Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Finite budget annealing for predictive trajectory control of four wheel steering cable handling vehicles

Wenliang Liu[1]
State Grid Fujian Electric Power Co., Ltd.
Yibin Zheng[1]
zgy12022024@163.com

State Grid Fujian Electric Power Co., Ltd.
Weiyuan Li[2]
State Grid Fujian Electric Power Co., Ltd.
Fenqian Wu[2]
State Grid Fujian Electric Power Co., Ltd.
Zhixiong Lin[1]
State Grid Fujian Electric Power Co., Ltd.

Article

Keywords: automated guided vehicle, four-wheel steering, model predictive control, simulated annealing, actuator modeling, finite budget optimization

Posted Date: November 18th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11512991/v1

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

Additional Declarations: No competing interests reported.

Finite budget annealing for predictive trajectory control of four wheel steering cable handling vehicles

Wenliang Liu[1], Yibin Zheng[1]*, Weiyuan Li[2], Fenqian Wu[2] & Zhixiong Lin[1]

[1] State Grid Xiamen Power Supply Company, State Grid Fujian Electric Power Co., Ltd., Xiamen 361004, Fujian, China

[2] State Grid Fujian Electric Power Co., Ltd., Fuzhou 350003, Fujian, China

*Correspondence: Yibin Zheng, zgy12022024@163.com

Abstract

Four-wheel-steering automated guided vehicles require accurate trajectory control with explicit actuator and geometric limits. We construct a fully specified numerical benchmark coupling a fourteen-state wheel and chassis plant to a reduced model predictive controller. Standard simulated annealing, chaos-initialized annealing, deterministic predictive control and a documented PID baseline are compared in four scenarios. Forty paired horizon searches use an equal budget of 300 objective evaluations, alongside twelve plant-perturbation cases and twenty-five payload–friction runs. The median final objective is 1.34903 for standard annealing and 1.33574 for the chaotic method, which wins 22 of 40 pairs; the paired uncertainty interval includes zero. Sinusoidal cross-track errors are 0.0222 m for chaos-initialized control and 0.0272 m for PID, but tracking, speed settling, electrical input and computation do not share a uniform ranking. Deterministic predictive control provides the smallest obstacle-route tracking error, while the chaotic method incurs greater modeled electrical input than PID. Archived parameters, code, trajectories and optimizer histories connect each result to an executable implementation. The findings support a conditional accuracy–actuation–computation tradeoff rather than a general advantage from chaotic initialization.

Keywords: automated guided vehicle; four-wheel steering; model predictive control; simulated annealing; actuator modeling; finite budget optimization

Introduction

High-voltage cable handling requires a transport platform to combine accurate motion with a useful load capacity and predictable actuator behavior. Cable drums, supported cable sections and installation fixtures can occupy a substantial fraction of an industrial aisle. A mobile chassis must therefore follow a reference while avoiding contact between its complete footprint and nearby equipment. Position error alone is an incomplete design criterion because an apparently accurate trajectory can require repeated steering reversals, high drive currents or excessive computation. These considerations motivate a control study in which the vehicle, actuator model, numerical optimizer and evaluation criteria are specified together rather than assessed through a small set of response curves.

Four-wheel steering provides several ways to coordinate chassis motion. Opposing front and rear steering can generate curvature with a short effective turning radius, while coordinated wheel directions can accommodate a lateral component of motion. However, finite steering range and steering rate restrict those possibilities. A platform with bounded wheel angles cannot execute an arbitrary body velocity instantaneously, particularly close to zero forward speed. The relationship between a commanded body twist and the four individual wheel velocities is consequently part of the control problem. Studies of four-wheel-steering trajectory control and independently actuated platforms provide relevant formulations for this relationship [10,12,16,17]. The present work evaluates that architecture within an explicit planar numerical benchmark for a cable-handling vehicle.

Model predictive control (MPC) is attractive for this application because it predicts future motion and penalizes several errors within one optimization problem. Position, heading, velocity, input effort and changes in command can be treated jointly, while actuator and obstacle conditions can be represented

through bounds or penalties. Related work has investigated physics-informed nonlinear prediction, integrated obstacle planning, curve-aware tracking and stability-oriented control [6–11]. These approaches illustrate the flexibility of predictive control, but their performance depends on the model, objective, reference and solver used in each study. An MPC label therefore does not identify a unique controller, and a comparison with PID is interpretable only when those implementation choices are available.

The numerical solver introduces a separate design question. Simulated annealing (SA) can accept a candidate with a higher objective and thereby move away from a local basin [26]. Its behavior under a short budget depends on initialization, proposal scale, temperature and the number of evaluations spent exploring rather than refining a candidate. A chaotic sequence can modify the initial pool or the subsequent proposals without requiring objective gradients. Nevertheless, deterministic sensitivity to an initial value is not evidence of improved optimization. The relevant outcomes are the final objective, the distribution of paired differences, the evaluations consumed and the resulting closed-loop commands. The direction of each comparison must also remain consistent with whether the objective is minimized or maximized.

A further difficulty is separating solver effects from controller effects. A CSA-MPC controller compared only with PID changes both the prediction-based control architecture and the optimization mechanism. Any improvement could arise from reference preview, the cost weights, command constraints or the predictive model rather than the chaotic sequence. Conversely, a lower objective on a frozen horizon does not necessarily translate into better tracking after the first command is applied to a nonlinear plant. We therefore compare SA-MPC and CSA-MPC directly, include a deterministic MPC solver using the same prediction problem, and retain a completely specified PID-based tracking controller. A random sixteen-candidate initialization is also used to examine whether the initial pool alone explains the optimizer comparison.

Vehicle mass and inertia affect the plant response, while current, voltage, steering and contact-force limits constrain the commands that can be realized. An unspecified motor map or a point-mass obstacle model can conceal effects that are visible only at wheel level. Heavy-duty AGV research has examined load transfer, friction variation, torque distribution and rollover behavior [20,21,23,24]. Those studies also indicate why the scope of a planar model must be stated carefully. A planar contact-force model can evaluate tracking and modeled traction demand, whereas a conclusion about roll dynamics requires additional states and validation.

This study addresses three questions. First, how do SA and CSA compare when they solve the same instantiated MPC problem under a common budget of 300 objective evaluations? Second, how do their closed-loop responses compare with PID and deterministic MPC in acceleration, turning, obstacle-route tracking and sinusoidal tracking? Third, how do the observed differences change when plant parameters, measurement noise, command delay and numerical integration are examined explicitly? The implementation records complete trajectories, wheel signals, optimizer histories and configuration files. The resulting contribution is an executable computational comparison within the stated physical scope.

Methods

Control architecture and numerical scope

The benchmark represents a four-wheel independently steered and driven vehicle moving on a horizontal surface. The nominal chassis mass is 1500 kg and the payload is 3000 kg, giving a total mass of 4500 kg. The wheel centers are placed at longitudinal coordinates of plus or minus 1.4 m and lateral coordinates of plus or minus 0.8 m relative to the center of mass. The collision footprint is a rectangle with half-length 1.6 m and half-width 0.9 m. The state contains three planar pose coordinates, three body velocity coordinates, four steering angles and four wheel angular velocities. Parameters are specified design values for this numerical benchmark.

The implementation distinguishes the nonlinear plant from the prediction model inside MPC. The plant integrates wheel rotation, steering motion and contact forces, whereas the optimizer propagates a

reduced six-state model of pose and body velocity. This distinction creates a defined prediction mismatch while keeping the finite-horizon problem small enough for repeated optimization. The same nonlinear plant and wheel allocator are used for all four controllers. Measurements are exact in the nominal experiments, and the perturbation experiment adds a specified noise sequence to the six states supplied to the outer controller. Figure 1 shows the feedback architecture: estimate or observe the state, predict a candidate trajectory, optimize the command sequence and apply its first element.

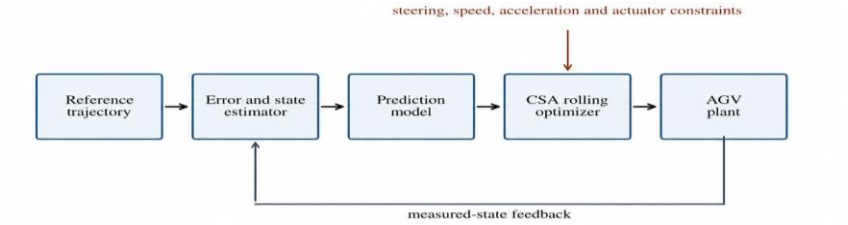


Figure 1. MPC feedback architecture. The numerical study uses a reduced predictor and a separate nonlinear wheel and chassis plant. Bounds and actuator responses are defined in Methods.

Table 1. Nominal chassis contact and sampling parameters.

Quantity	Value	Unit
Chassis mass	1500	kg
Payload mass	3000	kg
Total mass	4500	kg
Yaw inertia	5500	kg m²
Wheelbase and track	2.8 and 1.6	m
Collision footprint	3.2 × 1.8	m
Wheel radius	0.20	m
Effective wheel inertia	10	kg m²
Longitudinal slip coefficient	2500	N s/m
Lateral slip coefficient	4500	N s/m
Nominal friction coefficient	0.60	dimensionless
Rolling and quadratic drag	0.01 and 8	dimensionless; N s²/m²
Steering time constant	0.15	s
Steering angle and rate	±35 and ±40	degrees; degrees/s
Control and plant steps	0.05 and 0.005	s
Prediction and control horizons	20 and 6	samples
Center of gravity height	0.73	m; no roll state

The control sample interval is 0.05 s. The prediction horizon contains twenty samples, equivalent to 1.0 s, and the control horizon contains six command vectors. The final optimized command is held for the remaining fourteen prediction steps. Only the first vector is sent to the wheel allocator. The surviving five vectors and a repeated terminal vector form the next warm start. All controllers share bounds on the body command and its change between successive samples. Wheel steering and motor limits are implemented separately in the plant. Consequently, a bounded body command is not interpreted as a proof that every predicted trajectory satisfies all physical or geometric conditions.

Prediction model and receding implementation

The prediction state is ordered as global position X and Y, heading, forward body velocity, lateral body velocity and yaw rate. The manipulated command is the desired forward velocity, lateral velocity and yaw rate. The general nonlinear transition, nominal initial state and nominal input are written in Eqs. (1)–(3). In this implementation the pose derivative uses the current body velocity and heading, while each velocity follows its corresponding command through a first-order response. The three predictor time constants are 0.35, 0.40 and 0.45 s. Forward Euler discretization at the control sample interval defines the exact reduced transition used by every optimizer.

$$\tilde{\boldsymbol{\xi}}_0(k+1) = f\big(\tilde{\boldsymbol{\xi}}_0(k), \tilde{\mathbf{u}}_0(k)\big) \tag{1}$$

$$\tilde{\boldsymbol{\xi}}_0(0) = \boldsymbol{\xi}_0 \tag{2}$$

$$\tilde{\mathbf{u}}_0(k) = \mathbf{u}_0 \tag{3}$$

The first-order expansion in Eq. (4) explains the local state and input Jacobians. Equations (5) and (6) express the associated linear output model in deviation coordinates about a declared nominal trajectory. When absolute coordinates are used instead, the affine offset must be retained. The nonlinear shooting calculation does not replace this offset by zero; it evaluates the reduced transition directly. The matrix files supplied with the implementation contain A, B, C and the affine vector at the representative state X = 2.0 m, Y = 0.3 m, heading = 0.35 rad, forward velocity = 0.6 m/s, lateral velocity = 0.02 m/s and yaw rate = 0.08 rad/s.

$$\begin{aligned}\boldsymbol{\xi}(k+1) \approx\ & f\big(\tilde{\boldsymbol{\xi}}_0(k), \tilde{\mathbf{u}}_0(k)\big) \\ & + \left.\frac{\partial f}{\partial \boldsymbol{\xi}}\right|_0 \big(\boldsymbol{\xi}(k) - \tilde{\boldsymbol{\xi}}_0(k)\big) + \left.\frac{\partial f}{\partial \mathbf{u}}\right|_0 \big(\mathbf{u}(k) - \tilde{\mathbf{u}}_0(k)\big)\end{aligned} \tag{4}$$

$$\mathbf{x}(k+1) = \mathbf{A}_{k,t}\mathbf{x}(k) + \mathbf{B}_{k,t}\mathbf{u}(k) \tag{5}$$

$$\mathbf{y}(k \mid t) = \mathbf{C}_{k,t}\mathbf{x}(k \mid t) \tag{6}$$

The incremental formulation in Eqs. (7) and (8) augments the prediction state with the previous command. Its output-selection matrix is the augmented C matrix. For a locally frozen linearization, stacking the predictions gives Eqs. (9)–(13). The decision vector contains six increments, from the current sample through the fifth future sample, and the last dynamic-matrix exponent follows the difference between prediction and control indices. These indexing details matter because including an unintended seventh command changes the decision dimension and evaluation cost. The numerical optimizer uses eighteen absolute command coordinates with sequential slew projection, which is equivalent to enforcing the increment bounds under the declared hold convention.

$$\boldsymbol{\xi}(k+1 \mid t) = \widetilde{\mathbf{A}}_{k,t}\boldsymbol{\xi}(k \mid t) + \widetilde{\mathbf{B}}_{k,t}\Delta\mathbf{u}(k) \tag{7}$$

$$\eta(k \mid t) \ = \ C_{k,t}\, \xi(k \mid t) \tag{8}$$

$$\mathbf{Y}(t) = \psi_t \boldsymbol{\xi}(k) + \boldsymbol{\Theta}_t \Delta\mathbf{U}(k) \tag{9}$$

$$\mathbf{Y}(t) = [\boldsymbol{\eta}(t+1 \mid t), \boldsymbol{\eta}(t+2 \mid t), \ldots, \boldsymbol{\eta}(t+N_p \mid t)]^T \tag{10}$$

$$\Psi_t = \begin{bmatrix} CA \\ CA^2 \\ \vdots \\ CA^{N_p} \end{bmatrix} \tag{11}$$

$$\Delta U(k) = [\Delta u(k \mid k), \Delta u(k+1 \mid k), \ldots, \Delta u(k+N_c-1 \mid k)]^T \tag{12}$$

$$\Theta_t = \begin{bmatrix} CB & 0 & \ldots & 0 \\ CAB & CB & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{N_p-1}B & CA^{N_p-2}B & \ldots & CA^{N_p-N_c}B \end{bmatrix} \tag{13}$$

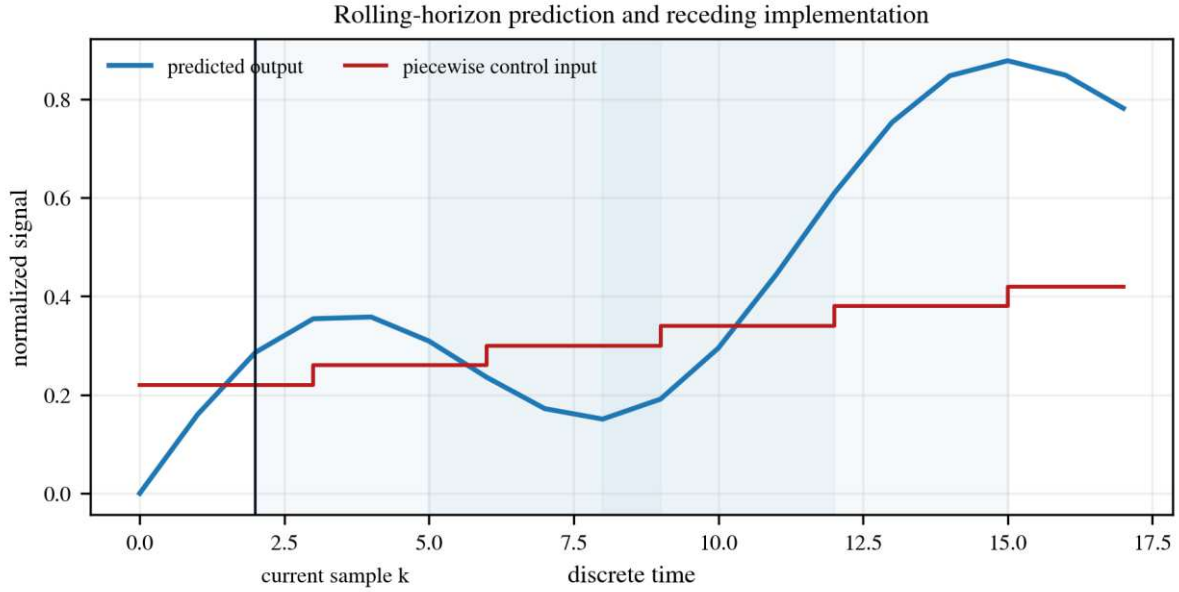


Figure 2. Rolling prediction and receding implementation. Normalized schematic signals illustrate the prediction and control horizons.

