

Path-Tracking Control for a Rear-Wheel-Drive Vehicle Governed by an Implicit ODE via State-Dependent Model Predictive Control

Javad Zolfaghari

gs64011@student.upm.edu.my

Universiti Putra Malaysia

A.S.M. Harithuddin

Universiti Putra Malaysia

A.A. Nuraini

Universiti Putra Malaysia

Raja Kamil

Universiti Putra Malaysia

Reza Jazar

Tarleton State University

Research Article

Keywords: Path-Tracking Control, State-Dependent MPC, Rear-Wheel-Drive Vehicle, Implicit ODE, Combined-Slip Tire Model

Posted Date: August 31st, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-10777655/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

Path-Tracking Control for a Rear-Wheel-Drive Vehicle Governed by an Implicit ODE via State-Dependent Model Predictive Control

Javad Zolfaghari^{*1}, A.S.M. Harithuddin¹, A.A. Nuraini², Raja Kamil³, and Reza Jazar⁴

¹Department of Aerospace Engineering, Universiti Putra Malaysia, Serdang, Selangor, Malaysia

²Department of Mechanical and Manufacturing Engineering, Universiti Putra Malaysia, Serdang, Selangor, Malaysia

³Department of Electrical and Electronic Engineering, Universiti Putra Malaysia, Serdang, Selangor, Malaysia

⁴Department of Mechanical, Environmental and Civil Engineering, Tarleton State University, Texas, USA

Abstract

Rear-wheel-drive vehicles operating near or beyond the tyre friction limit present two difficulties that this paper resolves together. The first is structural: because longitudinal load transfer feeds the vehicle's acceleration back into the tyre forces that produce it, the equations of motion are *implicit* – the state derivative appears on the right-hand side of its own governing equation. The second concerns the reference itself: the steady drifting condition the controller must track is not a fixed set-point but a *state-dependent* one, existing only on the vehicle's curvature-indexed equilibrium manifold. We address both issues at once with a State-Dependent Model Predictive Control (SDMPC) framework, cast as a single convex formulation that sits between linear predictive control and full nonlinear predictive control on the accuracy-versus-cost spectrum. Rather than linearising, we fold the implicit equation of motion of a single-track model with combined-slip tyre forces – together with actuator dynamics and Frenet-frame path-tracking kinematics – exactly into an augmented State-Dependent Coefficient representation, with the drifting equilibrium condition built directly into the dynamics. As a result, the receding-horizon prediction matrices carry no Taylor-series residual anywhere in the state space. We validate the controller in two stages. On a circular path, it regulates the augmented state onto a fixed drift equilibrium and recovers robustly across a severity ladder of off-equilibrium conditions. Reformulated as a direct output-tracking regulator, it then closes the loop on a self-intersecting figure-eight manoeuvre that demands a full sideslip-sign reversal through a zero-curvature transition where no drifting equilibrium exists at all. Together, the two studies show that a single exact prediction model can handle both equilibrium tracking and direct output tracking of geometries with no equilibrium, within one architecture for implicit, friction-limited vehicle dynamics.

Keywords: Path-Tracking Control, State-Dependent MPC, Rear-Wheel-Drive Vehicle, Implicit ODE, Combined-Slip Tire Model

^{*}Corresponding author. Email: gs64011@student.upm.edu.my

1 Introduction

When a rear-wheel-drive (RWD) vehicle is driven at or beyond the tyre friction limit, its behaviour departs from the familiar linear-handling regime and instead organises itself around a manifold of steady cornering and drifting equilibria, only some of which are stable [1, 2, 3]. Steering the vehicle along a prescribed geometric path under these conditions is therefore not a conventional trajectory-tracking problem: the controller must simultaneously regulate the vehicle onto a state that is itself a function of the path curvature, while contending with a plant whose equations of motion are *implicit*, since the normal loads that set the available tyre forces depend on the very longitudinal acceleration that those forces produce [4]. This coupling, often neglected in reduced two-state formulations that hold longitudinal velocity constant [5], becomes decisive precisely in the high-slip, high-acceleration regime where drifting and post-limit manoeuvres take place.

Much of the existing work on steady-state drifting and post-limit steering approaches the problem as one of stabilizing a single operating point: feedback linearization and phase-portrait-based framings regulate the vehicle onto a hand-picked equilibrium [6, 7, 18, 19], while State-Dependent Riccati Equation (SDRE) control has been used to hold a drift condition or track a path in limit-handling conditions [20]. These approaches show that a chosen drift equilibrium can be stabilized, but they generally freeze the vehicle model at, or close to, that single point, so their validity degrades away from it. A second line of work instead reschedules a locally linearized model along the predicted trajectory, as in linear time-varying MPC (LTV-MPC) [9, 8, 21]. Both the single-point and the rescheduled-linear families inherit a first-order Taylor residual that grows with distance from the linearization point and accumulates across the prediction horizon; for a vehicle sustaining large sideslip angles under combined-slip saturation, this residual can be large enough to destabilise the closed loop precisely when the manoeuvre is most demanding, for example during a yaw-rate sign reversal. Full nonlinear MPC (NMPC) removes the approximation altogether and has been demonstrated for autonomous drifting and combined drifting/yaw-moment control [22, 23, 24], and learning-based predictive schemes have pursued a related goal for aggressive racing manoeuvres [26]; however, NMPC requires solving a nonconvex programme online, at a computational cost typically one to two orders of magnitude larger than a comparable quadratic programme (QP), which is difficult to justify on embedded automotive hardware. What is needed, then, is a controller that reproduces the prediction accuracy of NMPC without giving up the convex, low-cost structure that makes LTV-MPC attractive – a formulation that trades off accuracy, robustness, and computational cost favourably rather than sacrificing one for the other.

State-Dependent Coefficient (SDC) methods offer exactly this compromise. By factorizing a smooth nonlinear vector field *exactly* as $\dot{\mathbf{z}} = \mathbf{A}(\mathbf{z})\mathbf{z} + \mathbf{B}(\mathbf{z})\mathbf{u}$, with $\mathbf{A}(\cdot)$ and $\mathbf{B}(\cdot)$ evaluated at, rather than linearized about, the current state, SDC representations underlie the State-Dependent Riccati Equation (SDRE) technique [10, 11] and, when combined with a receding-horizon QP, yield State-Dependent MPC (SDMPC): a controller that achieves the prediction accuracy of NMPC while retaining the convex, quadratic structure – and hence the computational tractability – of linear MPC [12, 13]. Beyond this accuracy-cost trade-off, the exactness of the SDC form carries a second, structural consequence that we exploit directly here: because $\mathbf{A}(\mathbf{z})\mathbf{z} + \mathbf{B}(\mathbf{z})\mathbf{u}$ reproduces the true right-hand side of the equations of motion at *every* state, it gives a natural and exact vehicle for resolving an *implicit* ODE – one in which the state derivative itself appears on the right-hand side – by augmenting the state vector with the derivative terms and absorbing the resulting algebraic coupling into the state-dependent coefficient matrices themselves, rather than through an approximate, locally-valid linearization of the implicit relation.

This paper builds on that idea and extends it, for the first time in the RWD drifting-control literature to our knowledge, from point stabilization of a single drifting equilibrium to genuine

spatial path-tracking under a moving, curvature-indexed reference, and further to direct output-tracking control of manoeuvres for which no such reference exists at all. This differs in kind, not merely in degree, from the NMPC-based drifting and friction-limit path-tracking controllers of Goh and co-workers [25, 24]: those formulations attain nonlinear prediction accuracy by solving a nonconvex programme online, whereas the SDMPAC controller developed here targets the same accuracy through an exact SDC factorisation evaluated inside a convex QP, so that NMPC-class path-tracking accuracy is pursued without inheriting its online nonconvex solve. Our contributions are fivefold. First, we adopt SDMPAC specifically as the controller occupying the favourable middle ground between low-cost, locally-valid linear/LTV-MPC schemes and high-accuracy but computationally prohibitive NMPC, since it recovers nonlinear prediction accuracy at close to linear computational cost; we quantify this trade-off explicitly against LTV-MPC and NMPC in terms of both prediction exactness and indicative solve time. Second, we recast the implicit equation of motion of a three-state RWD single-track model with Modified Elliptical Method (MEM) combined-slip tyre forces [14, 15], together with first-order actuator dynamics and Frenet-frame path-tracking kinematics, into a single ten-state SDC representation of the exact linear form $\dot{\mathbf{z}} = \mathbf{A}(\mathbf{z})\mathbf{z} + \mathbf{B}(\mathbf{z})\mathbf{u}$; this is our principal modelling contribution, since it resolves the implicit load-transfer coupling of the single-track RWD plant without any Taylor residual, at every point of the augmented state space. Third, because the augmented state itself carries the physical state derivatives as explicit components, the steady-state drifting condition $\dot{x}_i = 0$ that defines the equilibrium the controller must reach is embedded directly inside the SDC dynamics, rather than solved once offline and only weakly coupled to the online predictive law. Fourth, we validate the resulting successive-linearisation SDMPAC controller in two stages: first on a constant-curvature circular path, where it regulates the augmented state onto a fixed drift equilibrium and recovers, across a four-level severity ladder of off-equilibrium initial conditions, with a lateral error that grows gracefully rather than diverging as the initial deviation increases; and second on a highly dynamic, self-intersecting figure-eight manoeuvre requiring a full sideslip-sign reversal through a zero-curvature transition at which no steady drifting equilibrium exists. Fifth, for this second manoeuvre we generalise the moving-equilibrium formulation into a direct output-tracking regulator that prescribes the desired sideslip as an explicit, arc-length-scheduled target rather than looking it up from a single-valued equilibrium branch, extending the reach of the SDMPAC architecture to reference geometries for which the equilibrium-tracking formulation is not well posed.

The remainder of the paper is organised as follows. Section 2 presents the single-track vehicle model, the MEM combined-slip tyre formulation, and the resulting implicit equation of motion. Section 3 develops the actuator- and path-augmented SDC representation and the successive-linearisation SDMPAC controller built upon it, including its positioning relative to LTV-MPC and NMPC. Section 4 reports the closed-loop validation and robustness results on a circular reference path and demonstrates the output-tracking reformulation of the controller on the self-intersecting figure-eight manoeuvre. Section 5 concludes.

2 Single-Track Model of the Rear-Wheel-Drive Vehicle

2.1 Vehicle Equations of Motion

A single-track (bicycle) model of an RWD vehicle is considered (Figure 1). Two control inputs act on the vehicle: the front-wheel steering angle δ_f (positive anticlockwise) and the rear-wheel longitudinal slip ratio s_r . The front-wheel slip ratio s_f is not a controlled input: for an RWD vehicle, front-wheel slip arises only from rolling resistance and is negligible in normal operation, so $s_f = 0$ is adopted throughout, consistent with only the rear axle generating traction forces. The three state variables are the longitudinal velocity, lateral velocity, and yaw rate, expressed

in the vehicle-fixed frame,

$$x_1 \equiv v_x, \quad x_2 \equiv v_y, \quad x_3 \equiv r. \quad (1)$$

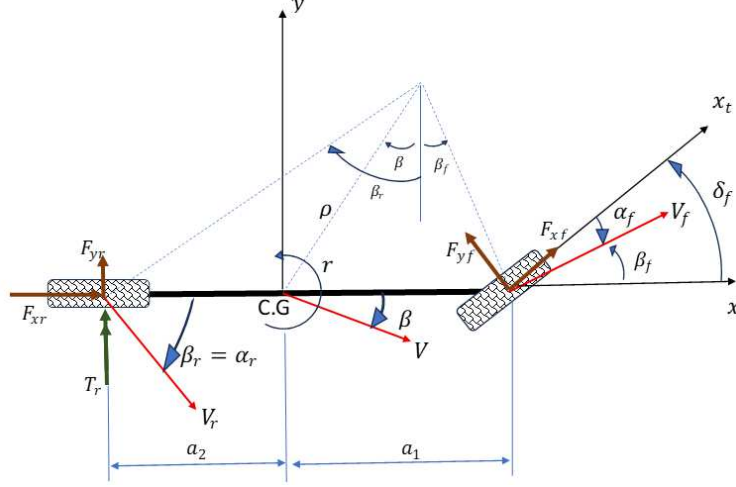


Figure 1: Single-track (bicycle) model of the rear-wheel-drive vehicle. The front axle is at distance a_1 from the centre of gravity and the rear axle at distance a_2 ; the steering angle δ_f and the yaw rate r are positive anticlockwise. The aerodynamic drag force opposes v_x .

