

Vehicle-Specific Configuration Space Distance Fields for Terrain-Aware Predictive Control of Heavy Robots

Tuukka Mattila

Tampere University

Mehdi Heydari Shahna

`mehdi.heydarishahna@tuni.fi`

Tampere University

Pauli Mustalahti

Tampere University

Jouni Mattila

Tampere University

Research Article

Keywords: field robotics, terrain traversability, signed distance fields, nonlinear model predictive control, skid-steer robots, terrain-aware navigation, online kinematic estimation

Posted Date: October 19th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11401816/v1>

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

[Read Full License](#)

Additional Declarations: The authors declare no competing interests.

Vehicle-Specific Configuration Space Distance Fields for Terrain-Aware Predictive Control of Heavy Robots

Tuukka Mattila

Faculty of Engineering and Natural Sciences
Tampere University

Mehdi Heydari Shahna*

Faculty of Engineering and Natural Sciences
Tampere University

Pauli Mustalahti

Faculty of Engineering and Natural Sciences
Tampere University

Jouni Mattila

Faculty of Engineering and Natural Sciences
Tampere University

Abstract

Heavy skid-steer robots can traverse terrain that appears occupied in a planar map when their wheels remain supported and low obstacles fit beneath the chassis. Exploiting these configurations requires a navigation model that distinguishes wheel contact regions from components with different ground clearances. This article develops a vehicle-specific terrain representation for predictive control of a 6000-kg skid-steer robot. Wheel-patch geometry, cross-wheel support consistency, body inclination, and component-wise underside clearance classify position-heading configurations. A periodic signed configuration space distance field (CDF) converts this classification into a continuous terrain cost with a conditional grid-level execution margin. The CDF enters a nonlinear model predictive controller coupled to online effective skid-steer estimation. Recorded MuJoCo comparisons on simple and rocky courses illustrate the tradeoff between reference tracking and body attitude during terrain negotiation. Circular recordings show increasing tracking deviation across increasing activation-distance labels, while one continuous run covers 11 nominal reference periods. Outdoor recordings establish execution on the skid-steer robot, with mean controller update times of 6.5–6.7 ms. Field-data replay quantifies kinematic calibration, and measured wheel responses characterize starts and reversals. Together, the recorded simulation trajectories and full-scale experiments support the integration of vehicle-specific terrain reasoning into predictive navigation.

Keywords: field robotics; terrain traversability; signed distance fields; nonlinear model predictive control; skid-steer robots; terrain-aware navigation; online kinematic estimation.

Nomenclature

Symbols are grouped by topic, following the structure of the article. The unit column gives the coherent SI unit, whereas values are quoted as in the article (e.g., angle limits in degrees). In that column, “—” denotes a dimensionless quantity and “mixed” a vector or matrix whose entries carry the SI units of the corresponding coordinates. An empty cell marks a set, index, operator, or cost quantity (cost weights and costs are numerical tuning quantities).

*Corresponding author: mehdi.heydarishahna@tuni.fi

Abbreviations

Abbreviation	Definition
CDF	Configuration space distance field
CPU	Central processing unit
CUDA	NVIDIA parallel computing platform and programming model for GPUs
D2H	Device-to-host (GPU-to-CPU) transfer (Table 4)
DT	Distance transformation (Table 10)
EDT	Euclidean distance transform
EKF	Extended Kalman filter
GNSS	Global navigation satellite system
GPU	Graphics processing unit
H2D	Host-to-device (CPU-to-GPU) transfer (Table 4)
HM	Height-map extraction (Table 10)
HPIPM	High-performance interior-point method (quadratic programming solver)
IMU	Inertial measurement unit
LiDAR	Light detection and ranging
MPC	Model predictive control
MPPI	Model predictive path integral control
MuJoCo	Multi-Joint dynamics with Contact (physics simulator)
NMPC	Nonlinear model predictive control
PBA+	Revised GPU implementation of the Parallel Banding Algorithm for exact Euclidean distance transforms
PI	Proportional–integral (control)
PID	Proportional–integral–derivative (control)
QP	Quadratic program
RAM	Random-access memory
RMS	Root mean square
RMSE	Root-mean-square error
$SE(2)$	Special Euclidean group of planar rigid-body poses (position and heading)
SI	International System of Units
SLAM	Simultaneous localization and mapping
SQP	Sequential quadratic programming
TSDF	Truncated signed distance field
<i>Outdoor run identifiers</i>	
K1	Circular-reference run with nominal kinematic parameters
K2	Circular-reference run with online-estimated kinematic parameters, initialized at nominal geometry
S1–S3	Straight-reference runs with online estimation (second session)
C1, C2	Circular-reference runs with online estimation (second session)

Symbols

Symbol	Description	Unit
<i>Platform, configuration, and motion model</i>		
x, y	Planar position of the reference point (center of the wheel rectangle) in the gravity-aligned map frame	m
ψ	Heading angle in the map frame	rad
$\mathbf{q}$	Planar configuration $[x, y, \psi]^\top$	mixed
$\mathbf{p}$	Planar position $[x, y]^\top$	m
ξ, ζ	Vehicle-fixed coordinates pointing forward and left, respectively	m

Continued on next page

Symbols (continued)

Symbol	Description	Unit
$R(\psi)$	Planar rotation from vehicle coordinates (ξ, ζ) to map coordinates	–
ω_L, ω_R	Left- and right-side wheel angular speeds	rad/s
ω	Side wheel-speed vector $[\omega_L, \omega_R]^\top$; commanded wheel speeds in the planning state and prediction model	rad/s
v	Longitudinal speed of the reference point in the vehicle frame, (1)	m/s
Ω	Yaw rate, $\dot{\psi} = \Omega$	rad/s
r	Nominal wheel radius, 0.411 m	m
α	Effective-wheel-radius scale; nominal value 1, clipped to $[0.5, 1.5]$ by the estimator	–
r_{eff}	Effective wheel radius αr	m
c	Effective half-track, $c > 0$; nominal value 0.976 m	m
X_v	Longitudinal offset, along ξ , of the effective rotation center from the reference point; nominal value 0	m
θ	Effective skid-steer parameters $[X_v, c, \alpha]^\top$	mixed
$\hat{\theta}$	Current estimate of θ , held constant along the prediction horizon	mixed
$f_q(q, \omega; \theta)$	Right-hand side of the effective skid-steer kinematics, (2)	mixed
χ	Planning state $[q^\top, \omega^\top]^\top$	mixed
u	Optimization input $\dot{\omega}$, the rate of the commanded wheel speeds	rad/s ²
<i>Height-map extraction</i>		
Φ_n	Signed distance of voxel n in a vertical TSDF column	m
W_n	TSDF weight of voxel n	–
z_n	Height of voxel n	m
Δ	Spacing of TSDF voxels, height-map cells, and configuration positions, 0.15 m	m
η_n	Interpolated surface height at a negative-to-positive crossing, (3)	m
$W_{\min}$	Minimum weight of both voxels for a valid crossing; 0.25 in the offline extraction	–
z_b	Height of the vertical band above the ground level in which valid crossings and near-surface voxels are evaluated; 3.5 m in the offline extraction	m
d_s	Near-surface threshold, $ \Phi_n < d_s$; 0.15 m in the offline extraction	m
n_s	Near-surface voxel count at which the highest near-surface voxel supplies the terrain height; 6 in the offline extraction	–
r_f	Filling radius for assigning a missing column the height of the nearest valid column; 0.5 m in the offline extraction, 20.5 m in the deployed implementation	m
$H(x, y)$	Extracted height of the terrain or obstacle surface relevant to the vehicle footprint	m
<i>Terrain admissibility</i>		
$\mathcal{G}$	Position–heading configuration grid	–
N_ψ	Number of heading samples: 100 onboard and in the terrain-processing benchmark, 36 in the offline recorded-map analysis	–
$\Delta\psi$	Heading step $2\pi/N_\psi$	rad
n_p	Number of samples in a 5×3 wheel patch, 15	–
$\ell_{j,i}$	Vehicle-frame offset $[\xi_{j,i}, \zeta_{j,i}]^\top$ of sample i in wheel patch j	m
$H_{j,i}$	Height-map sample at $\mathbf{p} + R(\psi)\ell_{j,i}$ (nearest-neighbor lookup), (4)	m
$\mathbf{H}_j$	Height samples of wheel patch j , $[H_{j,1}, \dots, H_{j,n_p}]^\top$	m
A_j	Least-squares design matrix of wheel patch j with rows $[\xi_{j,i}, \zeta_{j,i}, 1]$; its pseudoinverse $A_j^\dagger$ is precomputed	mixed
z	Height coordinate of a fitted plane, $z = a_j\xi + b_j\zeta + h_j$	m
a_j, b_j, h_j	Slopes and offset of the plane fitted to wheel patch j	–, –, m
ε_j	Maximum residual of wheel patch j from its fitted plane, (5)	m

Continued on next page

Symbols (continued)

Symbol	Description	Unit
$\pi_s(\xi, \zeta)$	Common support plane $a\xi + b\zeta + h_0$ fitted to the samples of the first three wheel patches	m
a, b, h_0	Slopes and offset of the support plane π_s	–, –, m
ε_s	Cross-wheel consistency residual of the fourth patch relative to π_s , (6)	m
φ, ϑ	Geometric roll and pitch of the support plane, (7); positive roll denotes a higher left side, positive pitch a lower front	rad
φ_j, ϑ_j	Geometric roll and pitch of the plane fitted to wheel patch j	rad
$\bar{\varphi}_w$	Wheel-patch inclination limit on $ \varphi_j $ and $ \vartheta_j $, 30°	rad
$\bar{\varepsilon}_w$	Wheel-patch residual limit, 0.05 m	m
$\bar{\varepsilon}_s$	Cross-wheel consistency threshold, 0.10 m	m
φ_-, φ_+	Support-plane roll limits, $\pm 20^\circ$	rad
ϑ_-, ϑ_+	Support-plane pitch limits, -20° and 30°	rad
$\mathbf{g}$	Map-frame center of an observed height-map cell in an underside region	m
ℓ_g	Vehicle-frame location $R(\psi)^\top(\mathbf{g} - \mathbf{p})$ of cell center $\mathbf{g}$	m
$\kappa(\cdot)$	Permissible terrain height above π_s by underside region: 0.20 m central chassis, 0.125 m bogie strips ($0.60 \text{ m} \leq \zeta \leq 0.90 \text{ m}$), 0.30 m outer envelope and stowed-manipulator footprint	m
$\mathcal{A}$	Set of admissible grid configurations	
$\mathcal{B}$	Set of rejected grid configurations, $\mathcal{G} \setminus \mathcal{A}$	
<i>Configuration space distance field and grid margin</i>		
$\rho(\mathbf{q}, \mathbf{q}')$	Heading-periodic metric between configurations $\mathbf{q}$ and $\mathbf{q}' = [x', y', \psi']^\top$, (10)	m
λ	Heading scale weighting heading against translation; 1 m/rad onboard, $\Delta/\Delta\psi$ in the offline recorded-map analysis	m/rad
γ	Node (configuration) of the grid $\mathcal{G}$	mixed
$D(\gamma)$	Signed CDF at grid nodes: distance to $\mathcal{B}$ for admissible nodes and negative distance to $\mathcal{A}$ for rejected nodes, (11)	m
$\text{dist}_\rho(\cdot, \cdot)$	Distance under ρ from a configuration to the nearest element of a set	m
B, T	Lower and upper halves, along the heading dimension, of the binary configuration volume; the periodic transform uses $[T, B, T, B]$	
$d(\mathbf{q})$	Heading-periodic trilinear interpolant of D queried by the controller	m
∇d	Gradient of d with respect to $\mathbf{q}$, estimated from the nearest-feature indices	mixed
r_g	Scaled-cell half diagonal $\frac{1}{2}\sqrt{2\Delta^2 + (\lambda\Delta\psi)^2}$	m
$\mathbf{q}^*$	Planned pose satisfying $d(\mathbf{q}^*) \geq m$ in Proposition 1	mixed
m	Lower bound on the CDF value at $\mathbf{q}^*$, $m > r_g$	m
δ	Bound on the deviation $\rho(\mathbf{q}, \mathbf{q}^*)$ of an executed pose $\mathbf{q}$, (12)	m
γ_ν, μ_ν	Corners of the interpolation cell of $\mathbf{q}^*$ and their trilinear weights (proof of Proposition 1)	mixed, –
β	Rejected grid node, taken as its periodic image nearest to $\mathbf{q}^*$ (proof of Proposition 1)	mixed
<i>Effective-kinematics estimation</i>		
$\mathbf{z}$	Augmented EKF state $[\mathbf{q}^\top, \boldsymbol{\theta}^\top]^\top$	mixed
$\hat{\mathbf{z}}^-, \hat{\mathbf{z}}^+$	Predicted and corrected estimates of $\mathbf{z}$	mixed
$\boldsymbol{\omega}_m$	Measured left- and right-side wheel-speed inputs; front-wheel measurements onboard	rad/s
$f_e(\mathbf{z}, \boldsymbol{\omega}_m)$	Augmented continuous-time model $[f_q^\top, \mathbf{0}^\top]^\top$ with random-walk parameters	mixed
T_e	Estimator prediction interval; 1 ms in the field-data replay	s
A, B	State and input Jacobians of the discretized EKF prediction, (13)	mixed
P, P^-, P^+	State covariance before prediction, after prediction, and after correction	mixed
P_0	Initial state covariance; $10^{-3}I_6$ in the implemented estimator	mixed

Continued on next page

Symbols (continued)

