passengers.

First, *rollover* can occur when the vehicle becomes disabled by tipping onto its sides. The risk of rollovers is substantially

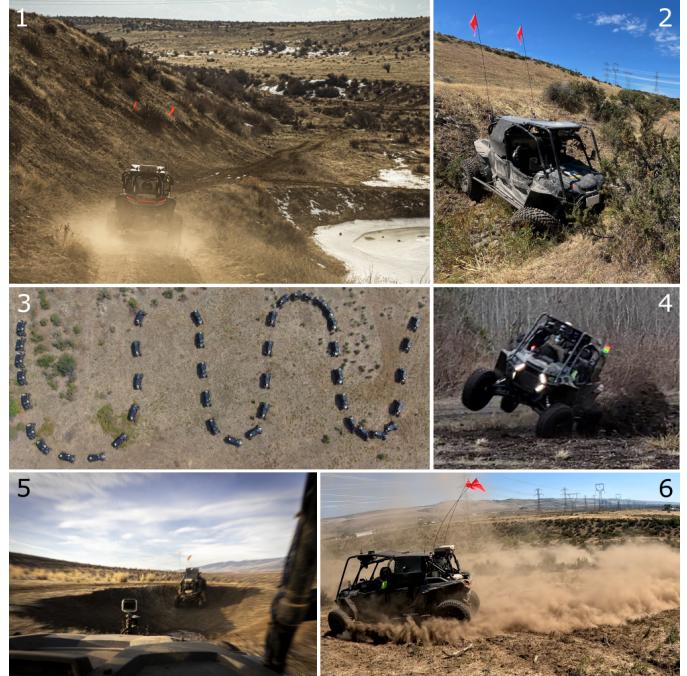


Fig. 1: (1) Robot in uneven terrain. (2) Poor ditch traversal. (3) Slalom on a steep hill. (4) Poor rollover handling. (5) Chasing robot through crater ditch. (6) Expert driver performing aggressive circles. Note that (2) and (4) are not outcomes of our approach.

higher in off-road driving where steep hills, evasive sharp turns, and vehicles with high centers of mass due to tall suspension systems are common. The second failure risk is associated with *ditches*. Often characterized by extreme impact forces, snapped or irreparably bent control arms (Fig. 1.2) are common failures due to poor ditch traversal.

Aggressive navigation through uneven terrain is further complicated by systems considerations such as imperfect perception, uncertainty, and real-time control. For instance, ground plane estimation is made difficult by the presence of obstacles (tall grass, short bushes and trees, rocks, etc.) in the scene [32, 31, 17, 3], which obfuscates the true ground location. As a result, predicted ground planes can often be noisy, inaccurate, and oversmoothed. Still, the downstream controller must be able to tune away the bias introduced by inaccurate but fast perception. Most importantly, at nominal speeds of 7–10 m/s, fast planning and optimization times

*Corresponding Author: than123@cs.washington.edu

are safety critical. Low control rates cause the system to plan through misrepresented or delayed semantic obstacles and terrain geometries which can be extremely dangerous. Evidently, the high-speed regime requires that perception and control be connected intimately to ensure safe operation. In addition to challenging sub-systems, high-speed off-road driving is a highly complex systems problem. To the authors' best knowledge, no prior work has demonstrated high speeds in uneven terrain on a full-scale system. We refer readers to our careful review of current literature across related off-road, on-road, small-scale, and low-speed systems in Section II.

We present two main contributions to off-road autonomy in this paper. First, we propose a physics-based framework to derive traversability constraints for a mobile robot on uneven terrain. We show that these constraints are time-independent and can be parallelized across both samples and the control horizon, enabling aggressive control through a *practical compromise between simple geometry-based methods and inertial dynamics models*. Second, we report our complete planning and control design with heuristics for off-road navigation validated on a full-scale Polaris RZR Turbo S4 equipped with onboard sensing and compute. The planning and control system consists of a planner based on the model predictive path integral (MPPI) algorithm [46, 48], and a low-level feedforward-feedback controller. Our method is qualitatively validated by expert driver behavior and traverses up to 22% faster than the geometry-based baseline on two and three kilometer comprehensive off-road courses with vegetation, steep hills, and sharp ditches.

II. RELATED WORK

Currently, there are few works which conduct autonomous off-road experimentation on a full-scale platform, fewer which demonstrate speeds beyond 7 m/s, and, to the authors' best knowledge, none which additionally deploy in real, uneven terrain. However, work on the same chassis, sensor, and compute platform does exist [16, 15, 45]. These works focus primarily on individual sub-systems, validate with offline data and methods, and do not report on system performance or speeds. Our work presents the most complete description and evaluation of a control system for the fastest full-scale, off-road, autonomous vehicle in uneven terrain to date. For more distant but related work, we note that due to lack of standardization in the off-road setting, difficulty of the environment and expected performance can vary drastically. Prior works on real, full-scale vehicles do not attempt to navigate in uneven terrain [7, 37]. However, even in mild terrain conditions, our operating speeds of 7–10 m/s are higher or comparable to other work. In [44], the system navigates autonomously at about 3.2 m/s and averages 4 interventions over 1.6 km. In the DARPA Grand Challenge, the fastest method achieves speeds between 9–11 m/s on largely flat terrain, slowing to 4–6 m/s for mild bumps [37]. Older work also demonstrate capabilities up to 9.7 m/s over flat ground and binary traversability [7].

Past works and systems reveal that high-speed navigation itself is not a new challenge. It is the combination of high

speeds and challenging, uneven terrain which current methods struggle with. Our real testing environment is most similar to the environment *simulated* by Lee et al. [26] where the proposed MPPI-based approach also averages 7–11 m/s in simulation. Importantly, the authors empirically discover that a cost must be applied to vertical accelerations and torques in order to maintain control. In the DARPA Subterranean Challenge, Fan et al. constrain velocity in the kinodynamic planner without rigorous justification but state that these constraints reduce the energy of interactions with the environment [11]. Evidently, there is a need for systems to require constraints due to the challenging terrain geometry of these unstructured environments. However, no principled justification has yet been provided. We aim to justify these heuristics such that they are motivated through physical modeling and constraints.

Existing work on real systems on uneven terrain but at low speeds generally assume the vehicle adheres to the ground at all times and constrain attitude angles as estimated by only the ground geometry [9, 11, 24]. This method is relatively fast, simple to implement, and effective at low speeds [19]. However, at higher speeds, the method becomes increasingly inaccurate and unsafe due to dynamic effects from unmodeled forces and vehicle inertia. On the other hand, complex dynamics models can explicitly predict wheel forces and trajectories but are often too slow for safe operation. For instance, [50] implements a 3D rigid-body model with tire forces in MPC at a control rate of 3 Hz, about one tenth of our target rate. In [39], the control rate is sufficient at 50 Hz but operates at speeds of 3 m/s over gentler and differentiable ground manifolds. Note that due to the architecture of common off-road perception systems, elevation maps are often outputs of complex neural networks [32, 15] or approximations to point clouds [24]. Neither representation admits a practical path toward real-time optimization using aforementioned methods.

Some works have pursued learning-based approaches to driving through uneven terrain. In [4], the authors propose using imitation learning for safe traversal of various types of ditches at 1 m/s in simulation. However, success rates drop by as much as 50% depending on which types of train and test geometries are selected. Learning-based dynamics modeling has also been proposed in the literature [16, 43, 49]. However, in the aggressive regime, collecting data near the safety limits of the vehicle can be dangerous and infeasible in practice. In [30], the authors seek to generalize low speed training data to higher speed tests using physics-based modeling, improving tight turns on flat ground over a basic bicycle model.

In summary, current methods fall short in our setting for three main reasons: (1) implementations on real systems lack models of dynamic effects, constraining autonomy to low speeds or mild terrain; (2) full vehicle dynamics models with explicit wheel forces are too computationally expensive; and (3) state-of-the-art real-time perception representations do not lend themselves to otherwise sufficiently fast gradient-based optimization. In this work, we propose contending with dynamic effects by estimating the implicit forces of sampled paths, similar in principle to methods in [27, 13, 19].

III. PHYSICS-BASED TRAVERSABILITY CONSTRAINTS

In automobile safety testing, metrics such as static stability factor [20, 35], lateral load transfer ratio [10, 25], and time to rollover [5, 8] quantify the risk of vehicle rollovers by focusing on the inertial forces which may tip the vehicle. This line of work supports the intuition that the ground and path geometry alone do not adequately represent the vehicle dynamics. However, these geometries, when coupled with gravitational forces and vehicle inertia, should contain sufficient information about feasible traversal speeds. In this section, we derive constraints on the vehicle by incorporating geometrical information in a physics model, which is used to inform cost function design for MPPI in Section IV.

A. Dynamics Model with Residual Torque

Following our rationale, our modeling objective will be to derive a relationship between the geometry of the environment and the relevant forces. As a motivating thought experiment, consider a marble rolling down a track shaped like a downwards-facing parabola. Guided by gravity, the marble will adhere to the track without issue. Once the marble is pushed, it begins to lose its grip on the track. Pushed harder still, the marble will begin to lift away from the track until its trajectory is entirely separated. For our vehicle, we are concerned with *any* wheel leaving the ground rather than the entire vehicle all at once. Thus, we are interested in constraining the rotational rate of the vehicle which is a tighter constraint on the state space. We proceed with the three-dimensional, rigid-body analogue of this thought experiment.

We conduct analysis with respect to an intermediate non-rotating frame \mathcal{A}_P with globally-fixed basis vectors and origin P fixed to some point on the vehicle. Vectors are demarcated by bold symbols as opposed to unbolted scalars. The \times operator is the cross product. Let $\mathbf{r}_{G/P}$ be the position of the vehicle's center-of-mass G relative to P . The relevant torques on the vehicle include: gravity ($\mathbf{r}_{G/P} \times m\mathbf{g}$, with m the mass of the vehicle), inertial torque (simplified as $\mathbf{r}_{G/P} \times \mathbf{a}$), and torques due to wheel forces $\boldsymbol{\tau}_P$. This leads us to the following torque-balance model,

$$\mathbb{I}_P \cdot \boldsymbol{\alpha} = \mathbf{r}_{G/P} \times m(\mathbf{g} - \mathbf{a}) + \boldsymbol{\tau}_P^{\text{res}}, \quad (1)$$

where $\boldsymbol{\alpha}$ is the angular acceleration and \mathbb{I}_P is the moment of inertia tensor about P . What is $\boldsymbol{\tau}_P^{\text{res}}$? In the following derivations, we will choose P such that the frictional (lateral) wheel forces have negligible torque contribution about P . As a result, $\boldsymbol{\tau}_P$ becomes directly proportional to the normal forces at the wheels. As it is also technically possible for the path dynamics to demand *negative* normal forces to satisfy the torque balance, we instead use the fictional “residual” torque $\boldsymbol{\tau}_P^{\text{res}}$. As we proceed, we vary our reference frame origin P (e.g. $P := L$ or $P := R$) depending on the failure mode of interest.

