

# Real-Time Trail-Braking Maneuver Generation for Off-Road Vehicle Racing

Changxi You Panagiotis Tsiotras

**Abstract**— Trail braking is a high-speed cornering technique that can, typically, be performed only by an expert driver. In this paper, we first learn a primitive high-speed cornering maneuver using a series of demonstrations obtained by solving the minimum-time cornering problem subject to different initial conditions. This primitive trajectory indicates that a typical trail-braking maneuver can be approximated by three segments, namely, entry corner guiding, steady-state sliding, and straight-line exiting. Based on this result, we divide a trail-braking maneuver into three stages. The middle sliding stage includes a segment of steady-state cornering that can be utilized to generate such trajectories for a variety of corner geometries. A flatness-based tracking controller is designed to generate the entry corner trajectory, and a feedback control stabilizes the vehicle at the exit.

**Keywords:** Differential flatness, trajectory learning, trajectory planning, trail-braking.

## I. INTRODUCTION

Trail-braking is a technique often used in rally racing, during which the vehicle is driven at high sideslip angles during cornering to shave off excess speed. In this paper we investigate real-time generation of a trail-braking maneuver.

Velenis in [1]–[3] modeled the trail-braking maneuver and showed that trail-braking can be generated as the solution of a minimum-time cornering problem subject to appropriate boundary conditions. Tavernini [4] investigated minimum-time cornering strategies for a vehicle with different transmission layouts (front-wheel-drive, rear-wheel-drive and all-wheel-drive) using different road surfaces, and showed that the minimum-time driving strategy under low-friction conditions turned out to be an aggressive high-drift cornering maneuver. Hindiyeh [5] analyzed the stability of the vehicle under high sideslip drifting conditions and revealed the existence of unstable equilibria corresponding to a steady-state cornering maneuver. The unstable equilibria during steady-state cornering were also shown by Yi [6], who proposed a hybrid tire/road model and analyzed the effect of the longitudinal slip on the lateral stability.

The above papers indicate that trail-braking may be approximately modeled using a steady-state cornering process. Nevertheless, most prior work is based on optimal trajectory optimization techniques and limited work exists

on trajectory planning and motion control to generate trail-braking maneuver in real time.

To better understand trail-braking, this paper first generates a series of demonstrations of trail-braking trajectories by solving the minimum-time cornering problem subject to several different initial conditions [2]. Based on these demonstrations, we learn a primitive trajectory using an iterative expectation-maximization (EM) algorithm, by utilizing an unscented Kalman filter (UKF) along with dynamic time warping (DTW) to align the time indexing of all the demonstrations. The result is a primitive trajectory, which can be used as the prototype trajectory to follow during trail-braking. This primitive trail-braking maneuver indicates the existence of a segment of sustained steady-state cornering. This observation leads to a decomposition of trail-braking into three stages, namely, guiding, sliding and exiting. Subsequently, we design a hybrid-mode control strategy for the three stages separately, using a combination of linear and nonlinear control techniques.

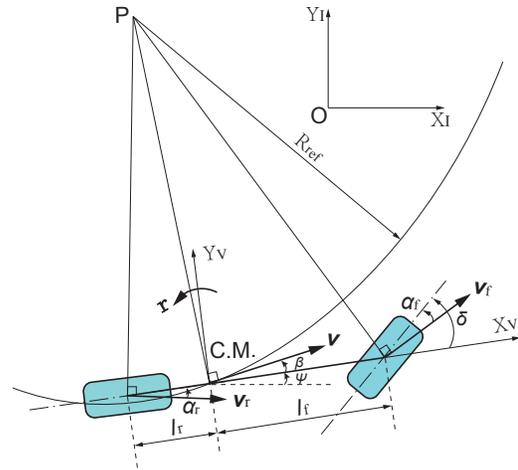


Fig. 1: Vehicle model.

## II. VEHICLE MODELING

The single-track vehicle model for control design and trajectory optimization used in this work is shown in Figure 1. In this figure,  $X_I - O - Y_I$  and  $X_B - CG - Y_B$  denote the inertial frame and the body frame (fixed on the vehicle), respectively. Furthermore,  $V_f$ ,  $V_r$  and  $V$  denote the velocities at the front and rear wheels and the vehicle's center of gravity (CG), and  $\alpha_f$ ,  $\alpha_r$  and  $\beta$  denote the sideslip angles of the front and rear wheels and the CG, respectively. The parameters  $l_f$  and  $l_r$  denote the distances of the CG to the front and rear axles,  $f_{ij}$  ( $i = F, R$

C. You is a PhD candidate at the School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0150, USA. Email: cyou6@gatech.edu

P. Tsiotras is a Professor at the School of Aerospace Engineering and the Institute for Robotics & Intelligent Machines, Georgia Institute of Technology, Atlanta, GA 30332-0150, USA. Email: tsiotras@gatech.edu

and  $j = x, y$ ) denote the longitudinal and lateral friction forces at the front and rear wheels, and  $\psi$  and  $r$  denote the yaw angle and the yaw rate of the vehicle, respectively. Finally,  $\delta$  is the steering angle of the front wheel.