The planar equations of motion, from Newton's second law in the rotating vehicle-fixed frame, are

$$\begin{aligned} \dot{x}_1 &= \frac{1}{m} (F_{xf} \cos \delta_f + F_{xr} - F_{yf} \sin \delta_f - C_A x_1^2) + x_2 x_3, \\ \dot{x}_2 &= \frac{1}{m} (F_{yf} \cos \delta_f + F_{yr} + F_{xf} \sin \delta_f) - x_1 x_3, \\ \dot{x}_3 &= \frac{1}{I_z} (a_1 F_{yf} \cos \delta_f + a_1 F_{xf} \sin \delta_f - a_2 F_{yr}), \end{aligned} \quad (2)$$

where C_A is the aerodynamic drag coefficient (lateral aerodynamic forces are neglected at the velocities considered), m is the vehicle mass, and I_z is the yaw moment of inertia; the terms $+x_2 x_3$ and $-x_1 x_3$ are the centripetal-acceleration components arising from the rotating frame. The normal loads on the front and rear axles, accounting for longitudinal-acceleration-induced load transfer, are

$$\begin{aligned} F_{zf} &= mg \frac{a_2}{a_1 + a_2} - m \frac{h}{a_1 + a_2} (\dot{x}_1 - x_2 x_3), \\ F_{zr} &= mg \frac{a_1}{a_1 + a_2} + m \frac{h}{a_1 + a_2} (\dot{x}_1 - x_2 x_3), \end{aligned} \quad (3)$$

with h the centre-of-gravity height and wheelbase $L = a_1 + a_2$. Because (3) couples the normal forces to $\dot{x}_1$, and $\dot{x}_1$ depends in turn on F_{zf} and F_{zr} through (2), the overall ODE system is *implicit* rather than explicit – a structural feature that is central to the SDC controller design of Section 3. The vehicle parameters used throughout are listed in Table 1.

Table 1: Vehicle parameters.

Parameter	m (kg)	I_z (kg m ²)	a_1 (m)	a_2 (m)	h (m)	C_A (N s ² m ⁻²)
Value	800	1000	1.35	1.5	0.9	0.2

2.2 Combined-Slip Tyre Forces via the Modified Elliptical Method

The tyre forces F_{xf} , F_{xr} , F_{yf} and F_{yr} are modelled using the Modified Elliptical Method (MEM), a closed-form, differentiable combined-slip formulation [14, 15]. The friction-circle coupling between longitudinal and lateral force is captured by multiplicative saturation factors, giving

$$\begin{aligned} F_{xf} &= \mathbf{MEM}_x(s_f, \alpha_f) F_{zf}, & F_{yf} &= \mathbf{MEM}_y(s_f, \alpha_f) F_{zf}, \\ F_{xr} &= \mathbf{MEM}_x(s_r, \alpha_r) F_{zr}, & F_{yr} &= \mathbf{MEM}_y(s_r, \alpha_r) F_{zr}, \end{aligned} \quad (4)$$

where s and α denote longitudinal slip ratio and slip angle, and

$$\mathbf{MEM}_x(s, \alpha) = \psi(s) \phi(\alpha), \quad \mathbf{MEM}_y(s, \alpha) = -\psi'(\alpha) \phi'(s). \quad (5)$$

Here $\psi(s)$ and $\psi'(\alpha)$ are the pure-longitudinal and pure-lateral force functions, and $\phi(\alpha)$, $\phi'(s)$ are the lateral and longitudinal saturation factors that encode the elliptical friction-circle constraint; unlike the Magic Formula, the MEM provides closed-form partial derivatives of the tyre force with respect to slip and slip angle, which is exploited directly in the SDC evaluation of Section 3. The effective longitudinal contact-patch velocities are

$$v_{xtf} = v_x \cos \delta_f + (v_y + a_1 r) \sin \delta_f, \quad v_{xtr} = v_x, \quad (6)$$

and the slip angles follow the standard bicycle-model kinematics,

$$\alpha_f = \arctan\left(\frac{v_y + a_1 r}{v_x}\right) - \delta_f, \quad \alpha_r = \arctan\left(\frac{v_y - a_2 r}{v_x}\right). \quad (7)$$

The pure-slip build-up functions and the saturation factors are

$$\psi(s) = \text{sgn}(s) \left[C_s s_s K(s, s_s, r_3, r_4) H(s, s_s) + (C_s s_s - b_1) J(s, s_s, r_1, r_2) (1 - H(s, s_s)) \right], \quad (8)$$

$$\psi'(\alpha) = \text{sgn}(\alpha) \left[C_\alpha \alpha_s K(\alpha, \alpha_s, r'_3, r'_4) H(\alpha, \alpha_s) + (C_\alpha \alpha_s - b_2) J(\alpha, \alpha_s, r'_1, r'_2) (1 - H(\alpha, \alpha_s)) \right], \quad (9)$$

$$\phi(\alpha) = \left[1 - D_{s\alpha} \left(\frac{S(\alpha - \alpha_s)}{\alpha_s} \right)^2 \right]^{1/n} J'(\alpha, \alpha_s, h_1, h_2), \quad (10)$$

$$\phi'(s) = \left[1 - D_{\alpha s} \left(\frac{S(s - s_s)}{s_s} \right)^2 \right]^{1/n'} J'(s, s_s, h'_1, h'_2), \quad (11)$$

with the auxiliary functions

$$\begin{aligned} J(p, q, u, v) &= \exp\left(-\left|\frac{p-q}{v}\right|^u\right), & K(p, q, u, v) &= 1 - \exp\left(-u \left(\frac{p-q}{q}\right)^v\right), \\ J'(p, q, u, v) &= \begin{cases} 1, & |p| \leq q, \\ J(p, q, u, v), & |p| > q, \end{cases} & S(p, q) &= \begin{cases} -q, & p < -q, \\ p, & -q \leq p \leq q, \\ q, & p > q, \end{cases} \end{aligned} \quad (12)$$

and $H(p, q) = 1$ for $|p| \leq q$, $H(p, q) = 0$ for $|p| > q$, selecting the pre- or post-saturation branch of the force curve. The MEM parameters, calibrated against Magic Formula reference data [14, 15, 16], are listed in Table 2; front and rear tyre parameters are identical in this study.

Table 2: MEM tyre parameters (identical front and rear).

C_s		C_α		s_s	α_s	b_1	b_2	n	n'		
8.6845		5.8555		0.12	0.16	0.7727	0.64	-1.95	-1.6		
r_1	r_2	r_3	r_4	r'_1	r'_2	r'_3	r'_4	h_1	h_2	h'_1	h'_2
0.8	0.7	2.1	1.15	1.25	0.25	2	1.1	0.8	0.42	0.86	0.35

2.3 Implicit Structure of the Equations of Motion

Combining (2)–(12), the equations of motion take the implicit form

$$\dot{x}_i = f_i(x_1, x_2, x_3, \dot{x}_1, s_r, \delta_f), \quad i = 1, 2, 3, \quad (13)$$

which is implicit because $\dot{x}_1$ appears on the right-hand side through the normal-load expressions (3): the state and control alone do not determine $\dot{x}$ without simultaneously solving for $\dot{x}_1$. This implicit coupling, retained throughout the controller design below rather than removed by an explicit surrogate, is the structural feature that motivates the State-Dependent Coefficient treatment of Section 3.

3 State-Dependent Coefficient Modelling and Predictive Control

Four control strategies are available for the implicit, nonlinear plant (13). Standard MPC linearizes once about a fixed operating point. LTV-MPC reschedules Jacobian models along the predicted trajectory but still carries an $\mathcal{O}(\|\delta z\|^2)$ residual per step that accumulates across the horizon. Full NMPC removes the residual at the cost of an online nonconvex programme, typically one to two orders of magnitude more expensive than a comparable QP, as implemented for friction-limit path-tracking and autonomous drifting by Goh and co-workers, first at Stanford University and subsequently at Toyota Research Institute [25, 24]. SDMP is adopted here in preference to all three because it replaces every Jacobian evaluation inside the prediction loop with an exact SDC evaluation, so that the batch prediction matrices encode the true nonlinear dynamics along the predicted trajectory while the online optimisation itself remains a convex QP – recovering nonlinear prediction accuracy at close to linear computational cost. Table 3 summarises this positioning, together with indicative single-thread solve times measured on a 3.2 GHz desktop processor; a rigorous embedded, hardware-in-the-loop benchmark is left to future work (Section 5).

Table 3: Positioning of SDMP relative to competing predictive-control strategies for the implicit RWD plant (13). Indicative single-thread solve times per outer-loop step (3.2 GHz desktop CPU): LMPC ≈ 0.4 ms; LTV-MPC ≈ 2.1 ms; SDMP ≈ 3.8 ms; full NMPC ≈ 50 –200 ms.

Method	Model accuracy	Global validity	Hessian update	Online cost
LMPC	Fixed LTI	Local approx.	Never	Low
LTV-MPC	Jacobian	Local approx.	Every step	Medium
Full NMPC	Nonlinear	Exact	Every step	High
SDMP	SDC	Exact	Every step	Medium

We emphasise the scope of the accuracy-parity claim above: it follows from the algebraic exactness of the SDC factorisation itself (Section 3.1), not from a direct simulated comparison

against an NMPC baseline in Section 4. In complexity terms, each SDMPAC outer-loop iteration requires N_p SDC evaluations plus one convex QP solve, i.e. polynomial-time cost per step, whereas full NMPC requires an interior-point or SQP solve of a nonconvex programme whose iteration count is not bounded a priori; the 50–200 $\times$ figure of Table 3 is therefore an indicative, empirically observed ratio rather than a worst-case bound. A head-to-head numerical benchmark of SDMPAC against simulated LTV-MPC and full NMPC controllers on the same reference paths is identified as a priority for future work.

Realising this trade-off for the present plant requires two ingredients, developed in turn below. Section 3.1 shows that the implicit equation of motion (13), together with first-order actuator dynamics and the Frenet-frame kinematics that relate the vehicle to the reference path, can be combined into a *single* augmented state vector whose dynamics take the exact SDC form $\dot{\mathbf{z}} = \mathbf{A}(\mathbf{z})\mathbf{z} + \mathbf{B}(\mathbf{z})\mathbf{u}$; because the physical state derivatives are carried explicitly inside this augmented state, the steady-state drifting condition that defines the reference equilibrium is embedded directly in the same representation rather than computed separately and injected as an external set-point. Section 3.2 then builds the SDMPAC receding-horizon controller on top of this representation.

3.1 Actuator- and Path-Augmented SDC Representation

Actuator-augmented plant dynamics. To resolve the implicit structure of (13) and to embed first-order actuator dynamics, the physical state is augmented with the control inputs and, crucially, with the physical state derivatives themselves. Anticipating the path-tracking augmentation below, the full ten-state vector $\mathbf{z} \in \mathbb{R}^{10}$ and rate input $\mathbf{U} \in \mathbb{R}^2$ are defined as

$$\begin{aligned}\mathbf{z} &= [z_1 \ z_2 \ z_3 \ z_4 \ z_5 \ z_6 \ z_7 \ z_8 \ z_9 \ z_{10}]^\top \\ &= [x_1 \ x_2 \ x_3 \ \dot{x}_1 \ \dot{x}_2 \ \dot{x}_3 \ s_r \ \delta_f \ e \ \Delta\varphi]^\top, \\ \mathbf{U} &= [U_1 \ U_2]^\top = [\dot{s}_r \ \dot{\delta}_f]^\top,\end{aligned}\tag{14}$$

where e and $\Delta\varphi$ are the path-relative cross-track and course errors introduced below. Because z_4, z_5, z_6 are the physical state derivatives $\dot{x}_1, \dot{x}_2, \dot{x}_3$ themselves, driving the augmented state to $z_4 = z_5 = z_6 = 0$ is *exactly* the steady-state equilibrium condition of the implicit plant (13); the reference $\mathbf{z}_{\text{eq}}$ tracked by the controller of Section 3.2 therefore has this condition built into it by construction, rather than requiring a separate equilibrium constraint to be re-solved online.