B. Rollover

On-road, rollovers generally occur when a turn is made so sharply that large lateral forces are induced as a result of

vehicle inertia. Without sufficient downward force to counteract this lateral or tipping force, this results in vehicle rollover. Rollover prevention for automobiles is well-studied and methods in essence constrain the normal forces at the vehicle's wheels [25, 5, 41, 13, 19].

Let \mathcal{B}_P represent the body frame with origin P which rigidly rotates with the vehicle. Let the basis unit vectors be \mathbf{b}_1 pointed forward in the direction of vehicle motion, \mathbf{b}_2 pointed to the left, and \mathbf{b}_3 pointed upwards away from the ground, completing the right-handed frame. Without loss of generality, we begin with modeling left rollover so that we are interested in the rotational forces which might cause the vehicle to roll counterclockwise along the \mathbf{b}_1 direction. Let L lay on the line from the front left to rear left wheels while aligning laterally with the center of mass such that $\mathbf{r}_{G/L} \cdot \mathbf{b}_1 = 0$. Now using the body frame \mathcal{B}_L for left rollover analysis, we want to bound $\boldsymbol{\tau}_L^{\text{res}}$ such that the wheel forces are always acting in support of the vehicle. This suggests that $\boldsymbol{\tau}_L^{\text{res}} \cdot \mathbf{b}_1 < 0$ (see Fig. 2 for diagram), i.e.,

$$[\mathbb{I}_L \cdot \boldsymbol{\alpha} - \mathbf{r}_{G/L} \times m(\mathbf{g} - \mathbf{a})] \cdot \mathbf{b}_1 < 0. \quad (2)$$

Observe that if these constraints were violated such that $\boldsymbol{\tau}_L^{\text{res}} \cdot \mathbf{b}_1 \geq 0$, our torque-balance would require the wheel forces to *pull* the vehicle down or, more realistically, result in **wheel airtime**. Likewise for right rollovers, the analysis is similarly conducted about R on the opposite side of the vehicle but with a change of sign such that $\boldsymbol{\tau}_R^{\text{res}} \cdot \mathbf{b}_1 > 0$.

2D Case: We consider the non-accelerating two-dimensional case and make the simplifying assumption that our rolling angular acceleration is small, $\boldsymbol{\alpha} \cdot \mathbf{b}_1 \approx 0$. Let P be either L or R such that $\mathbf{r}_{G/P} \cdot \mathbf{b}_1 = 0$ (see Fig. 3).

Note we can also decompose the acceleration \mathbf{a} by differentiating the vehicle velocity $\mathbf{v} := v\mathbf{b}_1$,

$$\begin{aligned} \mathbf{a} &= \frac{d}{dt}\mathbf{v} = \frac{d}{dt}[v]\mathbf{b}_1 + v\frac{d}{dt}[\mathbf{b}_1] \\ &= \dot{v}\mathbf{b}_1 + v\boldsymbol{\omega} \times \mathbf{b}_1 \end{aligned} \quad (3)$$

where we have used the fact that $\frac{d}{dt}[\mathbf{b}_1] = \boldsymbol{\omega} \times \mathbf{b}_1$ [21]. The residual torque from (2) can be combined with the decomposition from (3) and our non-acceleration assumption such that

$$\begin{aligned} \boldsymbol{\tau}_P^{\text{res}} \cdot \mathbf{b}_1 &= [\mathbb{I}_P \cdot \boldsymbol{\alpha} - \mathbf{r}_{G/P} \times m(\mathbf{g} - \mathbf{a})] \cdot \mathbf{b}_1 \\ &= -[\mathbf{r}_{G/P} \times m(\mathbf{g} - \mathbf{a})] \cdot \mathbf{b}_1 \\ &= -[\mathbf{r}_{G/P} \times m(\mathbf{g} - \dot{v}\mathbf{b}_1 - v\boldsymbol{\omega} \times \mathbf{b}_1)] \cdot \mathbf{b}_1 \end{aligned} \quad (4)$$

We note the vector product identity $(\mathbf{a} \times \mathbf{b}) \cdot \mathbf{c} = (\mathbf{a} \times \mathbf{c}) \cdot \mathbf{b}$ which allows us to further simplify such that

$$(\mathbf{r}_{G/P} \times \dot{v}\mathbf{b}_1) \cdot \mathbf{b}_1 = (\dot{v}\mathbf{b}_1 \times \mathbf{b}_1) \cdot \mathbf{r}_{G/P} = 0$$

A final vector triple product identity $(\mathbf{a} \times \mathbf{b}) \times \mathbf{c} = (\mathbf{a} \cdot \mathbf{c})\mathbf{b} - (\mathbf{a} \cdot \mathbf{b})\mathbf{c}$ allows further algebraic simplification,

$$\begin{aligned} \mathbf{r}_{G/P} \times (\boldsymbol{\omega} \times \mathbf{b}_1) &= (\mathbf{r}_{G/P} \cdot \mathbf{b}_1)\boldsymbol{\omega} - (\mathbf{r}_{G/P} \cdot \boldsymbol{\omega})\mathbf{b}_1 \\ &= -(\mathbf{r}_{G/P} \cdot \boldsymbol{\omega})\mathbf{b}_1 \end{aligned}$$

where we have used the fact that $\mathbf{r}_{G/P} \cdot \mathbf{b}_1 = 0$.

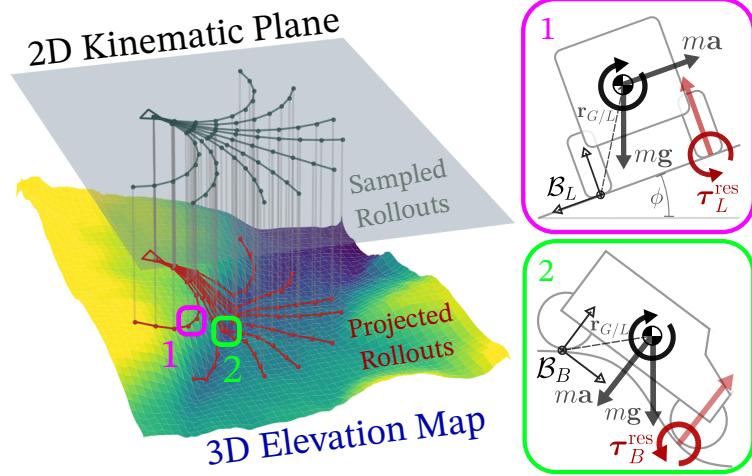


Fig. 2: Rollouts are sampled in the 2D kinematic plane then constrained by their implicit dynamics on the elevation map (left). Frames and forces for rollover analysis (rear-view, upper-right) and for ditch analysis (side-view, bottom-right).

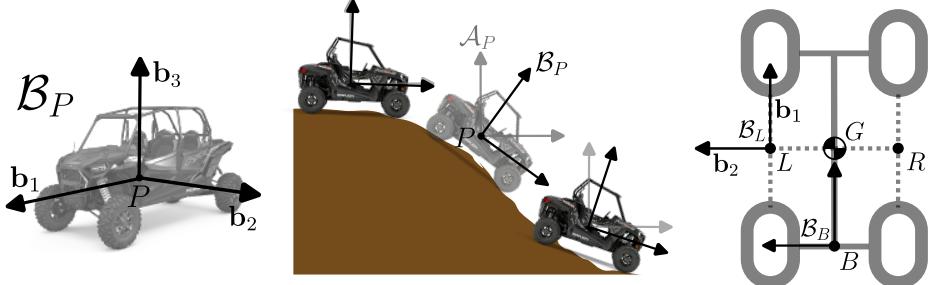


Fig. 3: Illustrations of body frame \mathcal{B}_P on vehicle (left), dynamics of \mathcal{B}_P and path frame \mathcal{A}_P (middle), and analysis frames \mathcal{B}_B and \mathcal{B}_L (right).

Reintroducing our simplifications into (1) gives

$$\tau_P^{\text{res}} \cdot \mathbf{b}_1 = -m(\mathbf{r}_{G/P} \times \mathbf{g}) \cdot \mathbf{b}_1 - mv(\mathbf{r}_{G/P} \cdot \boldsymbol{\omega}). \quad (5)$$

We can further simplify and scalarize by rewriting our basis vectors using known state quantities. Let ϕ be the roll angle of the vehicle such that $\mathbf{g} = g \sin \phi \mathbf{b}_2 - g \cos \phi \mathbf{b}_3$ with gravitational constant g . Let the position of the center of gravity in the body frame be given by $\mathbf{r}_{G/P} = P_2 \mathbf{b}_2 + P_3 \mathbf{b}_3$. For simplicity, we approximate the angular velocity here to be such that $\boldsymbol{\omega} \approx \omega_3 \mathbf{b}_3 \approx v\kappa \mathbf{b}_3$ where κ is the steering curvature. Then substituting our expressions for \mathbf{g} , $\boldsymbol{\omega}$, and $\mathbf{r}_{G/P}$ into (5) gives,

$$\tau_P^{\text{res}} \cdot \mathbf{b}_1 = -[P_2 g \cos \phi + P_3(v^2 \kappa + g \sin \phi)]$$

Depending on whether $P := L$ or $P := R$, the sign of P_2 and direction of inequality in (2) changes (Fig. 2). Both constraints can be simplified as

$$|P_3(v^2 \kappa + g \sin \phi)| < P_2 g \cos \phi, \quad (6)$$

which gives our physics-based rollover constraint in terms of our kinematic state and elevation map angles.