Symbol	Description	Unit
Q	Process-noise covariance added per prediction step; $\text{diag}(10^{-6}, 10^{-6}, 10^{-6}, 10^{-6}, 10^{-6}, 10^{-8})$ in the implemented estimator	mixed
R_u	Covariance of the measured wheel-speed inputs; $10^{-4}I_2$ in the implemented estimator	rad^2/s^2
R	Pose-measurement noise covariance; $10^{-3}I_3$ in the implemented estimator	mixed
C	Measurement matrix $[I_3 \ 0]$	—
K	Kalman gain	mixed
ν	Innovation: position difference and wrapped heading difference	mixed
<i>Predictive control</i>		
N	Number of prediction intervals; 200 onboard	—
h	Prediction (forward-Euler) step; 0.25 s onboard, giving a 50 s horizon	s
$\mathbf{q}_i, \mathbf{u}_i, \mathbf{u}_i, \chi_i$	Predicted pose, wheel-speed command, input, and planning state at prediction step i	mixed
$f_d(\chi_i, \mathbf{u}_i; \hat{\theta})$	Discrete prediction model, (15)	mixed
$\mathbf{p}_i^{\text{ref}}, \psi_i^{\text{ref}}, \omega_i^{\text{ref}}$	Reference position, heading, and wheel speed at prediction step i ; $\omega_i^{\text{ref}} = \mathbf{0}$ onboard	m, rad, rad/s
L_i	Stage cost, (16)	
L_N	Terminal cost: position, heading, wheel-speed, and CDF penalties at $i = N$	
$w_p, w_\psi, w_\omega, w_u, w_c$	Nonnegative weights on position error, heading error, wheel-speed deviation, wheel-speed rate, and CDF penalty; onboard values (2.0, 0.4, 0.2, 0.5, 80.0)	
d_a	CDF activation distance, below which the terrain penalty is active; 0.6 m onboard; 0.2 m, 0.6 m, and 1.0 m in the circular simulation comparison	m
$\omega_{\max}$	Bound on the commanded wheel speeds; 0.8 rad/s onboard	rad/s
$u_{\max}$	Slew-rate bound on the commanded wheel speeds; 0.3 rad/s ² onboard	rad/s ²
$\hat{\mathbf{q}}_k$	Estimated pose used at control update k	mixed
ω_k^{cmd}	Wheel-speed command at control update k ; $\omega_0^{\text{cmd}} = \mathbf{0}$	rad/s
$\bar{\mathbf{q}}_i$	Pose at prediction step i of the preceding SQP iterate	mixed
d_i	CDF value $d(\bar{\mathbf{q}}_i)$	m
$\mathbf{g}_i$	Nearest-feature-based estimate of the CDF gradient at $\bar{\mathbf{q}}_i$	mixed
$\tilde{d}_i$	Linearized CDF value $d_i + \mathbf{g}_i^\top (\mathbf{q}_i - \bar{\mathbf{q}}_i)$, (18)	m
$H_{c,i}^{\text{GN}}$	Gauss–Newton Hessian of the terrain cost at prediction step i	
δt_k	Elapsed time interval at control update k	s
β_k	Interpolation factor $\min(\delta t_k/h, 1)$ of the command and pose updates, (19)–(20)	—
$\mathbf{u}_{0 k}^*, \omega_{1 k}^*, \mathbf{q}_{1 k}^*$	First optimal input, and first predicted wheel-speed command and pose, of the solution computed at control update k	rad/s ² , rad/s, mixed
$\mathbf{e}$	Planar position error $\mathbf{p} - \mathbf{p}^{\text{ref}}$ in the tuning balances	m
$\nabla_p d$	Gradient of d with respect to planar position	—
d_p	Maximum available CDF margin of a passage, $d_p < d_a$	m
e^*	Indicative position error at which the terrain and position costs are equal, (22)	m
<i>Evaluation</i>		
τ	Elapsed time from the first valid timestamp of a simulation recording	s
$\mathbf{p}_{\text{line}}^{\text{ref}}(\tau), \mathbf{p}_{\text{circle}}^{\text{ref}}(\tau)$	Nominal straight-line and circular simulation references, (23)–(24)	m
f	Frequency of the circular simulation reference, 0.007 Hz	Hz
T	Period $1/f \approx 142.857$ s of the circular simulation reference	s
e_i	Time-indexed planar deviation $\ \mathbf{p}_i - \mathbf{p}_i^{\text{ref}}\ _2$ at retained frame i	m
n	Number of retained frames in an evaluation window	—
$e_{\text{RMS}}, e_{\text{max}}$	RMS and maximum of e_i over an evaluation window, (25)	m
e_{last}	Tracking error at the final included frame	m

Continued on next page

Symbols (continued)

Symbol	Description	Unit
$ \phi _{\max}, \theta _{\max}$	Maximum absolute body roll and pitch, extracted from the simulator pose relative to world vertical (reported in degrees)	rad
<i>Indices, operators, and accents</i>		
i	Prediction-step index ($i = 0, \dots, N$); also sample index in a wheel patch ($i = 1, \dots, n_p$) and retained-frame index in (25)	
j	Wheel index: 1 front left, 2 front right, 3 rear left, 4 rear right	
k	Control-update index	
n	Voxel index along a vertical TSDF column	
ν	Corner index of an interpolation cell (proof of Proposition 1)	
$(\cdot)^\top$	Transpose	
$(\cdot)^\dagger$	Pseudoinverse	
$\dot{(\cdot)}$	Time derivative	
$\partial(\cdot)/\partial(\cdot)$	Partial derivative (Jacobian)	
$\hat{(\cdot)}$	Estimate	
$(\cdot)^{\text{ref}}, (\cdot)^{\text{cmd}}$	Reference and commanded values	
$(\cdot)_{i k}^*$	Optimal value at prediction step i of the solution at control update k	
$(\cdot)_{0:N}$	Sequence over prediction steps $0, \dots, N$	
$[\cdot; \cdot]$	Vertical stacking of vectors	
$ \cdot $	Absolute value, applied componentwise to vectors	
$\ \cdot\ , \ \cdot\ _2$	Euclidean norm	
$\ \cdot\ _\infty$	Maximum (infinity) norm	
$[s]_+$	Positive part $\max(0, s)$	
$\text{wrap}(\cdot)$	Angle wrapping to $(-\pi, \pi]$	
$\text{atan2}(\cdot, \cdot)$	Four-quadrant inverse tangent	
$\text{diag}(\cdot)$	Diagonal matrix with the listed entries	
$\simeq$	Approximate equality (local model)	
I, I_2, I_3, I_6	Identity matrix; the subscript gives its dimension	
$\mathbf{0}$	Zero vector or matrix	

1 Introduction

Heavy mobile machines operating in construction, forestry, and mining must negotiate terrain that is both a support surface and a source of obstacles. For a skid-steer machine, the relevant question is often how to traverse an irregularity: a rock may fit beneath the central chassis, interfere with a lower component, or obstruct a wheel, depending on the vehicle’s position and heading. This distinction is especially useful when navigating through scattered rocks or along a confined route, where suitable wheel placement can preserve a traversable corridor. It also makes navigation a coupled geometric and control problem. A planned configuration must provide wheel support and body clearance, while the commanded motion must remain compatible with skid steering and the response of the wheel drives. The practical challenge is to express these requirements in a representation that can be updated from perception and queried repeatedly within predictive control.

Terrain assessment provides the first part of this connection. Surveys describe a broad range of geometric, semantic, and experience-based traversability measures (Borges et al., 2022; Shu et al., 2025). For ground robots in three-dimensional environments, Krüsi et al. (2017) assess traversability by fitting robot-sized planar patches directly to point-cloud maps and integrate this assessment with motion planning and trajectory optimization. Xue et al. (2023) fuse successive LiDAR frames to construct terrain models and assess traversability through geometric connectivity between neighboring terrain regions. For articulated planetary rovers, approximate clearance evaluation uses interval bounds on wheel heights to assess attitude, suspension configuration, and body clearance, including obstacle straddling (Otsu et al., 2020). Ho et al. (2016) use Gaussian-process

regression to predict rover attitude and chassis configuration from terrain geometry, including partially occluded and deformable terrain. At the scale of construction equipment, the terrain navigation system of Guan et al. (2022) combines semantic-geometric mapping, footprint-aware Hybrid A*, and tracking on a 49-ton excavator. TerrainNet learns ground, ceiling, and semantic map layers for off-road predictive navigation and uses wheel-location elevations in attitude-related costs (Meng et al., 2023). These studies establish the value of interpreting terrain through the geometry and operating requirements of the vehicle.

Heading-dependent representations make vehicle-specific terrain assessment available to motion planning. SEB-Naver constructs a GPU-based $SE(2)$ terrain-risk representation and uses signed-distance values and gradients in trajectory optimization, followed by tracking MPC (Li et al., 2025). Capsizing-guided planning evaluates orientation-dependent terrain conditions when optimizing trajectories for subsequent NMPC tracking (Zhang et al., 2025). In a recent preprint, Shi et al. (2026) construct yaw-specific traversable regions and use an A*-string-pulling-A* strategy for geometric pathfinding. Li et al. (2024) developed configuration space distance fields for manipulation planning, providing joint-space distance queries and gradients relative to robot-object contact configurations. The present work develops a periodic, vehicle-specific distance field that combines local wheel support, cross-wheel consistency, and component-wise underside clearance. Its position and heading derivatives enter the skid-steer NMPC directly, allowing terrain negotiation and reference tracking to be optimized within the same control problem.

Predictive terrain reasoning has also been demonstrated through complete field systems. Han et al. (2024) combine sampling-based MPC, physics-derived rollover and ditch-crossing costs, wheel-coplanarity penalties, and feedforward-feedback execution on a full-size off-road vehicle. Their results emphasize the interaction between terrain geometry, motion selection, and tracking at operating speed. For articulated machines, long short-term planning first samples contact schedules and whole-body keyframes and then refines them through short-horizon MPC; its demonstrations include a hydraulic walking excavator (Jelavic et al., 2023). These systems motivate a representation appropriate to the machine’s locomotion mechanism. Here, continuous skid-steer motion is described by a planar position-heading configuration, while terrain checks evaluate the supporting surface and clearance beneath the vehicle.

For skid-steer equipment, the prediction model and command interface are equally consequential. Hierarchical NMPC with superellipsoid separation has been used for obstacle avoidance and trajectory tracking on a compact track loader (Moran et al., 2024). Using the same skid-steer robot as the present study, Paz et al. (2025) investigate multiple-shooting NMPC with pose and wheel measurements, bounded wheel motion, low-level tracking, and explicit treatment of dead-zone effects. Gaussian-process residual models within stochastic model predictive path integral control (MPPI) further combine terrain-dependent motion prediction, uncertainty propagation, path following, and obstacle avoidance (Trivedi et al., 2025). These are closely related formulations of predictive mobile-machine navigation. The present study concentrates on the interface between detailed support and clearance geometry and receding-horizon wheel commands, using a configuration space distance field to combine terrain adaptation and reference tracking in the same objective.

Effective steering parameters connect those commands to observed motion. Skid-steer models based on instantaneous centers of rotation can be updated from measured motion to account for changing wheel-terrain interactions (Pentzer et al., 2014). Zuo et al. (2024) jointly estimate navigation states and skid-steer parameters from vision and wheel encoders, optionally aided by an IMU, and analyze parameter observability. Okawara et al. (2024) integrate LiDAR, IMU, and wheel measurements in a factor graph with online kinematic calibration and wheel-odometry uncertainty estimation. Complementary work estimates undesired vehicle-level skidding from IMU and wheel-encoder measurements using deep learning (Nourizadeh et al., 2024). These results provide a foundation for adapting effective kinematics during operation. In the present control hierarchy, estimated pose and steering parameters update the NMPC prediction, and recorded trajectories make their influence on prediction error measurable.

The control system connects perception and motion planning to wheel-command execution. Wang et al. (2024) demonstrate upper-level NMPC with coupled left/right current-velocity mappings and PID compensation for hydraulic skid-steer tracking. Complementary approaches address slip compensation through neural adaptive tracking (Shahna et al., 2025a) and hydraulic wheel actuation through torque-observer-based valve

control with input–output constraints (Shahna et al., 2025b). Learned dynamic local-model networks have also supported feedforward control of hydraulic excavator assistance functions, including coupled inputs and measured disturbances (Greiser et al., 2024). At the navigation level, Shahna et al. (2026) integrate stereo visual pose estimation, constrained reinforcement-learning-based motion planning, robust adaptive wheel control, and supervisory safe-return logic. The present study complements these developments by expressing vehicle-specific wheel support and underside clearance as a configuration space distance field for NMPC. Accordingly, the wheel-command interface and measured wheel responses form part of the evaluation, connecting terrain-aware motion generation to the measured wheel response.

This article develops and evaluates that connection for a 6000-kg skid-steer robot. A height map is converted into admissible and rejected vehicle configurations by testing wheel-patch support, cross-wheel consistency, predicted attitude, and component-specific underside clearance. A signed configuration space distance field (CDF) then represents proximity to the rejected set in a metric that accounts for both translation and heading. Its values and gradients allow NMPC to trade reference tracking against movement toward geometrically admissible configurations. The central contribution is the construction and use of this vehicle-specific interface within an executable navigation hierarchy. The article makes three contributions:

1. *A terrain field built from wheel support and component clearance.* Local terrain tests distinguish the surfaces supporting the wheels from the space available beneath individual underside components. Their classification produces a heading-periodic CDF. A conditional grid-level margin result relates the interpolated field value, grid resolution, and execution deviation, giving an explicit interpretation of the available configuration margin.
2. *Direct integration with skid-steer predictive control.* A one-sided CDF penalty is incorporated into the tracking NMPC cost to discourage low-margin and geometrically inadmissible configurations. The formulation combines this terrain penalty with bounded wheel-speed commands, bounded command slew rates, and updated effective steering parameters. Field values and gradients allow the optimizer to adjust position and heading over the prediction horizon. An implementation algorithm specifies the map-update, estimation, optimization, and command-execution sequence.
3. *Complementary recorded simulation and full-scale experimental evidence.* Archived MuJoCo trajectories quantify tracking and body attitude across simple- and rocky-course variants, circular activation-distance labels, and a continuous multi-lap run. Recorded full-scale experiments evaluate computational timing, prediction with estimated kinematics, attitude agreement, trajectory behavior, and wheel-command response.

The presentation begins with the mechanical platform, available measurements, and prediction model. It then develops the terrain representation and control formulation in Sections 3 and 4. Evaluation follows the same progression from recorded MuJoCo simulations in Section 5 to real-machine experiments in Section 6, connecting the geometric design choices to measured execution.

2 Robot, Sensing, and Motion Model

2.1 Mechanical platform and sensing

As shown in Fig. 1, the experimental platform is a 6000-kg four-wheel skid-steer robot with a rear-mounted electrified manipulator. Differential motion of the left and right wheel pairs produces skid steering; the navigation interface accepts one wheel-speed command per side. The vehicle has a 1.45 m wheelbase, a 1.952 m track, and a nominal wheel radius of 0.411 m. The navigation geometry represents the manipulator in a fixed, stowed posture. The central chassis and the lateral bogie regions have different underside clearances, making the location of a terrain feature within the footprint important even when the vehicle center and heading change only slightly.