The objective contains weighted output errors, deviation of the command from the reference body twist, and changes in the six optimized commands. A terminal error penalty is added at the twentieth prediction sample. The error weights for X, Y, heading, forward velocity, lateral velocity and yaw rate are 12, 30, 16, 4, 4 and 2. Acceleration uses zero longitudinal position weight and a forward-velocity weight of 12 so that the speed-response test is not dominated by a position deficit accumulated during acceleration. The input-reference weights are 0.03, 0.05 and 0.05; the increment weights are 0.25, 0.50 and 0.80. Terminal error uses twice the corresponding state weight. The complete objective is divided by twenty, which fixes the scale used by the annealing temperature.

Table 2. Complete MPC weights prediction constants and command bounds.

Component	Numerical value	Definition
Q diagonal	12, 30, 16, 4, 4, 2	X, Y, heading, vx, vy, yaw rate
Acceleration Q	0, 30, 16, 12, 4, 2	Same state order
R diagonal	0.03, 0.05, 0.05	Command minus reference twist

Component	Numerical value	Definition
S diagonal	0.25, 0.50, 0.80	Six command increments
Terminal multiplier	2	Multiplies corresponding Q
Prediction time constants	0.35, 0.40, 0.45 s	Forward, lateral, yaw
Input lower bounds	0, −0.35, −0.35	m/s, m/s, rad/s
Input upper bounds	1.5, 0.35, 0.35	m/s, m/s, rad/s
Input slew limits	2.0, 0.8, 1.4 per s	Derivative of each input
Clearance margin and weight	0.25 m; 10000	Squared shortfall penalty
Steering penalty	100	Squared angle excess in radians
Cost normalization	Divide total by 20	One unit reference scale per SI coordinate

Steering-demand and obstacle-clearance violations enter as additional squared penalties. For each candidate body command, the wheel velocity directions are calculated before steering clipping, and angle excess beyond 35 degrees is penalized with weight 100. Predicted footprint clearance below 0.25 m receives weight 10000. These terms are soft penalties. In contrast, command amplitude and slew bounds are imposed by projection for every candidate, and the plant applies its own hard wheel-angle, steering-rate, voltage and current limits. A finite budget can therefore return a command with a nonzero soft penalty, and actual clearance is evaluated from the simulated footprint rather than inferred from the optimizer status.

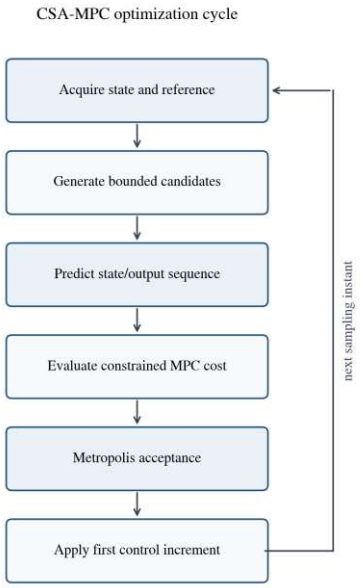


Figure 3. CSA-MPC calculation cycle. Candidate projection enforces command amplitude and slew limits, while steering demand and obstacle clearance enter the stated cost penalties.

Annealing construction and computational budget

The optimizer minimizes the horizon objective directly. The generic fitness transformations in Eqs. (14) and (15) illustrate how a truncated affine mapping can be constructed for minimization or maximization, respectively. They are not substituted for the objective in the implemented comparison. The reported final value is always the smallest evaluated horizon cost, so a lower value is preferable. This convention is maintained in the paired differences, convergence plots and tables. A numerical tie is defined by an absolute difference no greater than 1×10^{-10}, avoiding a change in ranking caused only by the printed precision of a table.

$$\mathrm{Fit}_{\min}(f) = \begin{cases} c_{\max} - f, & f < c_{\max}, \\ 0, & f \geq c_{\max}, \end{cases} \tag{14}$$

$$\mathrm{Fit}_{\max}(f) = \begin{cases} f + c_{\min}, & f > c_{\min}, \\ 0, & f \leq c_{\min}, \end{cases} \tag{15}$$

Chaos initialization uses the logistic recursion in Eq. (16). Figures 4 and 5 illustrate its parameter-dependent regimes and sensitivity to a small initial perturbation. The implemented fully chaotic sequence uses double precision and a starting value generated inside the interval from 0.05 to 0.95. Each new coordinate advances the recursion once. The value is then mapped into a window around the warm start with a half-range equal to one quarter of the corresponding input range. Sequential amplitude and slew projection produces the candidate actually evaluated. The deterministic logistic sequence can have nonuniform coverage and short numerical cycles.

$$x_{k+1} = \mu x_k (1 - x_k), \qquad 0 \leq x_k \leq 1, \quad 0 \leq \mu \leq 4 \tag{16}$$

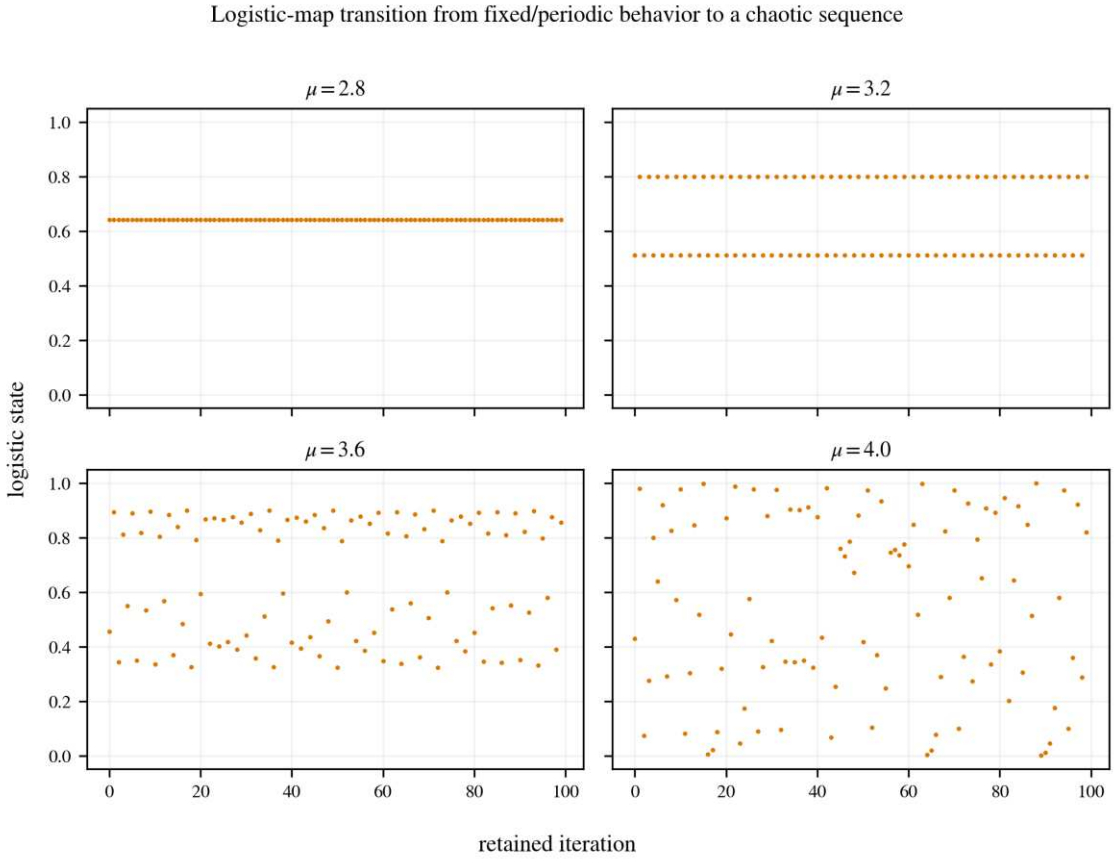


Figure 4. Logistic-map regimes for different parameter choices. Changes in the map parameter produce fixed, periodic and irregular sequences.

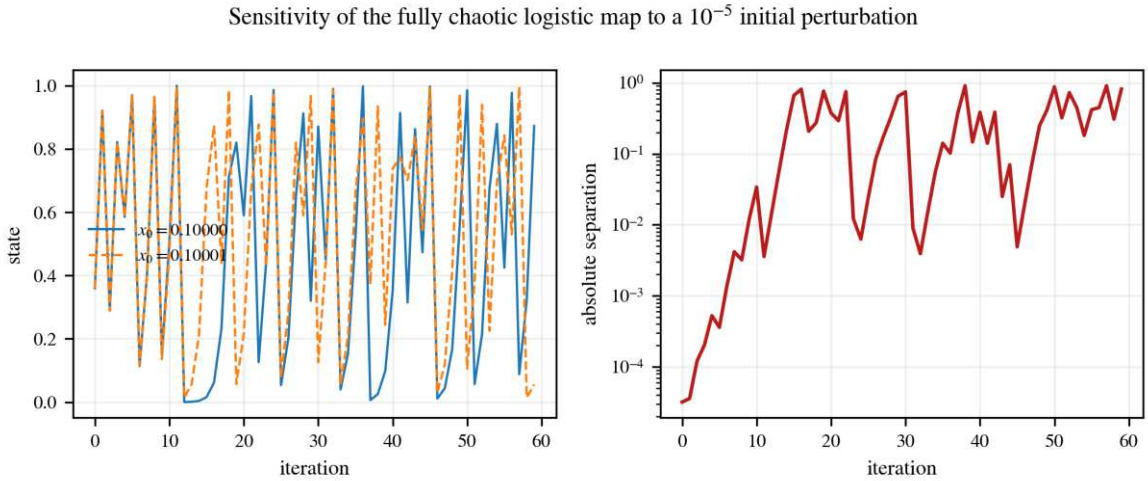


Figure 5. Sensitivity of the fully chaotic logistic map to a small initial perturbation. The paired curves and separation illustrate deterministic sequence divergence.

SA accepts a non-increasing objective unconditionally and accepts an increase according to the Metropolis probability in Eq. (17). Figures 6 and 7 show the explanatory local-basin and algorithm illustrations. The initial temperature is 0.2 in the units of the normalized objective. Temperature decreases by a factor of 0.985 per evaluation index and is bounded below by 10^{-5}. Equation (18) gives the positive-temperature relation for an alternative calibration from a positive representative cost increase and desired acceptance probability. The implemented fixed temperature makes the SA and CSA settings directly inspectable without estimating a different initial temperature from each algorithm's candidate pool.

$$P(x \to y) = \begin{cases} 1, & J(y) \le J(x), \\ \exp\left[-\dfrac{J(y) - J(x)}{T}\right], & J(y) > J(x), \end{cases} \tag{17}$$

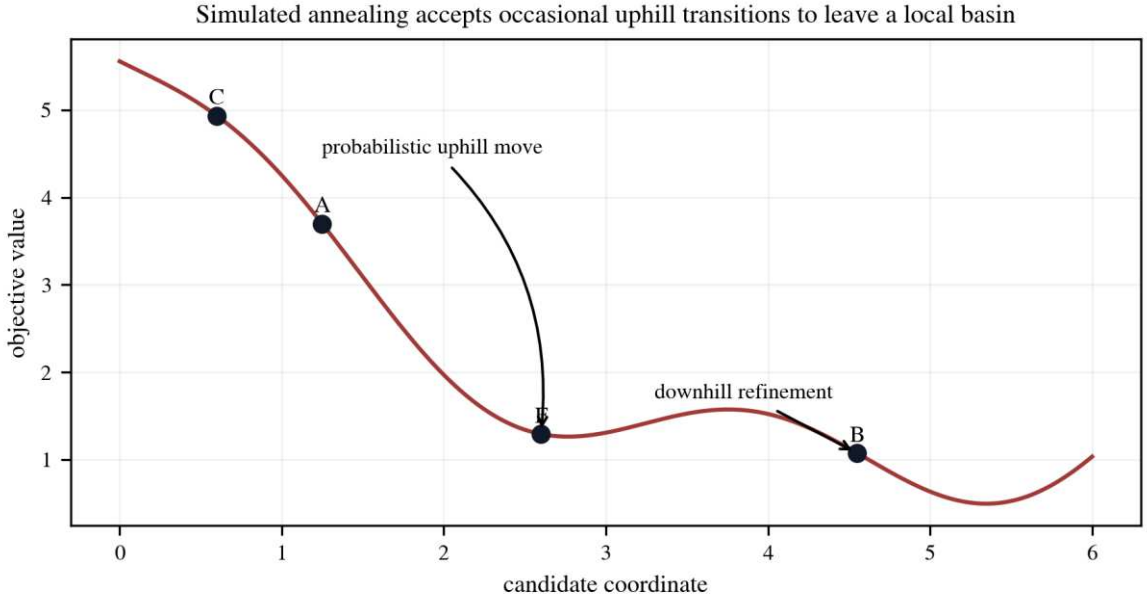


Figure 6. Simulated-annealing search principle. An accepted uphill move can leave a local basin before later refinement. The drawn cost profile is conceptual.

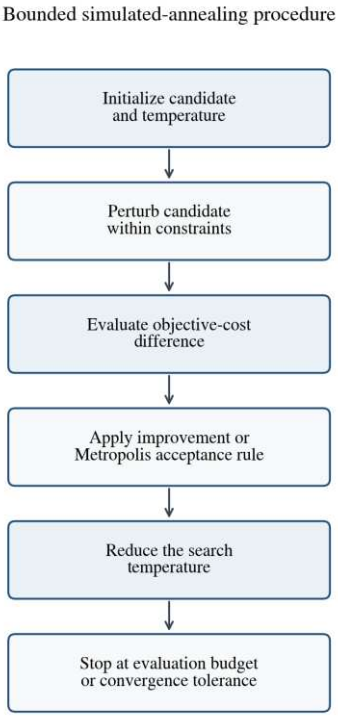


Figure 7. Bounded simulated-annealing workflow. Every evaluated initial or proposed candidate counts toward the common finite budget.

$$T_0 = -\frac{\Delta J^*}{ln(p_0)} > 0, \quad \Delta J^* > 0, \quad 0 < p_0 < 1 \tag{18}$$

$$C_k' = \mu C_k(1 - C_k) \tag{19}$$

$$C_k' = 4C_k(1 - C_k) \tag{20}$$

Standard SA evaluates the projected warm start once and then uses 299 proposal evaluations. CSA evaluates the same warm start and fifteen logistic candidates before using its remaining 284 proposal evaluations. The random-pool ablation also evaluates sixteen initial candidates, replacing the fifteen logistic points with independent uniform coordinates in the same warm-start window. Thus all initial evaluations count toward the same total budget of 300. Initial-pool selection, Gaussian proposals, acceptance uniforms and the extra chaotic component use explicitly separated random streams derived from the paired seed. Corresponding ordinary proposal and acceptance streams have the same initial states, while accepted trajectories can diverge as the algorithms encounter different costs.

Table 3. Fully specified annealing settings and random streams.

Setting	Value
Decision dimension and budget	18 coordinates; 300 evaluations including initialization
SA and CSA initial counts	1 and 16; common warm start included
Random-pool ablation	16 candidates including the same warm start
Initial temperature and cooling	0.2; factor 0.985 per evaluation index
Temperature floor	0.00001
Initial pool half-range	0.25 of each input range around the warm start

Setting	Value
Gaussian proposal scale	Linear decrease from 0.12 to 0.02 of input range
Full-sequence proposal probability	0.20; otherwise one three-coordinate block
Extra chaotic component	Probability 0.25; amplitude 0.08 of input range
Logistic map and start	z(next) = 4z(1−z); z0 drawn from [0.05, 0.95]
Random generator	C++ mt19937_64; separate initialization, proposal, acceptance and chaos streams
Paired seeds	202610234000 through 202610234039
Tie tolerance	Absolute cost difference ≤ 1×10^{-10}

Each proposal modifies either one three-component command block or the full sequence, with a full-sequence probability of 0.2. The Gaussian scale decreases linearly from 0.12 to 0.02 of the relevant input range across the evaluation budget. CSA adds a logistic perturbation of magnitude 0.08 of that range with probability 0.25 for each modified coordinate. The best evaluated candidate is retained even if the current annealing state accepts a worse value. Figure 8 summarizes the combined initialization and proposal procedure. The schematic projection refers to the command amplitude and slew conditions; obstacle clearance remains a separately evaluated soft condition.