C. Ditches

In traversing ditches, vehicle-ground separation almost surely leads to extreme impact forces as the vehicle wheels

are reintroduced to the ground plane. Thus, like rollover prevention, we aim to discourage this separation in the first place. Our approach to the ditch constraint is similar to rollover prevention but instead focused on the front-back (about \mathbf{b}_2) rotational forces. Let B be between the rear left and right wheels, aligned with the center of mass such that $\mathbf{r}_{G/B} \cdot \mathbf{b}_2 = 0$. The residual torque τ_B^{res} should again always act in support of the vehicle such that $\tau_B^{\text{res}} \cdot \mathbf{b}_2 < 0$ (see Fig. 2). However, we find that this sole constraint is insufficient for ditch handling. In fact, it is also necessary to impose that the residual torque is not overly excessive. Namely, $\tau_B^{\text{res}} \cdot \mathbf{b}_2 > \tau_{\min}^{\text{res}}$ where τ_{\min}^{res} is a safety parameter. Observe that τ_B^{res} is directly related to the normal force at the front wheels. Thus, τ_{\min}^{res} corresponds to the greatest allowable normal force at the front wheels that we are willing to incur for the sake of mechanical stresses or rider comfort. We can summarize our constraint with the following bound:

$$\tau_{\min}^{\text{res}} < \tau_B^{\text{res}} \cdot \mathbf{b}_2 < 0. \quad (7)$$

2D Case: We derive a form for this inequality that is computable under our MPPI framework in the two-dimensional case. Again using the path frame decomposition from (3) and [21], the residual torque about B can be written as

$$\tau_B^{\text{res}} = \mathbb{I}_B \cdot \boldsymbol{\alpha} - \mathbf{r}_{G/B} \times m(\mathbf{g} - \dot{v}\mathbf{b}_1 - v\boldsymbol{\omega} \times \mathbf{b}_1).$$

In frame \mathcal{B}_B , we have $\mathbf{r}_{G/B} = B_1\mathbf{b}_1 + B_3\mathbf{b}_3$. We consider a strictly pitching maneuver such that $\boldsymbol{\omega} \approx \omega_2\mathbf{b}_2$ and $\mathbf{g} = g\sin\theta\mathbf{b}_1 - g\cos\theta\mathbf{b}_3$ where θ is the pitch of the vehicle. Substituting these terms,

$$\tau_B^{\text{res}} \cdot \mathbf{b}_2 = \bar{I}_{22}\alpha_2 + B_1(v\omega_2 + g\cos\theta) + B_3(\dot{v} + g\sin\theta), \quad (8)$$

where \bar{I}_{22} is the specific product of inertia about \mathbf{b}_2 . For conciseness, define $\tau_{\text{ditch}}^{\text{res}} := \tau_B^{\text{res}} \cdot \mathbf{b}_2$ such that reintroducing (7) gives us our simplified bound in terms of our kinematic state, elevation map, and allowable torque parameter τ_{\min}^{res} ,

$$\tau_{\min}^{\text{res}} < \tau_{\text{ditch}}^{\text{res}} < 0. \quad (9)$$

In summary, we have derived values for both rollover prevention and ditch handling that are strictly functions of our sampled state $\mathbf{x} = (x \ y \ \psi)$ and controls $\mathbf{u} = (v \ \kappa)$ and whose constraint computations are simple box bounds. Importantly, these constraints can be computed in-place, are time-independent, and are parallelizable across rollout sampling *and* the control horizon. This is critical for the fast and efficient computation of these constraints on our full system whose details we provide in the next section. A theoretical and empirical analysis is also made to better determine the errors introduced from our simplifying assumptions, which can be found in Section VII-D.

IV. SYSTEM OVERVIEW

Our system can be abstracted into an upstream perception model, mid-level sample-based MPC planner (MPPI), and a low-level controller (Fig. 4). Our robotic platform is a modified Polaris RZR Turbo S4, fitted with LiDAR and RGB-D cameras as well as four NVIDIA GeForce RTX 3080 GPUs. Note the control algorithm utilizes only one of these GPUs.

A. Perception

The perception model receives data from the onboard LiDARs and cameras as input. At a rate of about 10 Hz, it outputs two grid-maps with a radius of 50 meters centered at the vehicle's current location. One of these grid-maps, the costmap, provides semantic information about the environment (e.g. trees, grass, rocks) [6, 22, 36]. The second grid-map, the elevation map, provides elevation predictions at each corresponding location in the map. This is a common representation for ground geometry where sensor input is provided by LiDAR or stereo cameras [12, 14, 38, 42]. As done in [50, 16, 39], we use these elevations to calculate predicted attitude angles (roll ϕ and pitch θ) as demonstrated in Fig. 2. These angles are used to inform our autonomy in case of potential violations to our constraints.

B. MPPI

1) *Preliminaries:* Let \mathbf{x}_t and \mathbf{u}_t represent state and control at time t . For a given control horizon H , we randomly sample N open-loop control sequences $\{\mathbf{u}_{t:t+H-1}^i\}_{i=1}^N$. For each sequence, we compute their cumulative costs over the horizon $C(\mathbf{u}_{t:t+H-1}^i; \mathbf{x}_t) := \sum_{h=0}^{H-1} c(\mathbf{x}_{t+h}, \mathbf{u}_{t+h})$ using the predicted trajectory rollout given by the bicycle kinematics model [23]. Without a priori knowledge of the environment, a

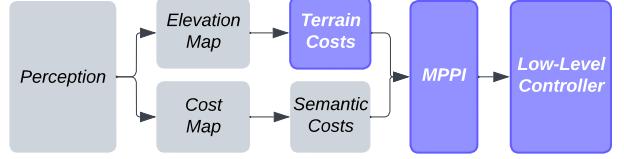


Fig. 4: Information flow across perception and control. The focus of this work are highlighted in purple.

model of the terrain dynamics can be impractical or altogether inaccessible. As inaccurate global models are known to result in poor performance for MPC [48], the abstracted bicycle model leverages the lower-level PI controller to better handle unmodeled terrain dynamics. With this model, our state can be represented by $\mathbf{x}_t = (x_t \ y_t \ \psi_t)$ where $(x_t \ y_t)$ is the position of the center of mass and ψ_t is the heading. Our control is then represented by $\mathbf{u}_t = (v_t \ \kappa_t)$ where v_t is the linear velocity and κ_t is the steering curvature. To obtain the optimal control command for the vehicle, we use the update law given by the model-predictive path integral (MPPI) control algorithm [47, 48, 1]. This family of MPC algorithms have become increasingly popular due to their flexibility with non-differentiable cost functions and parallelizability, especially with a kinematic model [1, 48].

2) *Fast and Parallel Model Rollouts:* We implement MPPI sampling, kinematic rollout, and cost function evaluation in a TorchScript PyTorch module running on an NVIDIA RTX 3080 GPU. Cost computations are vectorized across samples and the control horizon such that one iteration of MPPI samples are evaluated altogether in parallel. Our average MPPI computational time for one iteration is 29.6 ± 0.9 ms or about 33.8 Hz.

3) *Cost Function:* We evaluate costs of sampled kinematic rollouts against the elevation map. Our cost function ¹ is given by:

$$c(\mathbf{x}, \mathbf{u}) := w_1 c_{\text{rollover}}(\mathbf{x}, \mathbf{u}) + w_2 c_{\text{airtime}}(\mathbf{x}, \mathbf{u}) + w_3 c_{\text{bump}}(\mathbf{x}, \mathbf{u})$$

where each w_i is a hand-tuned weight, typically a large positive value to help disqualify unsafe trajectories from the control optimization.

Incorporating our rollover bound (6), the rollover cost term is a cumulative sum over the constraint violations,

$$c_{\text{rollover}}(\mathbf{x}_{t+h}, \mathbf{u}_{t+h}) := \sum_{k=t}^{t+h} RR(k) \cdot \mathbf{1}\{RR(k) > RR_{\max}\} \quad (10)$$

where,

$$RR(k) := \frac{|v_k^2 \kappa_k - g \sin \phi(k)|}{\cos \phi(k)} \quad (11)$$

and $RR_{\max} \leq g P_2 / P_3$ is the tunable parameter to control the vehicle's aggressiveness with rollover risk.

For ditch handling, in addition to the maximum allowable torque magnitude τ_{\max}^{res} , we introduce a tunable upper bound τ_{\max}^{res} such that $\tau_{\text{ditch}}^{\text{res}} < \tau_{\max}^{\text{res}} < 0$. While in theory, τ_{\max}^{res} can be

¹The full cost function includes more costs for handling the costmap, goal progress, etc. which we provide detail in appendix VII-B

0, this corresponds to the vehicle “hovering” above the ground, which can result in loss of control or instability. As the linear accelerations \dot{v}_t sampled from MPPI can be noisy, we remove \dot{v}_t from (9) to obtain a more tunable bound. We also compute ω_2 and α_2 through forward numerical differencing of $\theta(t)$. Our time-indexed $\tau_{\text{ditch}}^{\text{res}}$ can be computed as

$$\tau_{\text{ditch}}^{\text{res}}(t) = \bar{I}_{22}\alpha_2(t) + P_1v_t\omega_2(t) - P_3g \sin \theta(t) - P_1g \cos \theta(t)$$

For conciseness, define $\delta_{\min}(t) := \tau_{\min}^{\text{res}} - \tau_{\text{ditch}}^{\text{res}}(t)$ and $\delta_{\max}(t) := \tau_{\text{ditch}}^{\text{res}}(t) - \tau_{\max}^{\text{res}}$. As with rollover, we employ the cumulative sum of violations,

$$c_{\text{airtime}}(\mathbf{x}_{t+h}, \mathbf{u}_{t+h}) := \sum_{k=t}^{t+h} \delta_{\max}(k) \cdot \mathbf{1}\{\delta_{\max}(k) > 0\} \quad (12)$$

$$c_{\text{bump}}(\mathbf{x}_{t+h}, \mathbf{u}_{t+h}) := \sum_{k=t}^{t+h} \delta_{\min}(k) \cdot \mathbf{1}\{\delta_{\min}(k) > 0\} \quad (13)$$

The cumulative (running) sum in (6) and (9) deserve special attention. This heuristic is important because violations of our constraint almost surely imply violations of our modeling assumptions later in the horizon. For instance, if the vehicle is predicted to leave the ground, subsequent predictions have higher uncertainty and should be penalized accordingly. The cumulative sum also strongly penalizes sooner violations over later violations in the horizon.

4) Tuning: We find rollover parameter RR_{\max} is theoretically and empirically correlated with lateral force on the vehicle. To tune this parameter, we begin at $RR_{\max} := 2$ and increase the value to $RR_{\max} := 3.4$ where the human safety operator deems the behavior too risky. This is a qualitative assessment due to the diversity of terrain conditions but can generally include occurrences such as slipping, tipping, or instability.

We find ditch parameter τ_{\max}^{res} correlates with reductions while τ_{\min}^{res} correlates with increases in normal force as perceived by the operator. Starting with $\tau_{\max}^{\text{res}} := -5$ and $\tau_{\min}^{\text{res}} := -7$, this bound is widened to $\tau_{\max}^{\text{res}} := -3$ and $\tau_{\min}^{\text{res}} := -10$ where the safety operator deems the behavior too risky or intolerable. If the operator feels the vehicle lifting away from the ground too quickly, τ_{\max}^{res} is lowered. If the operator deems ground impact too forceful, τ_{\min}^{res} is raised.