The equations of motion of the model can be expressed in a body-fixed frame with the origin at CG as follows [7]

$$\dot{V} = \frac{1}{m} (f_{Fy} \sin \varphi + f_{Fx} \cos \varphi + f_{Ry} \sin \beta + f_{Rx} \cos \beta), \quad (1a)$$

$$\dot{\beta} = -r + \frac{1}{mV} (f_{Fy} \cos \varphi - f_{Fx} \sin \varphi + f_{Ry} \cos \beta - f_{Rx} \sin \beta), \quad (1b)$$

$$\dot{r} = \frac{1}{I_z} ((f_{Fy} \cos \delta + f_{Fx} \sin \delta) \ell_f - f_{Ry} \ell_r), \quad (1c)$$

where  $\varphi = \beta - \delta$ , where  $m$  is the total vehicle mass,  $I_z$  is the moment of inertia of the vehicle about the vertical axis through its CG, and the control is chosen as  $u = [\delta, f_{Fx}, f_{Rx}]^T$ . The lateral tire forces  $f_{Fy}$  and  $f_{Ry}$  are calculated by

$$f_{iy} = D_i \sin(C_i \text{atan}(B_i \alpha_i)), \quad i = F, R, \quad (2)$$

where  $D_i, C_i$  and  $B_i$  are constants, and the tire sideslip angles are given by

$$\alpha_F = \delta - \text{atan}\left(\frac{V \sin \beta + \ell_f r}{V \cos \beta}\right), \quad \alpha_R = -\text{atan}\left(\frac{V \sin \beta - \ell_r r}{V \cos \beta}\right).$$

### III. OPTIMAL TRAJECTORY GENERATION

We consider a 90deg cornering with the road geometry as shown in Figure 2. In this figure  $S_1$  and  $S_2$  denote the lengths of the two straight road segments before and after the corner, and  $R_{\text{ref}}$  and  $O$  denote the radius and the center of the centerline of the corner, respectively. The vehicle enters from Point A and exits from Point D with certain initial and final velocities.

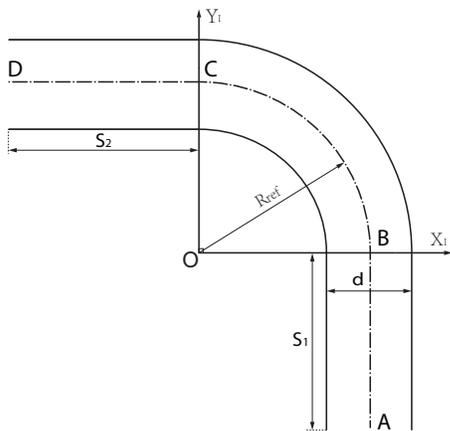


Fig. 2: Road geometry.

Based on the results of [3] we generate several optimal minimum-time cornering trajectories by minimizing the time  $t_f$  for the vehicle to drive from location A to location D, subject to the constraints

$$-d/2 < \Delta s < d/2 \quad \forall t \in [t_0, t_f], \quad (3)$$

and for several initial conditions. For instance, by assigning the road geometry parameters  $S_1 = S_2 = 5$  [m],  $R_{\text{ref}} = 10$  [m],  $d = 2$  [m] (see Figure 2), and the fifteen different initial positions and velocities shown in Table I, the previous minimization problem can be solved numerically. The results, for all fifteen trajectories are shown in Figure 3.

TABLE I: Initial conditions.

Initial Position A [(m,m)]	(9.5, -5), (10, -5), (10.5, -5)
Initial Velocity $V_0$ [m/s]	6, 7, 8, 9, 10

### IV. TRAJECTORY LEARNING

In this section we present an algorithm to learn a primitive trail-braking trajectory from the demonstrations shown in Figure 3. The algorithm is based on the approach initially proposed in [8], which assumes that each demonstration is an independent, noisy observation of some (unknown) primitive trajectory, along with a possible time reparameterization.