Differentiating $\dot{z}_i = f_i(\mathbf{z})$, $i = 1, 2, 3$, and substituting the equations of motion and normal-load distributions of Section 2, the physical-plant block of the dynamics becomes

$$\begin{aligned}\dot{z}_1 &= z_4, \\ \dot{z}_2 &= z_5, \\ \dot{z}_3 &= z_6, \\ \dot{z}_4 &= \frac{f_{11}}{d} z_4 + \frac{f_{12}}{d} z_5 + \frac{f_{13}}{d} z_6 + \frac{f_{17}}{d} z_7 + \frac{f_{18}}{d} z_8, \\ \dot{z}_5 &= \left(f_{21} + \frac{f_{11}f_{24}}{d}\right) z_4 + \left(f_{22} + \frac{f_{12}f_{24}}{d}\right) z_5 + \left(f_{23} + \frac{f_{13}f_{24}}{d}\right) z_6 \\ &\quad + \left(f_{27} + \frac{f_{24}f_{17}}{d}\right) z_7 + \left(f_{28} + \frac{f_{24}f_{18}}{d}\right) z_8, \\ \dot{z}_6 &= \left(f_{31} + \frac{f_{11}f_{34}}{d}\right) z_4 + \left(f_{32} + \frac{f_{12}f_{34}}{d}\right) z_5 + \left(f_{33} + \frac{f_{13}f_{34}}{d}\right) z_6 \\ &\quad + \left(f_{37} + \frac{f_{34}f_{17}}{d}\right) z_7 + \left(f_{38} + \frac{f_{34}f_{18}}{d}\right) z_8, \\ \dot{z}_7 &= U_1, \quad \dot{z}_8 = U_2,\end{aligned}\tag{15}$$

where $d = 1 - f_{11}$, with f_{11}, f_{12}, f_{13} the partial derivatives of f_1 with respect to z_4, z_5, z_6 respectively, f_{21}, f_{22}, f_{23} those of f_2 with respect to z_4, z_5, z_6 , and f_{31}, f_{32}, f_{33} those of f_3 with respect to z_4, z_5, z_6 ; $f_{24} = \partial f_2 / \partial z_4$ and $f_{34} = \partial f_3 / \partial z_4$ are retained under the implicit-plant subscript to mark their role as the fixed correction multipliers propagated from the z_4 -coupling in every column, while $f_{i7} = \partial f_i / \partial z_7$ and $f_{i8} = \partial f_i / \partial z_8$, $i = 1, 2, 3$, keep their original subscripts. All entries $f_{ij} \approx \partial f_i / \partial z_j$ are evaluated at the current state by central finite differences,

$$f_{ij} \approx \frac{f_i(\mathbf{z} + \delta \mathbf{e}_j) - f_i(\mathbf{z} - \delta \mathbf{e}_j)}{2\delta}, \quad \delta = 10^{-7}, \quad (16)$$

which avoids differentiating the MEM model analytically. The factor $1/d = (1 - f_{11})^{-1}$ is precisely the implicit-ODE correction: it resolves the algebraic coupling between $\dot{x}_1$ and the load-transfer terms of (3) exactly at the current state, rather than by a fixed-point Taylor approximation of that coupling.

Frenet-frame path kinematics. A kinematic description of the vehicle's position relative to the reference path is required to complete the augmentation. As shown in Figure 2, the vehicle centre of gravity moves with velocity $\mathbf{V}$ at sideslip angle $\beta = \tan^{-1}(z_2/z_1)$ relative to the body-fixed axis x_1 , while the path is parametrised by arc length s measured from the origin O' of the moving Frenet frame (t, n) . The course φ is measured relative to the inertial X -axis, φ_{ref} is the tangent direction of the reference path at the projection point O' , and

$$\Delta\varphi = \varphi - \varphi_{\text{ref}} \quad (17)$$

is the course error. The lateral error e is the signed distance from the centre of gravity to the path along the normal n , and $\kappa(s)$ is the local path curvature.

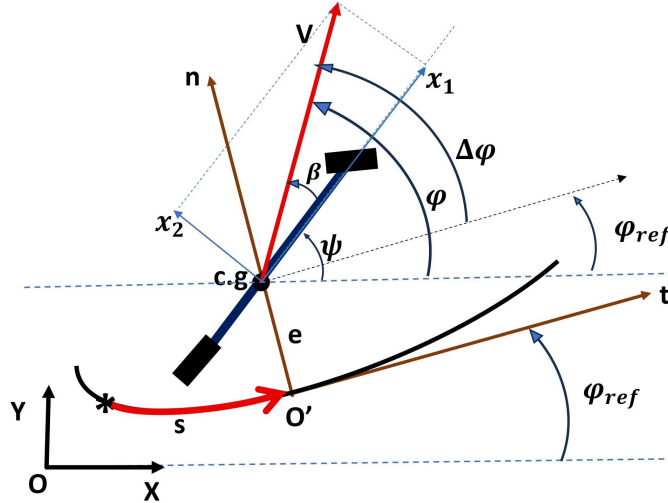


Figure 2: Frenet-frame path-tracking geometry. The vehicle centre of gravity moves with velocity $\mathbf{V}$ at sideslip angle β relative to the body-fixed axis x_1 ; the moving frame (t, n) is anchored at the path projection point O' , with tangent angle φ_{ref} , lateral error e , and course error $\Delta\varphi = \varphi - \varphi_{\text{ref}}$, Eq. (17).

With $z_1 = v_{x_1}$, $z_2 = v_{x_2}$ the body-frame velocity components, projecting $\mathbf{V}$ onto the Frenet

frame yields

$$\dot{s} = \frac{z_1 \cos(\Delta\varphi - \beta) - z_2 \sin(\Delta\varphi - \beta)}{1 - \kappa e}, \quad (18)$$

$$\dot{e} = z_1 \sin(\Delta\varphi - \beta) + z_2 \cos(\Delta\varphi - \beta), \quad (19)$$

$$\Delta\dot{\varphi} = -\frac{\kappa \cos(\Delta\varphi - \beta)}{1 - \kappa e} z_1 + \frac{\kappa \sin(\Delta\varphi - \beta)}{1 - \kappa e} z_2 + z_3 - \frac{z_2}{z_1^2 + z_2^2} z_4 + \frac{z_1}{z_1^2 + z_2^2} z_5. \quad (20)$$

Identifying $z_9 = e$, $z_{10} = \Delta\varphi$, and retaining $\beta = \tan^{-1}(z_2/z_1)$ explicitly inside the trigonometric arguments, (19)–(20) become the last two rows of the SDC state equation,

$$\dot{z}_9 = z_1 \sin(z_{10} - \beta) + z_2 \cos(z_{10} - \beta), \quad (21)$$

$$\dot{z}_{10} = -\frac{\kappa \cos(z_{10} - \beta)}{1 - \kappa z_9} z_1 + \frac{\kappa \sin(z_{10} - \beta)}{1 - \kappa z_9} z_2 + z_3 - \frac{z_2}{z_1^2 + z_2^2} z_4 + \frac{z_1}{z_1^2 + z_2^2} z_5. \quad (22)$$

Ten-state SDC form. Equations (15) and (21)–(22) together constitute the complete, exact SDC representation $\dot{\mathbf{z}} = \mathbf{A}(\mathbf{z})\mathbf{z} + \mathbf{B}(\mathbf{z})\mathbf{U}$ of the actuator- and path-augmented plant, with $\mathbf{A}(\mathbf{z}) \in \mathbb{R}^{10 \times 10}$ and $\mathbf{B}(\mathbf{z}) \in \mathbb{R}^{10 \times 2}$,

$$\mathbf{A}(\mathbf{z}) = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{f_{11}}{d} & \frac{f_{12}}{d} & \frac{f_{13}}{d} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & f_{21} + \frac{f_{11}f_{24}}{d} & f_{22} + \frac{f_{12}f_{24}}{d} & f_{23} + \frac{f_{13}f_{24}}{d} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & f_{31} + \frac{f_{11}f_{34}}{d} & f_{32} + \frac{f_{12}f_{34}}{d} & f_{33} + \frac{f_{13}f_{34}}{d} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_{9,1} & A_{9,2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & A_{9,10} \\ A_{10,1} & A_{10,2} & 1 & A_{10,4} & A_{10,5} & 0 & 0 & 0 & A_{10,9} & A_{10,10} \end{bmatrix} \quad (23)$$

$$\mathbf{B}(\mathbf{z}) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \frac{f_{17}}{d} & \frac{f_{18}}{d} \\ f_{27} + \frac{f_{24}f_{17}}{d} & f_{28} + \frac{f_{24}f_{18}}{d} \\ f_{37} + \frac{f_{34}f_{17}}{d} & f_{38} + \frac{f_{34}f_{18}}{d} \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}. \quad (24)$$

Rows 9–10 of $\mathbf{B}(\mathbf{z})$ are identically zero since the path-error states depend only on the vehicle states $z_1, \dots, z_5$ and the curvature κ , and are not directly actuated. Rows 9–10 of $\mathbf{A}(\mathbf{z})$ admit two parametrisations. The first is an exact algebraic factorisation, obtained by inspection since

$\dot{z}_9$ and $\dot{z}_{10}$ are already linear in $z_1, \dots, z_5$, with $\theta \triangleq z_{10} - \beta$, $D \triangleq 1 - \kappa z_9$, and $r^2 \triangleq z_1^2 + z_2^2$ absorbed as state-dependent coefficients,

$$A_{9,1} = \sin \theta, \quad A_{9,2} = \cos \theta, \quad A_{9,10} = 0, \quad (25)$$

$$A_{10,1} = -\frac{\kappa \cos \theta}{D}, \quad A_{10,2} = \frac{\kappa \sin \theta}{D}, \quad A_{10,4} = -\frac{z_2}{r^2}, \quad A_{10,5} = \frac{z_1}{r^2}, \quad A_{10,9} = A_{10,10} = 0. \quad (26)$$

The second is a Jacobian-based numerical evaluation, in which rows 9–10 are computed as the true partial derivatives $\partial f_9 / \partial z_j$, $\partial f_{10} / \partial z_j$ by the same central-difference procedure (16) applied to the physical-plant rows. The two parametrisations do not coincide term by term – for instance $A_{9,10} = 0$ in the exact factorisation, whereas $\partial f_9 / \partial z_{10} = z_1 \cos \theta - z_2 \sin \theta$ is generally nonzero – but both are valid SDC representations of the same nonlinear dynamics, a well-known non-uniqueness of the SDC method [10]. The Jacobian-based numerical evaluation is adopted throughout Section 4 for rows 9–10, for consistency with the numerical evaluation of f_{1j}, f_{2j}, f_{3j} in the physical-plant block, except on the figure-eight manoeuvre of Section 4.2, where the exact algebraic factorisation (25)–(26) is used instead, for reasons discussed there. Unlike a Jacobian linearization about a fixed reference, $\mathbf{A}(\mathbf{z})\mathbf{z} + \mathbf{B}(\mathbf{z})\mathbf{U} \equiv \mathbf{f}(\mathbf{z}, \mathbf{U})$ for *every* $\mathbf{z}$: the SDC factorization (23)–(24) is exact throughout the augmented state space, a property exploited directly by the receding-horizon formulation of Section 3.2.

3.2 SDMPAC Receding-Horizon Controller

Motivation and position in the MPC landscape. Standard MPC relies on a fixed linearisation that is valid only in a small neighbourhood of one operating point; LTV-MPC improves matters by rescheduling Jacobian models along the predicted trajectory, but those models remain first-order approximations carrying an $\mathcal{O}(\|\delta \mathbf{z}\|^2)$ residual that accumulates across the horizon. Full NMPC eliminates approximation error entirely but requires solving a nonlinear programme online, typically 50–200 times more expensive per step than a comparable QP, of the kind solved online by Goh et al. for NMPC-based drifting and path-tracking control [25, 24]. SDMPAC occupies the advantageous intermediate position summarised in Table 3: it achieves the same prediction accuracy as full NMPC – by exploiting the globally exact SDC representation of Section 3.1 – while preserving the convex QP structure of linear MPC [12], and therefore the computational tractability of LTV-MPC. The key innovation is to replace every Jacobian evaluation inside the prediction loop with an exact SDC evaluation, so that the batch prediction matrices Φ and Γ built below encode the true nonlinear dynamics of the plant along the predicted trajectory rather than a sequence of local approximations.