5) Handling Control Delays: In practice, the issued control command $\hat{\mathbf{u}}_t$ can not be immediately executed on the vehicle due to software and hardware delays. This means that we actually have $\mathbf{x}_{t+1} = f(\mathbf{x}_t, \hat{\mathbf{u}}_{t-\tau})$ for some delay $\tau \geq 0$. To handle delays, we consider the MPC optimization problem of horizon H from a projected future state $\tilde{\mathbf{x}}_t$ (rather than the true state \mathbf{x}_t). The projected future state $\tilde{\mathbf{x}}_t$ is given by rolling out the previously issued controls $(\hat{\mathbf{u}}_{t-\tau}, \dots, \hat{\mathbf{u}}_{t-1})$ starting from state \mathbf{x}_t . These prepended controls effectively simulate the control commands that are issued but yet to be executed on the system.

6) Sampling Control Sequences: The original MPPI algorithm samples the control sequences $\{\mathbf{u}_{t:t+H-1}^i\}_{i=1}^N$ from a Gaussian distribution centered around a nominal control sequence given by the shift operator [47, 48, 1, 46]. In our implementation, we sample raw control sequences $\{\tilde{\mathbf{u}}_{t:t+H-1}^i\}_{i=1}^N$

from a mixture of Gaussian distributions², and process the raw control sequences to generate sampled control sequences. When necessary, these Gaussians help the optimization with: (1) denser sampling around the nominal trajectory, (2) abrupt slow-downs, and (3) resetting local convergence on high-cost regions.

a) Sampling feasible control sequences: The sampled changes in linear velocity and curvature must be limited due to the vehicle’s throttle, brake, and steering limitations. Additionally due to static friction, the steering is unchanged at linear velocities near zero.

We process the raw control sequences $\{\tilde{\mathbf{u}}_{t:t+H-1}^i\}_{i=1}^N$ so that feasibility constraints are satisfied. Let $\tilde{\mathbf{u}}_{t+h}^i = (\tilde{v}_{t+h}^i, \tilde{\kappa}_{t+h}^i)$, we iteratively compute the processed control sequence $\mathbf{u}_{t+h}^i = (v_{t+h}^i, \kappa_{t+h}^i)$,

$$v_{t+h}^i = \text{clip}(\tilde{v}_{t+h}^i, v_{t+h-1}^i \pm \Delta v_{\max}), \\ \kappa_{t+h}^i = \begin{cases} \kappa_{t+h-1}^i & \text{if } |v_{t+h}^i| < v_{\min}, \\ \text{clip}(\tilde{\kappa}_{t+h}^i, \kappa_{t+h-1}^i \pm \Delta \kappa_{\max}) & \text{otherwise,} \end{cases} \quad (14)$$

where $\text{clip}(a, b \pm c) := \max(\min(a, b+c), b-c)$ for $a, b, c \in \mathbb{R}$ and $c > 0$. Δv_{\max} and $\Delta \kappa_{\max}$ are the maximum changes of linear velocity and curvature, respectively, per time step, and v_{\min} is the minimum speed that allows change of steering. We take $(v_{t-1}^i, \kappa_{t-1}^i)$ as the control issued at the previous step, $\hat{\mathbf{u}}_{t-1}$, for all i .

C. Low-Level Controller

Off-road environments have wildly varying ground conditions (friction, compliance, looseness, roughness, etc.) which can make direct optimization over throttle extremely challenging as this requires a second-order model over the environment. As done in the DARPA Subterranean and DARPA Grand Challenge [18, 40], we depend upon a low-latency controller to mitigate these unmodeled dynamics. We employ a low-level PI feedback controller to attain desired speeds from MPPI using vehicle throttle and brake:

$$u = -K_P(v - v_t) - K_I \int (v - v_t) dt + c_1 g \sin \theta + c_2 v$$

where v_t is the desired linear velocity from MPPI, and v is the actual velocity of the vehicle. The feedforward terms $c_1 g \sin \theta + c_2 v$ compensate for the acceleration in the b_1 direction from gravity and the acceleration due to drag and friction. We determine the linear coefficients, c_1 and c_2 empirically from data and tune the proportional and integral gains K_P and K_I . u is then converted to throttle (when u is positive) or brake (when u is negative) using a pre-identified fixed mapping. Finally, for the steering angle, we directly translate the desired curvature κ_t from MPPI to the steering angle command with a pre-identified fixed mapping.

²See appendix VII-A for more details on the mixture distributions used.

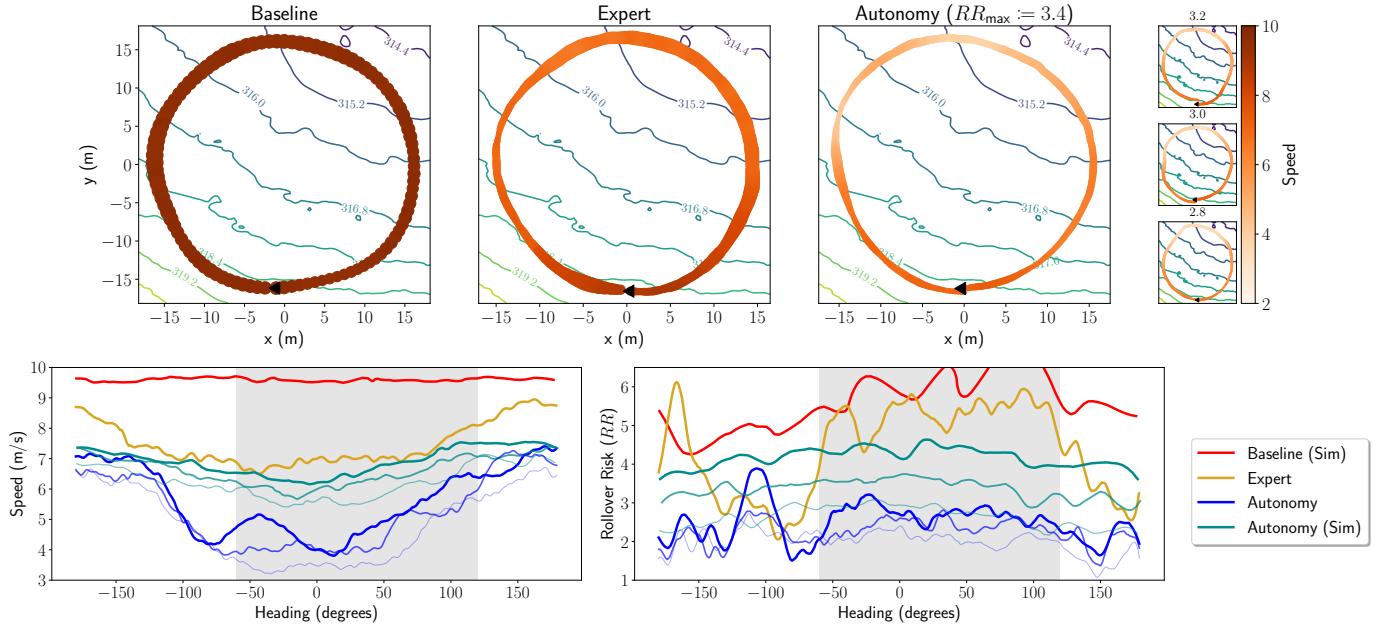


Fig. 5: *Top row*. Spatial plot over elevation contour map of 15-meter radius circles on a 10-degree incline hill. Results for each experiment are averaged over at least five loops. Speed and RR are reflected by the line color and marker width, respectively. Autonomy results of decreasing RR_{\max} control parameter are shown in the right column. *Bottom row*. Plots of speed and rollover risk as a function of heading angle. Off-camber turning highlighted by grey box. Thicker and opaquer autonomy lines correspond with higher RR_{\max} settings (Autonomy $RR_{\max} \in \{3.0, 3.2, \mathbf{3.4}\}$; Autonomy (Sim) $RR_{\max} \in \{4, 5, \mathbf{6}\}$). Autonomy (Sim) simulates MPPI for higher than tolerable RR_{\max} values. Baseline results are simulated due to unsafe behavior.

V. RESULTS

Our experiments consist of two groups of isolated tasks and three (two plus one reverse) complex courses. The first isolated task involves completing circles on slopes and is meant to test the robot's ability to avoid rollovers. The second task is crossing various ditches. We evaluate the geometry-based method (the baseline)³, the physics-based method (ours; also referred to as the autonomy), and human-driven results (the expert). We refer readers to the paper's website for videos of our experiments and general off-road performance.

A. Rollover

We consider a baseline algorithm which uses a cost that penalizes roll angles above 20 degrees, similar to [9, 11, 24]. We compare the expert to the baseline in simulation (Fig. 5). While the expert consistently operates at about $RR(t) \leq 6$, the baseline goes well above this threshold and in theory can surpass this arbitrarily as the slope angle never exceeds 20 degrees. At $RR_{\max} \approx 6$, the expert is already experiencing loss of control from slipping and tilting. Far exceeding this threshold, the baseline would likely cause a rollover or severe loss of control in the real world.

We also compare the expert to our method, with parameter $RR_{\max} := 3.4$ (Fig. 5). For safety reasons, we do not autonomously operate the real vehicle at $RR_{\max} > 3.4$. Instead, we conduct real and simulated tests at 6 different settings of

RR_{\max} to show that autonomous behavior indeed trends safely toward expert behavior with increasing values of RR_{\max} . In particular, the expert drives fastest through the on-camber turn and slowest through the off-camber turn (also observable in the slalom in Fig. 1.3). Note that MPPI often operates below its RR_{\max} setting due to the penalizing tails of the control Gaussian [48]. The expert also likely drives faster during the on-camber turn due to suspension loading which is not modeled in our constraints. Nonetheless, the autonomy consistently and safely remains below expert speeds with some tunable margin as desired.

B. Ditches

Similar to the rollover task, we seek to qualitatively evaluate our method in ditch crossing by comparing it with the expert (Fig. 6). Without explicit velocity constraints, this experiment verifies that the vehicle adequately traverses through ditches in a safe manner. The tests are conducted over three distinct ditches with varying geometries. In these experiments, the autonomy's maximum speed was limited to 5 m/s for safety.

We note a couple differences between the Autonomy and Expert that can be attributed to the Expert's access to prior knowledge about the environment. The autonomy's slower crossing speeds in the River (reverse) and Crater ditches are largely due to the sharp exiting ridges. Proceeding at the expert-executed speeds would result in infringing on the ditch upper-bound τ_{\max}^{res} and excessively reducing contact with the ground. As the expert has prior knowledge about what exists

³for details about the geometry-based method see appendix VII-C



Fig. 6: Robotic platform crossing the V-Ditch (pictured above; driver is present only for safety purposes). Average velocity is plotted as a function of the position through the ditch over the elevation profile (brown). Individual ditch traversals are plotted transparently.

beyond the ridge, they are more comfortable with momentarily reducing control authority in return for speed. Secondly, the expert decelerates much earlier than the autonomy before the crossing. Note that the descent angle of the ditch is often occluded from the vehicle’s LiDAR and camera during the approach and can result in delayed sensing of the ditch geometry. This results in the autonomy’s sharp decelerations in the V-Ditch and river crossings whose descent angles are initially occluded but revealed last-minute. However, as the expert has knowledge of the occluded regions of the ditch, they are able to make more comfortable decelerations in anticipation.