#### A. Generative Model

Suppose that we are given  $M$  representative demonstrations of length  $N_k$  for  $k = 0, 1, \dots, M-1$ . Each trajectory is assumed to be a discrete sequence of states  $x_j^k$  and controls  $u_j^k$ , which are composed into the augmented state vector  $y_j^k = [x_j^k, u_j^k]$ ,  $j = 0, 1, \dots, N_k - 1$ ,  $k = 0, 1, \dots, M-1$ . We then define the "hidden" target trajectory  $z^*$  of length  $T$ , which is denoted by  $z_t^* = [x_t^*, u_t^*]$ ,  $t = 0, 1, \dots, T-1$ . The hidden trajectory  $z_t^*$  must satisfy the system dynamics in (1). After discretization, and assuming some external noise, yields

$$z_{t+1}^* = f^*(z_t^*) + w_t^{(z)}, \quad (4a)$$

$$\eta_{t+1} = \eta_t + w_t^{(\eta)}, \quad (4b)$$

where  $w_t^{(z)} \sim \mathcal{N}(0, \Sigma^{(z)})$  and  $w_t^{(\eta)} \sim \mathcal{N}(0, \Sigma^{(\eta)})$  are Gaussian process noises. The value of  $\Sigma^{(\eta)}$  determines the smoothness of  $u^*$  for the hidden trajectory.

The demonstrations are independently observed from (4). The observations are therefore given by

$$y_j^k = z_{\tau_j^k}^* + w_j^{(y)}, \quad (5)$$

where  $w_j^{(y)} \sim \mathcal{N}(0, \Sigma^{(y)})$  is Gaussian observation noise. Here  $\tau_j^k$  is the time index in the hidden trajectory to which the observation  $y_j^k$  is mapped. Since  $\tau_j^k$  are not observed, we assume the following distribution with parameters  $d_i^k$

$$\mathbb{P}(\tau_{j+1}^k | \tau_j^k) = \begin{cases} d_1^k & \text{if } \tau_{j+1}^k - \tau_j^k = 1, \\ d_2^k & \text{if } \tau_{j+1}^k - \tau_j^k = 2, \\ d_3^k & \text{if } \tau_{j+1}^k - \tau_j^k = 3, \\ 0 & \text{otherwise,} \end{cases} \quad (6)$$

with  $\tau_0^k = 0$ , where  $\sum_{i=1}^3 d_i^k = 1$  and  $d_i^k \geq 0$ . The most likely hidden trajectory is obtained by solving the following

maximization problem

$$\max_{\tau, d, \Sigma^{(\cdot)}} \log \mathbb{P}(y; \tau, d, \Sigma^{(\cdot)}), \quad (7)$$

where  $\mathbb{P}$  is the joint likelihood of the observed trajectories  $y$  for the learned parameters  $\tau, d, \Sigma^{(\cdot)}$ .

### B. Primitive Trail-Braking Trajectory

After implementing the previous Trajectory Learning Algorithm we obtain the result shown in Figure 3.

The primitive trajectory in Figure 3 contains a segment of a circle that is tangent to the inner road boundary, and the vehicle seems to keep a constant side-slip angle  $\beta$  during the cornering. This observation leads to the conjecture that the primitive trajectory includes a segment of steady-state cornering. Steady-state cornering is characterized by a constant radius  $R^{ss}$ , negotiated at a constant speed  $V^{ss}$ , constant yaw rate  $r^{ss} = \frac{V^{ss}}{R^{ss}}$ , and constant side-slip angle  $\beta^{ss}$  [7]. Based on these results, we divide the trail-braking trajectory into three segments. The first segment allows the vehicle to complete the transition from straight-line driving to steady-state cornering. Before the vehicle enters the corner, the vehicle gradually changes its velocity using certain steering and accelerating/braking operations, until it reaches the target steady state near the entry of the corner. The vehicle then maintains steady-state cornering during the second segment, until it gets (close) to the exit of the corner. Finally, in the last segment, the vehicle leaves steady-state cornering and is steered back to straight-line driving after the vehicle exits the corner.

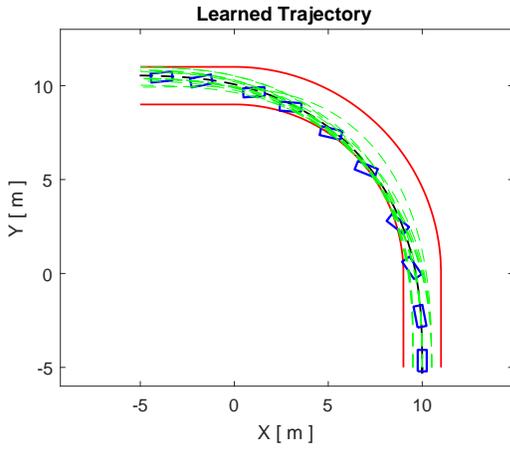


Fig. 3: Multiple demonstrations and the learned primitive trajectory.

We generate a high-speed, high-sideslip trail-braking maneuver using this idea. To this end, we first need to calculate the equilibrium for steady-state cornering. The main problem occurs after we obtain the target steady state since one has to guide the vehicle from a certain initial condition to the target steady state when the vehicle enters the steady state corner. This task is challenging, since we need to find a feasible trajectory, such that the vehicle reaches the target steady-state exactly when

it enters the corner. In order to solve this problem, in this paper we take advantage of the differential flatness of the vehicle dynamics to design the corresponding tracking control.