The sensor installation comprises front-mounted LiDAR and stereo cameras, GNSS antennas, wheel-speed measurements, inertial measurements, and pressure channels (Fig. 2). Visual simultaneous localization and mapping (SLAM) supplies the navigation pose at approximately 30 Hz. The recorded SLAM trajectory is also used as the pose reference for the motion analyses. Terrain processing starts from a truncated signed distance field (TSDF) map exported by nvblox (Millane et al., 2024). Measured angular speeds from the left and right front wheels provide the motion input to the onboard kinematic estimator, while the external pose provides its correction. Further details of the platform’s low-level control system are presented by Paz et al. (2025).

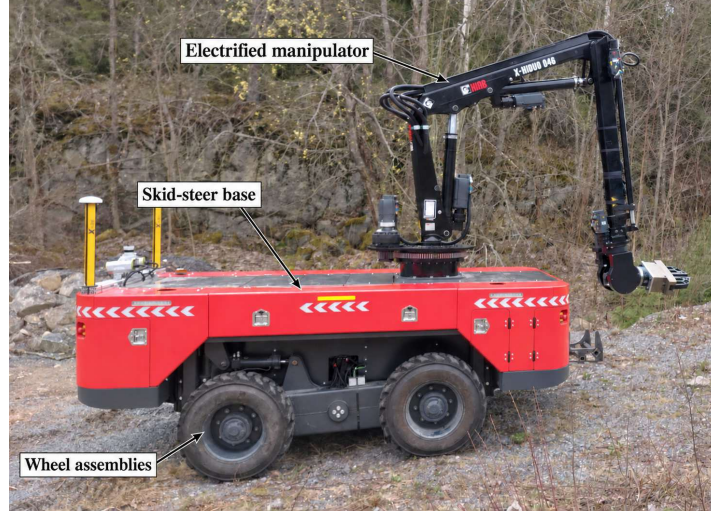


Figure 1: Skid-steer robot with its rear-mounted electrified manipulator and wheel assemblies.

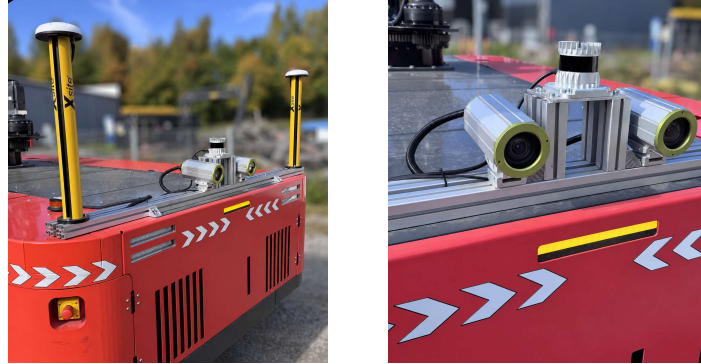


Figure 2: Experimental skid-steer robot. Left: front view of the vehicle and sensor installation. Right: LiDAR mounted above the stereo cameras.

2.2 Frames and effective skid-steer kinematics

Let $\mathbf{q} = [x, y, \psi]^\top$ denote the planar configuration in a gravity-aligned map frame. The model reference point is the center of the wheel rectangle. Vehicle-fixed coordinates ξ and ζ point forward and left, respectively, and $\boldsymbol{\omega} = [\omega_L, \omega_R]^\top$ contains the left- and right-side wheel angular speeds. An effective instantaneous-center-of-rotation model describes the relationship between wheel motion and planar motion:

$$v = \frac{\alpha r}{2}(\omega_L + \omega_R), \quad \Omega = \frac{\alpha r}{2c}(\omega_R - \omega_L), \quad (1)$$

$$\dot{\mathbf{q}} = \begin{bmatrix} v \cos \psi + X_v \Omega \sin \psi \\ v \sin \psi - X_v \Omega \cos \psi \\ \Omega \end{bmatrix} = f_q(\mathbf{q}, \boldsymbol{\omega}; \boldsymbol{\theta}), \quad \boldsymbol{\theta} = [X_v, c, \alpha]^\top. \quad (2)$$

Here $c > 0$ is the effective half-track, X_v is the longitudinal offset of the effective rotation center, and $r_{\text{eff}} = \alpha r$ is the effective wheel radius. The nominal radius $r = 0.411$ m is fixed, and the implemented estimator represents the effective radius through the dimensionless scale α . These parameters summarize the wheel–terrain interaction and the wheel-signal scale within a low-speed planar model. Nominal differential-drive geometry corresponds to $(X_v, c, \alpha) = (0, 0.976, 1)$. Estimating the parameters online allows the predictor to follow the effective turning behavior of the machine (Zuo et al., 2024; Okawara et al., 2024).

The planning state augments the pose with commanded wheel speeds, $\chi = [\mathbf{q}^\top, \boldsymbol{\omega}^\top]^\top$, and the optimization input is their rate $\mathbf{u} = \dot{\boldsymbol{\omega}}$. Measured wheel speeds drive state estimation, whereas the previous wheel-speed command initializes the wheel states used by NMPC. This preserves continuity of the commanded speed profile under the slew-rate constraints. Reinitializing that profile from noisy or lagging wheel measurements would perturb the command state and could restrict command buildup during wheel-drive start-up transients.

2.3 Geometry and execution architecture

Table 3 gives the geometric classification and control settings. The underside model contains a central chassis region, two lower bogie strips, and an outer envelope. Their clearance thresholds are measured relative to the terrain support plane defined in Section 3. The overall vehicle dimensions describe the physical platform, whereas the envelope is the selected geometric screening region.

Table 3: Platform geometry, reported onboard settings, and offline recorded-map grid.

Quantity	Value
Mass; overall length $\times$ width	6000 kg; 3.706×2.2 m
Wheelbase; track; nominal wheel radius r	1.45; 1.952; 0.411 m
Grid spacing Δ	0.15 m
Heading count N_ψ : onboard / offline recorded-map analysis	100 / 36
Onboard heading scale λ	1 m/rad
Chassis / bogie / envelope clearance thresholds	0.20 / 0.125 / 0.30 m
Bogie-strip lateral position $ \zeta $	[0.60, 0.90] m
Chassis and strip length; envelope dimensions	2.7 m; 3.9×2.4 m
Support-plane roll; pitch limits	$\pm 20^\circ$; $[-20^\circ, 30^\circ]$
Wheel-patch inclination; residual limits	30° ; 0.05 m
Cross-wheel consistency threshold	0.10 m
Onboard prediction intervals N ; step h	200; 0.25 s
Onboard wheel-command speed; slew-rate bounds	0.8 rad/s; 0.3 rad/s ²
Onboard weights $(w_p, w_\psi, w_\omega, w_u, w_c)$	(2.0, 0.4, 0.2, 0.5, 80.0)
Onboard CDF activation distance d_a	0.6 m
Onboard maximum SQP / QP iterations	5 / 5
Onboard solver stopping tolerances	10^{-20}

Terrain processing and predictive control operate at separate rates (Fig. 3). Terrain processing extracts a height map, classifies position–heading configurations, and publishes a signed CDF. The controller queries its values and gradients while optimizing wheel commands using the latest pose and effective kinematic parameters.

The approximately 30×30 m mapping grid is centered on the vehicle. Its horizontal axes are aligned with the vehicle’s initial heading and retain that orientation as the vehicle turns. The offline reconstruction in Section 6 evaluates $177 \times 177 \times 36$ configuration samples. Recorded timing shows terrain updates near 5 Hz and command updates near 150 Hz. Section 5 analyzes the recorded simulator trajectories and reports the observed external-timestamp update rates. The deployed system runs on a Nuvo-9160GC computer equipped with an Intel Core i9-13900E processor, an NVIDIA RTX 3050 GPU, and 32 GB of RAM. The configuration classifier evaluates position–heading samples in parallel on the GPU, and the signed distance transform is also

computed on the GPU. Height extraction and assembly of the periodic heading volume run on the CPU. The terrain-processing stages and their execution times are described in Section 3.6.

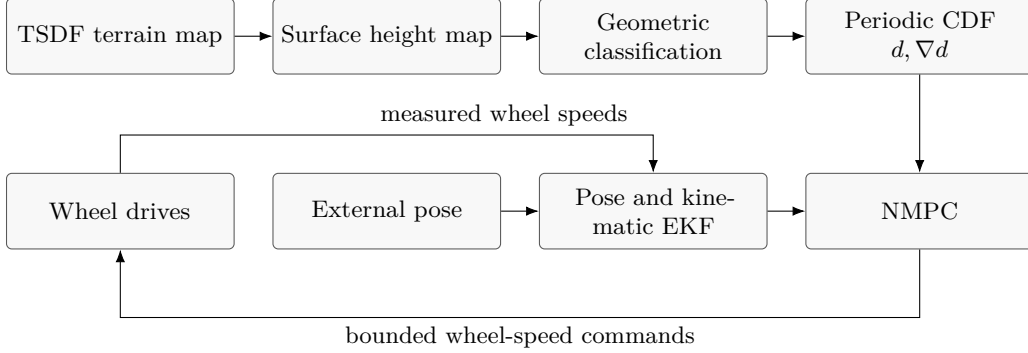


Figure 3: Terrain assessment is compiled into a CDF outside the fast control loop. The estimator uses measured wheel motion and external pose; NMPC executes a bounded command profile through the wheel drives.

3 Vehicle-Specific Terrain Admissibility

The terrain representation answers a configuration-dependent question: can the wheels find locally consistent support while the ground remains below each part of the underside? The evaluation uses the geometry and coordinate convention of Section 2, a fixed manipulator posture, and the thresholds in Table 3. Its output is a scalar field over position and heading that can be queried throughout the prediction horizon.

3.1 From a TSDF to a height map

Let Φ_n and W_n denote the signed distance and weight of voxel n in a vertical TSDF column, with height z_n and spacing Δ . A negative-to-positive crossing between observed adjacent voxels, $\Phi_n < 0 \leq \Phi_{n+1}$, gives the interpolated surface height

$$\eta_n = z_n - \frac{\Phi_n}{\Phi_{n+1} - \Phi_n} \Delta. \quad (3)$$

A crossing is valid when both weights are at least $W_{\min}$. The valid crossing with the lowest vertical position defines the ground level. Within a vertical band of height z_b above it, the extraction identifies the highest valid crossing and counts observed near-surface voxels satisfying $|\Phi_n| < d_s$. If this count reaches n_s , the highest near-surface voxel supplies the terrain height; otherwise the highest valid crossing is used. The voxel-count branch captures observed vertical structures whose upper surfaces lie outside the sensor view and therefore do not produce a valid surface crossing. The resulting height $H(x, y)$ represents the terrain or obstacle surface relevant to the vehicle footprint, rather than solely the lowest ground surface.

The offline extraction evaluated in this paper uses $W_{\min} = 0.25$, $z_b = 3.5$ m, $d_s = 0.15$ m, and $n_s = 6$. Missing columns are assigned the height of the nearest valid column within $r_f = 0.5$ m; more distant columns remain unknown. The deployed field implementation used $r_f = 20.5$ m to accommodate missing height data beneath the vehicle, where the LiDAR cannot directly observe the ground, particularly during mapping startup. Filled heights are extrapolated terrain values rather than additional observations. Height-map cells and configuration positions share the spacing $\Delta = 0.15$ m.

3.2 Wheel patches and support consistency

Configurations $\mathbf{q} = [x, y, \psi]^\top$ are sampled on a position–heading grid $\mathcal{G}$ and assigned binary admissibility labels, with heading step $\Delta\psi = 2\pi/N_\psi$. The onboard implementation uses $N_\psi = 100$; the offline recorded-map

analysis in Section 6 uses $N_\psi = 36$. The positional domain is reduced so that the wheel sampling patches lie inside the map. Write $\mathbf{p} = [x, y]^\top$ and let $R(\psi)$ be the planar rotation from vehicle coordinates (ξ, ζ) to map coordinates. For wheel j , a 5×3 patch contains $n_p = 15$ samples at spacing Δ , with offsets $\ell_{j,i} = [\xi_{j,i}, \zeta_{j,i}]^\top$. Nearest-neighbor lookup provides

$$H_{j,i} = H(\mathbf{p} + R(\psi)\ell_{j,i}), \quad i = 1, \dots, n_p. \quad (4)$$

The wheel order is front left, front right, rear left, and rear right. Every wheel sample must have an assigned height after the stated filling procedure.

A least-squares plane $z = a_j\xi + b_j\zeta + h_j$ is fitted to each patch. Let A_j contain rows $[\xi_{j,i}, \zeta_{j,i}, 1]$ and let $\mathbf{H}_j = [H_{j,1}, \dots, H_{j,n_p}]^\top$. Then

$$[a_j, b_j, h_j]^\top = A_j^\dagger \mathbf{H}_j.$$

Because the sampling coordinates are fixed in the vehicle frame, the pseudoinverse $A_j^\dagger$ is precomputed at initialization. The same construction is used for the common support plane. Only the sampled heights change between configuration evaluations. Its inclination describes the local supporting surface, while its maximum residual

$$\varepsilon_j = \max_i |H_{j,i} - a_j\xi_{j,i} - b_j\zeta_{j,i} - h_j| \quad (5)$$

measures departures from locally planar support. This distinguishes a smooth incline from an edge, depression, or small rock occupying the wheel patch.

A common support plane $\pi_s(\xi, \zeta) = a\xi + b\zeta + h_0$ is then fitted to the $3n_p$ samples of the first three wheels. The fourth patch provides the cross-wheel consistency test

$$\varepsilon_s = \max_i |H_{4,i} - \pi_s(\ell_{4,i})|. \quad (6)$$

Using a fixed wheel triple gives a consistent geometric reference at every grid configuration. The residual tests whether the fourth wheel patch agrees with the support plane fitted to the first three patches. The threshold $\bar{\varepsilon}_s = 0.10$ m permits limited noncoplanarity of the wheel support heights and rejects larger geometric inconsistencies.