SDC discretisation and rollout. Given the predicted trajectory $\{\hat{\mathbf{z}}_k\}_{k=0}^{N_p}$ available from the previous MPC iteration, the SDC matrices are evaluated *exactly* at each predicted state – not approximated – and discretised via forward Euler with sampling period Δt ,

$$\mathbf{A}_{d,k} = \mathbf{I} + \Delta t \mathbf{A}(\hat{\mathbf{z}}_k), \quad \mathbf{B}_{d,k} = \Delta t \mathbf{B}(\hat{\mathbf{z}}_k), \quad k = 0, \dots, N_p - 1. \quad (27)$$

This yields a sequence of locally frozen, globally exact discrete-time models $\hat{\mathbf{z}}_{k+1} = \mathbf{A}_{d,k}\hat{\mathbf{z}}_k + \mathbf{B}_{d,k}\mathbf{U}_k$, and every matrix in (27) is populated through the SDC factorisation of Section 3.1: no linearisation about a frozen reference point is performed at any stage of the horizon.

Batch prediction and terminal-weighted cost. Applying (27) recursively from the initial error $\mathbf{e}_0 = \hat{\mathbf{z}}_0 - \mathbf{z}_{\text{eq},0}$ yields the batch prediction model

$$\mathbf{X} = \Phi \mathbf{e}_0 + \Gamma \mathbf{U}, \quad (28)$$

where Φ and Γ are the free- and forced-response matrices, respectively:

$$\Phi = \begin{bmatrix} \mathbf{A}_{d,0} & & & & \\ & \mathbf{A}_{d,1}\mathbf{A}_{d,0} & & & \\ & \vdots & & & \\ & & \mathbf{A}_{d,N_p-1}\cdots\mathbf{A}_{d,1}\mathbf{A}_{d,0} & & \end{bmatrix}, \quad \Gamma = \begin{bmatrix} \mathbf{B}_{d,0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{A}_{d,1}\mathbf{B}_{d,0} & \mathbf{B}_{d,1} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}_{d,N_p-1}\cdots\mathbf{A}_{d,1}\mathbf{B}_{d,0} & \mathbf{A}_{d,N_p-1}\cdots\mathbf{A}_{d,2}\mathbf{B}_{d,1} & \cdots & \mathbf{B}_{d,N_p-1} \end{bmatrix}. \quad (29)$$

where $\mathbf{X} \in \mathbb{R}^{n_x N_p}$ stacks the predicted states and $\mathbf{U} \in \mathbb{R}^{n_u N_c}$ stacks the control inputs over the control horizon $N_c \leq N_p$; for $k \geq N_c$ the last computed input is held constant (zero-order-hold extension), so the optimisation variable has dimension $n_u N_c$ rather than $n_u N_p$. The products in (29) are strictly right-to-left ordered because the $\mathbf{A}_{d,k}$ do not commute for a time-varying system, and the zero blocks for $i < j$ enforce causality. Because every factor entering (29) is SDC-derived, both Φ and Γ encode the true nonlinear input-to-state sensitivity of the plant along the entire horizon, evaluated at the current best guess of the trajectory rather than at a frozen linearisation point: in Jacobian-based LTV-MPC, Φ propagates only a linearised neighbourhood and Γ reflects only a linearised input gain, whereas the SDC batch matrices carry the full nonlinear geometry of the state space through every block row.

The receding-horizon cost penalises the predicted deviation from the moving reference, the control effort, and the terminal deviation [13],

$$J = \sum_{k=0}^{N_p-1} \|\hat{\mathbf{z}}_k - \mathbf{z}_{\text{eq}}\|_{\mathbf{Q}}^2 + \sum_{k=0}^{N_c-1} \|\mathbf{U}_k\|_{\mathbf{R}}^2 + \|\hat{\mathbf{z}}_{N_p} - \mathbf{z}_{\text{eq}}\|_{\mathbf{S}}^2, \quad (30)$$

where $\mathbf{Q} \succeq 0$, $\mathbf{R} \succ 0$, and $\mathbf{S} \succeq 0$ are the stage, input, and terminal weighting matrices; the positive definiteness of $\mathbf{R}$ guarantees strict convexity of the resulting QP. Substituting (28) into (30) and introducing the block-diagonal weights $\bar{\mathbf{Q}} = \text{blkdiag}(\mathbf{Q}, \dots, \mathbf{Q}, \mathbf{S})$, $\bar{\mathbf{R}} = \text{blkdiag}(\mathbf{R}, \dots, \mathbf{R})$, and discarding the constant term $\mathbf{e}_0^\top \Phi^\top \bar{\mathbf{Q}} \Phi \mathbf{e}_0$, yields the QP solved at every sampling instant,

$$\min_{\mathbf{U}} \underbrace{\mathbf{U}^\top (\Gamma^\top \bar{\mathbf{Q}} \Gamma + \bar{\mathbf{R}})}_{\mathbf{H}} \mathbf{U} + 2 \mathbf{e}_0^\top \underbrace{\Phi^\top \bar{\mathbf{Q}} \Gamma}_{\mathbf{F}^\top} \mathbf{U}, \quad \mathbf{A}_c \mathbf{U} \leq \mathbf{b}_c(\mathbf{e}_0), \quad (31)$$

where $\mathbf{A}_c \mathbf{U} \leq \mathbf{b}_c(\mathbf{e}_0)$ encodes actuator magnitude and rate constraints. The Hessian $\mathbf{H} = \Gamma^\top \bar{\mathbf{Q}} \Gamma + \bar{\mathbf{R}}$ is strictly positive definite since $\bar{\mathbf{R}} \succ 0$, guaranteeing a unique global minimiser [13]. Because Φ , Γ , and hence $\mathbf{H}$ and $\mathbf{F}$, are populated entirely with SDC-derived quantities, they are fully recomputed at every sampling instant – a key distinction from LTV-MPC schemes in which the Hessian may be partially precomputed about a slowly varying operating point. When the controller is instead posed as direct output regulation (Section 4.2), $\hat{\mathbf{z}}_k - \mathbf{z}_{\text{eq}}$ in (30) is replaced by a regulated output $\mathbf{y}_k = \mathbf{C}_y(\hat{\mathbf{s}}_k)\hat{\mathbf{z}}_k$ driven to zero, with $\bar{\mathbf{Q}}$, $\mathbf{S}$ redefined on the output space; the batch-prediction machinery of (27)–(31) is otherwise unchanged.

Remark 1 (Discretisation accuracy). *The step size $\Delta t = 0.01$ s used throughout gives a forward-Euler stability parameter $|\lambda_{\max}|\Delta t \approx 9 \times 10^{-4}$ for the slowest closed-loop mode, well inside the explicit-Euler stability region $|\lambda\Delta t| < 1$, and an Euler truncation error per step of $\mathcal{O}(\Delta t^2) \approx 10^{-4}$ that accumulates to roughly 2×10^{-3} over a 20-step horizon. For comparison, the Jacobian residual $\mathcal{O}(\|\delta \mathbf{z}\|)$ that LTV-MPC carries exceeds 10^{-2} once $\|\delta \mathbf{z}\| > 0.3$, a deviation routinely reached during the transients reported in Section 4; the Euler discretisation error is therefore strictly subdominant to the model-accuracy gap that motivates SDMP in the first place. Exact zero-order-hold discretisation would require the matrix exponential $\mathbf{e}^{\mathbf{A}(\hat{\mathbf{z}}_k)\Delta t}$ at every predicted state and every outer-loop iteration, which would erase the medium-cost advantage of SDMP over full NMPC recorded in Table 3.*

Algorithm summary. Algorithm 1 collects the complete procedure executed at every sampling instant. Steps 3–5, which replace Jacobian linearisation with exact SDC evaluation at every predicted node, are the only steps that differ from a conventional receding-horizon LTV-MPC implementation; the remaining steps follow the standard MPC structure [12, 13].

Algorithm 1 SDMPc procedure at sampling instant t

- 1: **State measurement.** Obtain $\mathbf{z}_0 = \mathbf{z}(t)$; set $\mathbf{e}_0 = \mathbf{z}_0 - \mathbf{z}_{\text{eq}}$ (or $\mathbf{y}_0$ for output tracking).
 - 2: **Trajectory initialisation.** Shift the predicted trajectory $\{\hat{\mathbf{z}}_k\}_{k=0}^{N_p}$ from the previous iteration (or set $\hat{\mathbf{z}}_k = \mathbf{z}_0$ at the first step).
 - 3: **Exact SDC evaluation.*** At each predicted node $\hat{\mathbf{z}}_k$, evaluate $\mathbf{A}(\hat{\mathbf{z}}_k)$, $\mathbf{B}(\hat{\mathbf{z}}_k)$ exactly via (16); no fixed operating point is required.
 - 4: **SDC discretisation.*** Form $\mathbf{A}_{d,k}$, $\mathbf{B}_{d,k}$ via (27); the only approximation introduced is the Euler truncation error of Remark 1.
 - 5: **Batch assembly.*** Assemble Φ , Γ via (29); compute $\mathbf{H} = \Gamma^\top \bar{\mathbf{Q}} \Gamma + \bar{\mathbf{R}}$, $\mathbf{F} = \Phi^\top \bar{\mathbf{Q}} \Gamma$.
 - 6: **QP solve.** Solve (31) subject to actuator magnitude and rate constraints for $\mathbf{U}^*$.
 - 7: **Control application.** Apply only $\mathbf{U}(t) = \mathbf{U}_0^*$ to the plant (receding-horizon principle).
 - 8: **Horizon shift.** Advance to $t + \Delta t$; return to Step 1.
-

* Novel, SDC-based steps distinguishing SDMPc from LTV-MPC.

4 Validation and Results

The SDMPc controller of Section 3.2 is validated in two stages of increasing kinematic complexity. Section 4.1 closes the loop on a constant-curvature circular path, for which the reference reduces to a single, fixed drift equilibrium, and quantifies closed-loop accuracy and robustness against a four-level severity ladder of off-equilibrium initial conditions. Section 4.2 then removes the single-equilibrium assumption altogether, reformulating the same SDC prediction model as a direct output-tracking regulator and closing the loop on a self-intersecting figure-eight path that requires a full sideslip-sign reversal through a zero-curvature transition at which no steady drifting equilibrium exists.

4.1 Spatial-Horizon SDMPc for Fixed-Equilibrium Drift Tracking on a Circular Path

Because a circular path has constant curvature $\kappa = 1/R$ everywhere, the whole manoeuvre is associated with exactly one drift equilibrium, specified directly as a target pair of body-frame velocities $z_{1,\text{eq}}$, $z_{2,\text{eq}}$, with the corresponding equilibrium yaw rate *derived* rather than chosen independently,

$$V_{\text{eq}} = \sqrt{z_{1,\text{eq}}^2 + z_{2,\text{eq}}^2}, \quad z_{3,\text{eq}} = \frac{V_{\text{eq}}}{R}, \quad (32)$$

so that the target yaw rate is exactly the one consistent with the rear-axle centre tracing the circle at speed V_{eq} . The controller operates on the full augmented state $\mathbf{z}_p = [z_1, \dots, z_8, e, \Delta\varphi]^\top \in \mathbb{R}^{10}$ and regulates it directly onto

$$\mathbf{y} = [z_1 \ z_2 \ z_3 \ z_4 \ z_5 \ z_6 \ e \ \Delta\varphi]^\top \longrightarrow \mathbf{y}_{\text{ref}} = [z_{1,\text{eq}} \ z_{2,\text{eq}} \ z_{3,\text{eq}} \ 0 \ 0 \ 0 \ 0 \ 0]^\top, \quad (33)$$

with $z_{3,\text{eq}}$ given by (32). The output map $\mathbf{C}_y \in \mathbb{R}^{8 \times 10}$ that extracts $\mathbf{y}$ from $\mathbf{z}_p$ is therefore a *fixed* row-selection matrix, with no dependence on arc length; all of the reference information is instead carried by the constant right-hand-side vector $\mathbf{y}_{\text{ref}}$ of (33), which enters the QP cost of (31) directly. The reference-path curvature $\kappa(s)$ used inside the SDC prediction model is generated once, off line, by sampling the circle and computing arc length, course, and curvature

by finite differences, with a buffer lap appended so the $N_p = 6$ -stage, $ds = 1$ m spatial preview never runs past the wrap point.