## V. DIFFERENTIALLY FLATNESS TRAJECTORY GENERATION

It has been previously shown [9]–[12] that the single-track vehicle model is differentially flat [13], [14]. Here we take advantage of the differential flatness property to design the trajectory to be followed, but we take into account the road condition and the steering capacity of the vehicle.

### A. Differential Flatness of Vehicle Model

In this section we show that the equations (1) of the vehicle model are differentially flat with respect to a particular output. This property is stated in Theorem 5.1.

*Theorem 5.1:* The vehicle model in (1) is differentially flat with respect to the following output [10]

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} V \cos \beta \\ V \sin \beta - \left( I_z / m \ell_f \right) r \end{bmatrix}. \quad (8)$$

*Proof:* Recall from (2) that the lateral tire forces  $f_{iy}$  ( $i = F, R$ ) are smooth functions of the sideslip angles  $\alpha_i$  at the front and rear wheels, respectively. We can then derive the state of the system in (1) in terms of the flat output in (8) as follows

$$V = \sqrt{y_1^2 + \left( y_2 + \frac{I_z r}{m \ell_f} \right)^2}, \quad \beta = \text{atan} \left( \frac{m \ell_f y_2 + I_z r}{m \ell_f y_1} \right), \quad (9)$$

where  $r = r(y_1, y_2, \dot{y}_2)$  is given by solving the following implicit equation,

$$\begin{aligned} & D_R \sin \left( C_R \text{atan} \left( B_R \text{atan} \left( \frac{(m \ell_f \ell_r - I_z) r - m \ell_f y_2}{m \ell_f y_1} \right) \right) \right) \\ & - \frac{m \ell_f (\dot{y}_2 + y_1 r)}{\ell_f + \ell_r} \triangleq g(y_1, y_2, \dot{y}_2, r) = 0. \end{aligned} \quad (10)$$

Equations (9)-(10) give the expressions of the state  $x$  in terms of the flat output in (8). The control  $u = [\delta, f_{Fx}, f_{Rx}]^T$  can be recovered from (8) by solving the following equations

$$\begin{aligned} & f_{Fy}(\delta, y_1, y_2, \dot{y}_2) \cos \delta \ell_f + f_{Fx} \sin \delta \ell_f - I_z \dot{r}(y_1, \dot{y}_1, y_2, \dot{y}_2, \dot{y}_2) \\ & - f_{Ry}(y_1, y_2, \dot{y}_2) \ell_r = 0, \\ & m \left( \dot{y}_1 - y_2 r(y_1, y_2, \dot{y}_2) - \frac{I_z}{m \ell_f} r^2(y_1, y_2, \dot{y}_2) \right) - f_{Rx} \\ & + f_{Fy}(\delta, y_1, y_2, \dot{y}_2) \sin \delta - f_{Fx} \cos \delta = 0, \\ & \Gamma(f_{Fx}, f_{Rx}) = 0, \end{aligned}$$

where the rear wheel lateral tire force is given by

$$f_{Ry}(y_1, y_2, \dot{y}_2) = \frac{\dot{y}_2 m \ell_f + y_1 r(y_1, y_2, \dot{y}_2) m \ell_f}{\ell_f + \ell_r}, \quad (11)$$

and where  $\Gamma(f_{Fx}, f_{Rx})$  is a force distribution function related to the specific drive type of the vehicle. The choice of  $\Gamma(f_{Fx}, f_{Rx})$  depends on the for drive type (All-Wheel-Drive, Front-Wheel-Drive, Rear-Wheel-Drive). Here we assume a RWD and hence we set  $f_{Fx} = 0$ . We can then determine

the remaining control variables  $\delta$  and  $f_{R_x}$  by solving the following equations

$$f_{F_y}(\delta, y_1, y_2, \dot{y}_2) \cos \delta \ell_f - I_z \dot{r}(y_1, \dot{y}_1, y_2, \dot{y}_2) \quad (12a)$$

$$- f_{R_y}(y_1, y_2, \dot{y}_2) \ell_r = 0, \quad (12b)$$

$$m \left( \dot{y}_1 - y_2 r(y_1, y_2, \dot{y}_2) - \frac{I_z}{m \ell_f} r^2(y_1, y_2, \dot{y}_2) \right) - f_{R_x} \quad (12c)$$

$$+ f_{F_y}(\delta, y_1, y_2, \dot{y}_2) \sin \delta = 0, \quad (12d)$$

and we have all controls as functions of the flat outputs and their derivatives. ■

## VI. TRAJECTORY PLANNING

As mentioned earlier, based on the trajectory learning result in Section IV, we assume that a trail-braking maneuver consists of three stages: 1) An entry stage before the vehicle enters the corner; 2) a sliding stage where the vehicle passes through the corner at a steady-state; and 3) an exiting stage after the vehicle leaves the corner. We consider the following scenario of high-speed, high-slip cornering maneuver, as shown in Figure 4.