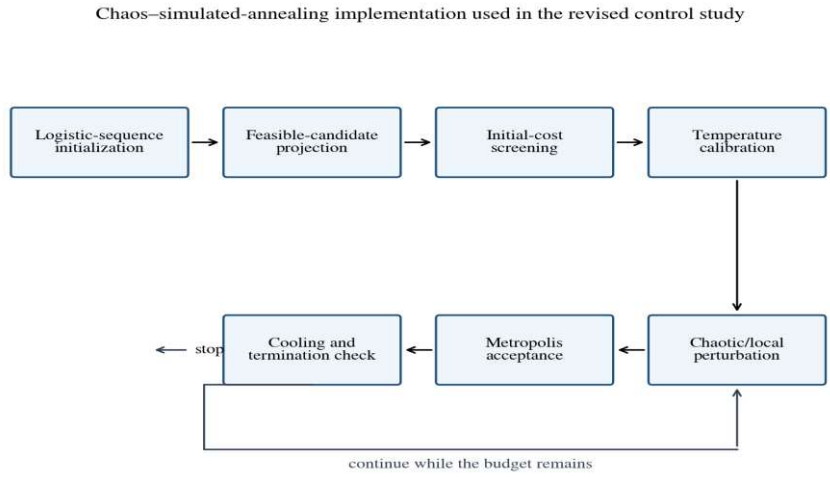


Figure 8. Chaos and annealing implementation. The executed configuration specifies the initialization pool, proposal streams, cooling and projection in Table 3 and the source code.

The deterministic MPC comparator uses L-BFGS-B on the identical eighteen-coordinate objective and input box. The objective wrapper also applies the same sequential slew projection. Finite-difference objective queries count toward the 300-evaluation allowance. A hard wrapper stops further evaluation at that count and returns the best candidate already observed; earlier numerical convergence can use fewer evaluations. This comparator is designated D-MPC because it is deterministic for a given state and warm start. It is a local numerical baseline, and neither the method nor its termination status establishes a global optimum for the projected, penalized horizon problem.

Nonlinear vehicle and actuator model

Figure 9 defines the body coordinates, wheel numbering and collision rectangle. Positive body x points forward and positive body y points left. Wheels are ordered front left, front right, rear left and rear right. Positive yaw is counterclockwise. For a wheel at body coordinates x_i and y_i, its local chassis velocity is obtained from the translational body velocity and yaw rate. Resolving that velocity along and normal to the current steering direction gives the longitudinal and lateral contact velocities. Equation (21) defines the

wheel-frame mapping. This same geometry is used for wheel allocation and force accumulation, so the front and rear yaw contributions do not rely on separate bicycle approximations.

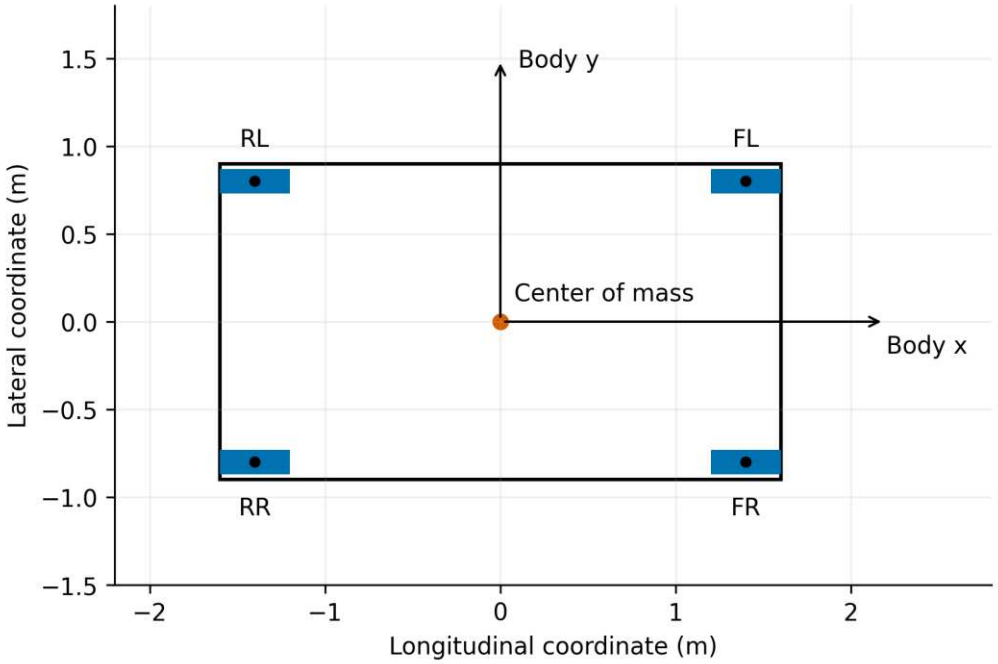


Figure 9. Instantiated vehicle geometry in body coordinates. Wheel centers are at (±1.4, ±0.8) m and the collision rectangle is 3.2 by 1.8 m. FL, FR, RL and RR denote front left, front right, rear left and rear right.

Figure 10 presents the spatial relationship between the rectangular deck envelope and the four independently steered wheel modules. The wheel centers share the coordinates used in the force and yaw-moment equations. The center-of-gravity marker uses the 0.73 m height listed in Table 1. Figure 11 gives the corresponding dimensioned plan: the 3200 by 1800 mm envelope contains a 2800 mm wheelbase and a 1600 mm wheel-center track. These views connect the body coordinate system to the geometric lever arms used by the controller and plant. Deck thickness and mounting details are schematic visual elements.

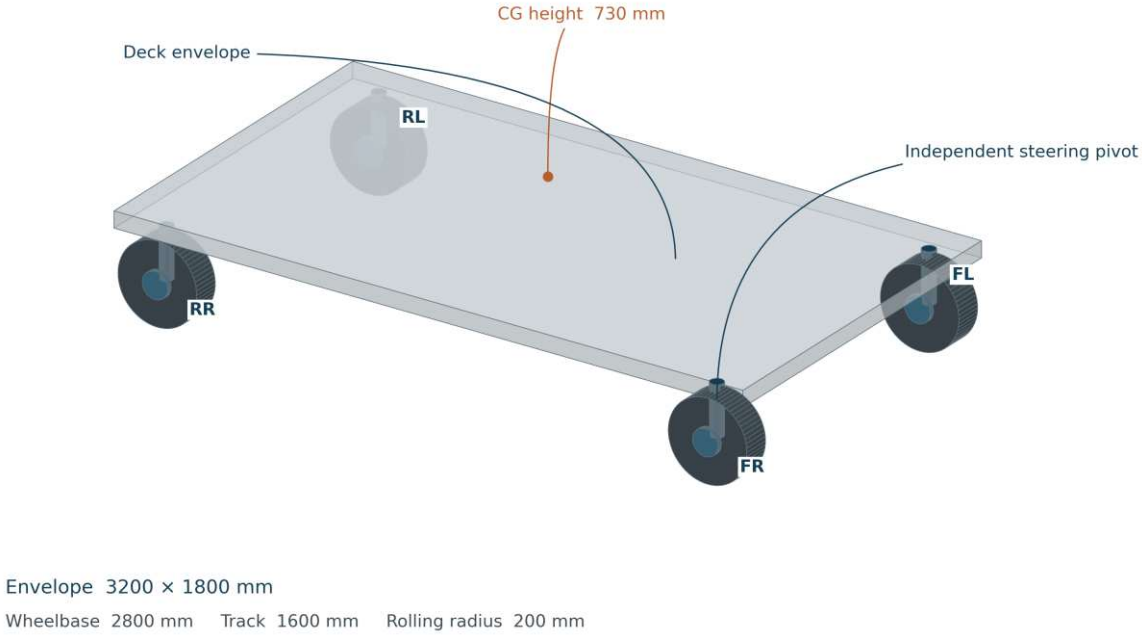


Figure 10. Three-dimensional structural schematic of the four-wheel-steering platform. The deck envelope, wheel centers, rolling radius and center-of-gravity marker follow Table 1. Deck thickness, hub details and mounting posts provide a schematic spatial arrangement.

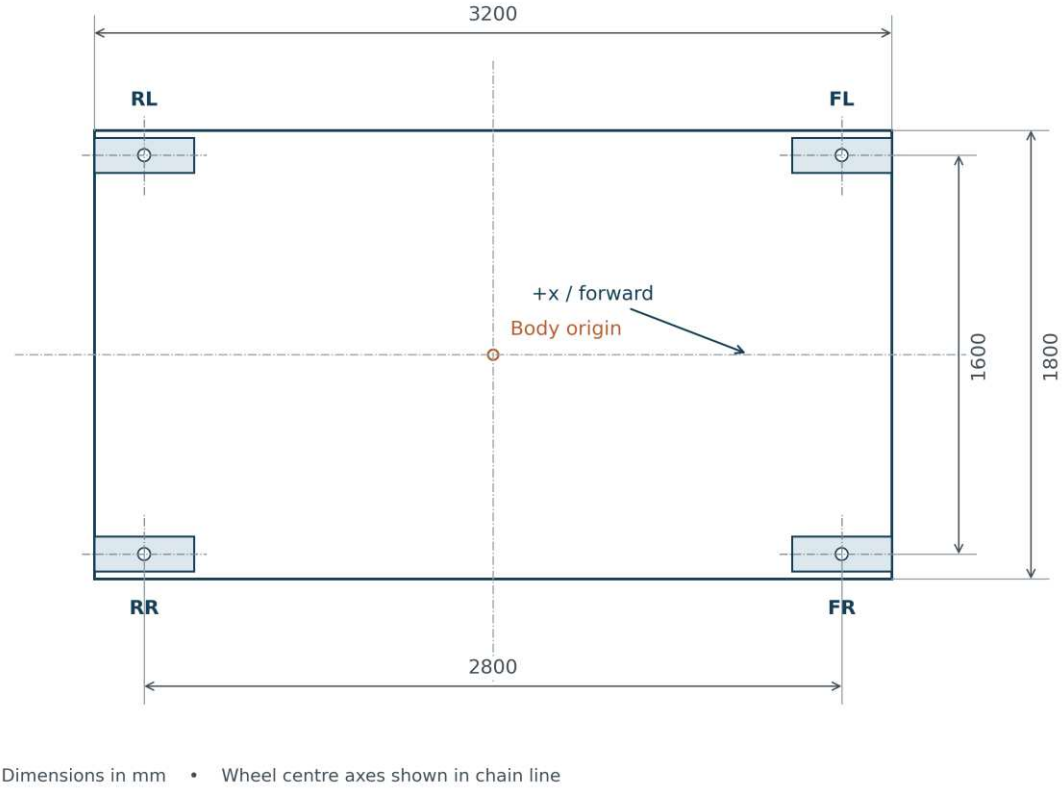


Figure 11. Dimensioned CAD plan of the chassis envelope and wheel locations. Dimensions are in millimeters. The envelope measures 3200 by 1800, the wheelbase is 2800 and the wheel-center track is 1600. Chain lines locate the body axes and steering pivots.

$$v_{xi} = v_x - r y_i, \quad v_{yi} = v_y + r x_i$$
$$\begin{bmatrix} v_{li} \\ v_{ti} \end{bmatrix} = \begin{bmatrix} \cos\delta_i & \sin\delta_i \\ -\sin\delta_i & \cos\delta_i \end{bmatrix} \begin{bmatrix} v_{xi} \\ v_{yi} \end{bmatrix} \tag{21}$$

The wheel allocator maps the requested body twist to desired wheel directions and signed angular speeds. If a wheel direction lies outside a forward-facing half-plane, its angle is shifted by pi and its speed sign is reversed before the steering limit is applied. The resulting desired angle is clipped to plus or minus 35 degrees. Actual steering follows a first-order response with a nominal time constant of 0.15 s and a rate cap of 40 degrees/s. This implementation makes finite steering response visible in the plant even when the reduced predictor changes yaw or lateral velocity more quickly. It also exposes requests that cannot be realized exactly after steering clipping.

$$\delta_i^* = clip(fold_{\pi/2}\{atan2(u_y + u_r x_i, u_x - u_r y_i)\}, \pm\delta_{max})$$
$$\dot{\delta}_i = clip((\delta_i^* - \delta_i)/\tau_\delta, \pm\dot{\delta}_{max}) \tag{22}$$

Figure 12 shows the wheel module and its steering sector. The rolling axis is horizontal, whereas the steering axis changes the wheel direction in the body plane. The drawing separates the hub from the wheel to make those axes visible. The 200 mm rolling radius is the same value used to convert wheel angular velocity to peripheral speed and to convert longitudinal contact force to wheel torque. The top view illustrates the common plus or minus 35 degree steering limit, independently applied at all four wheel locations.

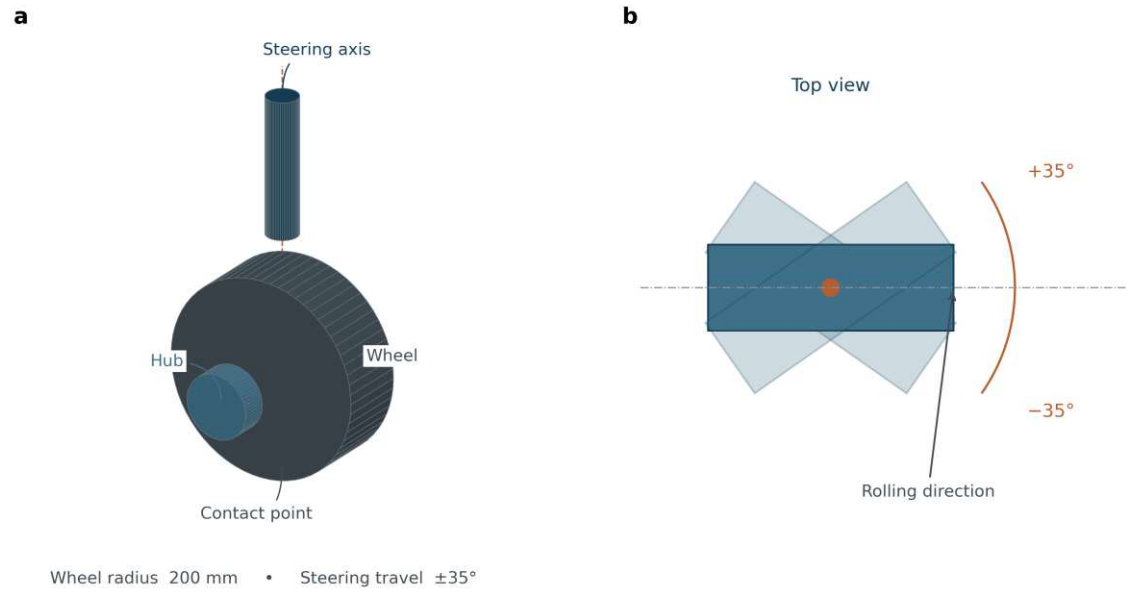


Figure 12. Wheel-module geometry and steering motion. Panel a separates the wheel, hubs and steering axis for visibility. Panel b shows the rolling direction and the admissible steering sector of plus or minus 35 degrees. Wheel radius is 200 mm; hub and mounting details are schematic.

Longitudinal contact demand is proportional to the difference between wheel peripheral speed and longitudinal contact velocity, with coefficient 2500 N s/m. Lateral demand opposes lateral contact velocity with coefficient 4500 N s/m. The combined two-dimensional demand is projected onto a friction disk whose radius is the friction coefficient multiplied by the wheel's normal load. All four static normal loads are equal to one quarter of vehicle weight. The nominal friction coefficient is 0.60. This regularized velocity-slip model is defined at zero speed and avoids division by a small longitudinal velocity. The contact coefficients are prescribed model values.

$$\begin{aligned} F_i^* &= \begin{bmatrix} C_x(r_w\omega_i - v_{li}) \\ -C_y v_{ti} \end{bmatrix} \\ F_i &= F_i^* \, min\{1, \mu mg/(4\|F_i^*\|)\} \\ F_i &= 0 \, when \, F_i^* = 0 \end{aligned} \tag{23}$$

The motor model uses an algebraic electrical relation with a 48 V terminal-voltage bound, resistance 0.08 ohm, torque constant 0.16 N m/A and back-emf constant 0.16 V s/rad. The gear ratio is 20 and mechanical transmission efficiency is 0.90. Motor current is limited to plus or minus 80 A. A wheel-speed voltage controller has gain 4.0 V s/rad and includes back-emf feedforward evaluated at the desired wheel speed. Wheel rotation uses an effective inertia of 10 kg m², including a declared reflected drive inertia, and viscous coefficient 0.02 N m s/rad. Electrical inductance and thermal states are omitted; the current limit is an imposed benchmark constraint rather than a continuous-duty rating.

$$\begin{aligned} V_i &= clip\{K_v(\omega_i^* - \omega_i) + K_e g\omega_i^*, \pm V_{max}\} \\ I_i &= clip\{(V_i - K_e g\omega_i)/R_m, \pm I_{max}\} \\ \tau_i &= \eta g K_t I_i, \;\; J_w\dot{\omega}_\iota = \tau_i - r_w F_{li} - b_w\omega_i \end{aligned} \tag{24}$$

Table 4. Motor wheel and steering actuator settings.