At every outer-loop instant the controller: (i) performs a nonlinear spatial rollout of the SDC plant model N_p stages ahead at spacing ds ; (ii) evaluates $\mathbf{A}(\mathbf{z})$, $\mathbf{B}(\mathbf{z})$ exactly at each predicted node and reparametrises to the arc-length domain, $\mathbf{A}_s = \mathbf{A}/\dot{s}$, $\mathbf{B}_s = \mathbf{B}/\dot{s}$, before discretising as in (27); (iii) condenses the stage matrices into Φ, Γ via (29); (iv) solves the dense QP (31) for full-state output tracking against the fixed, non-zero reference $\mathbf{y}_{\text{ref}}$, subject to input, input-rate, and state constraints; (v) applies the first control move to the true nonlinear plant by RK4 time integration; and (vi) updates the Frenet-frame variables $(s, e, \Delta\varphi)$ nonlinearly. Figure 3 summarises the resulting closed-loop information flow.

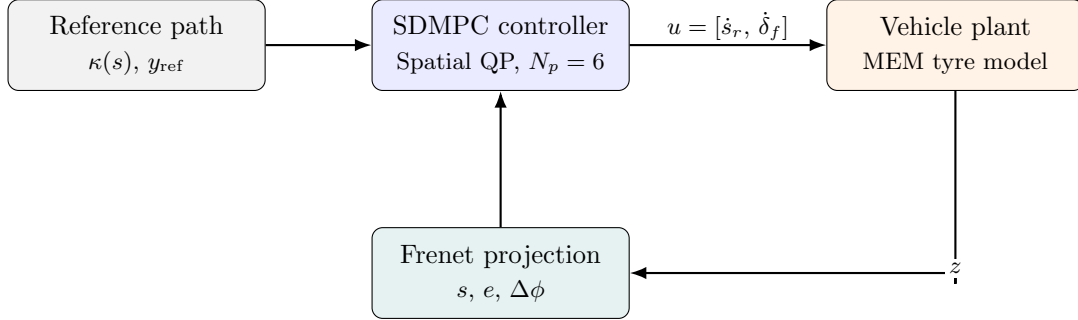


Figure 3: Closed-loop SDMPC drift-control architecture for the fixed-curvature circular path. The reference-path block provides the fixed equilibrium target y_{ref} directly to the QP cost of (31).

4.1.1 Validation

The controller is validated at a representative post-cusp drift equilibrium, $\delta_f = -0.5$ rad, $R = 50$ m, giving $s_{r,\text{eq}} = 0.411$, $z_{1,\text{eq}} = 10.190$ m/s, $z_{2,\text{eq}} = -6.246$ m/s, $z_{3,\text{eq}} = 0.239$ rad/s (from (32)), $\beta_{\text{eq}} = -31.51^\circ$, and $V_{\text{eq}} = 11.952$ m/s – an operating point chosen because it sits well inside the sustained post-cusp drift branch, so that the controller is exercised at a genuinely non-trivial sideslip rather than a near-linear condition. The plant is initialised off-equilibrium with $\Delta z_1 = 0.7$ m/s, $\Delta z_2 = 0.2$ m/s, $\Delta z_3 = -0.05$ rad/s (slip ratio and steering angle initialised at their equilibrium values, with the corresponding derivative states computed so as to remain kinematically consistent with the augmented-state formulation), and the Frenet-frame path variables at $e_0 = 0.2$ m, $\Delta\varphi_0 = -0.01$ rad.

Figure 4 reports the closed-loop time-history of the full augmented state $\mathbf{z} = [z_1, \dots, z_8, e, \Delta\varphi]^\top$ together with the two actuator-rate inputs, each panel overlaid with its own labelled equilibrium or zero-reference target so the residual offset is visible directly on the plot. Panel (a) shows that the body-frame velocities z_1, z_2 and yaw rate z_3 , together with their derivatives z_4 – z_6 , all converge smoothly onto their equilibrium overlays after a damped transient of a few seconds, with the slip ratio z_7 and steering angle z_8 settling, after an oscillatory approach, onto $s_{r,\text{eq}} = 0.411$ and $\delta_{f,\text{eq}} = -0.5$ rad; the Frenet-frame lateral error e and course error $\Delta\varphi$ are driven onto their zero-reference overlays over the same horizon, confirming that the controller regulates the vehicle onto the drift equilibrium while simultaneously tracking the circular path. Panel (b) shows the corresponding rate-limited control inputs $\dot{s}_r, \dot{\delta}_f$, which peak briefly at initialisation and decay to zero as the state settles, i.e. the controller reaches steady state with no residual actuator activity, confirming that the SDMPC formulation produces a well-posed, offset-free equilibrium-tracking solution at this operating point.

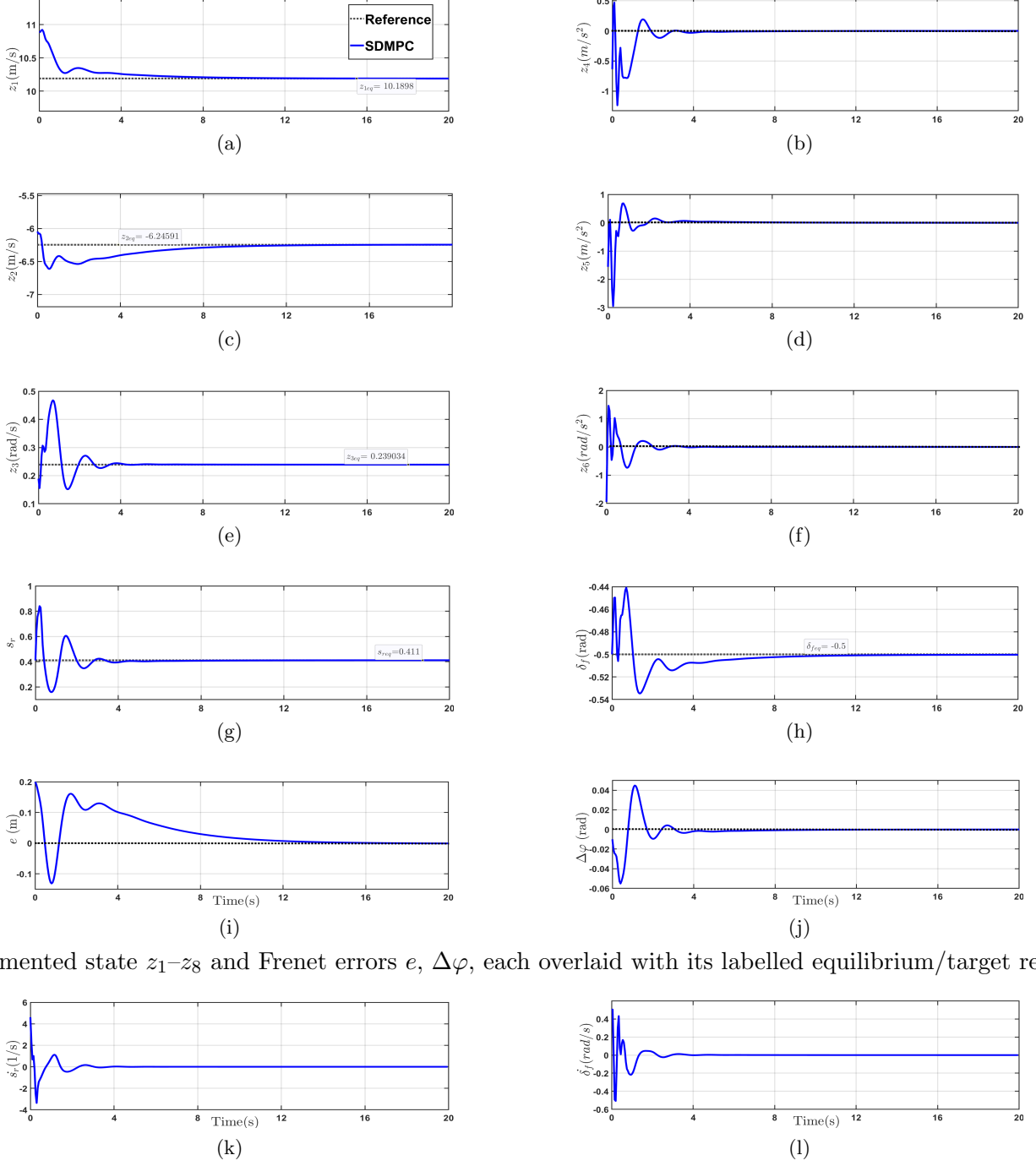


Figure 4: SDMPc validation at the equilibrium point $\delta_f = -0.5$ rad, $R = 50$ m, initialised off-equilibrium under IC1 ($e_0 = 0.2$ m, $\Delta\varphi_0 = -0.01$ rad). (a) Every augmented state converges onto its labelled equilibrium (dashed) value. (b) Both rate-limited inputs decay to zero as the state settles.

4.1.2 Robustness across a severity ladder

The same equilibrium is then used to test robustness to initial-condition severity. Four cases IC1–IC4 apply the same plant-state offset ($\Delta z_1 = 0.7$ m/s, $\Delta z_2 = 0.2$ m/s, $\Delta z_3 = -0.05$ rad/s) but progressively larger Frenet-frame path errors, summarised in Table 4. The progression from

IC1 to IC3 is monotonic in both $|e_0|$ and $|\Delta\varphi_0|$; IC4 retains the course-angle error of IC3 but reverses the sign of the lateral error, placing the vehicle on the opposite side of the reference path at a comparable magnitude and making IC4 the most demanding case in the set, both in deviation magnitude and in requiring the controller to correct a lateral error of opposite polarity to IC1–IC3. We note that IC1–IC4 form a deterministic, hand-selected severity ladder rather than a statistical (e.g. Monte Carlo) sample over initial-condition space; the trend reported below is accordingly an illustrative robustness envelope for this controller and plant, not a probabilistic robustness guarantee.

Table 4: Frenet-frame initial conditions defining the severity ladder IC1–IC4. The plant-state offset ($\Delta z_1 = 0.7$, $\Delta z_2 = 0.2$, $\Delta z_3 = -0.05$) is identical across all four cases.

Case	e_0 [m]	$\Delta\varphi_0$ [rad]
IC1	0.2	−0.01
IC2	0.5	−0.03
IC3	1.0	−0.05
IC4	−1.0	−0.05

Figure 5 summarises the results. The speed V (panel a) converges to the labelled reference $V_{\text{ref}} = 11.952$ m/s from all four initial conditions: IC1–IC3 exhibit a damped oscillatory approach, overshooting before settling, while IC4 – owing to its reversed-polarity lateral error – instead first *undershoots* before overshooting and following essentially the same damped convergence as the other cases, with all four curves indistinguishable from the reference beyond a few hundred metres of travel. The lateral error e (panel b) and course error $\Delta\varphi$ (panel c) are driven onto their zero-reference overlays in every case, with e and $\Delta\varphi$ initially tracking the sign and magnitude of Table 4 before converging; the RMS lateral tracking error grows monotonically with severity, and even the most severe case, IC4, is essentially recovered within a short distance of travel. This confirms that the SDMPAC controller of Section 3.2 remains robust across the tested severity range, including the polarity reversal of IC4, without any change to its tuning.