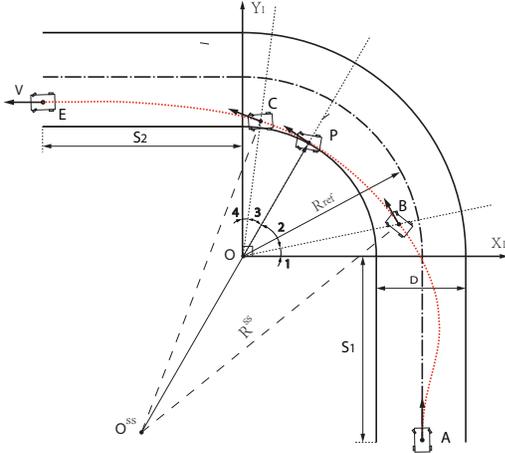


Fig. 4: Road geometry and trail-braking trajectory.

Figure 4 shows the feature of the “late apex”, where the vehicle exits the corner next to the inner edge of the road. As shown in this figure, the vehicle postpones the tangent point  $P$  to the second half of the corner and removes the inner/outer constraint lines. The steady-state cornering begins from  $B$ , passes through  $P$  and ends up at  $C$ . We introduce  $\angle 1, \dots, \angle 4$  to describe the position and length of the arc  $BPC$ . The angles  $\angle 1$  and  $\angle 4$  denote the vehicle’s late entering and early exiting positions of the steady-state cornering.

We compute the steady-state of the vehicle  $x^{ss} = [V^{ss}, \beta^{ss}, r^{ss}]^T$  having the desired cornering speed  $V^{ss}$  or the desired sideslip angle  $\beta^{ss}$  following [7].

### A. Guiding Trajectory

In order to determine a feasible trajectory of the vehicle from  $A$  to  $B$ , we claim that the trajectory must be designed to satisfy equation (10). To clarify the idea, let us assume that one has designed a trajectory  $x(t) = [V(t), \beta(t), r(t)]^T$ ,

$t \in [t_0, t_f]$ , and has calculated the flat outputs  $y_1(t)$  and  $y_2(t)$  using the trajectory  $x(t)$  following (8). However, it is not possible to recover  $x(t) = [V(t), \beta(t), r(t)]^T$  using the flat outputs  $y_1(t)$  and  $y_2(t)$ , unless  $x(t)$  is designed to satisfy equation (10).

We summarize this condition in Proposition 6.1.

*Proposition 6.1:* A trajectory defined by  $x(t) = [V(t), \beta(t), r(t)]^T$ ,  $t \in [t_0, t_f]$ , can be recovered from the output in (8) if and only if  $y_1(t)$ ,  $y_2(t)$  and  $r(t)$  satisfy equation (10).

*Proof:* The proof is straightforward and is omitted. ■

A feasible trajectory from  $A$  to  $B$  can then be designed as follows: First, a path is computed using purely geometric methods [15]. Specifically, let the coordinate  $X(t)$  be a function of  $Y(t)$  and suppose the path  $X(Y)$  is sufficiently smooth (i.e., a polynomial function). Next, the direction of the velocity is defined by the tangential of the path at each point. We assume that  $Y(t)$  is smooth and the velocity  $V(t)$  is determined by  $V(t) = \sqrt{1 + (\partial X / \partial Y)^2} \dot{Y}(t)$ . According to Proposition 6.1,  $\psi(t)$  must satisfy the boundary conditions  $\psi(t=0) = \psi_A, \psi(t=t_f) = \psi_B, \dot{\psi}(t=0) = r_A, \dot{\psi}(t=t_f) = r^{ss}$  and the nonlinear constraint  $g(\dot{\psi}(t)) = 0$  where  $g$  was defined in (10).

## VII. CONTROL DESIGN

A switching control is designed to achieve the tasks for the three different stages. In the guiding stage, a tracking controller is designed such that the vehicle is driven to reach the desired steady-state at the entry of the corner. Afterwards, a stabilizing controller is designed such that the vehicle follows the steady-state along the desired path. This stabilizing controller that keeps the vehicle along the arc  $BPC$  can be computed using standard methods (i.e., LQR theory). In the exiting stage, the vehicle leaves the corner and switches to a new control mode depending on the specific task. In this paper we design a state feedback controller that aligns the posture of the vehicle to be parallel with the road in the exiting stage. This exit controller aims at stabilizing the vehicle along the straight line. This controller can be designed using standard stabilization methods about the new straight-line equilibrium  $\hat{x}^{ss} = [\hat{V}^{ss}, \hat{\beta}^{ss}, \hat{r}^{ss}]^T = [V^{ss}, 0, 0]^T$ , where  $V^{ss}$  takes the value of the exit velocity of the vehicle after the corner. In order to align the vehicle to be parallel with the road, we include a target yaw angle  $\psi^{ss}$  into the state vector, that is,  $x^{ss} = [V^{ss}, 0, 0, \psi^{ss}]^T$ .