C. Complex Course

Our ultimate objective is to autonomously navigate through complex terrain. To evaluate this, we run the autonomy through the Tour, the Ruot (backwards Tour), and the Baja (Fig. 7). To evaluate method performance, we are interested in three key metrics:

1) Interventions: An intervention occurs when the human safety operator (on standby during autonomous operation) must make a manual maneuver (e.g. quickly stop or evade)

to reset the vehicle due to poor or dangerous decision-making. More interventions generally correlate with “worse” autonomy.

2) Course Time: The course time is the total time it takes the vehicle to complete the course by reaching every waypoint in order. An aim of aggressive driving is to (safely) reduce travelling time.

3) Ditch False-Positives: A falsely identified ditch (false-positive or FP) is a common complication in the development of a ditch control method due in part to imperfect perception. Because safe ditch traversal speeds are typically around 1-3 m/s, unnecessary slowdowns from our nominal speeds of 7-10 m/s are detrimental to aggressive autonomy.

We find that the geometry-based baseline is more susceptible to ditch FPs despite best efforts in parameter tuning to maintain safe ditch traversal while reducing FPs. It is likely due to pure geometrical reasoning, which treats all detected ditches equally regardless of current speed or actual severity. Moreover, there is less flexibility in the tunable parameter space to compensate for this, since we have shown in section III the severity is necessarily correlated

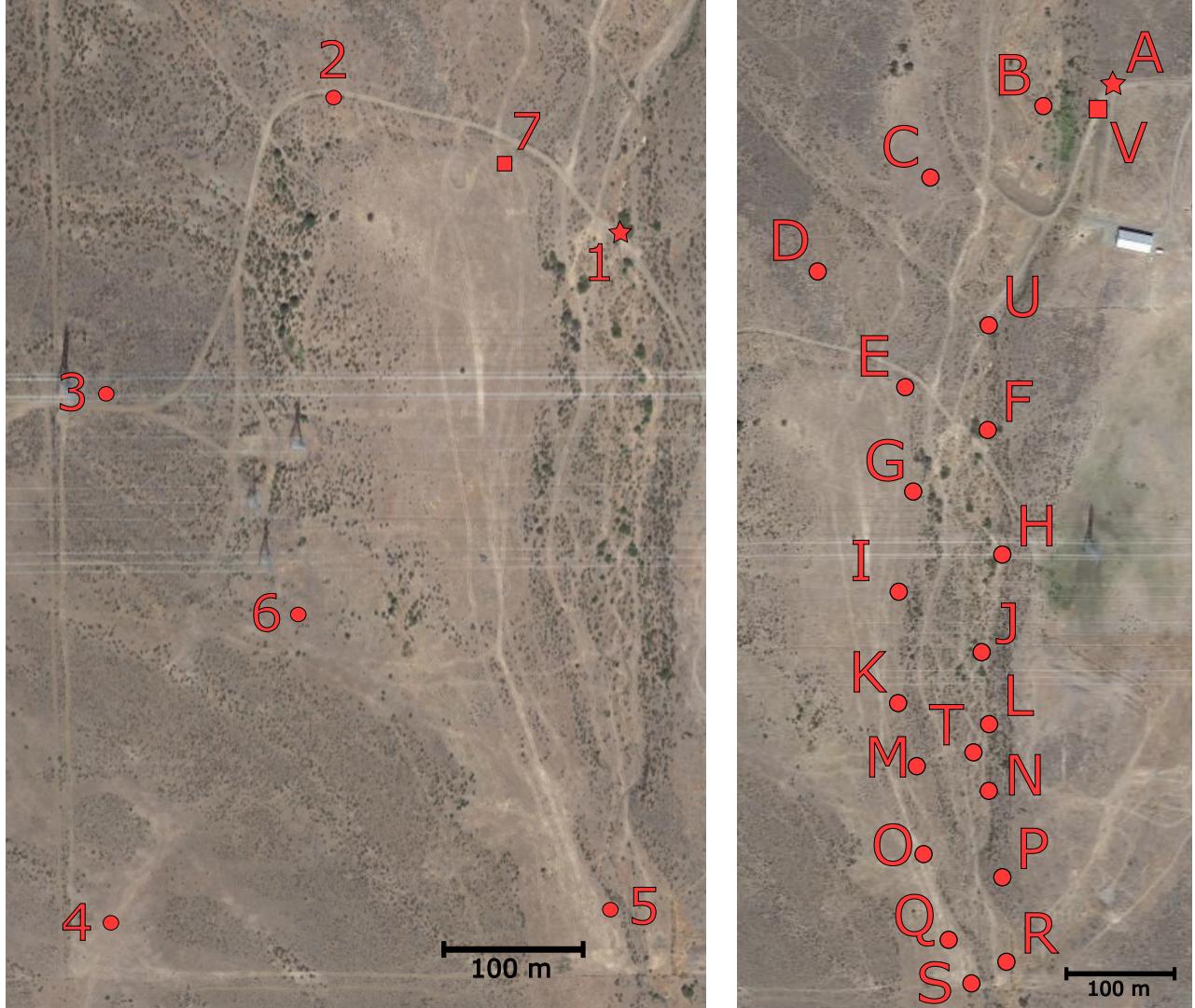


Fig. 7: Complex courses Tour (left) and Baja (right) with numbered and alphabetized waypoints. The Tour is approximately two kilometers in total distance with ± 80 m of elevation change. Reaching waypoints 4 through 7 require traversal of steep slopes and ditches. Waypoints 1 through 4 test trail-following and nominal behaviors (obstacle avoidance and basic navigation and control). Baja is approximately three kilometers and repeatedly crosses the region densest in ditches and vegetation from waypoints E through U.

with traversal speeds. The accumulation of multiple FPs inevitably increases the course time on the Tour for the baseline. With our method on the Ruot, we flag a potential FP encountered when cresting a steep hill (waypoint 5 to 4). Without slippage modeling, MPPI believes its command speed will result in airtime (12) at the crest. In reality, the vehicle lacks momentum and traction to cleanly crest the hill and begins to stall. An intervention is made to prevent damage to the transmission belt. On the Baja, which involves primarily ditches and obstacle avoidance, false-positives are less likely due to the dense presence of ditches. However, some of these ditches do not require full slowdowns for safe traversal. The baseline method does not accommodate this and results in longer course completion times due to over-slowing.

	Tour			Ruot		Baja	
	None	Base	Ours	Base	Ours	Base	Ours
Course Time	5:26	6:01	5:28	5:35	5:48	11:22	8:54
Ditch FP	N/A	8	0	6	1	0	0
Interventions	1	0	0	0	1	0	0

TABLE I: Run statistics on complex courses (see Fig. 7). Base indicates the geometry-based baseline. Our method is the physics-based approach in this work. “None” indicates no ditch-control method as a part of the autonomy.

VI. CONCLUSION

In this study, we propose a framework to augment 2D kinematic sampling and real-time elevation maps with physics-based constraints for safe and aggressive off-road autonomous driving. This framework yields two constraints addressing failures due to rollover and unsafe ditch handling which are fully parallelizable for sample-based MPC. Our control system design integrates this approach with a fast and efficient implementation of the MPPI algorithm. Finally, we validate and compare our method to a purely geometry-based baseline and an expert driver through real-world experiments. These experiments include autonomously traversing hills, ditches, and full complex courses on our full-scale robotic platform.

We identify a few limitations to our methods which we should be pursued in future work. While kinematic sampling is fast and parallelizable, uneven terrain with increasingly high-frequency features (e.g. rock beds) demand denser sampling methods to provide adequate signal for our constraints to behave as expected. Secondly, the impact force of a ditch cannot sometimes be strictly modeled as a path-induced normal force, i.e. collisions. Assuming such an environment can be properly perceived, predicting collisions like these at present can be computationally prohibitive. Our operating environments can also range widely with no prior information and our current methods can require significant tuning. Adaptive or learning-based methods for dynamics models for control can be highly impactful for improving system-wide performance in this setting. Finally, by comparing our method’s results to an expert driver, we observe similarities that demonstrate our approach’s ability to capture fundamental elements of safe yet aggressive control. However, beyond physical capabilities, the expert’s ability to reason about uncertainty sets it apart from the autonomy. Future work in this area can be hugely impactful in the real-world setting.

ACKNOWLEDGMENT

This research was developed with funding from the Defense Advanced Research Projects Agency (DARPA). The views, opinions and/or findings expressed are those of the author and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government. This material is also partially based upon work supported by the National Science Foundation Graduate Research Fellowship under Grant No. DGE-2140004. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

- [1] Mohak Bhardwaj, Balakumar Sundaralingam, Arsalan Mousavian, Nathan D Ratliff, Dieter Fox, Fabio Ramos, and Byron Boots. Storm: An integrated framework for fast joint-space model-predictive control for reactive manipulation. In *Conference on Robot Learning*, pages 750–759. PMLR, 2021.
- [2] Daniel W Carruth, Clayton T Walden, Christopher Goodin, and Sara C Fuller. Challenges in low infrastructure and off-road automated driving. In *2022 Fifth International Conference on Connected and Autonomous Driving (MetroCAD)*, pages 13–20. IEEE, 2022.
- [3] Mateo Guaman Castro, Samuel Triest, Wenshan Wang, Jason M. Gregory, Felix Sanchez, John G. Rogers, and Sebastian Scherer. How does it feel? self-supervised costmap learning for off-road vehicle traversability. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, page 931–938, May 2023. doi: 10.1109/ICRA48891.2023.10160856. URL <https://ieeexplore.ieee.org/abstract/document/10160856>.
- [4] Brian César-Tondreau, Garrett Warnell, Kevin Kochersberger, and Nicholas R Waytowich. Towards fully autonomous negative obstacle traversal via imitation learning based control. *Robotics*, 11(4):67, 2022.
- [5] Bo-Chiuan Chen and Huei Peng. Differential-braking-based rollover prevention for sport utility vehicles with human-in-the-loop evaluations. *Vehicle system dynamics*, 36(4-5):359–389, 2001.
- [6] Kashyap Chitta, Aditya Prakash, and Andreas Geiger. Neat: Neural attention fields for end-to-end autonomous driving. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, page 15773–15783, Montreal, QC, Canada, Oct 2021. IEEE. ISBN 978-1-66542-812-5. doi: 10.1109/ICCV48922.2021.01550. URL <https://ieeexplore.ieee.org/document/9710855/>.
- [7] D. Coombs, K. Murphy, A. Lacaze, and S. Legowik. Driving autonomously off-road up to 35 km/h. In *Proceedings of the IEEE Intelligent Vehicles Symposium 2000 (Cat. No.00TH8511)*, page 186–191, Dearborn, MI, USA, 2000. IEEE. ISBN 978-0-7803-6363-2. doi: 10.1109/IVS.2000.898339. URL <http://ieeexplore.ieee.org/document/898339>.
- [8] H Dahmani, M Chadli, A Rabhi, and A El Hajjaji. Vehicle dynamic estimation with road bank angle consideration for rollover detection: theoretical and experimental studies. *Vehicle System Dynamics*, 51(12):1853–1871, 2013.
- [9] Aniket Datar, Chenhui Pan, and Xuesu Xiao. Learning to model and plan for wheeled mobility on vertically challenging terrain. *arXiv preprint arXiv:2306.11611*, 2023.
- [10] M. Doumiati, A. Victorino, A. Charara, and D. Lechner. Lateral load transfer and normal forces estimation for vehicle safety: experimental test. *Vehicle System Dynam-*