Quantity	Value	Unit
Terminal voltage limit	48	V
Motor resistance	0.08	ohm
Torque and back-emf constants	0.16; 0.16	N m/A; V s/rad
Gear ratio and efficiency	20; 0.90	dimensionless

Quantity	Value	Unit
Current limit	±80	A
Wheel-speed voltage gain	4.0	V s/rad
Effective wheel inertia	10	kg m²
Wheel viscous coefficient	0.02	N m s/rad
Low-speed wheel torque cap	230.4	N m
No-load wheel speed	15	rad/s
Steering time constant	0.15	s
Steering angle and rate caps	±35; ±40	degrees; degrees/s

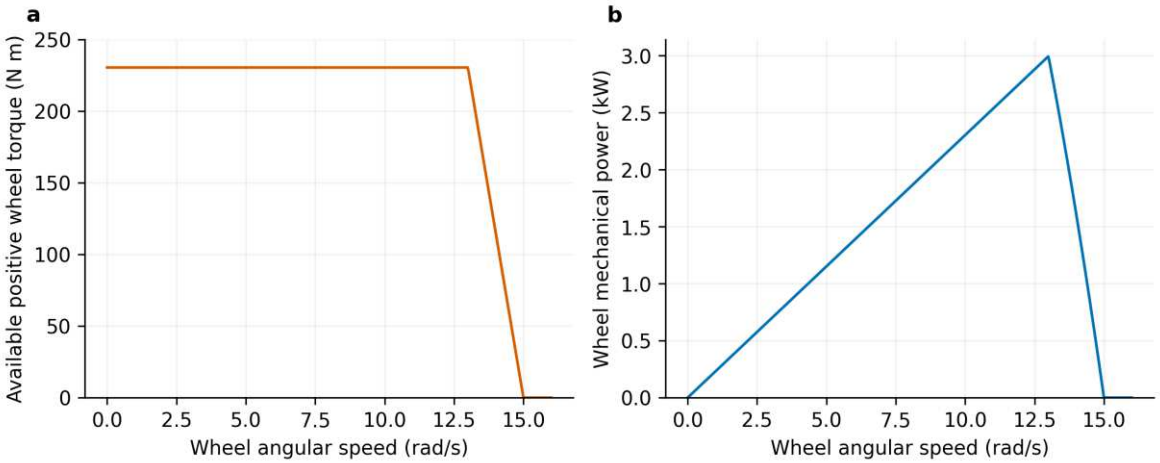


Figure 13. Calculated motor and gearing envelope from the declared electrical parameters. Panel a shows available positive wheel torque under voltage and current limits; panel b shows associated wheel mechanical power. The curves are model calculations.

The declared motor and gearing values imply a positive current-limited wheel torque of 230.4 N m at low speed and a no-load wheel speed of 15 rad/s under the voltage limit. Figure 13 plots this calculated drive envelope and its associated mechanical power. The envelope follows the stated algebraic equations. Actual torque in each run is computed from current rather than selected directly from the envelope. Electrical input energy is integrated over the plant substeps using terminal voltage multiplied by current for each motor. Positive input and net input are stored separately so that modeled regenerative intervals do not silently cancel all positive drive demand.

Planar chassis dynamics are given in Eq. (25). Contact forces are rotated into body coordinates and summed, while yaw moment uses each wheel's signed lever arm. The nominal yaw inertia is 5500 kg m². Body-frame acceleration includes the usual rotational coupling between forward, lateral and yaw velocity. Rolling resistance has coefficient 0.01 and uses a smooth speed denominator of 0.05 m/s; an additional quadratic drag coefficient of 8 N s²/m² is applied. Global pose is integrated using the instantaneous body velocity and heading. The complete fourteen-state plant is advanced with the explicit midpoint method at 0.005 s, giving ten integration substeps per control sample.

$$\begin{gathered} \dot{X} = \cos\psi\, v_x - \sin\psi\, v_y, \quad \dot{Y} = \sin\psi\, v_x + \cos\psi\, v_y \\ \dot{\psi} = r, \quad m\dot{v}_x = m r v_y + \Sigma F_{xi} - D_x \\ m\dot{v}_y = -m r v_x + \Sigma F_{yi} - D_y \\ I_{zz}\dot{r} = \Sigma(x_i F_{yi} - y_i F_{xi}) \end{gathered} \tag{25}$$

The nonlinear plant includes traction saturation but no vertical, pitch or roll states. Equal normal loads consequently exclude dynamic load transfer, and the supplied center-of-gravity height is descriptive

geometry rather than an active rollover model. The model evaluates planar tracking, actuation and contact-force outcomes; structural cable behavior, suspension response and rollover lie outside its state definition.

For the reduced predictor, the representative numerical A matrix includes the pose derivatives with respect to heading and body velocity. Its velocity diagonal entries are 0.857143, 0.875000 and 0.888889, while the corresponding nonzero entries of B are 0.142857, 0.125000 and 0.111111. C is the six-dimensional identity. The affine vector and the nine-state incremental matrices are included in machine-readable files. Figure 14 shows the numerical coupling structure. Central finite differences with step 10^{-6} agree with the analytical Jacobians to a maximum absolute error of 1.63e-10, providing a direct implementation check rather than relying on the symbolic appearance of a state-space equation.

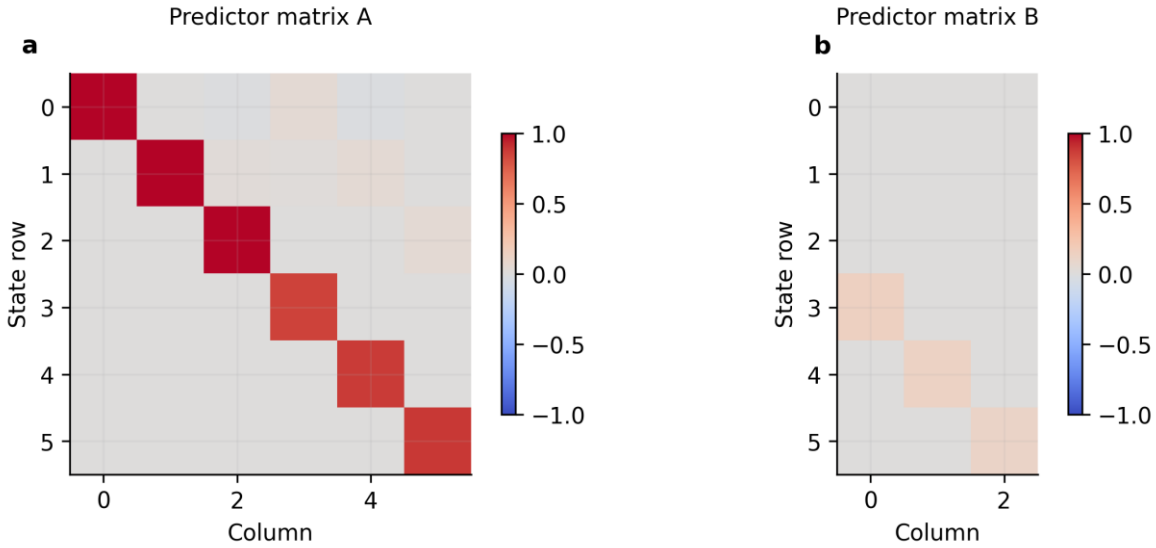


Figure 14. Numerical predictor matrices at the representative state and input specified in Methods. Panel a shows A and panel b shows B. Complete entries, the affine vector and augmented matrices accompany the code.

$$A = \begin{bmatrix} 1 & 0 & -0.011226 & 0.046969 & -0.017145 & 0 \\ 0 & 1 & 0.027838 & 0.017145 & 0.046969 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0.050000 \\ 0 & 0 & 0 & 0.857143 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.875000 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.888889 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0.142857 & 0 & 0 \\ 0 & 0.125000 & 0 \\ 0 & 0 & 0.111111 \end{bmatrix}, \quad C = I_6 \tag{26}$$

$$d = [0.003929, -0.009743, 0, 0, 0, 0]^T$$

PID baseline and shared interfaces

The PID-based comparator uses body-frame position and wrapped heading errors with reference feedforward. The proportional gains for forward position, lateral position and heading are 0.8, 0.9 and 1.8 per second. Integral gains are 0.04, 0.05 and 0.05 per second squared, and derivative gains are 0.08, 0.10 and 0.06. A first-order derivative filter has time constant 0.15 s. Additional velocity-error feedback gains are 0.3, 0.2 and 0.2. Integral states are limited to 0.4, 0.4 and 0.25 in their corresponding integrated-error units. Equation (27) describes the outer command before the shared projection is applied.

$$u^* = u_{ff} + K_p e + K_i \int e\, dt + K_d \dot{e}_{filtered} + K_v (v_{ref} - v) \tag{27}$$

Table 5. PID parameters and implementation shared with the plant.

Loop	Kp	Ki	Kd	Integral limit
Forward position	0.8	0.04	0.08	0.4
Lateral position	0.9	0.05	0.10	0.4
Heading	1.8	0.05	0.06	0.25
Acceleration speed	0.5	0.35	0.02	0.8
Velocity feedback	0.3; 0.2; 0.2	—	—	Forward; lateral; yaw
Derivative filter	0.15 s	—	—	All derivative channels
Anti-windup	Conditional integration	—	—	Shared amplitude and slew limits

Acceleration uses a dedicated forward-speed PID channel because longitudinal position is excluded from the acceleration MPC cost. Its proportional, integral and derivative gains are 0.5, 0.35 per second and 0.02 s, respectively, with integral limit 0.8 and the same derivative filter. Conditional integration updates an integral component only when its unsaturated command agrees with the projected command within numerical tolerance. This prevents the controller from accumulating an integral correction that cannot be applied under the shared amplitude and slew bounds. The PID gains are fixed throughout the benchmark.

All four outer controllers send the same three-component command to the same wheel allocator and plant. PID has reference feedforward at the current sample, while MPC uses the stated future reference window. The known reference schedule is the same source for both architectures; MPC's ability to use preview is an architectural difference and must not be attributed to CSA alone. Obstacle geometry and the detour generator are also shared. This arrangement avoids comparing a predictive tracking controller with a PID controller deprived of a collision-free route, while retaining the practical difference between a current-error control law and a horizon optimization.

Scenarios and geometric definitions

The acceleration scenario lasts 16 s and requests a forward speed of 0.8 m/s from 1 s onward. The vehicle starts at rest with zero wheel angle and speed. Because the reference is scheduled and visible within the prediction horizon, MPC may react before the nominal step time. Threshold-crossing times are therefore stored relative to the 1 s event and may be negative. The ten-to-ninety-percent interval uses the first actual crossings of the common 0.08 and 0.72 m/s levels over the complete record, rather than clipping both crossings to the post-step interval. Settling uses a band of plus or minus two percent of the common target after the step.

The turning scenario lasts 24 s. Its reference travels at 0.6 m/s along a straight line for 3 s and then follows a circle of radius 4 m with constant reference yaw rate 0.15 rad/s. The initial lateral displacement is minus 0.1 m, and the initial chassis and wheel velocities are consistent with the straight reference. Radius is evaluated from the simulated speed and yaw rate during established turning, so its error is measured relative to the requested radius. A smaller radius is not automatically considered better. Heading error, cross-track error and lateral force acceleration are retained to describe the maneuver without conflating compactness with tracking accuracy.

Table 6. Four fully defined nominal scenarios.

Scenario	Duration	Reference and initial condition
Acceleration	16 s	0.8 m/s scheduled step at 1 s; vehicle and wheels initially at rest
Turning	24 s	0.6 m/s; straight until 3 s then 4 m radius circle; initial Y offset −0.1 m
Obstacle route	32 s	X = 0.65t; disk (10, 0) m, radius 0.45 m, known at 2 s; 2.2 m quintic detour

Scenario	Duration	Reference and initial condition
Sinusoidal	30 s	X = 0.6t; Y = 0.6 sin(2πX/10); initial Y offset −0.15 m
Common termination	Fixed duration	Identical time grids; no outcome-dependent early stop

The obstacle scenario lasts 32 s and progresses along global x at 0.65 m/s. A disk centered at X = 10 m and Y = 0 has radius 0.45 m and becomes available to the common route generator at 2 s. The detour rises to Y = 2.2 m between X = 2 and 8 m, remains level until X = 12 m, and returns to the centerline by X = 18 m. The rising and falling segments use the quintic smoothstep polynomial, with analytically calculated first and second derivatives. The reference heading follows the tangent, and its forward body speed accounts for the increased arc length. This is a known static-obstacle route-tracking test rather than a test of object detection or an independently optimized avoidance planner.

$$\begin{aligned} s(z) &= 10q^3 - 15q^4 + 6q^5, \quad q = clip(z, 0, 1) \\ y_{ref}(x) &= 2.2\{s[(x-2)/6] - s[(x-12)/6]\} \end{aligned} \tag{28}$$

The collision calculation uses the oriented vehicle rectangle rather than the reference point alone. The obstacle center is transformed into the body frame; the signed distance to the rectangle is calculated from the positive parts of the two excess coordinates, with the interior term retained. Subtracting the disk radius gives footprint-to-obstacle clearance in Eq. (29). Negative clearance denotes intersection. The reference itself has a minimum footprint clearance of 0.85 m, which is checked before running the controllers. The nominal 0.25 m optimization margin is therefore substantially smaller than the geometric detour clearance, allowing the recorded results to distinguish route feasibility from tracking error around that route.

$$\begin{aligned} q_x &= |cos\,\psi\,\Delta x + sin\,\psi\,\Delta y| - L \\ q_y &= |-sin\,\psi\,\Delta x + cos\,\psi\,\Delta y| - W \\ clearance &= \sqrt{max(q_x, 0)^2 + max(q_y, 0)^2} + min\{max(q_x, q_y), 0\} - r_o \end{aligned} \tag{29}$$

The sinusoidal scenario lasts 30 s with X = 0.6t m and Y = 0.6 sin(2 pi X/10) m. Heading, forward speed and yaw rate are derived analytically from this time-parameterized path. The initial lateral displacement is minus 0.15 m. Since the tangent heading is nonzero at the origin, the initial signed cross-track error is not exactly the same as the imposed global lateral displacement. All controllers use identical initial pose, body velocity, steering and wheel-speed states. All scenarios use fixed observation intervals.

Paired optimization and plant perturbations

The forty-pair optimizer experiment freezes one sinusoidal prediction problem at t = 8 s. Its state is the reference state plus offsets of minus 0.2 m in X, plus 0.15 m in Y, plus 0.1 rad in heading, minus 0.1 m/s in forward velocity, plus 0.02 m/s in lateral velocity and plus 0.03 rad/s in yaw rate. The previous input is the reference body twist, and that twist is repeated across the six-command warm start. Every solver sees the same state, horizon, weights, bounds and active-obstacle flag. Seeds are the forty consecutive integers beginning at 202610234000, and the complete frozen problem is supplied as a JSON record.

For pair i, the primary effect is final CSA cost minus final SA cost, as defined in Eq. (30). A negative value favors CSA. We report both marginal medians and the median of the actual paired differences because those summaries are not interchangeable. A bootstrap interval for the paired median uses 20000 paired resamples with seed 202610234999. A two-sided Wilcoxon signed-rank test is also calculated from the forty differences. These statistics describe stochastic search on the one specified horizon problem. They are not treated as an inference over an unspecified population of vehicle models, reference paths or optimization landscapes.

$$\Delta_i = J_{CSA,i} - J_{SA,i} \tag{30}$$

The plant-perturbation experiment contains twelve cases, each simulated for 18 s of sinusoidal tracking under PID, SA-MPC and CSA-MPC. Payload is drawn uniformly from 1500 to 4500 kg, friction from 0.4 to 0.8, steering time constant from 0.10 to 0.25 s and motor resistance from 0.068 to 0.092 ohm. Yaw inertia scales in proportion to total mass. A zero- or one-sample command delay is selected for each case. Independent Gaussian measurement noise has standard deviations of 0.002 m in each position, 0.001 rad in heading, 0.003 m/s in each translational velocity and 0.002 rad/s in yaw rate. Within each case, controllers share the same plant parameters, delay and noise realization.

The deterministic sensitivity grid uses five payloads, from 1000 to 5000 kg in 1000 kg increments, and friction coefficients of 0.35, 0.45, 0.60, 0.75 and 0.90. Each of the twenty-five cells runs the actual nonlinear plant for 18 s using CSA-MPC and a common solver-seed stream. There is no measurement noise or command delay in this grid. The predictor and controller weights remain fixed while plant mass, inertia and friction change. The grid response follows from the specified model and excitation.