In this paper we therefore only focus on the control design for entry, guiding stage. That is, we consider only the design of the tracking controller that drives the vehicle from  $A$  to  $B$  following the trajectory designed in Section VI-A. To this end, let us denote the desired flat output as  $y^d(t)$  and denote the current output from the plant as  $y(t)$ . The tracking error is  $e(t) = y(t) - y^d(t)$  and the accumulated tracking error is  $\zeta(t) = \int_0^t e(\tau) d\tau + \zeta_0$ . The following design drives  $\zeta(t), e(t) \rightarrow 0$  as  $t \rightarrow \infty$  with proper

choices of  $\lambda$  and  $\nu$ ,

$$\dot{e}_1 = -\lambda_1 e_1 - \lambda_2 \zeta_1, \quad (13a)$$

$$\dot{e}_2 = -\nu_1 \dot{e}_2 - \nu_2 e_2 - \nu_3 \zeta_1. \quad (13b)$$

The values of  $\lambda$  and  $\nu$  can be determined so that the error dynamics is asymptotically stable. It follows from (13) that

$$\dot{y}_1 = \dot{y}_1^d - \lambda_1 e_1 - \lambda_2 \zeta_1, \quad (14a)$$

$$\dot{y}_2 = \dot{y}_2^d - \nu_1 \dot{e}_2 - \nu_2 e_2 - \nu_3 \zeta_1, \quad (14b)$$

and the control  $u$  is obtained by solving the following equations:

$$\mathcal{L}_f y_1(x, u) = \dot{y}_1, \quad \mathcal{L}_f^2 y_2(x, u) = \dot{y}_2, \quad (15)$$

The (local) solvability of equations in (15) is guaranteed following the implicit function theorem if and only if the following conditions are satisfied

$$\frac{I_z V \sin \beta (\partial f_{Ry} / \partial V) + I_z \cos \beta (\partial f_{Ry} / \partial \beta) + m V \ell_f (\partial f_{Ry} / \partial r)}{m^2 \ell_f^2} - \frac{V^2 \cos \beta}{\ell_f + \ell_r} \neq 0, \quad (16)$$

$$f_{Fy} \sin \delta - \frac{\partial f_{Fy}}{\partial \delta} \cos \delta \neq 0. \quad (17)$$

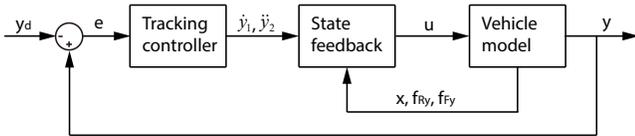


Fig. 5: Scheme of flatness based vehicle dynamics control.

One can refer to [10] for more analysis on the solvability of (15). The scheme of the controller design can be shown in Figure 5.

## VIII. RESULTS AND ANALYSIS

In this section we implement the proposed control architecture and analyze the results. Different trail-braking maneuvers are generated for different corner geometries.

### A. Trajectory and Controller Design

We first calculate the equilibrium for the steady-state cornering. Table II shows the equilibria for steady state-cornering with different speeds.

TABLE II: Equilibrium for steady-state cornering.

	1	2	3	4	5	6	7	8
$V^{ss}$ [m/s]	5	8.5	9.4	10.8	11.9	12.1	13.9	15
$\beta^{ss}$ [deg]	-2.8	-30	-20	-20	-28	-20	-23	-32
$r^{ss}$ [deg/s]	11.3	48.7	35.9	30.9	34.1	27.7	26.5	28
$R^{ss}$ [m]	25.4	10	15	20	20	25	30	30.7

A feasible trajectory that includes the target steady-state cornering process is designed following the approach of Section VI. In order to demonstrate the proposed trajectory planning and controller design, we take the

first equilibrium in Table II and use the segmentation in Figure 4, for instance. The radius  $R_{ref}$  of the corner is 10 [m], and the straight line segments have lengths  $S_1 = 20$  [m] and  $S_2 = 15$  [m]. The distance  $d$  between the inner and outer constraints is 2 [m].

The plots in Figure 6 show that the desired trajectories and the simulated trajectories agree well with each other.