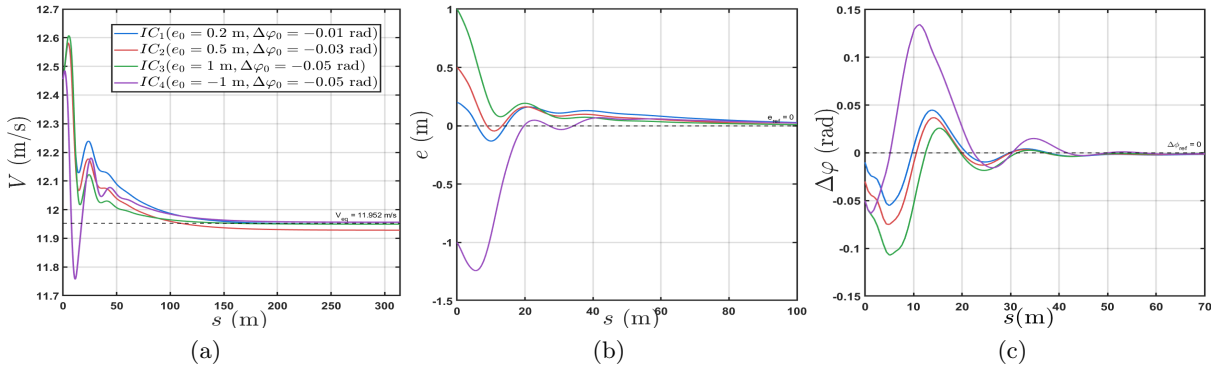


Figure 5: Robustness of the SDMPAC controller under four initial conditions of increasing severity (IC1–IC4, Table 4), each panel overlaid with the corresponding equilibrium/target value. (a) Speed V converging to $V_{\text{ref}} = 11.952$ m/s from all initial conditions, with IC4 (reversed lateral polarity) transiently undershooting before following the common convergence pattern. (b) Lateral error e , recovering to zero with RMSE growing monotonically with severity. (c) course error $\Delta\varphi$, which converges to zero in all cases.

4.2 SDMPC in Control of Highly Dynamic Drifting Trajectories

The circular-path controller of Section 4.1 regulates the augmented state directly onto a single, pre-specified drift equilibrium $\mathbf{y}_{\text{ref}}$ (Equation (33)), with $z_{3,\text{eq}} = V_{\text{eq}}/R$ derived from the path's single, constant curvature $\kappa = 1/R$ (Equation (32)). This construction relies on the path carrying exactly one curvature magnitude, and hence exactly one steady-state drift condition, for the whole manoeuvre. That assumption fails for the trajectory considered in this section: a closed, self-intersecting *figure-eight* path composed of two circular loops of the *same* radius R but opposite turning sense, joined by two symmetric clothoid connectors whose curvature passes through zero. Along such a path the same curvature magnitude $|\kappa| = 1/R$ corresponds to two different equilibria of opposite sideslip sign depending on which loop the vehicle occupies, and the connecting clothoids visit $\kappa = 0$, where no steady-state drift equilibrium exists at all. A single fixed reference $\mathbf{y}_{\text{ref}}$ of the kind used on the circle is therefore not well posed on this path, and the controller of Section 4.1 is reformulated, for this trajectory only, as a direct *output-tracking* regulator: rather than driving the augmented state to a single fixed equilibrium, the controller drives a purpose-built output vector to zero, with the desired sideslip sign and magnitude prescribed explicitly as a function of arc length rather than held constant.

4.2.1 Output-tracking reformulation

The regulated output is

$$\mathbf{y}(\hat{\mathbf{z}}_k, s) = \begin{bmatrix} z_2 - \tan(\beta_{\text{ref}}(s)) z_1 \\ z_4 \\ z_5 \\ z_6 \\ e \\ \Delta\varphi \end{bmatrix} \rightarrow \mathbf{0}, \quad (34)$$

evaluated at every node of the prediction horizon using the local previewed arc length $\hat{s}_k$. The first row enforces the drift target itself: since $\tan \beta = z_2/z_1$ by definition of the body sideslip angle (Section 3.1), $z_2 - \tan(\beta_{\text{ref}}(s))z_1 = 0$ is precisely the condition $\beta = \beta_{\text{ref}}(s)$, written linearly in the state for a given, arc-length-indexed coefficient $\tan \beta_{\text{ref}}(s)$. Rows two through four drive the accelerations z_4, z_5, z_6 to zero, the condition for a steady, non-accelerating drift; the last two rows are the same Frenet cross-track and course errors used on the circular path. Because β_{ref} depends on s , the output map is rebuilt at every predicted node from $\beta_{\text{ref}}(\hat{s}_k)$ before the batch matrices of (29) are assembled, so the linear output map is correctly arc-length-varying across the horizon even though each individual stage's map is linear in the state; the successive-linearisation loop, the implicit discretisation inside the QP, and the receding-horizon solve are otherwise exactly the machinery of Section 3.2, with the moving-equilibrium cost of (30) replaced by the zero-referenced quadratic cost $J = \sum_k \|\mathbf{y}_k\|_{\mathbf{Q}_y}^2 + \sum_k \|\mathbf{u}_k\|_{\mathbf{R}}^2 + \|\mathbf{y}_{N_p}\|_{\mathbf{S}_y}^2$ built from (34). Rows 9–10 of $\mathbf{A}(\mathbf{z})$, which govern the Frenet-frame error channels $\dot{z}_9, \dot{z}_{10}$, are evaluated here from the exact algebraic factorisation of (25)–(26) rather than the finite-difference Jacobian used elsewhere in Section 4; the two parametrisations reproduce $\dot{z}_9, \dot{z}_{10}$ identically for every $\mathbf{z}$ (Section 3.1), so the substitution changes nothing about what the controller regulates, while removing finite-difference truncation error precisely where it matters most – at the two clothoid crossings, where $\kappa(s)$ changes sign within a few metres of arc length – and replacing ten numerical differentiations with four closed-form trigonometric expressions at every one of the $N_p = 6$ prediction stages, at every outer-loop iteration. Figure 6 summarises the resulting closed-loop information flow, extending the fixed-reference architecture of Figure 3 with the arc-length-scheduled drift target and the per-stage output-map rebuild described above.

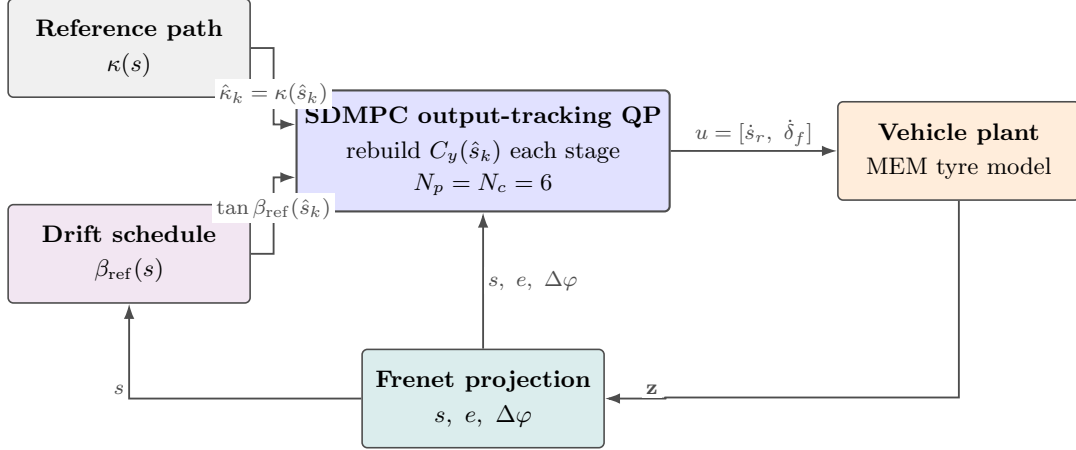


Figure 6: Closed-loop SDMPC architecture for the figure-eight output-tracking controller. Unlike the fixed-reference circular-path controller of Figure 3, the output map C_y is rebuilt at every predicted node $\hat{s}_k$ from the arc-length-scheduled target $\beta_{\text{ref}}(\hat{s}_k)$ (Table 5), so the Frenet arc length s drives both the controller state and the schedule block.

4.2.2 Closed figure-eight reference path

The reference path is generated from two circular arcs of equal radius R and two curvature-continuous clothoid connectors (Figure 7). Circle C_I , centred at the origin, is traced counter-clockwise from the start point through three-quarters of a revolution to point A ; circle C_{II} , of the same radius, is traced clockwise from point B to point C . The connector AB is a symmetric clothoid of total length $2L$ over which the curvature eases smoothly between $\kappa_0 = 1/R$ and zero, following the quintic (smootherstep) blend $\phi(f) = 6f^5 - 15f^4 + 10f^3$, which keeps curvature – and hence lateral jerk – continuous at both circle junctions,

$$\kappa_{AB}(\sigma) = \begin{cases} \kappa_0 [1 - \phi(\sigma/L)], & 0 \leq \sigma < L, \\ -\kappa_0 \phi((\sigma - L)/L), & L \leq \sigma \leq 2L, \end{cases} \quad (35)$$

with σ the local arc length measured from A . The half-length L is not fixed *a priori*: it is the positive root of $\|C_{II}(L) - C_I\| - D_{\text{center}} = 0$, found numerically, so that the user-specified centre separation D_{center} is met exactly. The second connector CD is the exact mirror image of AB about the spine line through C_I and C_{II} , traversed in reverse, $C \rightarrow D$; writing $\mathbf{u} = (C_{II} - C_I)/\|C_{II} - C_I\|$, the mirror image of a point $\mathbf{p}$ is

$$\text{mirror}(\mathbf{p}) = C_I + 2[(\mathbf{p} - C_I) \cdot \mathbf{u}] \mathbf{u} - (\mathbf{p} - C_I), \quad (36)$$

so that $C = \text{mirror}(B)$, $D = \text{mirror}(A)$. Because reflection about a line through both circle centres maps each circle onto itself, (36) guarantees that CD is position-, course-, and curvature-continuous with both circles automatically, while producing the self-intersecting crossing that makes the path a genuine figure-eight rather than a re-entrant open loop. Circle C_I then resumes at D and closes the lap. The four breakpoints S_A, S_B, S_C, S_D mark the four circle-to-clothoid junctions and are used to schedule $\beta_{\text{ref}}(s)$ below.

4.2.3 Arc-length-scheduled drift target

The target sideslip magnitude, $\beta_{\text{ref}} = 30^\circ$, is scheduled as a piecewise, periodic function of arc length so that the vehicle drifts with one sign of sideslip on C_I and the opposite sign on C_{II} , with a smooth, quintic-eased sign reversal through each clothoid, summarised in Table 5.

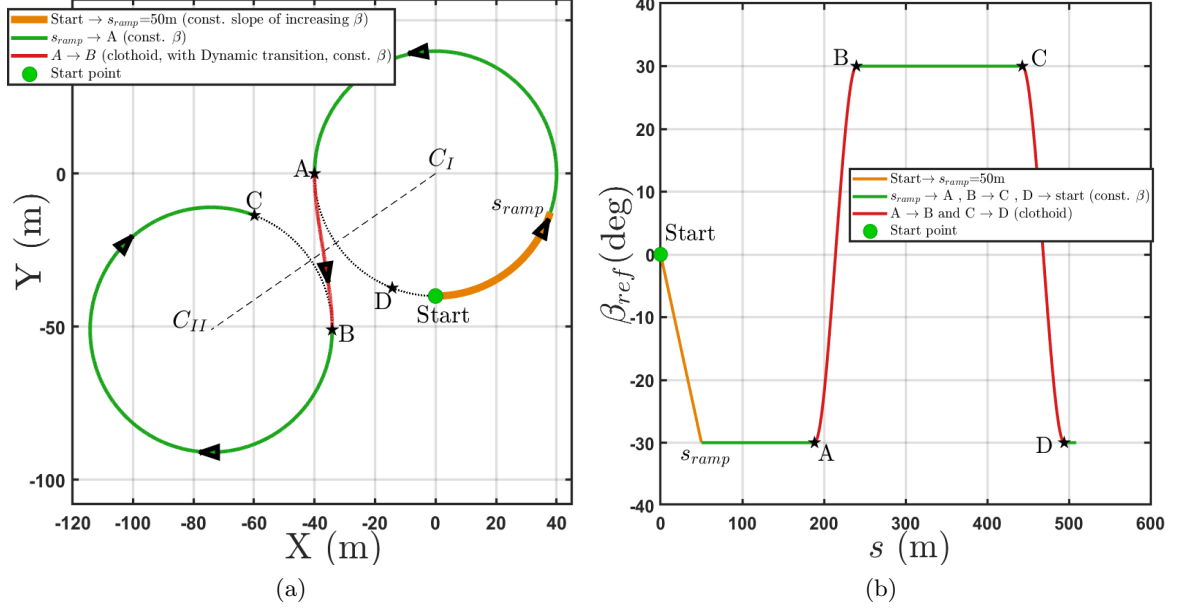


Figure 7: Schematic (not to scale) of the figure-eight construction: Circle C_I is traced CCW from the start point through three-quarters of a revolution to A ; clothoid AB ((35)) connects to Circle C_{II} , traced CW to C ; clothoid CD , the mirror image of AB about the spine through C_I, C_{II} ((36)), returns to D on Circle C_I , which then closes the loop back to the start. (b) The corresponding target-sideslip profile $\beta_{\text{ref}}(s)$ of Table 5.