The geometric roll and pitch of a plane with slopes (a, b) are defined by

$$\varphi = \text{atan2}(b, 1), \quad \vartheta = \text{atan2}\left(-a, \sqrt{1 + b^2}\right). \quad (7)$$

Here positive roll denotes a higher left side and positive pitch a lower front. Applying the same convention to (a_j, b_j) gives (φ_j, ϑ_j) for each wheel patch. These angles supply the terrain-inclination criteria. These quantities define the geometric inclination criteria used by the terrain classifier; they are distinct from prediction errors relative to measured vehicle attitude.

3.3 Component-specific clearance

The underside model partitions the footprint into the central chassis, two bogie strips, an outer envelope, and the stowed-manipulator footprint (Fig. 4). Each region has a permissible vertical terrain height κ above π_s : 0.20 m for the central chassis, 0.125 m for the bogie strips, and 0.30 m for the envelope and manipulator region. The bogie strips occupy $0.60 \text{ m} \leq |\zeta| \leq 0.90 \text{ m}$. Region membership is evaluated at map-cell centers.

For each observed cell center $\mathbf{g}$ in these regions, its vehicle-frame location is $\ell_g = R(\psi)^\top(\mathbf{g} - \mathbf{p})$, and clearance requires

$$H(\mathbf{g}) - \pi_s(\ell_g) \leq \kappa(\ell_g). \quad (8)$$

A configuration belongs to the admissible set $\mathcal{A}$ when its wheel samples are known, the clearance inequalities hold on observed underside cells, and

$$\begin{aligned} |\varphi_j|, |\vartheta_j| &\leq \bar{\varphi}_w, & \varepsilon_j &\leq \bar{\varepsilon}_w & (j = 1, \dots, 4), \\ \varepsilon_s &\leq \bar{\varepsilon}_s, & \varphi &\in [\varphi_-, \varphi_+], & \vartheta &\in [\vartheta_-, \vartheta_+]. \end{aligned} \quad (9)$$

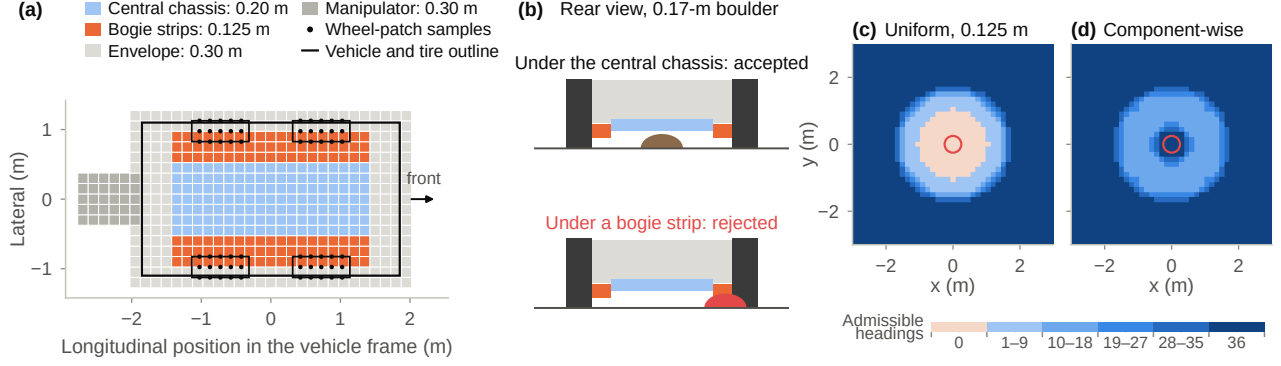


Figure 4: Illustrative vehicle-specific terrain evaluation. (a) Wheel sampling patches and underside regions on the 0.15 m grid. Labels give permissible terrain heights above the support plane. (b) A 0.17 m boulder fits below the central chassis and exceeds the bogie-strip allowance. (c,d) Number of admissible headings with uniform and component-specific clearance.

The rejected set is $\mathcal{B} = \mathcal{G} \setminus \mathcal{A}$. Thus wheel support requires assigned heights throughout each patch, whereas the underside test uses the available observed cells. The local residual limit is 0.05 m, the cross-wheel limit 0.10 m, and the remaining limits are given in Table 3.

This partition gives straddling a direct geometric interpretation. In the illustrative geometric example of Fig. 4, a 0.17 m boulder fits below the central chassis but exceeds the bogie allowance. A lateral or heading adjustment can place it in the central region while keeping the wheel patches clear. The uniform 0.125 m model rejects all headings at 149 nearby grid positions; the component-specific model retains an admissible heading at every position in the illustrated domain. The distinction arises from the vehicle geometry and the boulder’s position relative to it.

3.4 Heading-periodic configuration space distance field

Distance between configurations is measured by

$$\rho(\mathbf{q}, \mathbf{q}')^2 = (x - x')^2 + (y - y')^2 + \lambda^2 \text{wrap}(\psi - \psi')^2, \quad \lambda > 0, \quad (10)$$

where wrap maps angles to $(-\pi, \pi]$ and λ , measured in metres per radian, sets the relative weight of translation and heading. The distance-transform sampling intervals are $(\Delta, \Delta, \lambda\Delta\psi)$. The offline recorded-map analysis in Section 6 uses $N_\psi = 36$ and $\lambda = \Delta/\Delta\psi$, assigning the same distance to one positional step and one heading step. The deployed implementation uses $N_\psi = 100$ and $\lambda = 1 \text{ m/rad}$, giving a scaled heading interval $\lambda\Delta\psi = 0.06283 \text{ m}$. For $\gamma \in \mathcal{G}$, define

$$D(\gamma) = \begin{cases} \text{dist}_\rho(\gamma, \mathcal{B}), & \gamma \in \mathcal{A}, \\ -\text{dist}_\rho(\gamma, \mathcal{A}), & \gamma \in \mathcal{B}. \end{cases} \quad (11)$$

Here dist_ρ denotes distance to the nearest element of a set. Two Euclidean distance transforms (EDTs) produce the positive and negative distances and the corresponding nearest-feature indices. The indices are used to obtain distance-field gradients without finite differencing. The deployed GPU distance transform adapts the PBA+ implementation from CuPy (Okuta et al., 2017), based on the Parallel Banding Algorithm (Cao et al., 2010), with sampling intervals $(\Delta, \Delta, \lambda\Delta\psi)$ to account for the different spatial and heading resolutions. For even N_ψ , let B and T denote the lower and upper halves, respectively, along the heading dimension of the binary configuration volume. The transform is evaluated on the extended sequence $[T, B, T, B]$ to account for heading periodicity, treating configurations near opposite heading boundaries as neighbors. Distance to an empty set is formally infinite; the implementation uses a constant larger than the grid diameter for this special case. The negative branch retains distance variation within rejected regions, allowing the terrain penalty to guide local optimization when a predicted configuration is inadmissible. The controller queries the heading-periodic

trilinear interpolant $d(\mathbf{q})$ of the distance values and a gradient estimate obtained from the nearest-feature indices. The scalar interpolant is continuous and piecewise differentiable; the supplied gradient estimate is used to construct the local terrain model in the optimizer. Position queries outside the grid are clamped to its boundary. A field value therefore summarizes proximity to a classified geometric violation in both translation and heading. Its gradient supplies a local direction of increasing margin, including rotation when that is the effective way to align the vehicle with an admissible passage. The onboard optimal-control solver accesses the distance values and gradients through its generic external-cost interface.

3.5 Interpretation of the grid margin

Define the scaled-cell half diagonal

$$r_g = \frac{1}{2} \sqrt{2\Delta^2 + (\lambda\Delta\psi)^2}.$$

This quantity accounts for positional and heading discretization in the metric (10).

Proposition 1 (Conditional grid margin). *Let $\mathbf{q}^*$ lie within the positional grid domain and satisfy $d(\mathbf{q}^*) \geq m > r_g$. Every rejected grid node is at distance at least $\sqrt{m^2 - r_g^2}$ from $\mathbf{q}^*$. Consequently, an executed pose $\mathbf{q}$ within the domain has an admissible nearest grid node if*

$$\rho(\mathbf{q}, \mathbf{q}^*) \leq \delta < \sqrt{m^2 - r_g^2} - r_g. \quad (12)$$

Proof. Use scaled coordinates $(x, y, \lambda\psi)$ in the interpolation cell of $\mathbf{q}^*$, with corners γ_ν and trilinear weights μ_ν . For any rejected node β , choose its periodic image nearest to $\mathbf{q}^*$. The definition of D and the weighted-variance identity give

$$m^2 \leq \rho(\mathbf{q}^*, \beta)^2 + \sum_{\nu} \mu_{\nu} \|\gamma_{\nu} - \mathbf{q}^*\|^2 \leq \rho(\mathbf{q}^*, \beta)^2 + r_g^2.$$

The first inequality follows by bounding the interpolated signed distance by the weighted distances to that periodic image and applying Jensen’s inequality. If there are no rejected nodes, the conclusion is immediate. The nearest grid node to $\mathbf{q}$ lies within r_g of $\mathbf{q}$; the triangle inequality then proves (12). $\square$

The result concerns the terrain classification at grid nodes. At a planned pose with $d \geq d_a$, the corresponding deviation allowance is $\sqrt{d_a^2 - r_g^2} - r_g$, which is positive when $d_a > \sqrt{2} r_g$.

3.6 Computational cost of terrain processing

The terrain-processing implementation uses C and CUDA C++. The computational cost of the terrain-admissibility pipeline was evaluated using the same recorded TSDF snapshot shown in Fig. 9. The input contains $201 \times 201 \times 101$ voxels at 0.15 m spacing, and configuration classification uses $N_\psi = 100$ headings. The two benchmark systems comprise an NVIDIA RTX 3050 with an Intel Core i9-14900 running Ubuntu 24.04, and an NVIDIA RTX 4090 with an Intel Core i9-14900K running Ubuntu 24.04 under Windows Subsystem for Linux. CPU and GPU buffers are preallocated and reused. After one warm-up run, each system processes the fixed snapshot over 100 measured runs. Table 4 reports mean execution times and standard deviations. Height-map extraction is performed on the CPU. The resulting map is transferred to the GPU, where the configuration-classification kernel evaluates wheel support, support-plane consistency, attitude limits, and component clearance. The binary classification is returned to the CPU for periodic extension along the heading dimension. The extended volume is then uploaded for GPU distance transformation. Two EDTs process the binary classification and its complement: each mask is encoded, transformed using PBA+, and decoded into distances and nearest-feature indices. The results are combined into the signed CDF, and the distance and index arrays are returned to the CPU.

Table 4: Terrain-processing benchmark on the recorded TSDF snapshot. Values are mean $\pm$ standard deviation in milliseconds over 100 measured runs after one warm-up run. H2D and D2H denote host-to-device and device-to-host transfers. Stage subtotals and the complete pipeline are timed independently.

Operation	RTX 3050 system	RTX 4090 system
Height-map extraction (CPU)	4.762 ± 0.029	3.740 ± 0.247
<i>Configuration classification</i>		
H2D: height map	0.039 ± 0.001	0.059 ± 0.023
Classification kernel	20.543 ± 0.942	2.590 ± 0.020
D2H: binary classification	0.678 ± 0.004	0.457 ± 0.060
CUDA-stage subtotal	21.296 ± 0.989	3.222 ± 0.140
Periodic volume extension (CPU)	0.677 ± 0.011	0.548 ± 0.027
<i>Signed distance transformation</i>		
H2D: extended binary volume	1.285 ± 0.028	0.694 ± 0.022
Binary inversion	0.217 ± 0.040	0.068 ± 0.018
Encode: first EDT	0.310 ± 0.013	0.042 ± 0.001
PBA+: first EDT	12.675 ± 0.704	1.831 ± 0.007
Decode: first EDT	0.517 ± 0.022	0.097 ± 0.002
Encode: second EDT	0.311 ± 0.015	0.068 ± 0.002
PBA+: second EDT	14.201 ± 0.761	1.948 ± 0.009
Decode: second EDT	0.517 ± 0.021	0.097 ± 0.002
Signed-distance combination	0.453 ± 0.011	0.117 ± 0.002
D2H: signed distances	4.786 ± 0.005	2.503 ± 0.025
D2H: nearest-feature indices	4.784 ± 0.003	2.485 ± 0.018
CUDA-stage subtotal	40.209 ± 1.723	10.345 ± 0.212
Complete pipeline	66.957 ± 2.662	17.950 ± 0.485

The mean pipeline time differs by a factor of 3.73 between the tested systems. On the RTX 3050 system, the classification kernel and the two PBA+ passes together account for approximately 70.8% of the complete pipeline time. On the RTX 4090 system, the listed CPU–GPU transfers account for approximately 34.5%, compared with 17.3% on the RTX 3050 system. The measured stage subtotals need not equal the sums of their individual operations because their timing intervals also include unlisted overhead. Periodic extension is reported separately here; it is already included in the outer signed-distance-stage timer. These results compare the two complete systems, whose processors and execution environments also differ. The present measurements group wheel-patch sampling, plane fitting, support-consistency tests, attitude checks, and underside-clearance evaluation within one classification kernel. Their individual execution times are therefore not identified by this benchmark. A finer profile would measure these operations separately and report the number of configurations reaching each test, because early rejection changes the workload of subsequent checks. Possible reductions in transfer overhead include performing periodic extension on the GPU and evaluating whether the controller can reconstruct distances from nearest-feature indices. The latter requires the classification sign, grid spacing, and periodic heading metric to remain available. The execution-time benefit of either change requires a separate benchmark.

4 Predictive Control and Execution

The control layer uses the latest CDF to adjust a reference motion while accounting for the effective steering behavior of the machine. The estimator identifies that behavior from measured motion and wheel rotation. The optimizer predicts the commanded motion, and the command interface limits successive changes sent to the wheel drives.