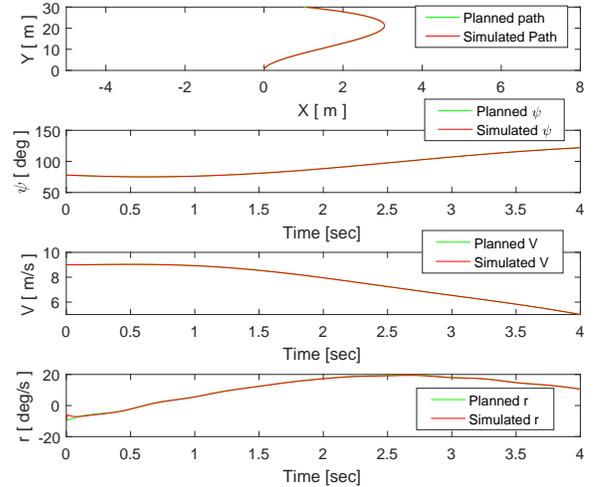


Fig. 6: The desired and simulated trajectories.

Since the system is differentially flat, the control  $u = [\delta, f_{Rx}]^T$  ( $f_{Fx} = 0$  and hence omitted) to achieve the desired trajectory can be recovered from the designed flat output by solving equations (12). The calculated control with the simulation result is shown in Figure 7.

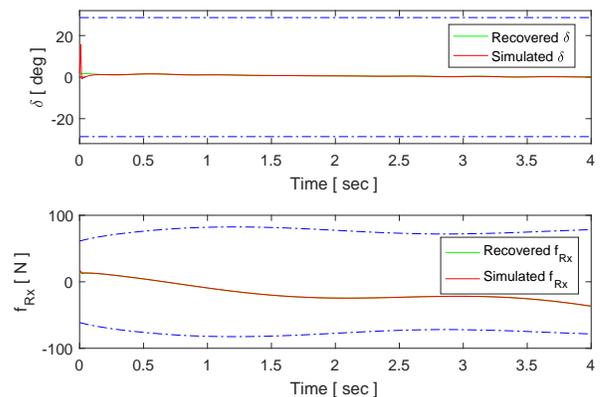


Fig. 7: The desired and simulated controls.

### B. Trail-Braking Trajectory Generation

In this section we generate trail-braking trajectories for a variety of corner geometries. We use the second and the third equilibria in Table II for steady-state cornering to achieve a high-speed, high-slip sliding maneuver.

The road geometry is defined using the parameters in Table III. It is worth mentioning that the other equilibria in Table II have steady-state cornering radius  $R^{ss} > 20$  [m], which seems too large and is not convenient to use for the road geometry in Table III, especially when the corner

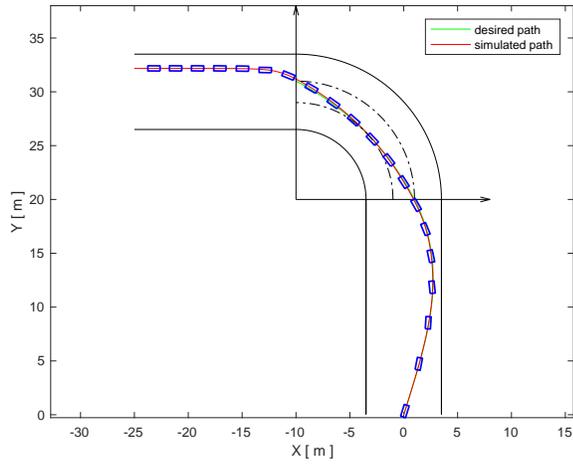


Fig. 8: Switching-mode control for steady-state cornering.

TABLE III: Road and trail-braking trajectory setup.

Road geometry	$S_1$ [m]	$S_2$ [m]	$R_{ref}$ [m]	$D$ [m]	Corner [deg]
Initial condition	15	15	10	5	60/90/120/180
	$V$ [m/s]	$\beta$ [deg]	$r$ [deg/s]	$(X, Y)$ [(m,m)]	$\psi$ [deg]
	15	0	0	$(R_{ref}, -S_1)$	90

angle is larger than 90 deg. For trail-braking with different corner angles, Table III provides appropriate choices of the equilibria and the angles  $\angle 1, \dots, \angle 4$  that determine the geometry of different trail-braking trajectories. We plan trajectories for different corner angles and implement the proposed switch-mode controller accordingly. The planned and simulated trajectories are shown in Figure 9.

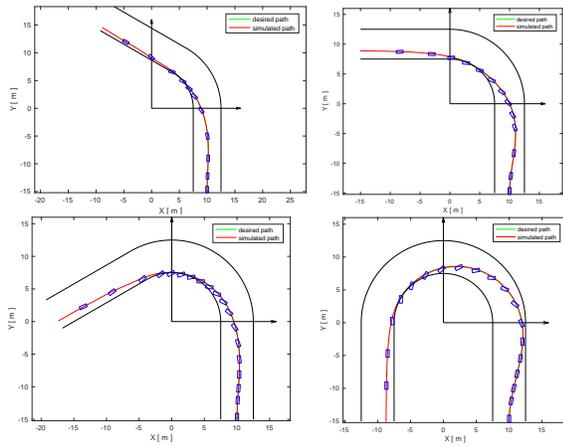


Fig. 9: Trail-braking maneuver generation for different road geometries.