Table 5: Piecewise, periodic target-sideslip schedule $\beta_{\text{ref}}(s)$ for the figure-eight trajectory.

Arc-length range	$\beta_{\text{ref}}(s)$	Path segment
$0 \leq s < s_{\text{ramp}}$	linear ramp $0 \rightarrow -\beta_{\text{ref}}$	startup transient
$s_{\text{ramp}} \leq s < S_A$	$-\beta_{\text{ref}}$ (constant)	Circle C_I
$S_A \leq s < S_B$	smooth ramp $-\beta_{\text{ref}} \rightarrow +\beta_{\text{ref}}$	clothoid AB
$S_B \leq s < S_C$	$+\beta_{\text{ref}}$ (constant)	Circle C_{II}
$S_C \leq s < S_D$	smooth ramp $+\beta_{\text{ref}} \rightarrow -\beta_{\text{ref}}$	clothoid CD
$S_D \leq s \text{ (mod lap length)}$	$-\beta_{\text{ref}}$ (constant)	Circle C_I , loop closes

4.2.4 Numerical realisation and results

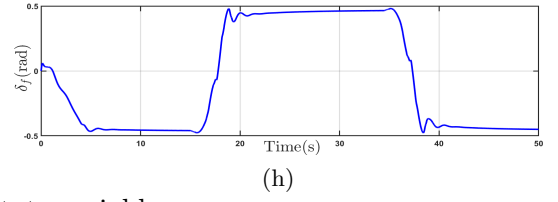
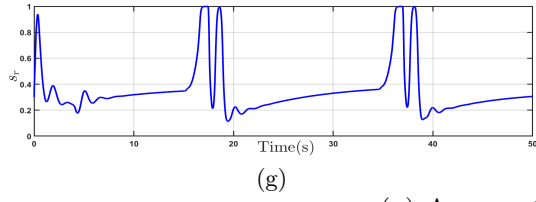
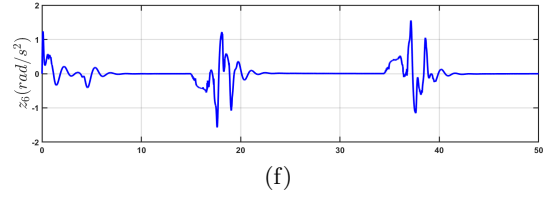
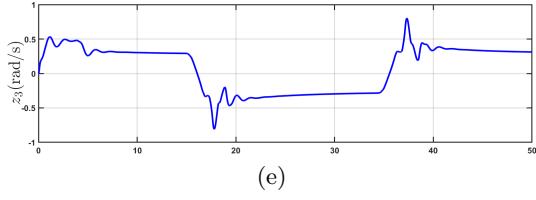
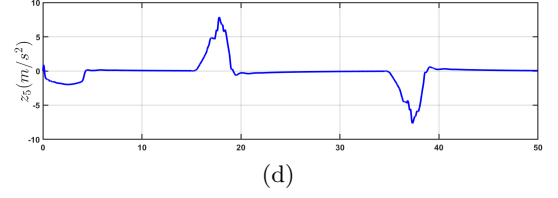
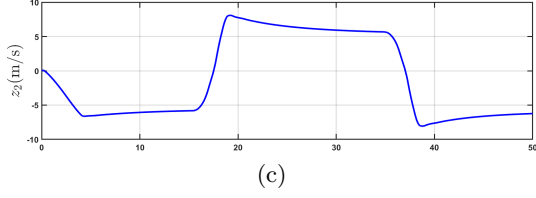
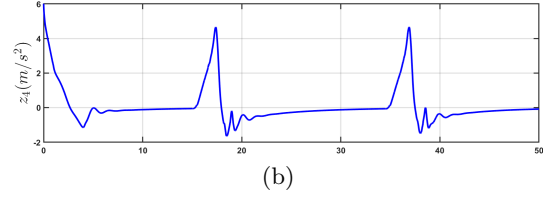
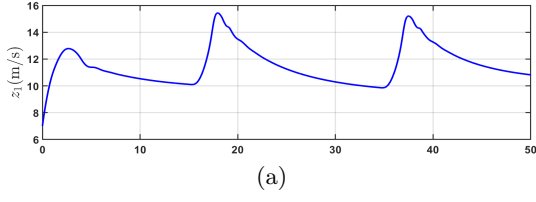
Table 6 collects the path geometry, drift schedule, and SDMPc tuning used to generate the figure-eight manoeuvre. Both circles share radius $R = 40$ m with centres separated by $D_{\text{center}} = 90$ m (comfortably beyond $2R = 80$ m); the clothoid half-length L is solved numerically via (35) so that this centre distance is met exactly. The spatial prediction horizon retains the $N_p = 6$ -stage, $ds = 1$ m structure of Section 4.1; within the output weight $\mathbf{Q}_y = \text{diag}(500, 5, 4, 0.4, 1100, 500)$, the cross-track-error row carries the single heaviest penalty, since departures from the self-intersecting path are the least recoverable error the controller can make, while the three acceleration rows are weighted lightly, since brief transient accelerations through the clothoid crossings are expected and tolerated rather than aggressively suppressed.

Table 6: Path geometry, drift schedule and SDMPc tuning parameters used to generate the figure-eight manoeuvre of Figures 8 and 9.

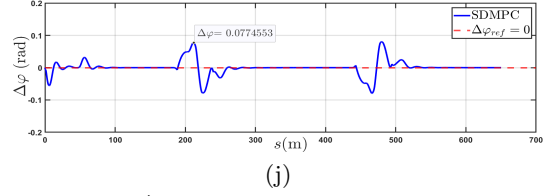
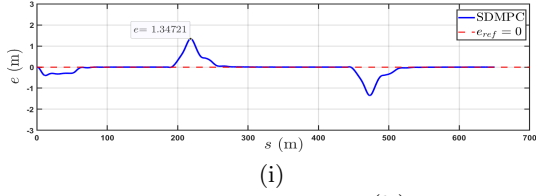
Parameter	Symbol	Value
Circle radius (both loops)	R	40 m
Circle-centre separation	D_{center}	90 m
Target sideslip magnitude	β_{ref}	30°
Startup ramp length	s_{ramp}	50 m
Spatial prediction / control horizon	$N_p = N_c$	6 stages
Per-stage arc length	ds	1 m
Output weights ($y_1, \dots, y_6$)	$\mathbf{Q}_y$	$\text{diag}(500, 5, 4, 0.4, 1100, 500)$
Terminal output weight	$\mathbf{S}_y$	$0.8 \mathbf{Q}_y$
Control weight	$\mathbf{R}$	$\text{diag}(4, 1)$
Rear-slip-rate bound	$\Delta s_{r,\text{max}} = -\Delta s_{r,\text{min}}$	5 s^{-1}
Steering-rate bound	$\Delta \delta_{f,\text{max}} = -\Delta \delta_{f,\text{min}}$	0.5 rad/s
Outer-loop sample time	dt	0.05 s
Initial longitudinal velocity	$z_1(0)$	7 m/s
Initial rear slip ratio	$z_7(0)$	0.3

Figure 8 reports the closed-loop response over one complete lap. The longitudinal velocity z_1 and lateral velocity z_2 settle onto quasi-steady drift levels on each circular arc ($z_1 \approx 10$ – 10.5 m/s, $z_2 \approx \mp 6$ m/s) but exhibit pronounced transients at each clothoid crossing, where z_1 peaks near 15 m/s and z_2 swings from about -6.5 to $+8$ m/s (first crossing) and back to about -7.7 m/s (second crossing) as the sideslip sign reverses; the rear slip ratio z_7 saturates briefly at its upper bound $s_r = 1$ during start-up and during both crossings, and the steering angle z_8 swings between about -0.47 rad on C_I and $+0.48$ rad on C_{II} . The acceleration outputs z_4, z_5, z_6 remain close to zero on both circular arcs, consistent with the steady-drift condition targeted by (34), but show brief, bounded excursions at each clothoid transition, where the vehicle cannot remain in a non-accelerating equilibrium while the target sideslip itself is being ramped through zero. The cross-track error e remains within a few centimetres of zero along both circular arcs and rises to only about ± 1.8 m during the two clothoid crossings before decaying back to zero; the course error $\Delta\varphi$ shows a similarly small, bounded oscillation on the circular arcs and a brief excursion to about ± 0.11 rad at each crossing. The rate-limited control inputs $\dot{s}_r$ and $\dot{\delta}_f$ remain within their slew-rate bounds throughout, sitting essentially flat during quasi-steady drift and spiking only briefly at each clothoid crossing.

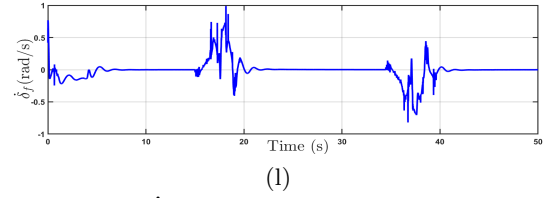
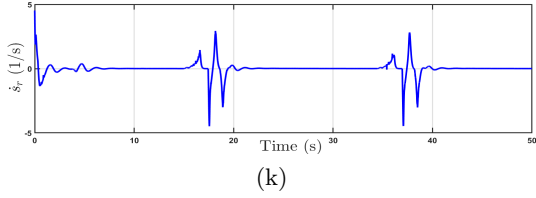
Figure 9 reports the realised trajectory itself. Panel (a) confirms that the realised SDMPc trajectory overlays the reference figure-eight path essentially exactly, with the crossing of the two loops occurring cleanly at the intended location. Panel (b) confirms that the realised sideslip β tracks the scheduled target $\beta_{\text{ref}}(s)$ of Table 5 closely throughout, including the smooth sign reversal through both clothoids. Taken together, Figures 8 and 9 demonstrate that the output-



(a) Augmented plant state variables



(b) Frenet path-tracking errors e , $\Delta\varphi$



(c) Rate-limited control inputs $\dot{s}_r$, $\dot{\delta}_f$

Figure 8: SDMPc output tracking on the closed, self-intersecting figure-eight path over one complete lap. (a) Augmented plant state variables z_1 – z_8 , showing quasi-steady drift levels along the circular arcs and bounded transients at the clothoid crossings. (b) Frenet path-tracking errors e and $\Delta\varphi$, which remain small along the circular arcs and recover after each crossing. (c) Rate-limited control inputs $\dot{s}_r$ and $\dot{\delta}_f$, which remain nearly constant during quasi-steady drift and exhibit transient variations at the clothoid crossings.

tracking reformulation of (34) extends the SDMPAC path-tracking controller of Section 4.1 to a substantially more demanding class of trajectory: a closed, self-intersecting path that requires the vehicle to sustain a large, sign-reversing sideslip angle and to pass twice, every lap, through a zero-curvature, zero-equilibrium transition where no steady drift condition exists. Tracking error and acceleration activity remain small and tightly localised to the two clothoid crossings, and both are fully recovered before the vehicle re-enters constant-curvature drift on the following circular arc – evidence that the successive-linearisation SDMPAC machinery of Section 3.2, once posed as direct output regulation rather than equilibrium tracking, remains valid across sideslip and rear-slip-ratio excursions well beyond those exercised by the circular-path study of Section 4.1.