4.1 Online effective-kinematics estimation

The kinematic model in (2) has parameters $\boldsymbol{\theta} = [X_v, c, \alpha]^\top$, with $c > 0$. An augmented extended Kalman filter (EKF) estimates $\mathbf{z} = [\mathbf{q}^\top, \boldsymbol{\theta}^\top]^\top$, using the measured left- and right-side wheel speeds as its two inputs. In the outdoor experiments, these inputs are the front-wheel measurements. The pose follows the kinematic model, and the effective parameters follow random walks. With estimator interval T_e , write the augmented continuous model as $f_e(\mathbf{z}, \boldsymbol{\omega}_m) = [f_q(\mathbf{q}, \boldsymbol{\omega}_m; \boldsymbol{\theta})^\top, \mathbf{0}^\top]^\top$, where f_q is the right-hand side of (2). Prediction uses

$$\begin{aligned}\hat{\mathbf{z}}^- &= \hat{\mathbf{z}} + T_e f_e(\hat{\mathbf{z}}, \boldsymbol{\omega}_m), \\ P^- &= A P A^\top + Q + B R_u B^\top, \\ A &= I + T_e \frac{\partial f_e}{\partial \mathbf{z}}, \quad B = T_e \frac{\partial f_e}{\partial \boldsymbol{\omega}_m}.\end{aligned}\tag{13}$$

A fresh pose observation supplies the correction with measurement matrix $C = [I_3 \ 0]$, covariance R , and gain $K = P^- C^\top (C P^- C^\top + R)^{-1}$. The innovation is the position difference and the wrapped heading difference. The state and covariance updates are $\hat{\mathbf{z}}^+ = \hat{\mathbf{z}}^- + K \boldsymbol{\nu}$ and $P^+ = (I - KC)P^-$. The scale estimate α is clipped to $[0.5, 1.5]$.

The implemented estimator uses the covariance settings

$$\begin{aligned}P_0 &= 10^{-3} I_6, \quad R = 10^{-3} I_3, \quad R_u = 10^{-4} I_2, \\ Q &= \text{diag}(10^{-6}, 10^{-6}, 10^{-6}, 10^{-6}, 10^{-6}, 10^{-8}).\end{aligned}\tag{14}$$

Entries refer to the stated state and input coordinates in SI units. In particular, Q is added per prediction step. The field-data replay uses $T_e = 1$ ms; Section 6.3 reports agreement with the recorded estimates at this setting. The initial pose is the first available pose observation, and the nominal parameter initialization is $(X_v, c, \alpha) = (0, 0.976, 1)$ with $r = 0.411$ m. The second field session starts from $(X_v, c, \alpha r) = (-0.35, 1.30, 0.34)$ m, matching the logged initialization.

The estimated quantities describe the mapping from measured wheel rotation to vehicle motion. Turning supplies information about X_v and c , and translational excitation helps separate the wheel-radius scale from the turning parameters Zuo et al. (2024). This interpretation also keeps the kinematic calibration distinct from the measured command-to-wheel response evaluated in Section 6.4.

4.2 Terrain-aware optimal control problem

Let $\boldsymbol{\chi} = [\mathbf{q}^\top, \boldsymbol{\omega}^\top]^\top$ contain the pose and the predicted wheel-speed commands, and let $\mathbf{u}$ be their rate of change. At each control update, the current parameter estimate is held along the prediction horizon. Forward Euler integration gives

$$\mathbf{q}_{i+1} = \mathbf{q}_i + h f_q(\mathbf{q}_i, \boldsymbol{\omega}_i; \hat{\boldsymbol{\theta}}), \quad \boldsymbol{\omega}_{i+1} = \boldsymbol{\omega}_i + h \mathbf{u}_i,\tag{15}$$

abbreviated as $\boldsymbol{\chi}_{i+1} = f_d(\boldsymbol{\chi}_i, \mathbf{u}_i; \hat{\boldsymbol{\theta}})$. The onboard controller uses $N = 200$ intervals of $h = 0.25$ s, giving a 50 s prediction horizon. For nominal straight motion with $\alpha = 1$, the wheel-speed limit corresponds to $r\omega_{\max} = 0.3288$ m/s, or 16.44 m over the horizon and 0.0822 m per prediction interval.

For a reference $(\mathbf{p}_i^{\text{ref}}, \psi_i^{\text{ref}}, \boldsymbol{\omega}_i^{\text{ref}})$, define

$$\begin{aligned}L_i &= w_p \|\mathbf{p}_i - \mathbf{p}_i^{\text{ref}}\|^2 + w_\psi (\psi_i - \psi_i^{\text{ref}})^2 + w_\omega \|\boldsymbol{\omega}_i - \boldsymbol{\omega}_i^{\text{ref}}\|^2 \\ &\quad + w_u \|\mathbf{u}_i\|^2 + w_c [d_a - d(\mathbf{q}_i)]_+^2, \quad [s]_+ = \max(0, s).\end{aligned}\tag{16}$$

All weights are nonnegative; the reported onboard values are in Table 3. Pose and reference headings are continuously unwrapped onto a common branch before forming their difference. The terrain term vanishes at margins of at least d_a and increases quadratically below that activation distance. Thus the optimizer can trade tracking, smooth commands, and geometric margin throughout the horizon.

The terminal cost L_N includes the position, heading, wheel-speed, and CDF state penalties evaluated at $i = N$, and omits the wheel-acceleration term. With the current command $\boldsymbol{\omega}_k^{\text{cmd}}$ and estimated pose $\hat{\mathbf{q}}_k$, the optimization is

$$\begin{aligned} \min_{\boldsymbol{\chi}_{0:N}, \mathbf{u}_{0:N-1}} \quad & h \sum_{i=0}^{N-1} L_i + L_N \\ \text{subject to} \quad & \boldsymbol{\chi}_0 = [\hat{\mathbf{q}}_k^\top, (\boldsymbol{\omega}_k^{\text{cmd}})^\top]^\top, \\ & \boldsymbol{\chi}_{i+1} = f_d(\boldsymbol{\chi}_i, \mathbf{u}_i; \hat{\boldsymbol{\theta}}), \quad i = 0, \dots, N-1, \\ & |\boldsymbol{\omega}_i| \leq \omega_{\max}, \quad i = 0, \dots, N, \\ & |\mathbf{u}_i| \leq u_{\max}, \quad i = 0, \dots, N-1. \end{aligned} \tag{17}$$

The inequalities apply componentwise, with $\omega_{\max} = 0.8 \text{ rad/s}$ and $u_{\max} = 0.3 \text{ rad/s}^2$. The onboard formulation uses $\boldsymbol{\omega}_i^{\text{ref}} = \mathbf{0}$, giving a wheel-speed damping term.

4.3 Field queries, optimization, and wheel commands

The controller uses sequential quadratic programming with a limited iteration budget. The reported onboard settings permit at most five SQP iterations and five iterations per QP solve. The solver is generated with *acados* and *CasADi* (Verschuere et al., 2022; Andersson et al., 2019), using partial condensing and HPIPM (Frison and Diehl, 2020). The onboard solver uses the generic external-cost interface. The external routine supplies the cost value, gradient, and Hessian approximation, including the CDF contribution and the analytic derivatives of the remaining quadratic terms. At the previous iterate's poses $\bar{\mathbf{q}}_i$, the controller queries $d_i = d(\bar{\mathbf{q}}_i)$ and the nearest-feature-based gradient estimate $\mathbf{g}_i$. The local representation used in an iteration is

$$d(\mathbf{q}_i) \simeq d_i + \mathbf{g}_i^\top (\mathbf{q}_i - \bar{\mathbf{q}}_i). \tag{18}$$

For the local model (18), write $\tilde{d}_i = d_i + \mathbf{g}_i^\top (\mathbf{q}_i - \bar{\mathbf{q}}_i)$. The terrain-cost gradient is $-2w_c[d_a - \tilde{d}_i]_+ \mathbf{g}_i$, and its Gauss–Newton Hessian away from the activation threshold is

$$H_{c,i}^{\text{GN}} = \begin{cases} 2w_c \mathbf{g}_i \mathbf{g}_i^\top, & \tilde{d}_i < d_a, \\ \mathbf{0}, & \tilde{d}_i > d_a. \end{cases}$$

At $\tilde{d}_i = d_a$, the gradient is zero, while the classical Hessian generally does not exist. Value and gradient queries carry the changing terrain geometry into a fixed generated solver. The preceding solution provides a warm start for the next solve. This reduces repeated optimization effort, although changes in the terrain field can make the inherited trajectory unsuitable. Because the CDF enters the objective as a soft penalty, the formulation does not impose collision avoidance as a hard constraint. The onboard implementation updates the command by interpolating toward the first predicted wheel state. With elapsed interval δt_k , define

$$\beta_k = \min\left(\frac{\delta t_k}{h}, 1\right), \quad \boldsymbol{\omega}_{k+1}^{\text{cmd}} = (1 - \beta_k) \boldsymbol{\omega}_k^{\text{cmd}} + \beta_k \boldsymbol{\omega}_{1|k}^*. \tag{19}$$

For $\delta t_k \geq 0$ and a feasible solution of (17), $\boldsymbol{\omega}_{1|k}^* = \boldsymbol{\omega}_k^{\text{cmd}} + h \mathbf{u}_{0|k}^*$. Hence the interpolation preserves

$$\|\boldsymbol{\omega}_{k+1}^{\text{cmd}}\|_\infty \leq \omega_{\max}, \quad \|\boldsymbol{\omega}_{k+1}^{\text{cmd}} - \boldsymbol{\omega}_k^{\text{cmd}}\|_\infty \leq \min(\delta t_k, h) u_{\max} \leq \delta t_k u_{\max}.$$

These bounds concern the commanded wheel speeds. When no fresh SLAM pose is available, the control pose is advanced by the analogous interpolation

$$\hat{\mathbf{q}}_{k+1} = (1 - \beta_k) \hat{\mathbf{q}}_k + \beta_k \mathbf{q}_{1|k}^*, \tag{20}$$

using unwrapped heading coordinates.

4.4 Terrain-cost tuning and margin selection

Two single-stage balances interpret the tradeoff between reference tracking and terrain margin. Assume $w_p > 0$ and $w_c > 0$ for these balances. With an active terrain penalty, planar error $\mathbf{e} = \mathbf{p} - \mathbf{p}^{\text{ref}}$, and nonzero $\nabla_p d$, stationarity of the position and terrain terms gives $w_p \mathbf{e} = w_c (d_a - d) \nabla_p d$. Its magnitude yields

$$d \simeq d_a - \frac{w_p}{w_c} \frac{\|\mathbf{e}\|}{\|\nabla_p d\|}. \quad (21)$$

This estimate describes a stationary pose dominated by these two terms. For a passage with maximum available margin $d_p < d_a$, its terrain cost equals the position cost at the indicative error level

$$e^* = \sqrt{\frac{w_c}{w_p}} (d_a - d_p). \quad (22)$$

The balance indicates how a larger terrain weight or activation distance can increase the reference deviation associated with entering a narrow admissible passage. These local tuning interpretations motivate the activation-distance comparison in Section 5.

4.5 Implementation sequence

The terrain, estimation, and control operations run at their respective update rates. Algorithm 1 specifies the data exchanged between them. Each control update queries the current published field; the observed map and command rates are reported in the field experiments.

Implementation algorithm: CDF-guided predictive execution

Inputs: vehicle geometry and thresholds; TSDF or height map; timestamped pose and wheel-speed observations; reference trajectory; prediction and command intervals.

Initialization: set the configuration grid, EKF state and covariances, initial command $\omega_0^{\text{cmd}} = \mathbf{0}$, and NMPC trajectory guess.

1. **Terrain update:** extract and locally fill H ; evaluate the four wheel patches, common support plane, attitude, and component clearance at every grid configuration; form $\mathcal{A}$ and $\mathcal{B}$; compute the periodic signed distance transform; publish D , its nearest-feature indices, and its grid coordinates.
 2. **Estimator prediction:** select the measured left- and right-side wheel-speed inputs (front-wheel measurements onboard) and apply (13). On a fresh pose observation, apply the wrapped-innovation correction and scale clipping.
 3. **Control update:** obtain the current pose and effective parameters, the latest field, and ω_k^{cmd} . Sample the reference horizon with continuous yaw.
 4. Set the initial optimization state to $[\hat{\mathbf{q}}_k; \omega_k^{\text{cmd}}]$. For each configured iteration, query the field value and gradient along the preceding iterate, form (18), and solve the resulting QP subproblem.
 5. Update the onboard command using (19) and send the two side commands to the wheel-drive interfaces. When no fresh SLAM pose is available, advance the control pose using (20). Retain the solution for the next update.
 6. Record pose, effective kinematic parameters, terrain-based roll and pitch estimates evaluated at the current estimated pose, commands, measured wheel speeds, and update timing; continue the three update processes.
-

Algorithm 1: Implementation algorithm. The estimator uses measured wheel motion; the optimizer and output interface propagate the commanded wheel motion.

5 Simulation evaluation

5.1 Simulator and recorded-data protocol

The simulation evaluation uses recordings of the skid-steer robot operating in MuJoCo. The full controller variants compute and query the CDF online during simulation. The simulator couples rigid-body contact dynamics in MuJoCo (Todorov et al., 2012) to wheel-speed PI control and actuator submodels for articulated mechanisms. Table 5 describes the archived simulator snapshot. The nominal physical-platform mass in Table 3 and the summed robot-body mass in the simulator are stated separately.

Table 5: Settings verified in the simulator model examined in this study. These describe the examined model and do not establish the configuration of every logged run.

Quantity	Snapshot value
MuJoCo version	3.8.0
Integration step; integrator	1 ms; implicitfast
Contact solver; maximum iterations; tolerance	Newton; 100; 10^{-8}
Summed robot-body mass	5873.34 kg
Wheel collision-cylinder radius; half-width	0.411 m; 0.16 m
Wheel collision contact dimension; friction tuple	3; (1, 0.005, 0.0001)
Suspension stiffness; damping	2×10^6 N/m; 5×10^4 N s/m
Wheel motor force range (unit transmission ratio)	$\pm 80\,000$ N m

Table 6: Recorded straight-course results over the first 300 s. e_{last} is the error at the final included frame, not an arrival or collision criterion. Roll and pitch columns report maximum absolute body angles. Values are descriptive statistics of one take per row.

Course	Variant	e_{RMS} (m)	e_{max} (m)	e_{last} (m)	$ \phi _{\text{max}}$ (deg)	$ \theta _{\text{max}}$ (deg)
Simple	Full	0.875	2.376	0.087	4.273	5.078
	No terrain cost	3.214	11.171	11.171	18.717	23.416
	Attitude only	3.217	11.170	11.170	13.915	16.799
Rocky	Full	0.723	2.308	0.140	3.697	5.614
	No terrain cost	4.690	14.081	14.081	9.027	31.851
	No wheel checks	0.602	1.946	0.134	2.916	8.585

Table 7: One-period circular tracking results for the activation-distance settings specified in the analysis script. Each row is one recording.

d_a label (m)	e_{RMS} (m)	e_{max} (m)	e_{last} (m)	$ \phi _{\text{max}}$ (deg)	$ \theta _{\text{max}}$ (deg)
0.2	0.381	0.876	0.160	4.644	6.214
0.6	0.772	1.683	0.157	4.614	6.197
1.0	1.087	2.034	0.162	5.166	6.208

The reported simulation analysis uses six straight-course recordings, three single-circle recordings with different activation distances, one continuous multi-lap recording, and one uneven-terrain recording for attitude-estimation evaluation. The variant names below follow the dataset names and plotting script. Table 3 reports the onboard controller settings. The simulation comparisons describe the recorded trajectory and attitude outcomes.