Metrics software and numerical checks

Signed cross-track error is the projection of position error onto the normal of the reference tangent. Along-track error is its tangent projection, and heading error is wrapped to the principal angular interval. Equation (31) defines mean absolute error, root-mean-square error and peak absolute error over all recorded samples, including startup. Euclidean position RMSE and terminal position error are also archived. These quantities answer different questions: a vehicle can have small cross-track error while lagging behind the time-parameterized reference. No statistical replication is inferred from the many time samples within a single nominal trajectory.

$$MAE = \frac{\sum_{i=1}^{M} |e_i|}{M}, \quad RMSE = \sqrt{\frac{\sum_{i=1}^{M} e_i^2}{M}} \qquad peak = max_i |e_i| \tag{31}$$

Actuator outcomes include maximum steering angle, sampled steering rate, current, voltage, wheel torque, contact-force utilization and longitudinal slip velocity. Electrical energy in Eq. (32) is integrated using midpoint motor power at each plant substep, rather than estimated only from the instantaneous current at a control update. Positive motor input is summed independently from net input. This distinction is useful when commands produce braking or current reversal. The energy calculation excludes auxiliary loads, battery losses, thermal behavior and cable deformation work, and is used only to compare the declared motor model under the same scenario.

$$E^{+} = \frac{1}{3600} \sum_{j=1}^{N_s} \sum_{i=1}^{4} h \, max(V_{ij} I_{ij}, 0) \quad (W\,h)$$
$$E_{net} = \frac{1}{3600} \sum_{j=1}^{N_s} \sum_{i=1}^{4} h \, V_{ij} I_{ij} \quad (W\,h) \tag{32}$$

For completeness, Eq. (33) states the exact normalized horizon objective assembled by the numerical kernel. All error and input coordinates use unit SI reference scales, and the terminal multiplier is two. The increment sum covers only the six optimized commands; the final command is held thereafter. Required wheel angles are evaluated before clipping, and the clearance term is included only when the obstacle is available. Thus the amplitude and rate projection, geometric penalty and tracking weights have separate mathematical roles and can be reconstructed independently from the parameter files.

$$
\begin{aligned}
20J \;=\; & \sum_{j=1}^{20} e_j^T Q e_j \\
& + \sum_{j=1}^{20} (u_j - u_{ref,j})^T R (u_j - u_{ref,j}) \\
& + \sum_{j=1}^{6} \Delta u_j^T S \Delta u_j \;+\; 2 e_{20}^T Q e_{20} \\
& + 100 \sum_{j=1}^{20} \sum_{i=1}^{4} max(|\delta_{req,ij}| - \delta_{max}, 0)^2 \\
& + 10000 \sum_{j=1}^{20} max(0.25 - clearance_j, 0)^2
\end{aligned}
\tag{33}
$$

The implementation combines Python orchestration with a C++17 numerical kernel. The tested software versions are Python 3.12, NumPy 2.3.5, SciPy 1.17.0, pandas 2.2.3 and Matplotlib 3.10.8. The archive includes the native source and build script, so the supplied shared library is not the only implementation. Optimizer timing covers the solver call and objective calculations, while the PID measurement covers its command update. These wall-clock measurements were obtained in a Linux x86_64 execution environment and are descriptive of that runtime; they exclude sensing, communications and target-hardware scheduling. Environment metadata and source hashes accompany the outputs.

Numerical checks include zero-input equilibrium, straight-command symmetry, finite-difference Jacobians, command and actuator bounds, friction-disk projection, and monotonic best-so-far optimizer traces. Integration sensitivity is assessed separately by replaying the archived sinusoidal CSA-MPC commands with a 0.0025 s plant step and comparing them with the 0.005 s trajectory. The controller is not reoptimized during this replay, which isolates the integration contribution from stochastic feedback changes. All generated trajectories, including perturbed cases and grid cells, are retained.

Results

Acceleration with a common scheduled target

Figure 15 compares forward velocity and its command under the common 0.8 m/s target. The PID response reaches a terminal speed of 0.8009 m/s and has 2.18 percent overshoot. SA-MPC and CSA-MPC finish at 0.7699 and 0.7769 m/s, respectively, while D-MPC finishes at 0.7864 m/s. The stochastic MPC responses are smoother in speed than in their underlying command sequence, illustrating the filtering supplied by motor and vehicle dynamics. Their zero positive overshoot should not be interpreted as superior regulation because a persistent negative speed error remains.

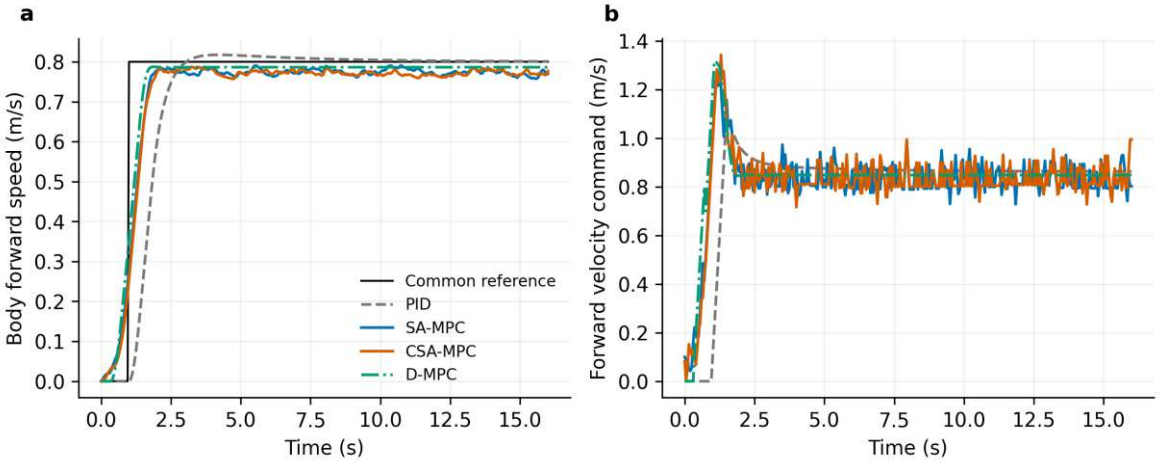


Figure 15. Common-target acceleration response over 16 s. Panel a compares the 0.8 m/s scheduled speed and actual body forward velocity. Panel b shows the issued velocity command. MPC can use the known schedule within its prediction window.

Table 7. Acceleration metrics at the same 0.8 m per second target.

Controller	t10 relative s	t90 relative s	10 to 90 s	2 percent settling s	Terminal m/s
PID	0.30	1.40	1.10	3.90	0.8009
SA-MPC	-0.35	0.70	1.05	Not reached	0.7699
CSA-MPC	-0.30	0.75	1.05	Not reached	0.7769
D-MPC	-0.35	0.50	0.85	0.75	0.7864

The threshold times in Table 7 are calculated at the same absolute speed levels, and the recorded relative times retain any anticipatory response caused by preview. This differs from comparing times to different controller-specific terminal values. PID remains inside the two-percent target band after 3.90 s following the step. D-MPC meets the same settling criterion after 0.75 s. SA-MPC and CSA-MPC do not satisfy it over the remaining 15 s observation interval. Thus an apparently short rise interval for a stochastic optimizer does not establish better final speed regulation. Its horizon cost and approximate velocity response can favor a command that improves the immediate prediction without eliminating the plant's steady mismatch.

The acceleration scenario also separates longitudinal timing from lateral tracking. The ideal symmetry of the PID and deterministic straight-line commands produces negligible lateral error, whereas stochastic command changes introduce small lateral excursions. These excursions remain within the reported wheel limits but are not a benefit of exploration. The acceleration results therefore provide an example in which derivative-free search can use a short computation time while producing a less desirable regulation outcome than a specified deterministic controller. They also show why no single response-time number is used to rank all controllers globally.

Turning response relative to the requested curvature

Figure 16 shows the turning paths and headings. All controllers follow the requested straight-to-circle transition and negotiate the approximately half-circle reference by the end of the 24 s record. Cross-track RMSE is 0.0207 m for PID, 0.0189 m for SA-MPC, 0.0200 m for CSA-MPC and 0.0176 m for D-MPC. The common initial lateral offset contributes to each full-record RMSE and sets a common initial maximum error. Differences in the terminal region should consequently be assessed alongside the shared startup condition rather than inferred from a visibly similar path plot alone.

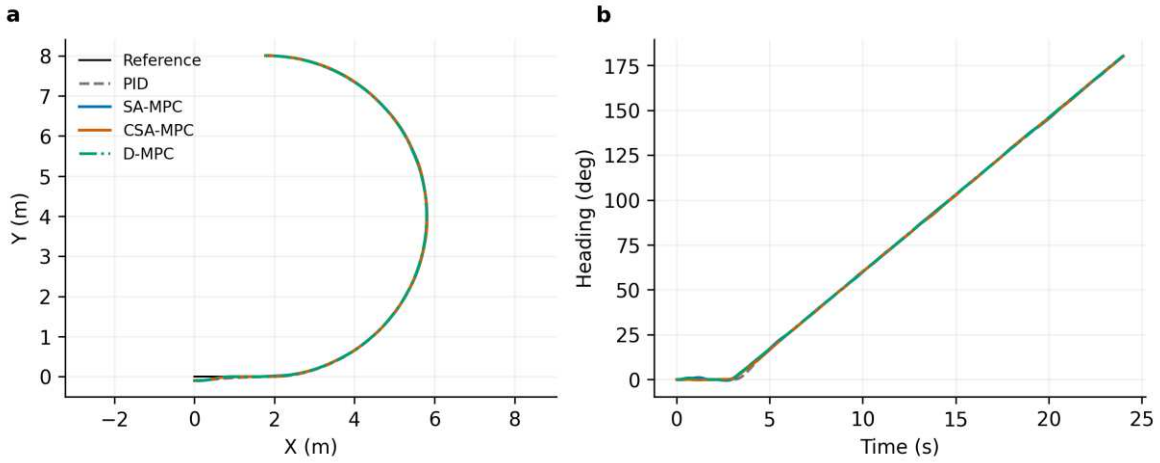


Figure 16. Tracking of the common 4 m radius turning reference. Panel a shows the paths and panel b the heading histories. All controllers use the same initial lateral displacement and reference speed.

Table 8. Turning results relative to the common 4 m radius.

Controller	Cross RMSE m	Terminal error m	Established radius m	Peak lateral m/s²

Controller	Cross RMSE m	Terminal error m	Established radius m	Peak lateral m/s²
PID	0.0207	0.0537	4.004	0.153
SA-MPC	0.0189	0.0184	3.990	0.249
CSA-MPC	0.0200	0.0197	4.027	0.272
D-MPC	0.0176	0.0109	4.001	0.317

The target radius is 4 m for every controller. The radius statistic in Table 8 uses established-turn speed and yaw rate, while heading and cross-track metrics describe the transition. CSA-MPC does not have the smallest turning RMSE in this experiment: both SA-MPC and D-MPC obtain smaller values. Its terminal position error is nevertheless substantially below that of PID. This combination indicates that the methods distribute error differently over time. A controller with a slightly worse integrated lateral error can still finish closer to the terminal reference, so terminal accuracy cannot replace an integrated measure of the complete maneuver.

Peak lateral force acceleration is small relative to the nominal friction capacity because the reference speed is low and the requested curvature is moderate. The contact model therefore operates away from its friction disk for most of the maneuver. This is consistent with the intended low-speed handling application, but it also limits what the turn can reveal about traction-limited behavior. The result supports a comparison of planar tracking and steering coordination at the specified curvature. It does not demonstrate a reduction in a physical rollover risk or establish that a smaller turning radius could be used safely with a real cable payload.

Tracking around the defined obstacle

Figure 17 includes the obstacle disk, selected oriented vehicle footprints and the clearance histories. The minimum footprint clearances are 0.8405 m for PID, 0.8362 m for SA-MPC, 0.8412 m for CSA-MPC and 0.8478 m for D-MPC. All are positive and exceed the 0.25 m penalty margin. No collision sample occurs in any of the four nominal obstacle runs. The common detour therefore provides sufficient geometric separation in the declared model, even though the tracking paths and wheel commands differ among controllers.

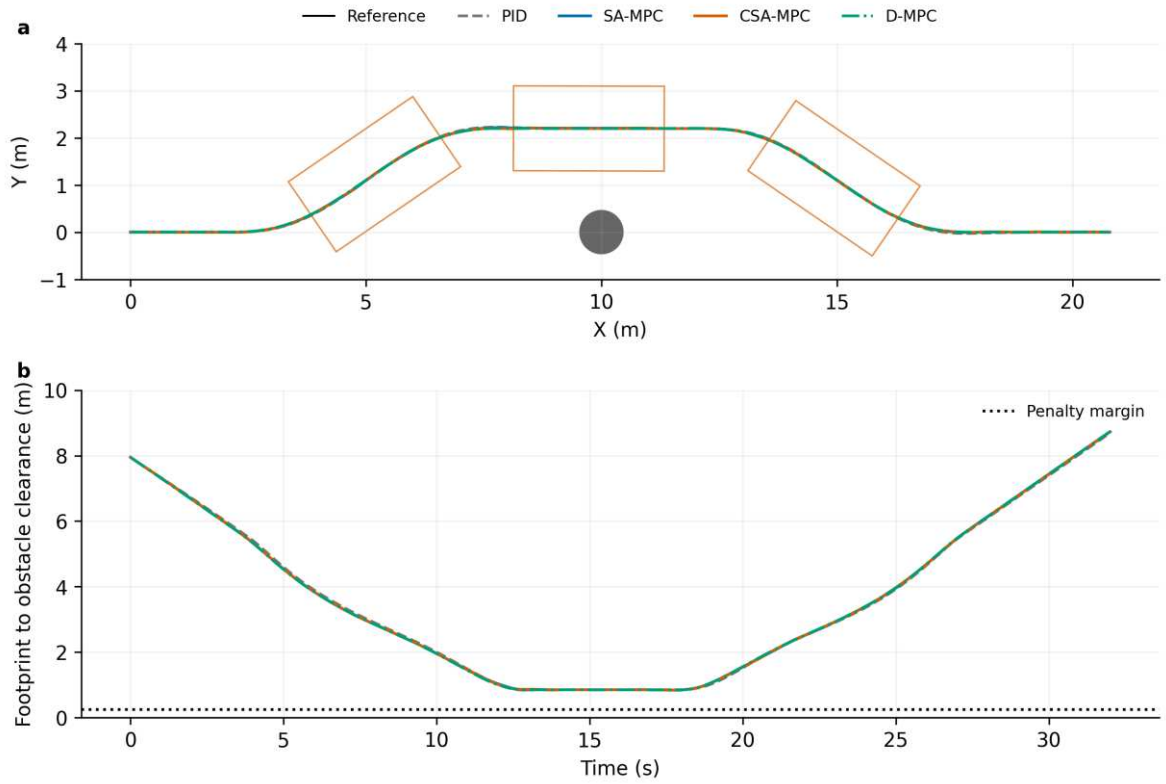


Figure 17. Tracking around the explicitly defined static obstacle. Panel a includes the disk, common route, controller paths and selected CSA-MPC vehicle footprints. Panel b reports signed footprint-to-disk clearance; the dotted line marks the 0.25 m penalty margin.

Table 9. Obstacle route tracking and full footprint clearance.

Controller	Cross RMSE m	Minimum clearance m	Collision samples	Terminal error m
PID	0.0139	0.8405	0	0.0566
SA-MPC	0.0064	0.8362	0	0.0248
CSA-MPC	0.0053	0.8412	0	0.0173
D-MPC	0.0015	0.8478	0	0.0110

Cross-track RMSE is 0.0139 m for PID, 0.0064 m for SA-MPC, 0.0053 m for CSA-MPC and 0.0015 m for D-MPC. CSA-MPC improves this metric relative to SA-MPC and PID, but D-MPC has the lowest error. The clearance differences are much smaller than the common route clearance. This means that the observed collision-free outcome is dominated by the shared detour geometry, rather than demonstrating that one optimizer independently discovers a safer avoidance path. The solver comparison is meaningful for tracking that route, while a perception or motion-planning comparison would require different experiments.

The recorded 2 s detection time is identical for all methods and is not estimated from the trajectory. Once the disk is available, each controller receives the same route definition. We therefore do not report a controller-specific obstacle detection or reaction latency. A curve that begins to move away from an obstacle at a different time can reflect reference following, vehicle dynamics or a different path position, and is not equivalent to a measured sensing response. Using the complete footprint and a shared information event removes that ambiguity from the present minimum-clearance assessment.