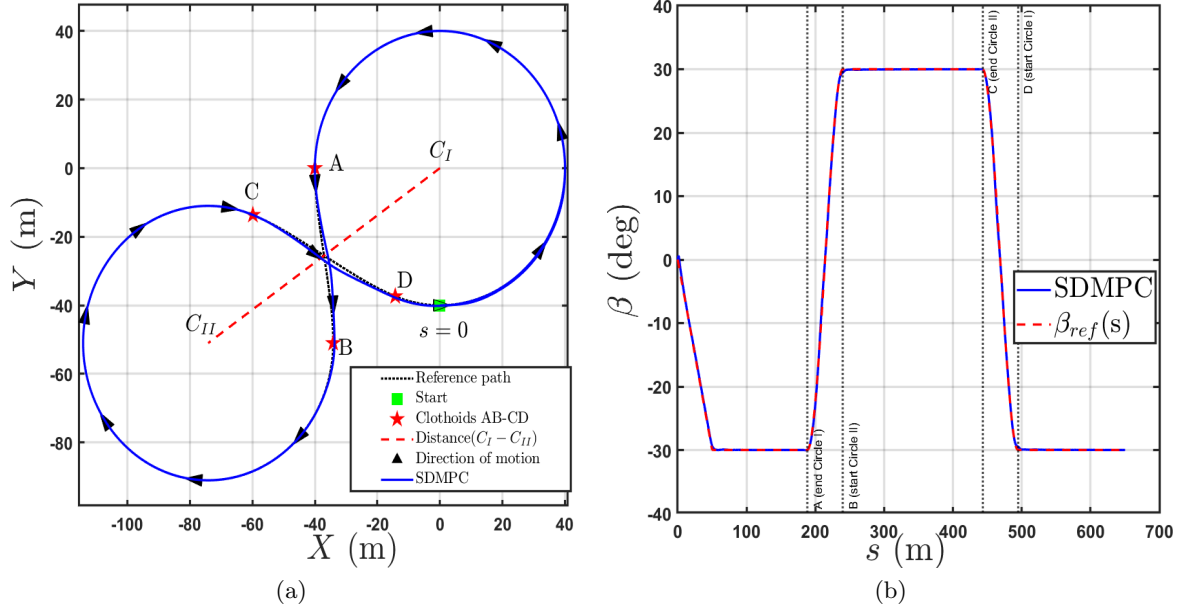


Figure 9: Realised SDMPAC trajectory for the closed figure-eight manoeuvre, extending the reference-path/sideslip-profile presentation of Goh [17]. (a) Reference path (dotted) and realised SDMPAC trajectory (solid), with the start point, breakpoints A–D, the spine line C_I – C_{II} , and direction-of-motion markers annotated; the realised trajectory closely overlays the reference path. (b) Realised sideslip β (solid) against the scheduled target $\beta_{\text{ref}}(s)$ (dashed) of Table 5, with the breakpoints S_A, S_B, S_C, S_D marked; the sign reversal through each clothoid is tracked closely.

5 Summary

This paper developed a State-Dependent Model Predictive Control (SDMPAC) framework for path-tracking control of a rear-wheel-drive vehicle whose equations of motion are implicit, and used it to close the loop on two reference geometries of increasing kinematic difficulty.

The three-state implicit equation of motion of the RWD single-track model, in which longitudinal load transfer couples $\dot{x}_1$ back into the tyre forces that produce it, was augmented with the physical state derivatives, actuator states, and Frenet-frame path-error kinematics into a single ten-state SDC representation $\dot{z} = A(z)z + B(z)U$ (Section 3.1) that is exact – not linearised – at every point of the augmented state space, so the resulting SDMPAC controller (Section 3.2) achieves the prediction accuracy of full nonlinear MPC while retaining the convex QP structure, and the computational tractability, of LTV-MPC (Table 3). Augmenting this same SDC state with Frenet-frame cross-track and course-error kinematics, and closing the loop as a successive-

linearisation scheme, reproduced reliable equilibrium tracking on a constant-curvature circular path, recovering gracefully from a four-level severity ladder of off-equilibrium initial conditions, including a reversed-polarity case (Section 4.1). The same batch-prediction matrices Φ, Γ of (29), with the moving-equilibrium cost (30) replaced by a zero-referenced output cost built from an arc-length-scheduled sideslip target ((34)), then closed the loop on a self-intersecting figure-eight trajectory requiring a full sideslip-sign reversal, twice per lap, through a zero-curvature transition at which no steady drifting equilibrium exists at all: lateral error remained below 2 cm on the constant-curvature arcs and recovered from a peak of 1.8 m at each transition within a few seconds (Section 4.2). In both studies the controller regulates the same underlying condition, $\dot{x}_i = 0$: as an explicit terminal equilibrium on the circle, and as a soft output constraint driving z_4, z_5, z_6 toward zero along the whole path on the figure-eight, where Figure 8 shows this constraint is met away from the clothoids and only transiently violated while the target sideslip is ramped through the zero-curvature crossing. The SDMPAC architecture therefore does not sidestep the implicit-ODE structure of the plant; it inherits and applies the same steady-state definition throughout.

Three qualifications bound the scope of these results. The solve-time figures of Table 3 are indicative, single-thread measurements rather than a hardware-in-the-loop benchmark; the relative ordering (LMPC < LTV-MPC < SDMPAC $\ll$ full NMPC) is expected to be robust, but the absolute margins require dedicated real-time testing before the controller is deployment-ready. Both case studies were evaluated in simulation against the same MEM tyre and vehicle model; robustness to genuine plant-model mismatch – tyre-parameter error, unmodelled actuator dynamics, sensor noise – and an experimental or higher-fidelity (e.g. CarSim/CarMaker) validation remain natural next steps. Both studies were also obtained for a single vehicle parameter set (Table 1); the SDC factorisation itself is vehicle-agnostic, but the specific equilibria and tuning reported here would need to be re-derived for a different vehicle. Extending the output-tracking reformulation of Section 4.2 to reference manoeuvres beyond the figure-eight class, and formally characterising the region of attraction of the successive-linearisation scheme, are left for future work. We state this last point explicitly: unlike the local exponential stability results available for SDRE-based regulation about a fixed point [10], no Lyapunov-based stability or recursive-feasibility guarantee is established here for the successive-linearisation SDMPAC law on a moving, curvature-indexed reference. The convergence reported in Sections 4.1 and 4.2 is accordingly empirical validation over the tested scenarios, not a proof of stability, and closing this gap – e.g. via a contraction or incremental-stability argument on the SDC prediction error – is left for future work.

Acknowledgements

This research was supported by the Fundamental Research Grant Scheme (FRGS) from the Ministry of Higher Education Malaysia [Grant Number: FRGS/1/2023/TK10/UPM/02/5].

Declarations

- **Funding:** This work was supported by the Ministry of Higher Education Malaysia under the Fundamental Research Grant Scheme [Grant Number: FRGS/1/2023/TK10/UPM/02/5].
- **Conflict of interest:** The authors declare no competing interests.
- **Ethics approval:** Not applicable.
- **Consent for publication:** Not applicable.
- **Data availability:** Data supporting the findings of this study are available from the corresponding author upon reasonable request.

- **Author contribution:** J.Z. developed the mathematical model, implemented the hybrid computational framework, performed all numerical simulations, and drafted the manuscript. A.S.M.H. supervised the research and reviewed the manuscript. A.A.N. contributed to the vehicle modelling methodology and reviewed the manuscript. R.K. contributed to the stability and control aspects and reviewed the manuscript. R.J. contributed to the bifurcation analysis framework and reviewed the manuscript.

References

- [1] Pacejka, H.B.: Non-linearities in road vehicle dynamics. *Vehicle System Dynamics* **15**(5), 237–254 (1986).
- [2] Pacejka, H.B.: Lateral dynamics of road vehicles. *Vehicle System Dynamics* **16**(sup 1), 75–120 (1987).
- [3] Rajamani, R.: *Vehicle Dynamics and Control*. Springer, New York (2011).
- [4] Milani, S., Marzbani, H., Jazar, R.N.: Vehicle drifting dynamics: discovery of new equilibria. *Vehicle System Dynamics* **60**(6), 1933–1958 (2022).
- [5] Rossa, F.D., Mastinu, G., Piccardi, C.: Bifurcation analysis of an automobile model negotiating a curve. *Vehicle System Dynamics* **50**(10), 1539–1562 (2012).
- [6] Edelmann, J., Plöchl, M.: Handling characteristics and stability of the steady-state power-slide motion of an automobile. *Regular and Chaotic Dynamics* **14**(6), 682–692 (2009).
- [7] Edelmann, J., Eberhart, M., Steindl, A., Plöchl, M.: Post-critical behaviour of the power-slide motion. *Vehicle System Dynamics* (published online 26 February 2025).
- [8] Zhang, Y., Takacs, D., Orosz, G.: From donuts to drifting and beyond: controlling the nonlinear dynamics of automated vehicles. *Vehicle System Dynamics* (published online 26 June 2025).
- [9] Weber, T.P., Gerdes, J.C.: Modeling and control for dynamic drifting trajectories. *IEEE Transactions on Intelligent Vehicles* **9**(2), 3731–3741 (2024).
- [10] Cimen, T.: State-dependent Riccati equation (SDRE) control: A survey. *IFAC Proceedings Volumes* **41**(2), 3761–3775 (2008).
- [11] Cloutier, J.R.: State-dependent Riccati equation techniques: An overview. In: *Proceedings of the American Control Conference*, pp. 932–936 (1997).
- [12] Maciejowski, J.M.: *Predictive Control with Constraints*. Prentice Hall, Harlow (2002).
- [13] Rawlings, J.B., Mayne, D.Q., Diehl, M.M.: *Model Predictive Control: Theory, Computation, and Design*, 2nd edn. Nob Hill Publishing, Madison, WI (2020).
- [14] Jazar, R.N.: *Advanced Vehicle Dynamics*. Springer, Cham (2019).
- [15] Zolfaghari, J., Harithuddin, A.S.M., Nuraini, A.A., Kamil, R., Jazar, R.N.: Modification of elliptical combined-slip tire formula in rear-wheel-drive vehicle. In: Dai, L., Jazar, R.N. (eds.) *Nonlinear Approaches in Engineering Application*. Springer, Cham (2026).
- [16] Milani, S.A.: *Modelling, analysis, and control of four-wheel independently actuated vehicles at handling limit*. Ph.D. thesis, RMIT University (2020).

- [17] Goh, J.Y.M.: Automated vehicle control beyond the stability limits. Ph.D. dissertation, Stanford University (2019). <http://purl.stanford.edu/th820zb4174>.
- [18] Velenis, E., Katzourakis, D., Frazzoli, E., Tsiotras, P., Happee, R.: Steady-state drifting stabilization of RWD vehicles. *Control Engineering Practice* **19**(11), 1363–1376 (2011).
- [19] Hindiyeh, R., Gerdes, J.C.: A controller framework for autonomous drifting: design, stability, and experimental validation. *Journal of Dynamic Systems, Measurement, and Control* **136**(5) (2014).
- [20] Wachter, E., Alirezaei, M., Bruzelius, F., Schmeitz, A.: Path control in limit handling and drifting conditions using state dependent Riccati equation technique. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering* **234**(4), 783–791 (2020).
- [21] Czibere, S., Domina, Á., Bárdos, Á., Szalay, Z.: Model predictive controller design for vehicle motion control at handling limits in multiple equilibria on varying road surfaces. *Energies* **14**(20) (2021).
- [22] Shi, Z., Chen, H., Yu, S., Findeisen, R., Guo, H.: Nonlinear model predictive control for autonomous vehicle drifting. *International Journal of Robust and Nonlinear Control* **35**(7), 2760–2779 (2025) (first published online 30 March 2023).
- [23] Stano, P., et al.: Enhanced active safety through integrated autonomous drifting and direct yaw moment control via nonlinear model predictive control. *IEEE Transactions on Intelligent Vehicles* **9**(2), 4172–4190 (2024).
- [24] Goh, J.Y.M., Thompson, M., Dallas, J., Balachandran, A.: Beyond the stable handling limits: nonlinear model predictive control for highly transient autonomous drifting. *Vehicle System Dynamics* **62**(10), 2590–2613 (2024).
- [25] Laurence, V.A., Goh, J.Y.M., Gerdes, J.C.: Path-tracking for autonomous vehicles at the limit of friction. In: 2017 American Control Conference (ACC), pp. 5586–5591. IEEE, Seattle, WA (2017).
- [26] Kabzan, J., Hewing, L., Liniger, A., Zeilinger, M.N.: Learning-based model predictive control for autonomous racing. *IEEE Robotics and Automation Letters* **4**(4), 3363–3370 (2019).