The recordings provide wheel-speed measurements, side commands, and vehicle poses. Startup samples are discarded, and elapsed time is measured from the first valid timestamp. Planar trajectories are expressed relative to their initial position and heading, without further trajectory registration. Body roll and pitch are extracted from the pose rotation relative to world vertical.

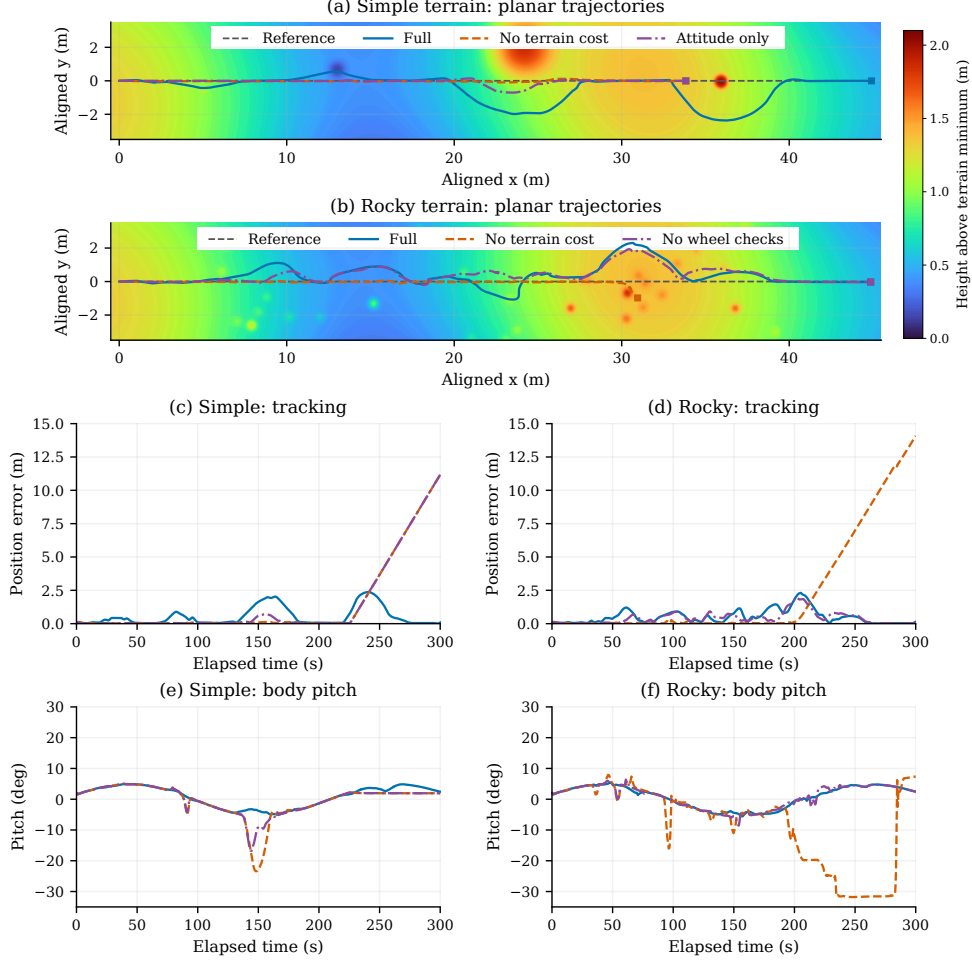


Figure 5: Recorded simple- and rocky-course comparisons. The top panels show aligned planar trajectories with equal spatial axis scales; squares mark the last included positions. The lower panels show time-indexed position error and world-frame body pitch over the common 300-s window. The reduced variant is attitude only on the simple course and no wheel checks on the rocky course. Background shading shows elevation from the saved simulator terrain assets, relative to each terrain’s minimum, aligned with the full variant’s initial-pose frame

The nominal references are reconstructed from the plotting script using elapsed time:

$$\mathbf{p}_{\text{line}}^{\text{ref}}(\tau) = [0.15 \min(\tau, 300), 0]^\top \text{ m}, \quad (23)$$

$$\mathbf{p}_{\text{circle}}^{\text{ref}}(\tau) = [5 \sin(2\pi f \tau), 5\{\cos(2\pi f \tau) - 1\}]^\top \text{ m}, \quad f = 0.007 \text{ Hz}. \quad (24)$$

The straight-course tracking comparisons use $0 \leq \tau \leq 300$ s, corresponding to a nominal 45-m traversal. Single-circle comparisons use one period $T = 1/f \approx 142.857$ s; the multi-lap analysis uses $0 \leq \tau < 11T$. These common windows omit the unequal trailing portions of the recordings. Tracking error is the time-indexed planar deviation $e_i = \|\mathbf{p}_i - \mathbf{p}_i^{\text{ref}}\|_2$, not the nearest distance to the reference curve. Reported statistics use every retained frame in the specified window:

$$e_{\text{RMS}} = \sqrt{\frac{1}{n} \sum_{i=1}^n e_i^2}, \quad e_{\text{max}} = \max_i e_i. \quad (25)$$

Repeated timestamps are retained because the pose continues to change between external updates. Thus, these are sample-weighted statistics of logged simulator states. The distinct external timestamps indicate an observed controller–simulator exchange rate of approximately 144–151 Hz over the active recordings.

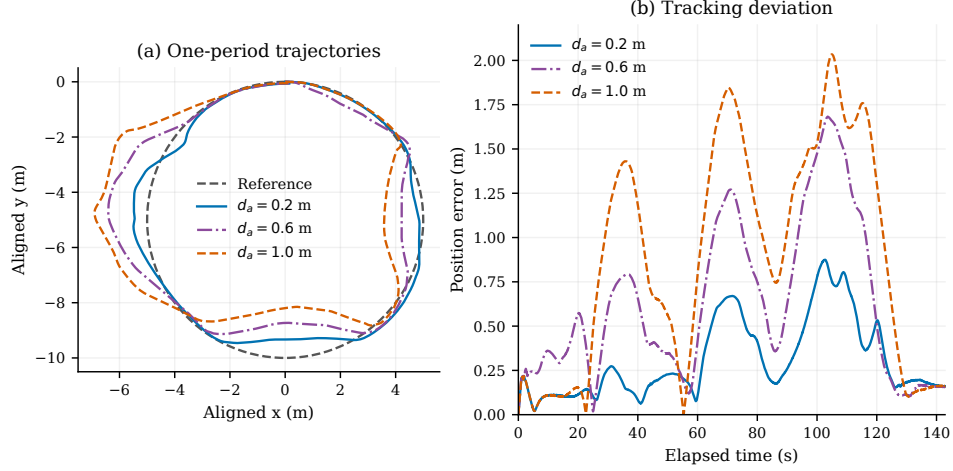


Figure 6: Circular trajectories and time-indexed position errors for the three activation-distance settings over one nominal period. The reference has radius 5 m and frequency 0.007 Hz.

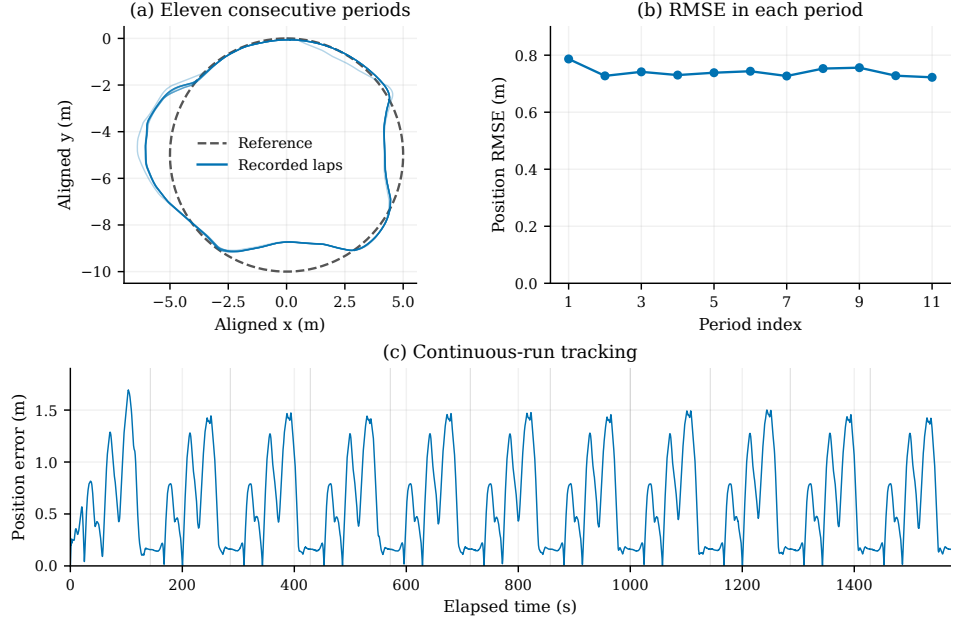


Figure 7: Eleven nominal reference periods from the continuous multi-lap recording: aligned trajectories, position RMSE by period, and the full position-error history. Vertical lines mark nominal period boundaries. All periods use the same initial alignment; no per-period registration is applied.

The archive provides trajectory and attitude evidence from one take per variant. It does not contain contact-force histories, executed CDF values, or solver-status histories, so collision frequency, achieved clearance, and satisfaction of the conditional margin in Proposition 1 are not inferred from these recordings. The comparisons below describe the recorded outcomes without estimating variation across independent trials.

5.2 Simple- and rocky-terrain comparisons

Table 6 and Figure 5 compare the full, no-terrain-cost, and reduced variants on the simple and rocky courses. During the common 300s evaluation window, the full variants make greater longitudinal progress than the

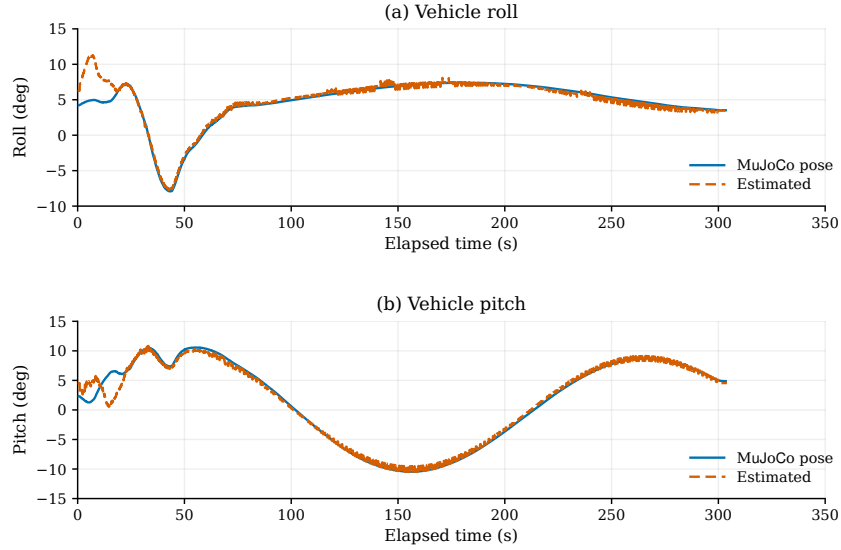


Figure 8: Attitude-estimation evaluation on uneven terrain. Terrain-based roll (top) and pitch (bottom) estimates evaluated at the current vehicle configuration are compared with the corresponding body angles extracted from the MuJoCo pose. Solid blue curves show the pose-derived angles; dashed orange curves show the terrain-based estimates.

variants without terrain cost. At the final included sample, their deviations from the nominal reference are 0.087 m and 0.140 m on the simple and rocky courses, respectively, compared with 11.171 m and 14.081 m for the variants without terrain cost. The tracking and body-angle statistics provide complementary descriptions of these recorded outcomes.

The attitude-only variant on the simple course reaches approximately the same final longitudinal position as the no-terrain-cost variant. Its peak absolute pitch is lower, 16.799° versus 23.416° , but both retain approximately 11.17 m final tracking error. On the rocky course, the variant labeled “no wheel checks” has lower tracking RMSE than the full variant (0.602 m versus 0.723 m) and higher peak absolute pitch (8.585° versus 5.614°). The full variant therefore does not minimize every reported tracking statistic; the recordings expose a tradeoff between reference following and body attitude.

5.3 Activation-distance comparison

The circular recordings compare the activation-distance settings $d_a = 0.2$ m, 0.6 m, and 1.0 m. Over one nominal period, tracking RMSE increases from 0.381 m to 0.772 m and 1.087 m, and maximum tracking error increases from 0.876 m to 1.683 m and 2.034 m (Table 7). Figure 6 shows the associated path deviations. Peak absolute pitch remains near 6.2° in all three recordings. These observations are consistent with a stronger departure from the nominal reference at larger activation distances, but do not quantify the associated obstacle clearance because the executed field values and contact histories are unavailable.

5.4 Continuous multi-lap tracking and attitude estimation

The continuous multi-lap dataset covers 11 complete nominal reference periods in the evaluation window, totaling approximately 1571.429 s. Its aggregate tracking RMSE is 0.742 m, with maximum error 1.694 m. The RMSE of individual periods ranges from 0.722 m to 0.787 m; the first and eleventh periods have RMSEs of 0.787 m and 0.722 m, respectively (Figure 7). The overlapping trajectories and period-wise errors show sustained repeated traversal within this continuous recording. These periods are successive portions of one run and are not independent experimental repetitions.

The uneven-terrain test evaluates the terrain-based attitude estimate at the vehicle’s current configuration. The estimated roll and pitch are obtained from the height-field support plane using (7) and compared with the body attitude extracted from the MuJoCo pose (Fig. 8). This comparison assesses agreement between the geometric support model and simulated vehicle attitude during traversal. The larger initial discrepancies are consistent with incomplete terrain observations beneath the vehicle, where missing heights are assigned from the nearest valid columns within the filling radius r_f .