Sinusoidal tracking and distributed error

Figure 18 compares the sinusoidal trajectories and signed cross-track errors. The full-record cross-track RMSE is 0.0272 m for PID, 0.0227 m for SA-MPC, 0.0222 m for CSA-MPC and 0.0226 m for D-MPC. Relative to the specified PID baseline, CSA-MPC reduces this metric by 18.23 percent. Its reduction

relative to SA-MPC is only 2.27 percent. These are descriptive differences between the stated nominal runs, rather than estimates of a controller-population effect or a significance test based on trajectory samples.

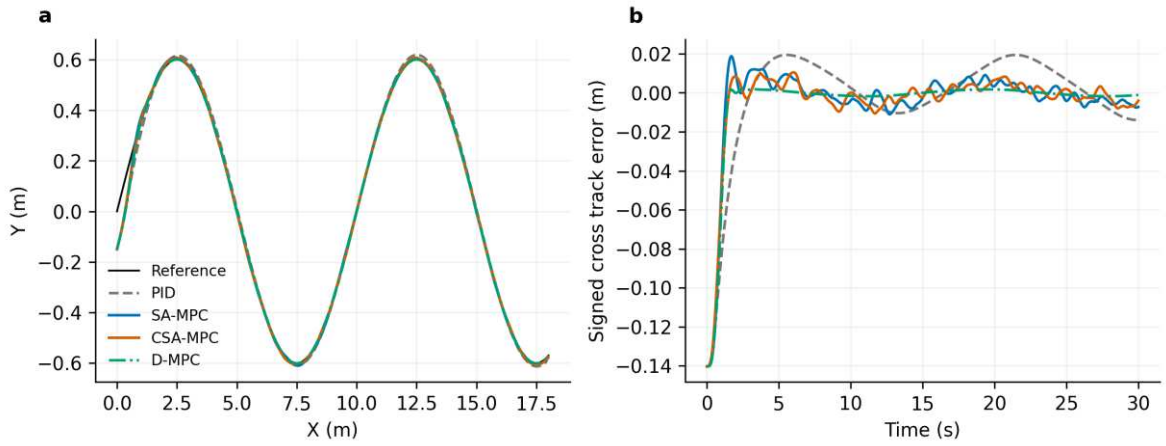


Figure 18. Sinusoidal-path tracking over the same 30 s interval. Panel a compares global paths, and panel b gives signed reference-normal position errors, including the common initial displacement.

Table 10. Sinusoidal tracking outcomes over the same 30 s record.

Controller	Cross MAE m	Cross RMSE m	Peak cross m	Position RMSE m	Terminal error m
PID	0.0154	0.0272	0.1404	0.0642	0.0542
SA-MPC	0.0087	0.0227	0.1404	0.0293	0.0202
CSA-MPC	0.0080	0.0222	0.1404	0.0292	0.0195
D-MPC	0.0055	0.0226	0.1404	0.0256	0.0109

The common initial displacement produces approximately the same peak cross-track error for all controllers. Subsequent differences therefore reflect how quickly and how smoothly that error is corrected and how closely later oscillations are followed. Mean absolute error, cross-track RMSE, Euclidean position RMSE and terminal error are reported separately in Table 10. D-MPC has a terminal Euclidean position error of 0.0109 m, compared with 0.0195 m for CSA-MPC. Hence the controller with the smallest cross-track RMSE does not also minimize every position-related outcome in the nominal sinusoid.

CSA-MPC and SA-MPC use higher-frequency command adjustments than D-MPC or PID. The vehicle and motor dynamics attenuate those adjustments in the path, but their actuator consequences remain visible later in the wheel and current plots. A nearly overlapping chassis trajectory is therefore insufficient to conclude that two controllers have equivalent implementation demands. The sinusoidal experiment is particularly useful for this distinction because the reference curvature changes continuously, requiring repeated coordination of front and rear steering while maintaining forward progress over a common 30 s interval.

Forty paired horizon optimizations

Figure 19 shows median best-so-far convergence with tenth-to-ninetieth-percentile bands and the final objective distributions. The marginal median final objectives are 1.34903 for SA, 1.33574 for CSA and 1.35752 for the random sixteen-candidate initializer. CSA's marginal median is 0.98 percent lower than SA's on this frozen horizon. Every annealing run uses exactly 300 evaluations, including initialization. The convergence histories contain 36000 evaluation-indexed records across the forty seeds and three methods, so early progress can be assessed without omitting the initial pool's computational cost.

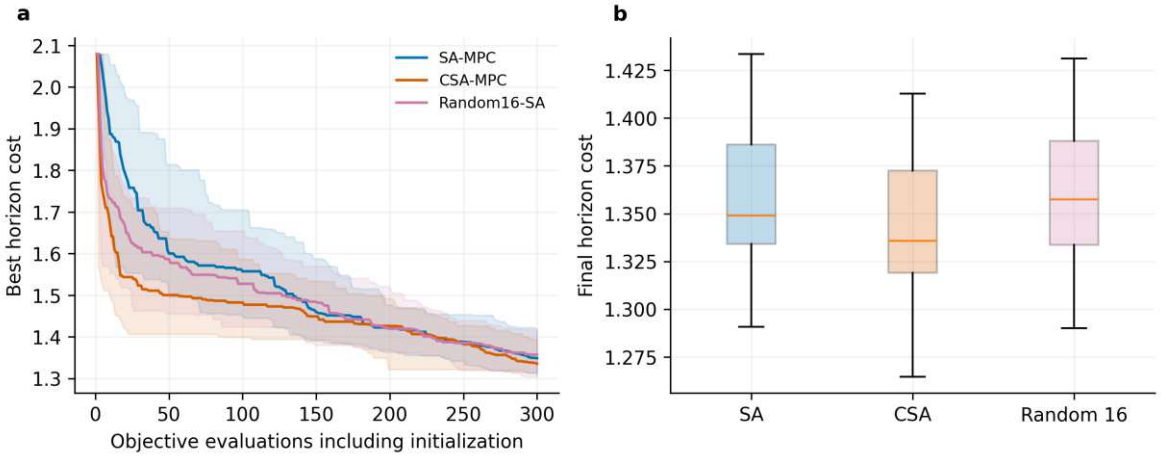


Figure 19. Forty seeded optimizations of one frozen MPC horizon with 300 evaluations per run. Panel a shows median best-so-far cost and 10th–90th percentile bands, including initialization. Panel b summarizes final objective values for SA, CSA and the random sixteen-candidate initializer.

Table 11. Forty paired frozen horizon optimizations and their uncertainty.

Outcome	Value
SA marginal median	1.349026
CSA marginal median	1.335744
Random sixteen-candidate median	1.357516
CSA wins; SA wins; ties	22; 18; 0
CSA win fraction	55.0 percent
Median paired CSA minus SA	-0.008175
Paired median bootstrap interval	[-0.032507, 0.004469]
Two-sided Wilcoxon p value	0.146089
Budget and paired instances	300 evaluations; 40 seeds on one fixed problem

CSA obtains a lower final objective in 22 pairs and SA in 18 pairs, with 0 numerical ties. The CSA win fraction is 55.0 percent. The median of the actual CSA-minus-SA paired differences is -0.00818, with bootstrap interval from -0.03251 to 0.00447. The two-sided Wilcoxon p value is 0.146. Since the paired interval includes zero and the test does not identify a strong difference at the conventional 0.05 threshold, the results do not support a statistically established optimization advantage for CSA on this problem. The modest marginal-median improvement and the paired uncertainty are both retained in the interpretation.

The random sixteen-candidate method has a slightly higher marginal median than both SA and CSA in this experiment. This outcome shows that the benefit of evaluating several initial points is not automatic under a fixed total allowance: initial screening reduces the number of subsequent proposals. It also prevents the early CSA curve from being interpreted solely as evidence that chaos covers the complete eighteen-dimensional domain more effectively. The initializer, its local window, the proposal mixture and the finite budget act together. A broader claim would require additional frozen problems varying curvature, initial error, active constraints and model dimensions.

Paired plant perturbations and payload grid

Figure 20 plots all twelve perturbed-plant cases and their paired CSA-minus-SA RMSE differences. Median cross-track RMSE is 0.0337 m for PID, 0.0298 m for SA-MPC and 0.0297 m for CSA-MPC. CSA-MPC has a smaller RMSE than PID in all twelve cases, but it improves over SA-MPC in only six. One case gives a more substantial CSA improvement, while several cases show very small differences in either

direction. The paired points make that heterogeneity visible instead of implying that the three marginal medians describe every operating condition.

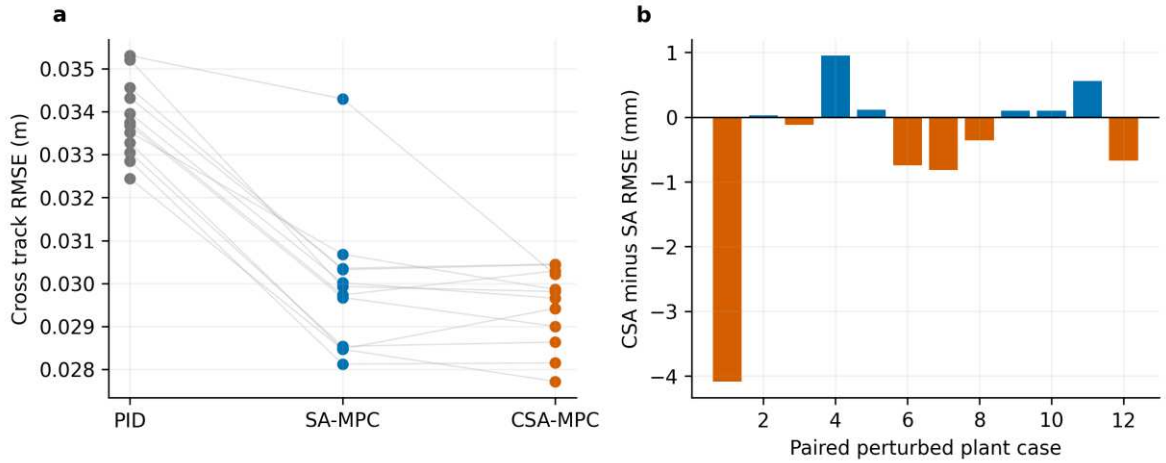


Figure 20. Actual nonlinear-plant perturbation experiment. Panel a connects the three controller RMSE values within each of twelve common parameter, delay and noise cases. Panel b shows CSA-minus-SA RMSE for each case; negative values favor CSA.

Table 12. Twelve actual perturbed plant cases using common noise and delay.

Controller	Median RMSE m	Minimum RMSE m	Maximum RMSE m	Cases
PID	0.0337	0.0324	0.0353	12
SA-MPC	0.0298	0.0281	0.0343	12
CSA-MPC	0.0297	0.0277	0.0305	12

These records are complete closed-loop simulations after changing the nonlinear plant and adding the specified noise and command delay. The outcome therefore supports conditional robustness of the specified controllers within this limited parameter family. It does not identify the true distribution of payloads or floor friction in a cable-installation site, because the uniform parameter ranges are benchmark design choices. The comparison also retains one fixed controller tuning and reduced predictor, so adaptation to identified payload or friction is not included.

Figure 21 reports the twenty-five-cell payload–friction grid. Cross-track RMSE ranges from 0.0275 to 0.0301 m and increases across the stated payload levels. At each payload, the RMSE is unchanged across the tested friction coefficients to the stored numerical precision. Maximum contact-force utilization remains below 0.276. Friction capacity is consequently inactive for this low-demand reference, even at the smallest coefficient. The flat error direction is a physically interpretable outcome of the specified force projection and excitation, rather than evidence that friction is irrelevant to every AGV maneuver.

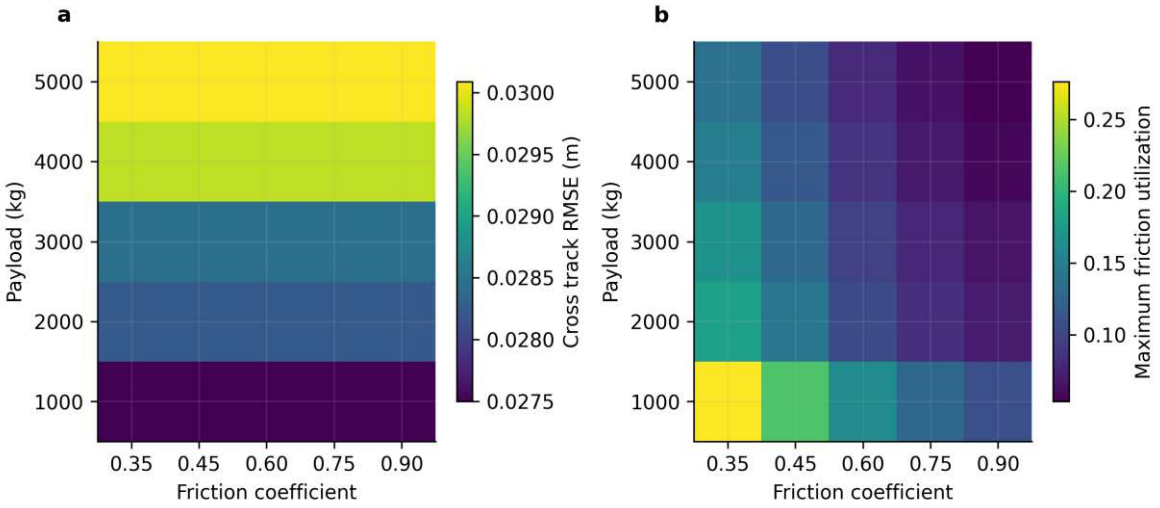


Figure 21. Twenty-five executed payload–friction grid cells for 18 s sinusoidal CSA-MPC runs. Panel a shows cross-track RMSE and panel b maximum friction utilization. The unchanged RMSE across friction at a given payload reflects an inactive force cap.

The payload effect is associated with fixed motor capability, proportional inertia scaling and a predictor that does not adapt to the changed plant. Increasing available friction cannot improve a trajectory whose demanded force remains well inside the friction disk. A more aggressive maneuver, a lower coefficient or a braking event would be needed to expose the active boundary. The grid is therefore a screening result for the declared low-speed reference. It is not extrapolated into a rollover envelope, and no smooth contour is used to imply additional simulated cells between the twenty-five archived parameter combinations.

Steering actuation and contact demand

Figure 22 shows the four steering angles and sampled rates for nominal sinusoidal CSA-MPC. The wheel angles change sign as path curvature changes, with front and rear wheels responding differently through their signed longitudinal lever arms. All recorded angles remain within 35 degrees, and the sampled steering rates remain within 40 degrees/s. The rate traces often approach the imposed limit because the stochastic command sequence changes more rapidly than the steering actuator can follow. This behavior is compatible with a smooth chassis path but represents a substantial actuator-demand characteristic of the chosen input-increment weighting and search budget.

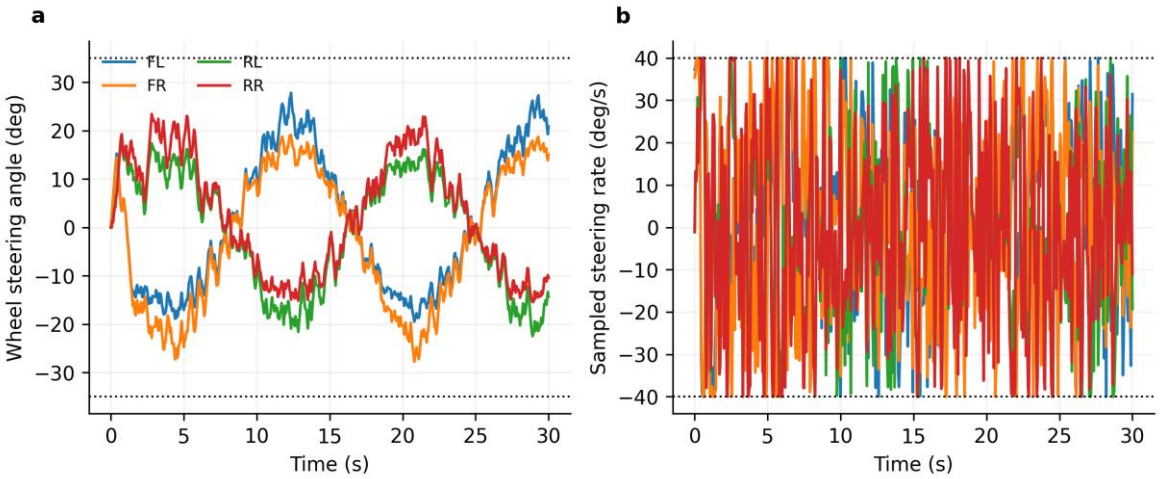


Figure 22. Wheel steering during nominal sinusoidal CSA-MPC. Panel a gives four actual steering angles and panel b sampled rates. Dotted lines mark ±35 degrees and ±40 degrees/s. The rate activity is retained without smoothing away command-induced variation.