These figures show the computed trail-braking trajectories for 60 deg, 90 deg, 120 deg and 180 deg cornering, respectively. In each figure, the simulated trajectory agrees well with the planned trajectory.

## IX. CONCLUSION

Trail-braking is a high-speed, high-slip cornering technique often used by expert rally racing drivers. This paper provides a methodology to generate trail-braking

maneuvers in semi-analytic form. We first use a trajectory learning technique to find a primitive trajectory that captures the “essence” of a trail-braking maneuver, using a series of trail-braking demonstrations. Based on this primitive trajectory, we segment the trail-braking trajectory into three stages, namely, the entry, sliding and exiting stages, and we show that the middle the sliding stage includes a segment of steady-state cornering. Based on the existence of steady-state equilibria and the fact that the vehicle dynamics are differentially flat, we then develop a switching-mode control for each stage using different control techniques to generate trail-braking.

## REFERENCES

- [1] E. Velenis, P. Tsiotras, and J. Lu, “Modeling aggressive maneuvers on loose surfaces: The cases of trail-braking and pendulum-turn,” in *European Control Conference*, Kos, Greece, July 2-5 2007, pp. 1233–1240.
- [2] —, “Trail-braking driver input parameterization for general corner geometry,” SAE Technical Paper, Tech. Rep., 2008.
- [3] —, “Optimality properties and driver input parameterization for trail-braking cornering,” *European Journal of Control*, vol. 14, no. 4, pp. 308–320, 2008.
- [4] D. Tavernini, M. Massaro, E. Velenis, D. I. Katzourakis, and R. Lot, “Minimum time cornering: the effect of road surface and car transmission layout,” *Vehicle System Dynamics*, vol. 51, no. 10, pp. 1533–1547, 2013.
- [5] R. Y. Hindiyeh and J. C. Gerdes, “Equilibrium analysis of drifting vehicles for control design,” in *ASME 2009 Dynamic Systems and Control Conference*. Hollywood, CA: American Society of Mechanical Engineers, October 12-14 2009, pp. 181–188.
- [6] J. Yi and E. H. Tseng, “Nonlinear stability analysis of vehicle lateral motion with a hybrid physical/dynamic tire/road friction model,” in *ASME Dynamic Systems and Control Conference*. Hollywood, CA: American Society of Mechanical Engineers, October 12-14 2009, pp. 509–516.
- [7] E. Velenis, E. Frazzoli, and P. Tsiotras, “Steady-state cornering equilibria and stabilisation for a vehicle during extreme operating conditions,” *International Journal of Vehicle Autonomous Systems*, vol. 8, no. 2-4, pp. 217–241, 2010.
- [8] A. Coates, P. Abbeel, and A. Y. Ng, “Learning for control from multiple demonstrations,” in *Proceedings of the 25th International Conference on Machine Learning*, Helsinki, Finland, July 5-9 2008, pp. 144–151.
- [9] Y. Wang, S. Shi, and L. Li, “Flatness-based vehicle coupled control for steering stability and path tracking,” in *Proceedings of SAE-China Congress 2015: Selected Papers*. Springer, 2016, pp. 49–60.
- [10] S. Fuchshumer, K. Schlacher, and T. Rittenschober, “Nonlinear vehicle dynamics control-a flatness based approach,” in *Proceedings of the 44th IEEE Conference on Decision and Control*, Seville, Spain, December 15 2005, pp. 6492–6497.
- [11] H. Souilem and N. Derbel, “Vehicle control by flatness,” *Proceedings Engineering & Technology-Vol.*, vol. 4, pp. 37–42, 2013.
- [12] J. Villagra, B. d’Andrea Novel, H. Mounier, and M. Pengov, “Flatness-based vehicle steering control strategy with sdr feedback gains tuned via a sensitivity approach,” *IEEE Transactions on Control Systems Technology*, vol. 15, no. 3, pp. 554–565, 2007.
- [13] M. Fliess, J. Lévine, P. Martin, and P. Rouchon, “Differential flatness and defect: an overview,” in *Workshop on Geometry in Nonlinear Control*, Banach Center Publications, Warsaw, 1993.
- [14] —, “Flatness and defect of non-linear systems: introductory theory and examples,” *International Journal of Control*, vol. 61, no. 6, pp. 1327–1361, 1995.
- [15] J.-w. Choi, R. Curry, and G. Elkaim, “Path planning based on bézier curve for autonomous ground vehicles,” in *Advances in Electrical and Electronics Engineering-IAENG Special Edition of the World Congress on Engineering and Computer Science*. San Francisco, CA: IEEE, October 22-24 2008, pp. 158–166.