6 Outdoor Experiments on the Skid-Steer Robot

6.1 Recorded trials and evaluation protocol

Nine outdoor runs were recorded on September 11 and 15, 2026. Seven contain the pose data required for motion evaluation: two circular runs with nominal (K1) and online-estimated (K2) kinematics, followed by three straight runs (S1–S3) and two circular runs (C1, C2) with online estimation. The two additional recordings contribute timing data. The circular reference has a 5-m radius and a nominal speed of approximately 0.11 m/s; the straight reference speed is 0.15 m/s. Time-indexed reference trajectories are reconstructed using the original recording-analysis conventions. Tracking errors below are therefore measured relative to these reference curves.

The 500-Hz recordings contain wheel commands and measurements, pose, effective kinematic parameters, attitude-estimate channels, pressure measurements, and GNSS data. Analysis ends at the final wheel motion in each run; this removes the terminal stationary intervals of 6.0 s, 2.0 s, and 1.3 s from S1–S3, respectively. The observed command and pose rates are 147–153 Hz and 29.8 Hz. Motion errors are evaluated using the recorded SLAM poses and the reconstructed reference trajectories. Table 8 summarizes all seven pose-equipped runs.

Table 8: Outdoor motion evaluation. Position and heading errors use the reconstructed time-indexed references. Command saturation denotes time within 0.001 rad/s of the bound; reverse denotes a side command below -0.01 rad/s.

Run	Reference	Duration (s)	Position error (m)		Heading RMS (°)	Saturation (%)	Reverse (%)
			RMS	Maximum			
K1	Circle	294	0.119	0.21	5.1	0	0
K2	Circle	290	0.077	0.22	4.6	0	0
S1	Straight	239	0.807	1.81	21.8	33	17
S2	Straight	239	0.595	1.58	18.9	19	11
S3	Straight	214	0.610	1.29	18.6	26	17
C1	Circle	301	0.662	1.77	22.4	14	20
C2	Circle	302	0.830	2.43	22.0	18	12

6.2 Terrain representation from a recorded map

The terrain pipeline was also evaluated offline on a recorded nvblox snapshot containing $201 \times 201 \times 101$ voxels at 0.15-m spacing (Fig. 9). Surface extraction recovers 47.8% of the horizontal columns. The extracted heights match the original routine exactly at all 19,299 observed columns. Local filling adds 3,451 cells; 264 have different choices among equidistant nearest neighbors in the two implementations. This observed-column comparison checks the surface-extraction stage on the recorded map.

Of 552 914 configurations with valid heights assigned to every wheel sample after local filling, 54.1% satisfy the complete classification, compared with 94.6% for attitude limits alone. Clearance, wheel-patch geometry, and cross-wheel consistency each reject 32–42% of these supported configurations. Among positions with valid wheel samples for every heading that admit at least one heading, 44% admit only a subset. This recorded scene demonstrates why the terrain cost depends on orientation as well as position. The component partition recovers 761 configurations relative to a uniform 0.125-m threshold in this snapshot.

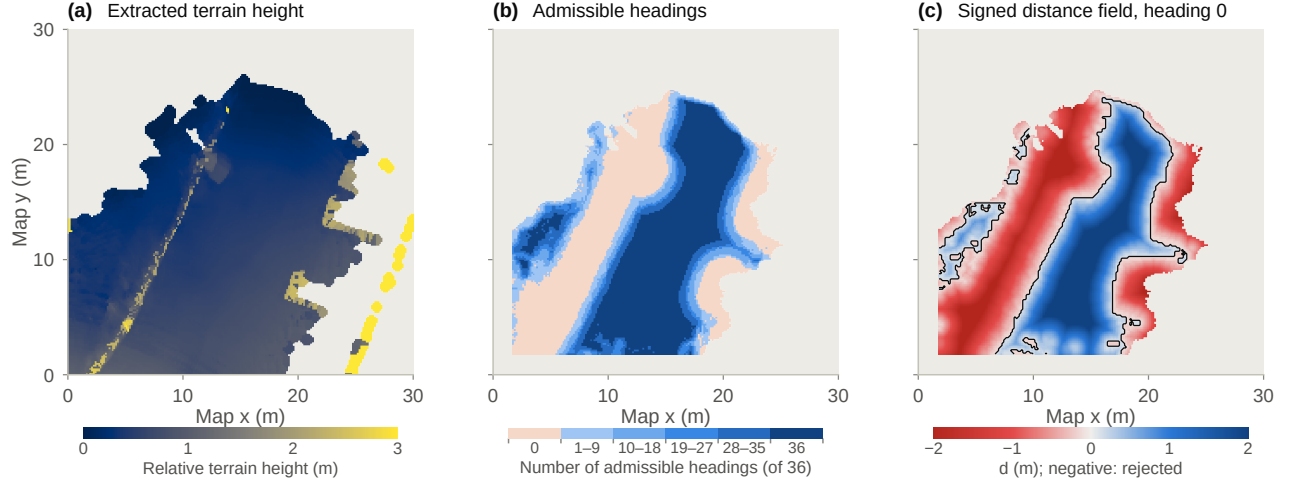


Figure 9: Configuration-space terrain assessment from a recorded TSDF snapshot: extracted height, number of admissible headings, and a signed CDF slice at zero heading. Gray regions in the height map indicate unavailable heights after local filling. The elongated structure has approximately 1.5 m median local relief.

6.3 Effective kinematics and prediction accuracy

Replaying the implemented EKF with recorded wheel speeds and poses reproduces the evolution of the six estimator-active runs. At the evaluated 1-kHz prediction setting, the RMS differences after the first 20 s are at most 0.003 m in X_v , 0.012 m in c , and 0.003 m in αr (Fig. 10a). This replay rate is the reconstruction setting used for comparison with the recorded estimates.

K2 starts from nominal geometry and ends at $(X_v, c, \alpha r) = (-0.349, 1.335, 0.344)$ m. The second-session runs start from $(-0.35, 1.30, 0.34)$ m and maintain effective turning centers 0.30–0.63 m behind the wheel-rectangle center and effective tracks 13–40% wider than the geometric track. The wheel-speed channel is converted using 0.3556 m, whereas the controller’s nominal radius is 0.411 m; the estimated αr consequently combines geometric calibration and the signal convention.

Prediction accuracy is evaluated by initializing model (2) from a recorded pose every second after the first 20 s, retaining starts with at least 20 s remaining, and integrating the subsequent measured wheel speeds for 10 s. Four parameter choices separate nominal geometry, wheel-signal scaling, fixed calibration, and online calibration (Table 9). The online estimates reduce mean position error by 63–73% relative to nominal geometry. Correcting the wheel-signal radius accounts for at least 58% of that reduction in each run. Relative to the fixed calibrated set, online estimates yield 4–23% smaller position errors in S1–C2, with comparable heading errors. Thus, the strongest measured effect is obtaining an effective calibration, with further position-prediction refinement during the second session. K1 and K2 have respective closed-loop tracking RMS values of 0.119 m and 0.077 m; Table 9 complements that descriptive pair with a within-recording model comparison.

6.4 Wheel-command response

The response from commanded to measured wheel speed characterizes the execution dynamics of the onboard system. The recorded commands have a maximum magnitude of 0.800 rad/s. The 99th-percentile slew rates computed from the recordings range from 0.30–0.32 rad/s²; the upper end exceeds the configured command slew-rate bound of 0.30 rad/s². First-order-plus-dead-time fits between commanded and measured wheel speeds on the two sides of the five second-session runs give fitted dead times of 0.21–0.28 s and gains of 0.95–1.06 in nine of ten fits. Mean speed deficits are 0.005–0.034 rad/s.

Starts and reversals exhibit longer response delays. Across 68 starts from rest, motion begins a median 1.1 s

Table 9: Ten-second prediction from measured wheel speeds. Each pair gives mean position error (m) and RMS heading error ($^{\circ}$). Parameters are fixed over each prediction window. The fixed set is $(X_v, c, \alpha r) = (-0.35, 1.30, 0.34)$ m.

Run	Nominal		Radius corrected		Fixed calibrated		Online estimates	
	m	$^{\circ}$	m	$^{\circ}$	m	$^{\circ}$	m	$^{\circ}$
K1	.206	8.2	.050	5.4	.029	.4	—	—
K2	.199	8.2	.046	5.4	.035	.4	.054	1.9
S1	.379	18.9	.179	12.4	.148	3.2	.142	3.7
S2	.388	13.9	.176	9.0	.138	2.6	.126	3.0
S3	.392	15.0	.180	9.6	.134	2.4	.108	2.6
C1	.274	16.2	.160	10.3	.101	2.1	.078	2.1
C2	.296	16.7	.145	10.9	.094	2.4	.085	2.3

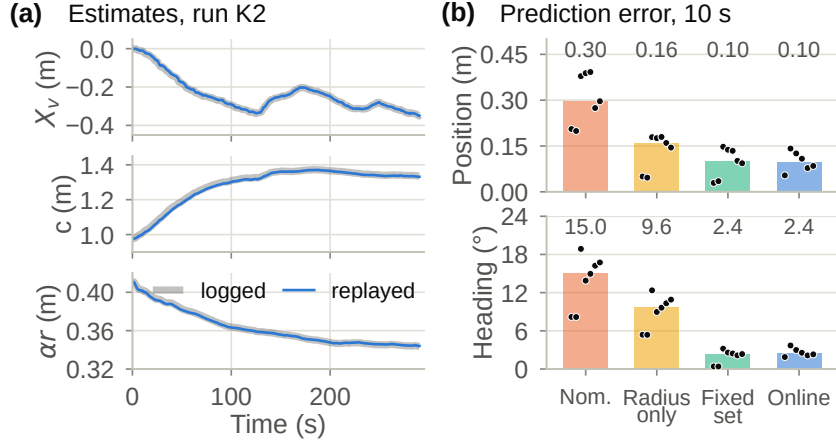


Figure 10: Effective kinematic calibration. (a) Recorded and replayed EKF estimates for K2. (b) Ten-second position and heading prediction errors for the four parameter sets in Table 9; bars show medians across runs and dots show individual runs.

after the command most recently exceeds 0.03 rad/s, with 10th–90th percentiles of 0.7–1.9 s. The command sampled 0.2 s before measured onset has median magnitude 0.27 rad/s. Of 49 commanded reversals, 33 produce a measured reversal, with median delay 1.5 s; the other 16 are shallow command excursions. Figure 11c shows representative pivot maneuvers. These event statistics characterize command-to-motion delays during starts and reversals.

Prediction driven by commanded speeds includes these response effects. Using the recorded online parameter estimates, mean position error is 0.04–0.06 m at 2 s, 0.17–0.26 m at 10 s, and 0.27–0.52 m at 20 s. Comparing these values with the measured-wheel predictions in Table 9 distinguishes prediction using commanded speeds from prediction using measured wheel motion.

6.5 Outdoor trajectories and attitude signals

The September 15 runs include differential wheel commands large enough to pivot the robot and depart from the nominal reference (Fig. 11). Commands reach their bounds for 14–33% of each run and command one side backward for 11–20%. Maximum position deviations range from 1.3 m to 2.4 m.

The straight runs exhibit commanded departures from their time-indexed references. Around the analyzed turns, the initiating command difference exceeds 0.5 rad/s when the position error is only 0.07–0.26 m. Successive poses around those turns change by at most 15 mm and 0.4° . These observations link the departures to commanded steering rather than abrupt changes in the recorded pose.

The decoded attitude-estimate channels follow the variations of the SLAM attitude, which remains within approximately $\pm 6^\circ$ on this site. Roll correlations are 0.88–0.96; pitch correlations are 0.88–0.97 in the five runs with pitch standard deviation above 1° . Mean estimate-minus-SLAM offsets are -0.5 – -0.4° in roll and -1.3 – -0.7° in pitch. After removing each run’s mean, the RMS differences are 0.5–1.0°. Reporting the offsets alongside the centered errors keeps absolute agreement and agreement in attitude variation distinct.

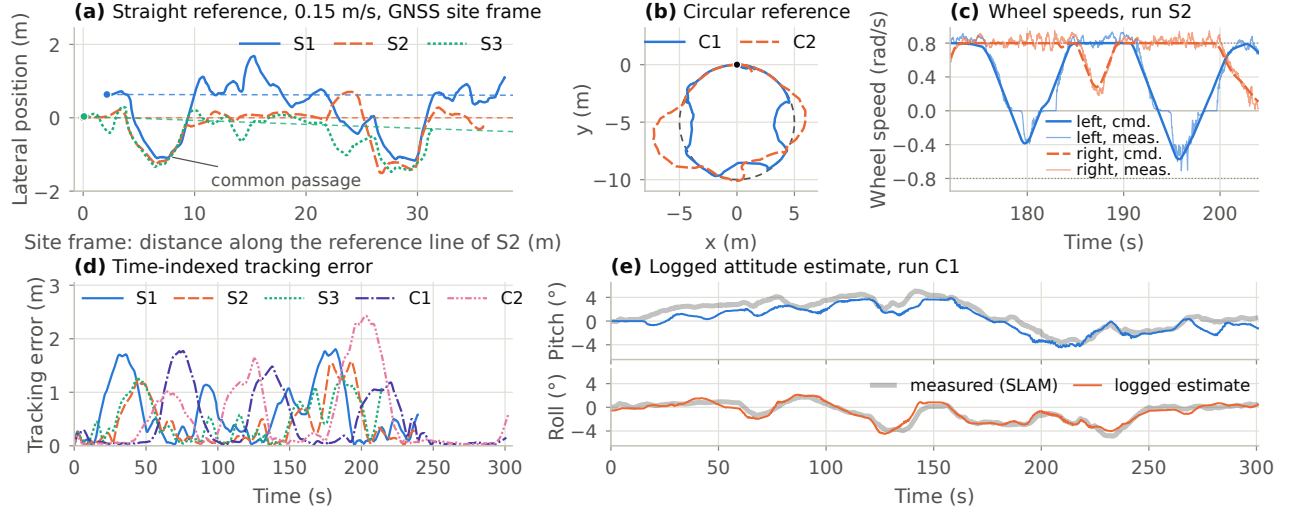


Figure 11: Outdoor trajectories and execution on September 15. (a) Straight runs in the common site frame; dashed lines are references. (b) Circular runs in their initial-pose frames. (c) Two pivot maneuvers in S2: thick lines are commands, thin lines are wheel measurements, and dotted lines are command bounds. (d) Reference-tracking error. (e) Online attitude estimates recorded during control and SLAM attitude.