- ics*, 47(12):1511–1533, Dec 2009. ISSN 0042-3114. doi: 10.1080/00423110802673091.
- [11] David D Fan, Kyohei Otsu, Yuki Kubo, Anushri Dixit, Joel Burdick, and Ali-Akbar Agha-Mohammadi. Step: Stochastic traversability evaluation and planning for risk-aware off-road navigation. *arXiv preprint arXiv:2103.02828*, 2021.
- [12] Péter Fankhauser, Michael Bloesch, and Marco Hutter. Probabilistic terrain mapping for mobile robots with uncertain localization. *IEEE Robotics and Automation Letters*, 3(4):3019–3026, Oct 2018. ISSN 2377-3766. doi: 10.1109/LRA.2018.2849506.
- [13] Thomas Fork, H. Eric Tseng, and Francesco Borrelli. Models and predictive control for nonplanar vehicle navigation. (arXiv:2104.08427), April 2022. URL <http://arxiv.org/abs/2104.08427>. arXiv:2104.08427 [cs, eess].
- [14] Bianca Forkel and Hans-Joachim Wuensche. Dynamic resolution terrain estimation for autonomous (dirt) road driving fusing lidar and vision. In *2022 IEEE Intelligent Vehicles Symposium (IV)*, page 1181–1187, Aachen, Germany, Jun 2022. IEEE Press. doi: 10.1109/IV51971.2022.9827214. URL <https://doi.org/10.1109/IV51971.2022.9827214>.
- [15] Jonas Frey, Shehryar Khattak, Manthan Patel, Deegan Atha, Julian Nubert, Curtis Padgett, Marco Hutter, and Patrick Spieler. Roadrunner - learning traversability estimation for autonomous off-road driving. (arXiv:2402.19341), March 2024. URL <http://arxiv.org/abs/2402.19341>. arXiv:2402.19341 [cs].
- [16] Jason Gibson, Bogdan Vlahov, David Fan, Patrick Spieler, Daniel Pastor, Ali-akbar Agha-mohammadi, and Evangelos A. Theodorou. A multi-step dynamics modeling framework for autonomous driving in multiple environments. (arXiv:2305.02241), May 2023. URL <http://arxiv.org/abs/2305.02241>. arXiv:2305.02241 [cs, eess].
- [17] Tianrui Guan, Zhenpeng He, Ruitao Song, Dinesh Manocha, and Liangjun Zhang. Tns: Terrain traversability mapping and navigation system for autonomous excavators. (arXiv:2109.06250), May 2022. URL <http://arxiv.org/abs/2109.06250>. arXiv:2109.06250 [cs].
- [18] Gabriel M. Hoffmann, Claire J. Tomlin, Michael Montemerlo, and Sebastian Thrun. Autonomous automobile trajectory tracking for off-road driving: Controller design, experimental validation and racing. In *2007 American Control Conference*, page 2296–2301, New York, NY, USA, July 2007. IEEE. ISBN 978-1-4244-0988-4. doi: 10.1109/ACC.2007.4282788. URL <http://ieeexplore.ieee.org/document/4282788/>.
- [19] Thomas M. Howard and Alonzo Kelly. Optimal rough terrain trajectory generation for wheeled mobile robots. *The International Journal of Robotics Research*, 26(2):141–166, 2007. doi: 10.1177/0278364906075328. URL <https://doi.org/10.1177/0278364906075328>.
- [20] Ronald L Huston and Fred A Kelly. Another look at the static stability factor (ssf) in predicting vehicle rollover. *International journal of crashworthiness*, 19(6):567–575, 2014.
- [21] Jeremy Kasdin and Derek Paley. *Engineering Dynamics*. Mar 2011. ISBN 978-0-691-13537-3. URL <https://press.princeton.edu/books/hardcover/9780691135373/engineering-dynamics>.
- [22] Alonzo Kelly, Anthony Stentz, Omead Amidi, Mike Bode, David Bradley, Antonio Diaz-Calderon, Mike Happold, Herman Herman, Robert Mandelbaum, Tom Pilarski, Pete Rander, Scott Thayer, Nick Vallidis, and Randy Warner. Toward reliable off road autonomous vehicles operating in challenging environments. *The International Journal of Robotics Research*, 25(5–6):449–483, May 2006. ISSN 0278-3649, 1741-3176. doi: 10.1177/0278364906065543.
- [23] Jason Kong, Mark Pfeiffer, Georg Schildbach, and Francesco Borrelli. Kinematic and dynamic vehicle models for autonomous driving control design. In *2015 IEEE Intelligent Vehicles Symposium (IV)*, page 1094–1099, Jun 2015. doi: 10.1109/IVS.2015.7225830.
- [24] Philipp Krüsi, Paul Furgale, Michael Bosse, and Roland Siegwart. Driving on point clouds: Motion planning, trajectory optimization, and terrain assessment in generic nonplanar environments. *Journal of Field Robotics*, 34(5):940–984, 2017.
- [25] Chad Larish, Damrongrit Piyabongkarn, Vasilios Tsourapas, and Rajesh Rajamani. A new predictive lateral load transfer ratio for rollover prevention systems. *IEEE Transactions on Vehicular Technology*, 62(7):2928–2936, Sep 2013. ISSN 1939-9359. doi: 10.1109/TVT.2013.2252930.
- [26] Hojin Lee, Taekyung Kim, Jungwi Mun, and Wonsuk Lee. Learning terrain-aware kinodynamic model for autonomous off-road rally driving with model predictive path integral control. *IEEE Robotics and Automation Letters*, 8(11):7663–7670, November 2023. ISSN 2377-3766, 2377-3774. doi: 10.1109/LRA.2023.3318190.
- [27] Xingyu Li, Bo Tang, John Ball, Matthew Doude, and Daniel W. Carruth. Rollover-free path planning for off-road autonomous driving. *Electronics*, 8(66):614, Jun 2019. ISSN 2079-9292. doi: 10.3390/electronics8060614.
- [28] Yugang Liu and Goldie Nejat. Robotic urban search and rescue: A survey from the control perspective. *Journal of Intelligent & Robotic Systems*, 72:147–165, 2013.
- [29] Liang Ma, Jianru Xue, Kuniaki Kawabata, Jihua Zhu, Chao Ma, and Nanning Zheng. Efficient sampling-based motion planning for on-road autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 16(4):1961–1976, Aug 2015. ISSN 1558-0016. doi: 10.1109/TITS.2015.2389215.
- [30] Parv Maheshwari, Wenshan Wang, Samuel

- Triest, Matthew Sivaprakasam, Shubhra Aich, John G. Rogers III, Jason M. Gregory, and Sebastian Scherer. Piaug – physics informed augmentation for learning vehicle dynamics for off-road navigation. (arXiv:2311.00815), November 2023. doi: 10.48550/arXiv.2311.00815. URL <http://arxiv.org/abs/2311.00815>. arXiv:2311.00815 [cs].
- [31] Daniel Maturana, Po-Wei Chou, Masashi Uenoyama, and Sebastian Scherer. Real-time semantic mapping for autonomous off-road navigation. In *Field and Service Robotics: Results of the 11th International Conference*, pages 335–350. Springer, 2018.
- [32] Xiangyun Meng, Nathan Hatch, Alexander Lambert, Anqi Li, Nolan Wagener, Matthew Schmittle, JoonHo Lee, Wentao Yuan, Zoey Chen, Samuel Deng, et al. Terrainnet: Visual modeling of complex terrain for high-speed, off-road navigation. *arXiv preprint arXiv:2303.15771*, 2023.
- [33] Hossein Mousazadeh. A technical review on navigation systems of agricultural autonomous off-road vehicles. *Journal of Terramechanics*, 50(3):211–232, 2013.
- [34] Brian Paden, Michal Čáp, Sze Zheng Yong, Dmitry Yershov, and Emilio Frazzoli. A survey of motion planning and control techniques for self-driving urban vehicles. *IEEE Transactions on Intelligent Vehicles*, 1(1):33–55, Mar 2016. ISSN 2379-8904. doi: 10.1109/TIV.2016.2578706.
- [35] Jia-Ern Pai. Trends and rollover-reduction effectiveness of static stability factor in passenger vehicles. Technical report, 2017.
- [36] Amirreza Shaban, Xiangyun Meng, JoonHo Lee, Byron Boots, and Dieter Fox. Semantic terrain classification for off-road autonomous driving. In Aleksandra Faust, David Hsu, and Gerhard Neumann, editors, *Proceedings of the 5th Conference on Robot Learning*, volume 164 of *Proceedings of Machine Learning Research*, pages 619–629. PMLR, 08–11 Nov 2022. URL <https://proceedings.mlr.press/v164/shaban22a.html>.
- [37] David Stavens, G. Hoffmann, and S. Thrun. Online speed adaptation using supervised learning for high-speed, off-road autonomous driving. January 2007.
- [38] Maximilian Stözl, Takahiro Miki, Levin Gerdes, Martin Azkarate, and Marco Hutter. Reconstructing occluded elevation information in terrain maps with self-supervised learning. *IEEE Robotics and Automation Letters*, 7(2):1697–1704, Apr 2022. ISSN 2377-3766. doi: 10.1109/LRA.2022.3141662.
- [39] Qifan Tan, Cheng Qiu, Jing Huang, Yue Yin, Xinyu Zhang, and Huaping Liu. Path tracking control strategy for off-road 4ws4wd vehicle based on robust model predictive control. *Robotics and Autonomous Systems*, 158:104267, Dec 2022. ISSN 09218890. doi: 10.1016/j.robot.2022.104267.
- [40] Rohan Thakker, Nikhilesh Alatur, David D. Fan, Jesus Tordesillas, Michael Paton, Kyohei Otsu, Olivier Toupet, and Ali-akbar Agha-mohammadi. Autonomous off-road navigation over extreme terrains with perceptually-challenging conditions. In Bruno Siciliano, Cecilia Laschi, and Oussama Khatib, editors, *Experimental Robotics*, page 161–173, Cham, 2021. Springer International Publishing. ISBN 978-3-030-71151-1. doi: 10.1007/978-3-030-71151-1_15.
- [41] Antonio Tota, Luca Dimauro, Filippo Velardocchia, Genny Paciullo, and Mauro Velardocchia. An intelligent predictive algorithm for the anti-rollover prevention of heavy vehicles for off-road applications. *Machines*, 10(1010):835, Oct 2022. ISSN 2075-1702. doi: 10.3390/machines10100835.
- [42] Rudolph Triebel, Patrick Pfaff, and Wolfram Burgard. Multi-level surface maps for outdoor terrain mapping and loop closing. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, page 2276–2282, Beijing, China, Oct 2006. IEEE. ISBN 978-1-4244-0258-8. doi: 10.1109/IROS.2006.282632. URL <http://ieeexplore.ieee.org/document/4058725/>.
- [43] Samuel Triest, Matthew Sivaprakasam, Sean J Wang, Wenshan Wang, Aaron M Johnson, and Sebastian Scherer. Tartandrive: A large-scale dataset for learning off-road dynamics models. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 2546–2552. IEEE, 2022.
- [44] Samuel Triest, Mateo Guaman Castro, Parv Maheshwari, Matthew Sivaprakasam, Wenshan Wang, and Sebastian Scherer. Learning risk-aware costmaps via inverse reinforcement learning for off-road navigation. *arXiv preprint arXiv:2302.00134*, 2023.
- [45] Carnegie Mellon University. Pushing limits of off-road autonomy in darpa’s racer program - national robotics engineering center - carnegie mellon university, 2022. URL <http://www.cmu.edu/nrec/media/news-stories-pages/news-stories-2022/darpa-racer.html>.
- [46] N Wagener, C Cheng, J Sacks, and B Boots. An online learning approach to model predictive control. In *Proceedings of Robotics: Science and Systems (RSS)*, 2019.
- [47] Grady Williams, Andrew Aldrich, and Evangelos A Theodorou. Model predictive path integral control: From theory to parallel computation. *Journal of Guidance, Control, and Dynamics*, 40(2):344–357, 2017.
- [48] Grady Williams, Nolan Wagener, Brian Goldfain, Paul Drews, James M Rehg, Byron Boots, and Evangelos A Theodorou. Information theoretic mpc for model-based reinforcement learning. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1714–1721. IEEE, 2017.
- [49] Xuesu Xiao, Joydeep Biswas, and Peter Stone. Learning inverse kinodynamics for accurate high-speed off-road navigation on unstructured terrain. *IEEE Robotics and Automation Letters*, 6(3):6054–6060, 2021.