Figure 23 gives motor current and terminal voltage at the control samples. Current reversals and near-limit samples occur during repeated wheel-speed corrections, while terminal voltage remains inside the 48 V limit. The current and torque values come from the electrical and wheel-rotation equations, so they can be traced to a particular command, wheel speed and simulated time. They are not independent hardware measurements. Their significance in the comparison is that a controller's apparently small path improvement can coexist with more active electrical control and a higher modeled input-energy requirement.

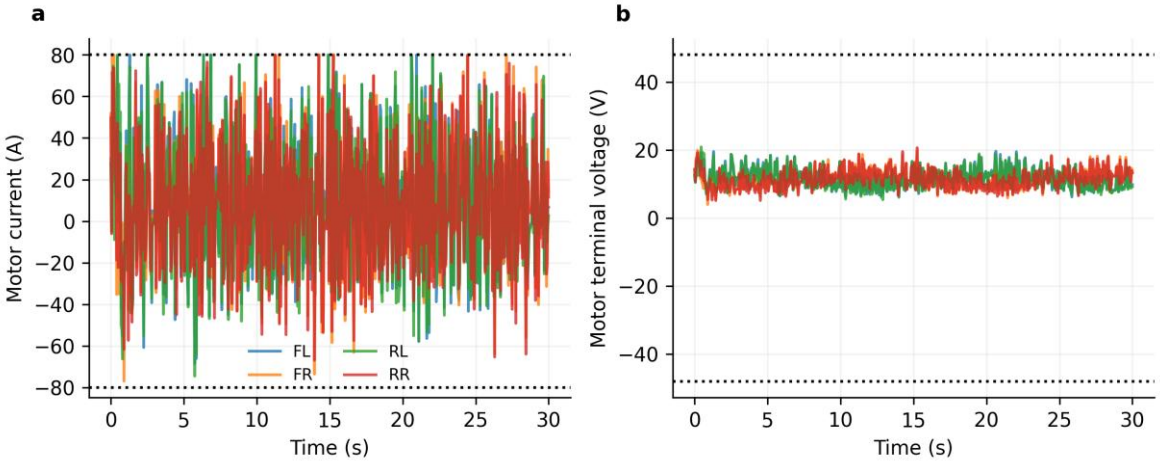


Figure 23. Motor electrical variables during nominal sinusoidal CSA-MPC. Panel a shows current and panel b terminal voltage. Dotted lines mark the model limits of ±80 A and ±48 V. Values are instantaneous control-sample diagnostics.

Figure 24 shows combined contact-force utilization and longitudinal slip velocity. The sinusoidal trajectory remains far from the friction cap, consistent with the grid result. Nonzero slip velocity is nevertheless present because the regularized longitudinal force law requires a difference between wheel peripheral speed and contact velocity to transmit force. A low utilization ratio therefore does not mean that wheel speed is exactly the no-slip kinematic value. This distinction explains why the reduced predictor can maintain a systematic forward-velocity residual even when the force limits are inactive and the geometric tracking error is small.

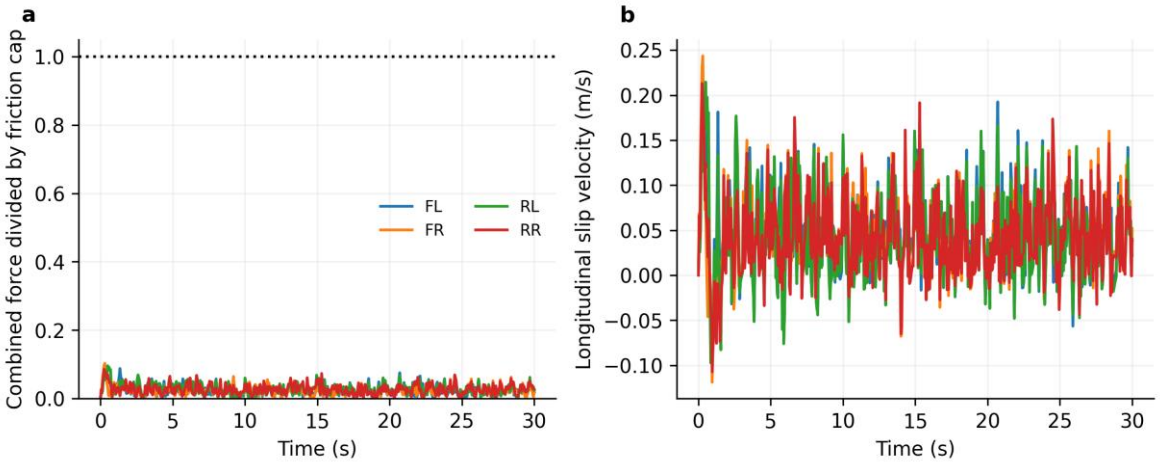


Figure 24. Wheel contact variables during nominal sinusoidal CSA-MPC. Panel a normalizes combined wheel force by its friction-disk cap. Panel b gives the longitudinal difference between peripheral wheel speed and contact velocity.

Computation paired effects and numerical accuracy

Figure 25 compares update-time distributions pooled over the four nominal scenarios. PID has the lowest measured update time, the native annealing kernels are typically below one millisecond per update, and D-MPC requires a substantially longer solver call in this implementation. The observed maximum times and counts above the 50 ms sample interval are retained in Table 13, including occasional

deterministic-solver excursions. These measurements are influenced by the runtime, finite-difference wrapper and native implementation choices. They do not establish that one algorithm will have the same ratio on an embedded processor or that unmeasured sensing and communication tasks fit inside the remaining sample time.

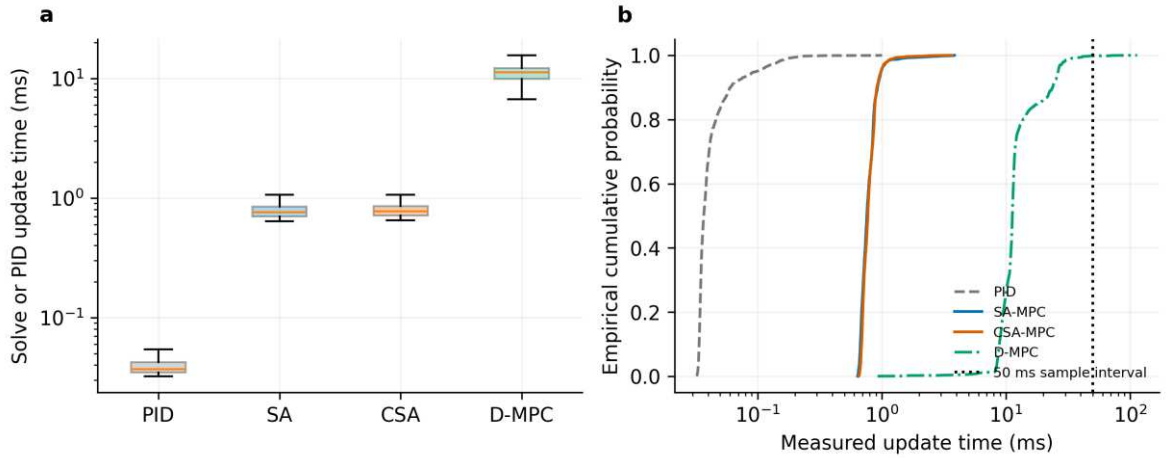


Figure 25. Measured command-update times pooled across the four nominal scenarios. Panel a shows distributions with a logarithmic axis and omitted boxplot outlier markers; panel b shows complete empirical cumulative distributions, including long updates. The vertical line marks the 50 ms sample interval.

Table 13. Measured update times pooled across four nominal runs.

Controller	Mean ms	95th percentile ms	Maximum ms	Over 50 ms	Updates
PID	0.047	0.099	1.004	0	2040
SA-MPC	0.799	0.988	3.854	0	2040
CSA-MPC	0.800	0.987	3.658	0	2040
D-MPC	13.173	26.366	113.928	4	2040

Figure 26 displays every paired final-objective difference from the frozen optimizer experiment. Both positive and negative differences occur over a range that is not represented by the marginal median alone. The distribution supports reporting a small central CSA improvement with appreciable seed dependence, rather than declaring uniform superiority. It also illustrates why marginal percentage differences, paired median differences and win fractions must be kept distinct. The raw forty-row paired file preserves the numerical values so that alternative effect summaries or an explicitly specified tie tolerance can be evaluated without digitizing a plot.

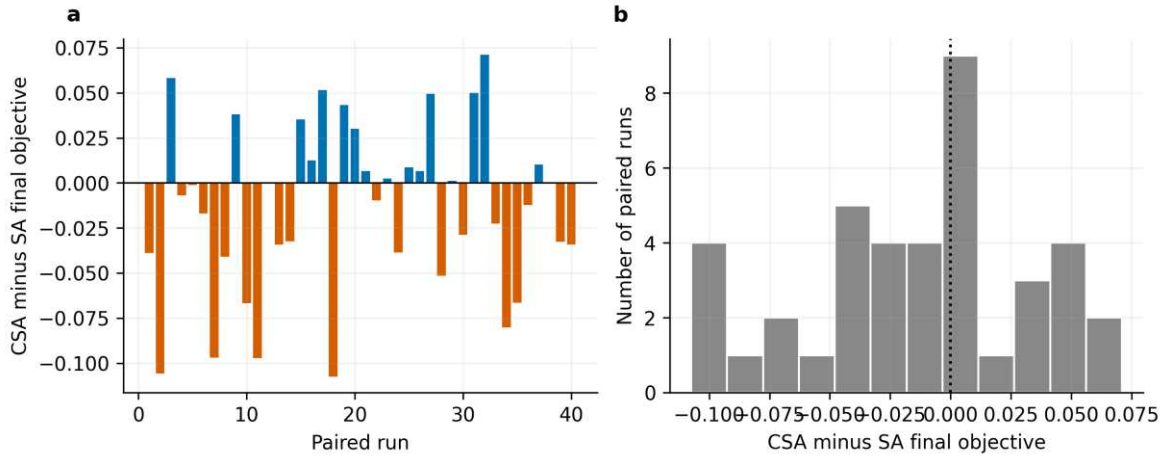


Figure 26. All forty CSA-minus-SA final-objective differences from the frozen horizon experiment. Panel a retains pair order and panel b summarizes the difference distribution. Negative values favor CSA and positive values favor SA.

Figure 27 reports the integration-step replay. Reducing the plant step from 0.005 to 0.0025 s while holding the archived commands fixed produces a maximum position difference of 3.467e-05 m and maximum heading difference of 2.978e-06 rad. These differences are small relative to the centimeter-scale tracking outcomes. They provide evidence that the nominal plant integration step is adequate for the displayed trajectories under the archived input sequence. They do not replace a model-validation experiment, and they do not measure the combined effect of numerical refinement and reoptimized stochastic feedback.

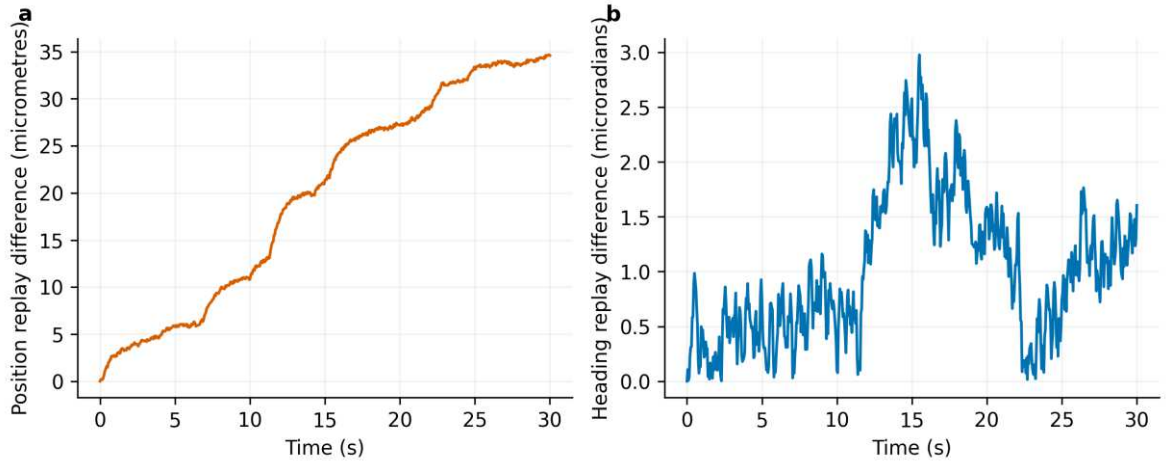


Figure 27. Plant-integration refinement under identical archived CSA-MPC commands. Panel a shows position differences between 0.005 and 0.0025 s midpoint steps; panel b shows heading differences. This is an input replay, not a reoptimized controller comparison.

Table 14. Numerical and implementation verification outcomes.

Check	Observed value	Criterion
Zero-input equilibrium	0	State change below 10^{-12}
Straight-command symmetry	0	Lateral and yaw change below 10^{-10}
Predictor Jacobian error	1.63e-10	Below 10^{-7}
Reference minimum clearance	0.850 m	At least 0.25 m
Maximum nominal steering angle	34.931 degrees	At most 35 degrees
Maximum nominal steering rate	40.000 degrees/s	At most 40 degrees/s

Check	Observed value	Criterion
Maximum nominal current	80.000 A	At most 80 A
Maximum nominal force utilization	0.1491	At most 1
Optimizer trace rows	36000	120 runs × 300 evaluations
Refinement maximum position difference	3.47e-05 m	Identical archived input replay

Tracking and electrical input tradeoffs

Figure 28 combines cross-track RMSE with positive electrical input energy in the sinusoidal scenario. The modeled energies are 2.945 W h for PID, 5.579 W h for SA-MPC, 5.367 W h for CSA-MPC and 3.187 W h for D-MPC. CSA-MPC therefore uses more modeled positive electrical input than PID and D-MPC despite its lower cross-track RMSE. The cost penalizes command magnitude and increments, but it does not directly minimize electrical energy. Its numerical optimum should not be interpreted as an energy optimum or a battery-life improvement.

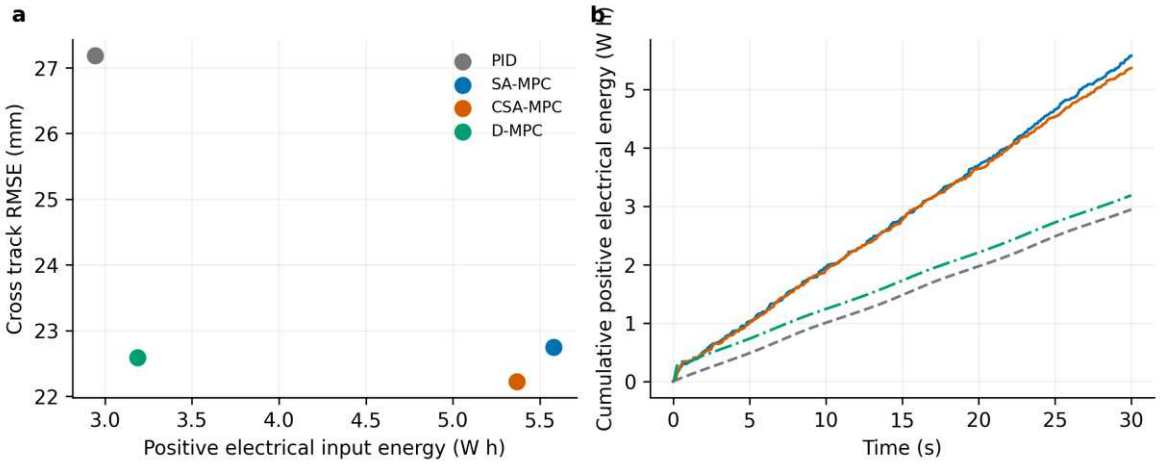


Figure 28. Nominal sinusoidal tracking and modeled electrical-input tradeoff. Panel a compares cross-track RMSE with positive motor input energy, while panel b shows cumulative positive energy integrated at plant substeps.

The energy accumulation curves are evaluated from internal-step motor power and include the consequences of high-frequency corrections that are visually attenuated in the chassis path. The comparison suggests two practical changes for future controller design: represent an actuator-consistent energy term in the prediction objective, and examine whether increasing the increment penalty or using a smoother parameterization reduces unnecessary electrical activity. These changes were not used to retune the reported experiment. Their value would have to be evaluated in a new matched comparison, because reducing command activity can also alter tracking, settling and the optimizer landscape.