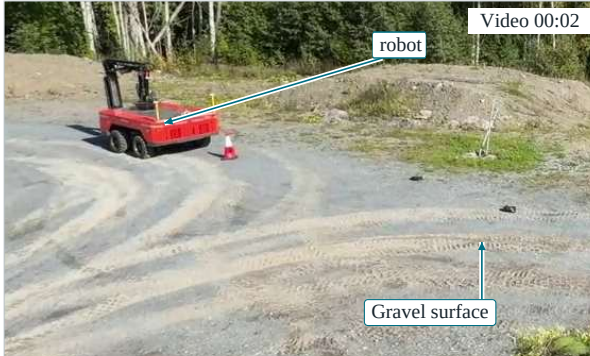
6.6 Onboard computation

The five timing exports contain 237 817 controller updates (Table 10). Across the five timing exports, mean controller update times are 6.5–6.7 ms; the largest observed update time is 28.5 ms. Height extraction, configuration classification, and distance transformation require respective mean times of 2.1–2.3 ms, 23.9–26.9 ms, and 49.9–51.4 ms. The onboard measurements and the fixed-snapshot benchmark in Section 3.6 show the same ordering of mean terrain-processing times: distance transformation, configuration classification, and height extraction. Their absolute execution times are not directly comparable because the processors and execution conditions differ. Terrain publication was deliberately capped at approximately 5 Hz. Between terrain publications, the faster predictive-control loop queries the latest available CDF.

Table 10: Recorded computation times in milliseconds. Height extraction (HM), classification, distance transformation (DT), and control report mean $\pm$ standard deviation. The maximum control-update time and number of controller updates are reported separately. The “No pose” recordings provide timing data but lack the pose data required for motion evaluation.

Export	HM	Classification	DT	Control	Control max.	Updates
K1	2.09 $\pm$ 0.49	24.68 $\pm$ 2.77	50.26 $\pm$ 2.54	6.54 $\pm$ 0.52	19.0	53 086
K2	2.16 $\pm$ 0.50	25.48 $\pm$ 2.47	50.18 $\pm$ 3.42	6.52 $\pm$ 0.57	28.5	45 421
No pose, 1	2.06 $\pm$ 0.44	24.20 $\pm$ 1.65	49.89 $\pm$ 2.00	6.61 $\pm$ 0.59	15.8	46 901
No pose, 2	2.10 $\pm$ 0.48	23.93 $\pm$ 2.72	50.36 $\pm$ 3.20	6.56 $\pm$ 0.53	17.6	47 896
C1	2.28 $\pm$ 0.55	26.89 $\pm$ 1.84	51.43 $\pm$ 2.13	6.72 $\pm$ 0.84	17.3	44 513

(a) Early traversal



(b) Heading change



(c) Progress across the test area



(d) Further heading adjustment



(e) Traversal near scattered rocks

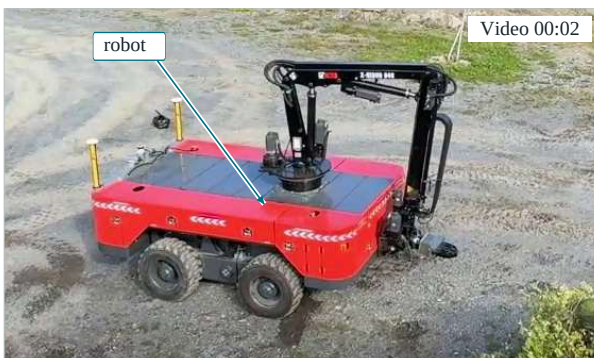


(f) Continued traversal



Figure 12: Sequential snapshots of the skid-steer robot during an outdoor straight-reference trial on gravel terrain with scattered rocks. Panels (a)–(f) show vehicle progression and heading changes at video-relative times of 2, 8, 14, 20, 26, and 32 s, respectively. Equal-size image crops follow the vehicle. The arrow and inset in panel (d) highlight a ground depression.

(a) Early loop segment



(b) Changing heading



(c) Far-side traversal



(d) Continued traversal



(e) Returning to the foreground



(f) Later loop segment



Figure 13: Sequential snapshots of the skid-steer robot during an outdoor circular-reference trial on gravel terrain. Panels (a)–(f) show vehicle progression and heading changes at video-relative times of 2, 16, 30, 44, 58, and 74 s, respectively. Equal-size crops follow the vehicle, and the cone provides a visible scene landmark.

7 Discussion and Conclusion

The central design choice is to evaluate terrain in the robot’s configuration space using the geometry that determines passage: the four wheel patches, their collective consistency, and the distinct underside regions. This preserves useful configurations in which a low obstacle lies between the wheels while screening configurations that place the same obstacle beneath a lower component. The periodic CDF provides one scalar value and its gradient to the predictive controller, keeping the detailed terrain classification outside the fast optimization loop. The evaluation combines recorded simulation trajectories, recorded terrain, and full-scale outdoor operation. In the MuJoCo recordings, the full variants have lower tracking RMSE and smaller peak pitch than the variants labeled no terrain cost on both straight courses. The rocky variant labeled no wheel checks follows the nominal reference more closely than the full variant while reaching a larger peak pitch, exposing a tradeoff that tracking error alone does not capture. The circular recordings associate larger activation-distance labels with greater reference deviation, while the continuous multi-lap recording shows sustained traversal over 11 nominal reference periods. The recorded terrain analysis demonstrates configuration-dependent admissibility in a measured scene. The outdoor experiments quantify timing, effective kinematics, wheel commands, and the measured wheel response, while field-data replay shows improved prediction with effective-parameter calibration relative to nominal geometry. Proposition 1 gives a conditional grid interpretation of an achieved CDF value. Its deviation allowance depends on the positional and scaled heading resolution; the activation-distance setting alone does not establish the achieved value along an executed trajectory. The analytical tuning balances and recorded activation-distance comparison together describe the interaction between reference tracking and the terrain penalty. The resulting framework connects component-level terrain classification to skid-steer predictive control, supported by analysis of recorded simulation trajectories and complementary full-scale operation.

Acknowledgments

This work was supported by the grant named the Post Docs in Companies (PoDoCo) program in Finland.

Data Availability

The data and code supporting this study are not publicly available.

Conflict of Interest

The authors declare no conflicts of interest.

References

- Andersson, J. A. E., Gillis, J., Horn, G., Rawlings, J. B., and Diehl, M. (2019). CasADi: A software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation*, 11(1):1–36.
- Borges, P. V. K., Peynot, T., Liang, S., Arain, B., Wildie, M., Minareci, M. G., Lichman, S., Samvedi, G., Sa, I., Hudson, N., Milford, M., Moghadam, P., and Corke, P. (2022). A survey on terrain traversability analysis for autonomous ground vehicles: Methods, sensors, and challenges. *Field Robotics*, 2:1567–1627.
- Cao, T.-T., Tang, K., Mohamed, A., and Tan, T.-S. (2010). Parallel banding algorithm to compute exact distance transform with the gpu. In *Proceedings of the 2010 ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, pages 83–90.
- Frison, G. and Diehl, M. (2020). HPIPM: A high-performance quadratic programming framework for model predictive control. *IFAC-PapersOnLine*, 53(2):6563–6569.

- Greiser, L., Demir, O., Hartmann, B., Hose, H., and Trimpe, S. (2024). Feedforward controllers from learned dynamic local model networks with application to excavator assistance functions. In *Proceedings of the 63rd IEEE Conference on Decision and Control (CDC)*, pages 4105–4112.
- Guan, T., He, Z., Song, R., Manocha, D., and Zhang, L. (2022). TNS: Terrain traversability mapping and navigation system for autonomous excavators. In *Proceedings of Robotics: Science and Systems*.
- Han, T., Liu, A., Li, A., Spitzer, A., Shi, G., and Boots, B. (2024). Model predictive control for aggressive driving over uneven terrain. In *Proceedings of Robotics: Science and Systems*.
- Ho, K., Peynot, T., and Sukkarieh, S. (2016). Nonparametric traversability estimation in partially occluded and deformable terrain. *Journal of Field Robotics*, 33(8):1131–1158.
- Jelavic, E., Qu, K., Farshidian, F., and Hutter, M. (2023). LSTP: Long short-term motion planning for legged and legged-wheeled systems. *IEEE Transactions on Robotics*, 39(6):4190–4210.
- Krüsi, P., Furgale, P., Bosse, M., and Siegwart, R. (2017). Driving on point clouds: Motion planning, trajectory optimization, and terrain assessment in generic nonplanar environments. *Journal of Field Robotics*, 34(5):940–984.
- Li, X., Xu, L., Huang, X., Xue, D., Zhang, Z., Han, Z., Xu, C., Cao, Y., and Gao, F. (2025). SEB-Naver: A SE(2)-based local navigation framework for car-like robots on uneven terrain. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5792–5799.
- Li, Y., Chi, X., Razmjoo, A., and Calinon, S. (2024). Configuration space distance fields for manipulation planning. In *Proceedings of Robotics: Science and Systems*, Delft, Netherlands.
- Meng, X., Hatch, N., Lambert, A., Li, A., Wagener, N., Schmittle, M., Lee, J., Yuan, W., Chen, Z., Deng, S., Okopal, G., Fox, D., Boots, B., and Shaban, A. (2023). TerrainNet: Visual modeling of complex terrain for high-speed, off-road navigation. In *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea.
- Millane, A., Oleynikova, H., Wirbel, E., Steiner, R., Ramasamy, V., Tingdahl, D., and Siegwart, R. (2024). nvblox: GPU-accelerated incremental signed distance field mapping. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 2698–2705.
- Moran, R., Bagley, S., Kasmann, S., Martin, R., Pasley, D., Trimble, S., Dianics, J., and Sopasakis, P. (2024). NMPC for collision avoidance by superellipsoid separation. *ASME Letters in Dynamic Systems and Control*, 4(4):041005.
- Nourizadeh, P., McFadden, F. J. S., and Browne, W. N. (2024). In situ skid estimation for mobile robots in outdoor environments. *Journal of Field Robotics*, 41(1):179–194.
- Okawara, T., Koide, K., Oishi, S., Yokozuka, M., Banno, A., Uno, K., and Yoshida, K. (2024). Tightly-coupled LiDAR-IMU-wheel odometry with online calibration of a kinematic model for skid-steering robots. *IEEE Access*, 12:134728–134738.
- Okuta, R., Unno, Y., Nishino, D., Hido, S., and Loomis, C. (2017). CuPy: A NumPy-compatible library for NVIDIA GPU calculations. In *Proceedings of Workshop on Machine Learning Systems (LearningSys) in The Thirty-first Annual Conference on Neural Information Processing Systems (NIPS)*.
- Otsu, K., Matheron, G., Ghosh, S., Toupet, O., and Ono, M. (2020). Fast approximate clearance evaluation for rovers with articulated suspension systems. *Journal of Field Robotics*, 37(5):768–785.
- Paz, A., Mustalahti, P., Dastranj, M., and Mattila, J. (2025). Real-time nonlinear model predictive control of heavy-duty skid-steered mobile platform for trajectory tracking tasks. arXiv preprint arXiv:2510.02976v1.
- Pentzer, J., Brennan, S., and Reichard, K. (2014). Model-based prediction of skid-steer robot kinematics using online estimation of track instantaneous centers of rotation. *Journal of Field Robotics*, 31(3):455–476.
- Shahna, M. H., Mustalahti, P., and Mattila, J. (2025a). Anti-slip AI-driven model-free control with global exponential stability in skid-steering robots. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6855–6862. IEEE.
- Shahna, M. H., Mustalahti, P., and Mattila, J. (2025b). Robust torque-observed control with safe input-output constraints for hydraulic in-wheel drive systems in mobile robots. *Control Engineering Practice*, 164:106459.
- Shahna, M. H., Mustalahti, P., and Mattila, J. (2026). Robust vision-based goal-reaching control for mobile robots using a hierarchical learning framework. *Robotics and Autonomous Systems*, 206:105710.

- Shi, S., Qu, K., Chen, C., Kast, I., Ma, Y., and Hutter, M. (2026). SE(2) navigation mesh. arXiv preprint arXiv:2607.01454v1.
- Shu, Y., Dong, L., Liu, J., Liu, C., and Wei, W. (2025). Overview of terrain traversability evaluation for autonomous robots. *Journal of Field Robotics*, 42(5):1724–1765.
- Todorov, E., Erez, T., and Tassa, Y. (2012). MuJoCo: A physics engine for model-based control. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5026–5033.
- Trivedi, A., Prajapati, S., Shirgaonkar, A., Zolotas, M., and Padir, T. (2025). Data-driven sampling based stochastic MPC for skid-steer mobile robot navigation. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 16619–16625.
- Verschueren, R., Frison, G., Kouzoupis, D., Frey, J., van Duijkeren, N., Zanelli, A., Novoselnik, B., Albin, T., Quirynen, R., and Diehl, M. (2022). acados—a modular open-source framework for fast embedded optimal control. *Mathematical Programming Computation*, 14(1):147–183.
- Wang, J., Liu, Z., Chen, H., Zhang, Y., Zhang, D., and Peng, C. (2024). Trajectory tracking control of a skid-steer mobile robot based on nonlinear model predictive control with a hydraulic motor velocity mapping. *Applied Sciences*, 14(1):122.
- Xue, H., Fu, H., Xiao, L., Fan, Y., Zhao, D., and Dai, B. (2023). Traversability analysis for autonomous driving in complex environment: A LiDAR-based terrain modeling approach. *Journal of Field Robotics*, 40(7):1779–1803.
- Zhang, W., Wang, Y., Lu, W., Zhang, P., Zhang, X., Wang, Y., and Wang, C. (2025). Capsizing-guided trajectory optimization for autonomous navigation with rough terrain. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 21360–21367.
- Zuo, X., Zhang, M., Wang, M., Chen, Y., Huang, G., Liu, Y., and Li, M. (2024). Visual-based kinematics and pose estimation for skid-steering robots. *IEEE Transactions on Automation Science and Engineering*, 21(1):91–105.