- [50] Siyuan Yu, Congkai Shen, and Tulga Ersal. Nonlinear model predictive planning and control for high-speed autonomous vehicles on 3d terrains. *IFAC-PapersOnLine*, 54(20):412–417, Jan 2021. ISSN 2405-8963. doi: 10.1016/j.ifacol.2021.11.208.

Supplementary Materials

Quantity	Assumption	Maximum Constraint Error	
		Theoretical	Empirical
RR_{\max}	$\alpha \cdot \mathbf{b}_1 \approx 0$	$\bar{I}_{11} \alpha_1 $	0.01
	$\omega \approx \omega_3 \mathbf{b}_3$	$ v \ \mathbf{r}_{G/P}\ \ \omega - \omega_3 \mathbf{b}_3\ $	0.9
	$\omega_3 \approx v\kappa$	$ P_3 v \omega_3 - v\kappa $	0.8
$\tau_{\min}^{\text{res}}, \tau_{\max}^{\text{res}}$	$\omega \approx \omega_2 \mathbf{b}_2$	$ v \ \mathbf{r}_{G/B}\ \ \omega - \omega_2 \mathbf{b}_2\ $	2.8
v	$v_{\text{cmd}} \approx v_{\text{actual}}$	$ v_{\text{cmd}} - v_{\text{actual}} $	0.15

TABLE II: Empirical Effects of Simplifying Assumptions on Constraints

VII. APPENDIX

A. Explicit Forms of Gaussians for Control Sampling

1) *Conventional Samples*: The conventional samples are drawn from a Gaussian distribution centered at the nominal control sequences,

$$\tilde{\mathbf{u}}_{t:t+H-1}^{\text{conv}} \sim \mathcal{N}(\bar{\mathbf{u}}_{t:t+H-1}, I_H \otimes \Sigma), \Sigma = \text{diag}([\sigma_1, \sigma_2]).$$

2) *Narrow Samples*: We find it to be beneficial to sample more control sequences that are close to the nominal control sequence. Therefore, we sample a number of control sequences with a smaller covariance,

$$\tilde{\mathbf{u}}_{t:t+H-1}^{\text{narrow}} \sim \mathcal{N}(\bar{\mathbf{u}}_{t:t+H-1}, I_H \otimes (s_n \cdot \Sigma)),$$

with $s_n \in (0, 1)$ being the narrow scaling coefficient.

3) *Scaled Samples*: We additionally sample around a scaled nominal control sequence which has smaller linear velocities. This allows the vehicle to effectively slow down while following a similar path as the nominal trajectory. Let $\bar{\mathbf{u}}_{t+h} = (\bar{v}_{t+h}, \bar{\kappa}_{t+h}^d)$ for any $h \in [0, H-1]$. We define the scaled nominal control sequence as

$$\bar{\mathbf{u}}_{t:t+H-1}^{\text{scaled}} := ((s_s \bar{v}_t, \bar{\kappa}_t^d), \dots, (s_s \bar{v}_{t+H-1}, \bar{\kappa}_{t+H-1}^d)),$$

with $s_s \in (0, 1)$ being scaling coefficients. The scaled samples are drawn from a Gaussian distribution centered at the scaled nominal control sequence,

$$\tilde{\mathbf{u}}_{t:t+H-1}^{\text{scaled}} \sim \mathcal{N}(\bar{\mathbf{u}}_{t:t+H-1}^{\text{scaled}}, I_H \otimes \Sigma).$$

4) *Reset Samples*: Finally, we include a small number of raw control sequences sampled from fixed-mean Gaussian distributions. These samples can reset the optimization if the nominal control sequence becomes high-cost due to, e.g., newly detected obstacles and terrain features.

$$\tilde{\mathbf{u}}_{t:t+H-1}^{\text{reset}, k} \sim \mathcal{N}(\mathbb{1}_H \otimes \bar{\mathbf{u}}_{\text{reset}}^k, I_H \otimes \Sigma).$$

In particular, we consider three fixed mean controls, $\mathbf{u}_{\text{reset}}^1 = (0, 0)$, $\mathbf{u}_{\text{reset}}^2 = (0, -\kappa_{\max}^d)$ and $\mathbf{u}_{\text{reset}}^3 = (0, \kappa_{\max}^d)$, where κ_{\max}^d is the maximum desired curvature. These fixed mean controls all use zero linear velocity but have different desired curvatures.

Parameter	Value
Temperature	1.0
No. of Samples	10,000
Horizon (s)	5.0
σ_1	4.0
σ_2	2.0
τ_{\min}^{res}	-10.0
τ_{\max}^{res}	-3.0
P_2	0.9
P_3	1.3
B_1	1.8
B_3	1.3
ϕ_{\max}	20°
θ_{\max}	30°

TABLE III: Table of relevant parameter settings. The temperature, number of samples, and horizon are with respect to the MPPI algorithm as laid out in [48].

B. System Parameters and Other Costs

Here, we include our settings for the primary design parameters of MPPI and a brief explanation of the other primary costs used in the system.

1) *Goal Costs*: In full course runs, the ‘goal’ point for MPPI is dynamically set by a higher level planner as the true goal points can be many kilometers apart. We have often found it useful to reconsider the mathematical measure of distance between two points. In short, we employ a time-to-go metric using the distance of the dubins path between points and our predicted speed.

$$c_{\text{goal}}(\mathbf{x}_t, \mathbf{u}_t) := w_{\text{goal}} \cdot d_{\text{dubins}}(\mathbf{x}_t, \mathbf{x}_{\text{goal}})/v_t$$

2) *Coplanarity Costs*: The fidelity of our elevation maps result in high uncertainty for areas with high frequency features (loose rock beds, sharp discontinuities, etc.). In addition to helping to ensure our analysis in section III remains valid, We encourage the vehicle to navigate toward low frequency regions by applying cost if the predicted wheel locations are non-coplanar. Let $\mathbf{x}_{\text{fl}}, \mathbf{x}_{\text{fr}}, \mathbf{x}_{\text{bl}}, \mathbf{x}_{\text{br}}$ be the four wheel locations and $A = (\mathbf{x}_{\text{fl}} \ \mathbf{x}_{\text{fr}} \ \mathbf{x}_{\text{bl}} \ \mathbf{x}_{\text{br}})^T$. Let $\mathbf{h} = (h(\mathbf{x}_{\text{fl}}) \ h(\mathbf{x}_{\text{fr}}) \ h(\mathbf{x}_{\text{bl}}) \ h(\mathbf{x}_{\text{br}}))^T$ where $h(\mathbf{x})$ is the elevation at \mathbf{x} . Then, the coplanarity cost is

$$c_{\text{coplanar}}(\mathbf{x}_t, \mathbf{u}_t) := w_{\text{coplanar}} \cdot \|\mathbf{h}_t - A_t A_t^\dagger \mathbf{h}_t\|$$

3) *Costmap Costs*: Semantically classified objects are assigned hand-tuned weights that are simply added to the final cost of the trajectory if any part of the vehicle footprint⁴ occupies the same cell. Finally, the cost is scaled with velocity.

$$c_{\text{costmap}}(\mathbf{x}_t, \mathbf{u}_t) := w_{\text{costmap}} v_t \cdot \sum_{\mathbf{x} \in \text{footprint}(\mathbf{x}_t)} \text{costmap}(\mathbf{x})$$

⁴ $\text{footprint}(\mathbf{x}_t)$ is the set of all $(x_t \ y_t)$ positions at time t that corresponds to some point on the vehicle (e.g. the four wheels, the front bumper, etc.)

C. Baseline Implementation

The geometry-based baseline is strictly a function of the spatial $(x_t \ y_t)$ path of the vehicle and the elevation map. Let $\bar{c}(\cdot)$ indicate cost functions used for the baseline.