Figure 29 gives one-step prediction residuals for the nominal sinusoidal CSA-MPC trajectory. The forward channel has a visible systematic component, while the lateral and yaw residuals vary with wheel coordination and steering response. The reduced model lacks the complete wheel slip and motor dynamics of the plant, so this mismatch is expected under the declared modeling hierarchy. The residual files identify where a more detailed predictor or an identified disturbance term could be useful. They also explain why obtaining a smaller reduced-model horizon cost cannot, by itself, guarantee the smallest realized tracking error or eliminate the acceleration scenario's speed bias.

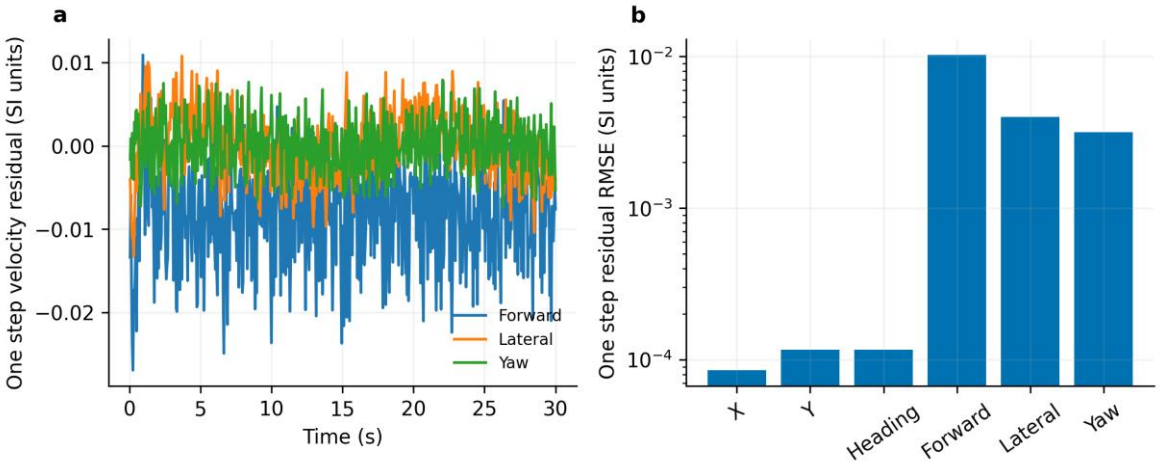


Figure 29. One-step reduced-predictor residuals on the nominal sinusoidal CSA-MPC trajectory. Panel a shows forward, lateral and yaw velocity residuals. Panel b gives residual RMSE by state; each channel is expressed in its corresponding SI unit.

Discussion

The principal result is that solver quality, chassis tracking and actuator demand do not share one universal ranking. CSA has a slightly lower marginal final cost than SA on the specified frozen horizon, yet the paired uncertainty includes zero and its win fraction is only modestly above one half. In closed loop it improves sinusoidal and obstacle-route cross-track RMSE relative to the stated SA run, but has a larger turning RMSE. The perturbed-plant comparison is similarly mixed between the two annealing methods. These findings support evaluating chaos initialization as a conditional numerical design choice rather than treating deterministic irregularity as a general source of control robustness.

The frozen optimizer experiment and the closed-loop experiments answer different questions. Freezing the state, previous command and reference makes the forty-pair comparison interpretable as a solver assessment. Once the first input is applied to the plant, SA-MPC and CSA-MPC visit different states, so their later objective values belong to different optimization problems. Comparing their complete objective histories sample by sample would then mix search quality with the states created by earlier commands. For that reason the paper uses a matched frozen problem for paired solver statistics and physical trajectory metrics for the closed-loop comparison. Neither analysis is substituted for the other.

Including D-MPC changes the interpretation of the PID comparison. D-MPC gives the smallest obstacle-route tracking error and a small terminal position error while requiring more measured computation in this implementation. PID regulates the common speed target more effectively than either stochastic MPC response over the observed acceleration interval and uses less modeled electrical input in the sinusoid. These outcomes indicate that a useful AGV design decision depends on the required balance between accuracy, regulation, energy and computation. A claim that CSA-MPC is globally superior would discard favorable baseline outcomes that remain important for a practical controller.

The energy result also shows the limitations of using an increment penalty as a surrogate for physical effort. Penalizing a change in body command discourages rapid variation, but the mapping from that change to current depends on wheel geometry, steering lag, motor back-emf and contact slip. A small command perturbation can generate appreciable electrical activity when its direction or timing changes the wheel-speed target. Conversely, some smooth commands can require sustained torque under rolling resistance or a heavy payload. An energy-aware extension should therefore use either a suitably reduced actuator model or a verified approximation derived from the same plant, with its prediction error reported alongside any energy claim.

The low-friction grid requires equally careful interpretation. Since maximum force utilization is below the cap, changing the cap does not change the applied forces and the trajectories remain identical across

friction coefficients at a given payload. This follows directly from the implemented saturation law. The result demonstrates that sensitivity depends on excitation and on whether a modeled condition becomes active. Testing a heavy AGV only on gentle reference paths can leave important traction limitations unexamined. Future work should include controlled braking, larger curvature demand and independently varied disturbance forces before discussing a broader friction-dependent operational envelope.

The obstacle result is primarily a tracking result around an explicitly feasible route. A static disk, known detection event and shared smooth detour remove uncertainty about what each controller is asked to follow. This is useful for isolating control behavior, but it does not test missed detections, object classification, moving obstacles or route replanning under occlusion. The complete footprint prevents a point-robot clearance error, yet the obstacle geometry is still idealized and the contact threshold is deterministic. A deployment study would need measured perception timing and uncertainty, an appropriate stopping supervisor and scenario-specific validation. Those additions should be evaluated on top of the declared control benchmark rather than assumed from a positive simulated clearance.

The planar plant likewise has a clear physical boundary. Equal wheel loads and fixed planar inertia are adequate for a specified force-allocation comparison, but they omit the load-transfer mechanisms studied in heavy-duty AGV literature [20,23,24]. A real cable payload may shift its center of mass, deform its support or introduce forces that are absent from this plant. Steering compliance, floor discontinuities, transmission backlash and motor temperature can also affect the realized motion. Introducing any of those effects would require new parameters and corresponding validation data.

The stochastic search design offers several directions for a more extensive numerical study. The current horizon has only eighteen decision coordinates, one warm-start strategy and one fixed evaluation budget. Other budgets could change the balance between a sixteen-point initialization and subsequent proposals. A larger control horizon could expose correlations in the logistic sequence or make a local warm-start window more consequential. Additional frozen problems with active steering penalties, near-obstacle states and different predictor mismatch would test whether the observed paired distribution is representative. These extensions should use problem-level replication and maintain evaluation accounting.

The parameter-perturbation study is more informative than adding noise to an already summarized error, because every case regenerates the physical trajectory and actuator signals. Nevertheless, twelve cases provide only limited coverage of a multidimensional parameter space. Their uniform sampling distributions are not measured operational frequencies, and the selected one-sample delay does not represent every sensing or network condition. The paired design reduces nuisance differences between controllers, but it does not turn the case family into a field-reliability distribution. Larger studies could use structured coverage, identify the important interactions and include failures as explicit outcomes, while retaining the complete records as done here.

Numerical Jacobian checks and integration replay support internal consistency of that implementation. They do not establish that a particular real vehicle has the assumed motor constants, contact coefficients or inertia. Experimental identification and independent trajectory measurements would be needed for that step. A useful next stage is to replace selected design assumptions with measured parameters while keeping the code, scenario definitions and comparison protocol fixed, making the effect of calibration visible.

For high-voltage cable handling, the immediate implication is to choose a controller by explicit operating requirements. If terminal speed regulation and low electrical activity dominate, the reported PID and deterministic outcomes deserve consideration. If a very short native solver call and centimeter-scale lateral tracking are preferred, annealing-based MPC is a candidate under the tested conditions, with its residual bias and command activity still requiring attention. The chaos component provides no general guarantee that resolves that tradeoff. The benchmark makes the alternatives concrete enough for subsequent actuator, hardware-in-the-loop and loaded-platform experiments to test them without changing the meaning of the reported metrics.

Conclusions

A complete numerical benchmark was constructed for four-wheel-steering cable-handling vehicle control, including a fourteen-state nonlinear plant, motor and contact equations, a six-state MPC predictor, numerical matrices, PID gains, reference trajectories and obstacle geometry. Four nominal scenarios, forty paired horizon optimizations, twelve paired plant-perturbation cases and twenty-five payload–friction runs were executed and archived. This specification connects the reported controller outcomes to inspectable model and software inputs.

CSA produced a slightly lower marginal median objective than SA on the frozen MPC problem and won 22 of 40 pairs, but the paired interval included zero. Its closed-loop benefits were also conditional: sinusoidal cross-track RMSE was lower than PID and SA-MPC, whereas turning RMSE, speed settling and electrical input did not show a uniform advantage. Deterministic MPC offered strong tracking with a longer measured solver call, while the fixed PID baseline retained favorable regulation and energy outcomes. The results support a transparent tradeoff assessment rather than a universal claim of chaos-enhanced optimality.

The numerical checks support internal consistency within the declared planar model. The inactive friction boundary, prediction residuals and actuator activity identify specific conditions that a more detailed model or physical experiment should examine. Future validation should calibrate the plant and actuator parameters, include relevant load-transfer and cable effects, and verify computation and clearance under representative operating conditions. The supplied code and raw outputs provide a defined basis for those extensions.

Funding

This work was supported by State Grid Fujian Electric Power Co., Ltd., through “A Study on Clustered Operation Technology for High-Voltage Cable Installation Robots” (52134024000R).

Author contributions

W.L. and Y.Z. conceived the engineering problem. Y.Z., W.L. and F.W. compiled vehicle and crane parameters. Z.L. and F.W. reviewed construction and cost constraints. All authors reviewed the analytical framework and approved the manuscript. The final contribution statement should be confirmed by all authors before submission.

Competing interests

The authors declare no competing interests.

Data availability

All data generated or analysed during this study are included in this published article.

References

1. Lackner, T., Hermann, J., Kuhn, C. & Palm, D. Review of autonomous mobile robots in intralogistics: State-of-the-art, limitations and research gaps. Procedia CIRP 130, 930–935 (2024). https://doi.org/10.1016/j.procir.2024.10.187

2. International Organization for Standardization. ISO 3691-4:2023 Industrial trucks—Safety requirements and verification—Part 4: Driverless industrial trucks and their systems, 2nd edn (ISO, 2023).

3. Liu, L., Wang, X., Yang, X., Liu, H., Li, J. & Wang, P. Path planning techniques for mobile robots: Review and prospect. Expert Syst. Appl. 227, 120254 (2023). https://doi.org/10.1016/j.eswa.2023.120254

4. Loganathan, A. & Ahmad, N. S. A systematic review on recent advances in autonomous mobile robot navigation. Eng. Sci. Technol. Int. J. 40, 101343 (2023). https://doi.org/10.1016/j.jestch.2023.101343

5. Brunke, L. et al. Safe learning in robotics: From learning-based control to safe reinforcement learning. Annu. Rev. Control Robot. Auton. Syst. 5, 411–444 (2022). https://doi.org/10.1146/annurev-control-042920-020211

6. Li, Y. & Liu, L. Physics-informed neural network-based nonlinear model predictive control for automated guided vehicle trajectory tracking. World Electr. Veh. J. 15, 460 (2024). https://doi.org/10.3390/wevj15100460

7. Feng, J., Yang, Y., Zhang, H., Sun, S. & Xu, B. Path planning and trajectory tracking for autonomous obstacle avoidance in automated guided vehicles at automated terminals. Axioms 13, 27 (2024). https://doi.org/10.3390/axioms13010027

8. Dong, D., Ye, H., Luo, W., Wen, J. & Huang, D. Collision avoidance path planning and tracking control for autonomous vehicles based on model predictive control. Sensors 24, 5211 (2024). https://doi.org/10.3390/s24165211

9. Manikandan, S., Ganesan, K., Hakak, S. & Gadekallu, T. R. Curve-aware model predictive control trajectory tracking for automated guided vehicle over on-road, indoor and agricultural land. Sustainability 14, 12021 (2022). https://doi.org/10.3390/su141912021

10. Liu, W., Wan, Y., Zhang, Q., Yu, Y., Liu, P. & Shi, Z. Trajectory tracking control of four-wheel steering automatic guided vehicle under the working condition of moving centroid. Proc. Inst. Mech. Eng. D 237, 691–705 (2022). https://doi.org/10.1177/09544070221082338

11. Xie, Y., Li, C., Jing, H., An, W. & Qin, J. Integrated control for path tracking and stability based on model predictive control for four-wheel independently driven electric vehicles. Machines 10, 859 (2022). https://doi.org/10.3390/machines10100859

12. Fu, W., Liu, Y. & Zhang, X. Research on accurate motion trajectory control method of four-wheel steering AGV based on Stanley-PID control. Sensors 23, 7219 (2023). https://doi.org/10.3390/s23167219

13. Wang, J., Fader, M. T. H. & Marshall, J. A. Learning-based model predictive control for improved mobile robot path following using Gaussian processes and feedback linearization. J. Field Robot. 40, 1014–1033 (2023). https://doi.org/10.1002/rob.22165

14. Li, Y. et al. Trajectory tracking of four-wheel driving and steering autonomous vehicle under extreme obstacle avoidance condition. Veh. Syst. Dyn. 62, 601–622 (2023). https://doi.org/10.1080/00423114.2023.2186249

15. Aizat, M., Qistina, N. & Rahiman, W. A comprehensive review of recent advances in automated guided vehicle technologies: Dynamic obstacle avoidance in complex environments toward autonomous capability. IEEE Trans. Instrum. Meas. 73, 1–25 (2023). https://doi.org/10.1109/TIM.2023.3338722

16. Ćaran, B., Milić, V., Švaco, M. & Jerbić, B. Optimal control-based algorithm design and application for trajectory tracking of a mobile robot with four independently steered and four independently actuated wheels. Actuators 13, 279 (2024). https://doi.org/10.3390/act13080279

17. Chu, Y., Wang, C., Zhou, X., Zhang, Z. & Zhao, W. Instantaneous center of rotation tracking control of four-wheel independent steering vehicles under large-curvature turning conditions. IEEE Trans. Intell. Transp. Syst. 25, 17965–17978 (2024). https://doi.org/10.1109/TITS.2024.3432885

18. Li, S., Yang, Z. & Li, B. Four-wheel steering vehicle trajectory tracking control based on PSO-optimized MPC. Phys. Scr. 99, 095252 (2024). https://doi.org/10.1088/1402-4896/ad6d16

19. Min, T., Lin, S. & Luo, Y. Mecanum wheel AGV trajectory tracking control based on efficient MPC algorithm. IEEE Access 12, 13763–13772 (2024). https://doi.org/10.1109/ACCESS.2024.3356583

20. Fu, W., Wang, X. & Zhang, X. Rollover stability of heavy-duty AGVs in turns considering variation in friction coefficient. Lubricants 11, 119 (2023). https://doi.org/10.3390/lubricants11030119

21. Jneid, M. S. & Harth, P. Integrated torque vectoring control using vehicle yaw rate and sideslip angle for improving steering and stability of all off-wheel-motor drive electric vehicles. Acta Polytech. Hung. 21, 87–106 (2024). https://doi.org/10.12700/APH.21.7.2024.7.6

22. Wan, Z., Xu, C., Li, B., Li, Y. & Ye, F. Trajectory tracking method of four-wheeled independent drive and steering AGV based on LSTM-MPC and fuzzy PID cooperative control. Electronics 14, 2000 (2025). https://doi.org/10.3390/electronics14102000

23. Li, X., Chen, S., Chen, X., Liu, B., Wang, C. & Su, Y. Trajectory tracking control strategy of 20-ton heavy-duty AGV considering load transfer. Appl. Sci. 15, 4512 (2025). https://doi.org/10.3390/app15084512

24. Li, X., Chen, X., Chen, S., Liu, B. & Wang, C. Trajectory tracking and driving torque distribution strategy for four-steering-wheel heavy-duty automated guided vehicles. Machines 13, 383 (2025). https://doi.org/10.3390/machines13050383

25. Zhang, L., Zhang, T., An, Y., Niu, T., Wang, S. & Wang, J. 4WIDS-MAPF: Multiagent pathfinding for four-wheel independent drive/steering robots. IEEE Internet Things J. 12, 38795–38803 (2025). https://doi.org/10.1109/JIOT.2025.3587737

26. Kirkpatrick, S., Gelatt, C. D., Jr. & Vecchi, M. P. Optimization by simulated annealing. Science 220, 671–680 (1983). https://doi.org/10.1126/science.220.4598.671