1) *Rollover*: Rollover costs using pure geometrical methods treat the vehicle statically and use the critical rollover angle as the costing threshold. However, in practice, we want to constrain the vehicle far below this threshold, and so it is more practical to simply define a hand-tuned angle limit. Let $\beta(t) := |\theta(t)| - \theta_{\max}$ and $\gamma(t) := |\psi(t)| - \psi_{\max}$. We apply the cumulative sum of violations,

$$\begin{aligned}\bar{c}_{\text{pitch}}(\mathbf{x}_{t+h}, \mathbf{u}_{t+h}) &:= \sum_{k=t}^{t+h} \beta(k) \cdot \mathbf{1}\{\beta(k) > 0\} \\ \bar{c}_{\text{yaw}}(\mathbf{x}_{t+h}, \mathbf{u}_{t+h}) &:= \sum_{k=t}^{t+h} \gamma(k) \cdot \mathbf{1}\{\gamma(k) > 0\}\end{aligned}$$

2) *Ditches*: This method inspects the geometry in the rollout for deviations from the current vehicle pitch. If a ditch is detected with some certainty, the max vehicle speed will be set to a pre-defined value. In the development of this method, many other approaches were implemented such as using second derivative values of pitch and detecting sudden low elevations. The following implementation performed the best. We use the pitch value at t to predict the expected elevation of the next time $t+1$ and compare this prediction with the actual elevation as provided by the elevation map $h(\mathbf{x}_{t+1})$. Namely,

$$\begin{aligned}ds &:= \sqrt{(x_{t+1} - x_t)^2 + (y_{t+1} - y_t)^2} \\ \delta h_{t+1}(\mathbf{x}_{t+1}) &:= \tan(\theta_t) - [h(\mathbf{x}_{t+1}) - h(\mathbf{x}_{t+1})] / ds\end{aligned}$$

Define the “ditch value” at t to be the max of δh_t over the footprint of the vehicle and “masked” by states with downward pitch. This mask reduces false-positives associated with uphills. Explicitly,

$$V_{\text{ditch}}(\mathbf{x}_t) := \max_{\mathbf{x} \in \text{footprint}(\mathbf{x}_t)} \delta h_t(\mathbf{x}) \cdot \mathbf{1}\{\theta_t > 0\}$$

Finally, if $V_{\text{ditch}}(\mathbf{x}_t) > V_{\max}$ anywhere in the nominal (optimized) rollout, the max sampled vehicle speed (see section IV-B6) is set to v_{ditch} at time t . We found that this sampling constraint performed better in practice for this method over costing, due in part to the false positive frequency.

D. Constraint Model Error Analysis

In our constraint model, some simplifying assumptions are made to obtain bounds that can be computed in-place without storing excessively large tensors. We summarize the influence of our assumptions by bounding their errors on (6) and (9).

Recall the unsimplified residual torque about P is given by:

$$\bar{\tau}_P^{\text{res}} = \bar{\mathbb{I}}_P \cdot \boldsymbol{\alpha} - \mathbf{r}_{G/P} \times (\mathbf{g} - \dot{v}\mathbf{b}_1 - v\boldsymbol{\omega} \times \mathbf{b}_1) \quad (15)$$

where we have divided by the mass of the vehicle as in (6) and (III-C).

1) *Rollover*: Recall we take $P \in \{L, R\}$. Our first assumption is negligible rolling acceleration, i.e. $\boldsymbol{\alpha} \cdot \mathbf{b}_1 \approx 0$. The error we expect from this assumption is clear from inspecting the first term of (15),

$$\bar{\mathbb{I}}_P \cdot \boldsymbol{\alpha} \cdot \mathbf{b}_1 = \bar{I}_{11}\alpha_1 \quad (R1)$$

where we have used a diagonal moment of inertia for simplicity. Next, we assume that the angular velocity is primarily in \mathbf{b}_3 , i.e. $\boldsymbol{\omega} \approx \omega_3 \mathbf{b}_3$. To see the effect of this assumption, we focus our attention to just the affected triple product term in (15),

$$\begin{aligned}\mathbf{r}_{G/P} \times (\boldsymbol{\omega} \times \mathbf{b}_1) &= \mathbf{r}_{G/P} \times [(\boldsymbol{\omega} - \omega_3 \mathbf{b}_3 + \omega_3 \mathbf{b}_3) \times \mathbf{b}_1] \\ &= \mathbf{r}_{G/P} \times (\omega_3 \mathbf{b}_3 \times \mathbf{b}_1) \\ &\quad + \mathbf{r}_{G/P} \times [(\boldsymbol{\omega} - \omega_3 \mathbf{b}_3) \times \mathbf{b}_1] \\ &\leq \underbrace{\mathbf{r}_{G/P} \times (\omega_3 \mathbf{b}_3 \times \mathbf{b}_1)}_{\text{Constrained by } RR_{\max}} \\ &\quad + \underbrace{\|\mathbf{r}_{G/P}\| \|\boldsymbol{\omega} - \omega_3 \mathbf{b}_3\|}_{\text{Error due to Simplifying Assumption}} \quad (16)\end{aligned}$$

where the first triple-product term in (16) is part of the quantity constrained in section III-B. The latter term of (16) gives the theoretical maximum error due to our assumption when multiplied with the remaining term v in (15),

$$|v| \|\mathbf{r}_{G/P}\| \|\boldsymbol{\omega} - \omega_3 \mathbf{b}_3\| \quad (R2)$$

Finally, we also assume that the bicycle model is accurate in estimating the angular velocity, i.e. $\omega_3 \approx \kappa v$. Beginning from (5),

$$\begin{aligned}v(\mathbf{r}_{G/P} \cdot \boldsymbol{\omega}) &\approx v(\mathbf{r}_{G/P} \cdot \omega_3 \mathbf{b}_3) \\ &= v[\mathbf{r}_{G/P} \cdot (v\kappa + \omega_3 - v\kappa)\mathbf{b}_3] \\ &= P_3 v^2 \kappa + P_3 v(\omega_3 - v\kappa) \quad (17)\end{aligned}$$

where again the first term in (17) is used in the original constraint (6) and the second term is the error term,

$$|P_3 v| |\omega_3 - v\kappa| \quad (R3)$$

2) *Ditches*: The assumption used in the derivation for our ditch constraints is $\boldsymbol{\omega} \approx \omega_2 \mathbf{b}_2$. Thus, the bound is equivalent to (16) except with $P := B$,

$$|v| \|\mathbf{r}_{G/B}\| \|\boldsymbol{\omega} - \omega_2 \mathbf{b}_2\| \quad (D1)$$

3) *Velocity Tracking*: The tracking assumption due to employing the low-level controller is simply

$$|v_{\text{actual}} - v_{\text{cmd}}| \quad (V1)$$

4) *Results*: We plot the relevant states obtained using the IMU and visual-inertial odometry during our experiments in Fig. 8 and summarize their largest empirical values in Table II.

E. Project Website

Footage of experiments can be found on our project website here: <https://sites.google.com/cs.washington.edu/off-road-mpc>.

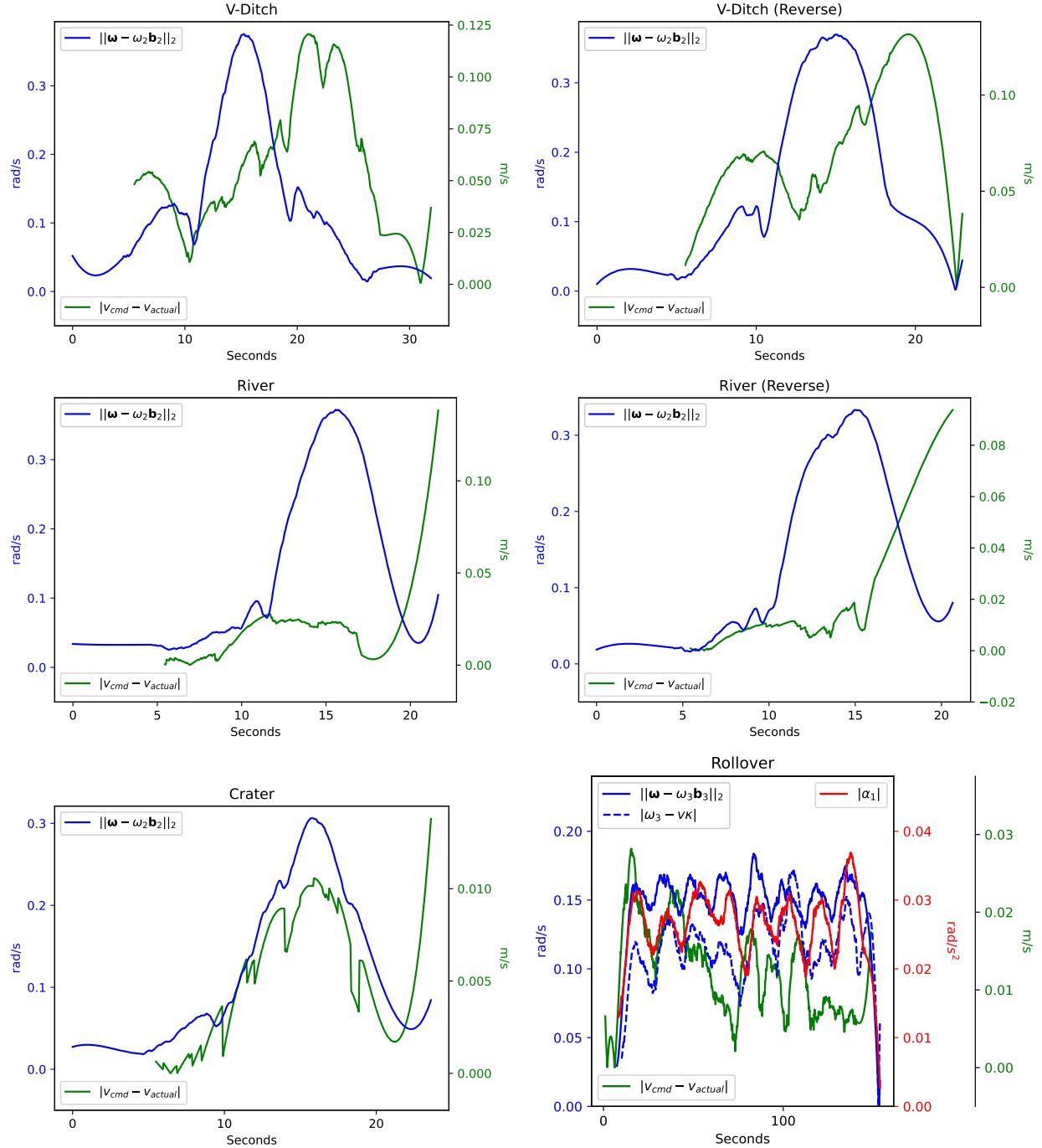


Fig. 8: Plots of empirical constraint error terms for V-Ditch, River, Crater, and rollover experiments. See Table II for these errors' effects on the constraint values.