LEAN: Local Energy-Aware Navigation for Skid-Steer Robots via Deep Reinforcement Learning

by Aguila-Rigordi, O., Torres-Torriti, M., Toro, R., Cheein, F.A. and Araya, M.

Copyright, publisher and additional Information: This is the author accepted manuscript. The final published version (version of record) is available online via IEEE Xplore.

DOI link to the version of record on the publisher's website

© 2026 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

Please refer to any applicable terms of use of the publisher.

Harper Adams University

Aguila-Rigordi, O., Torres-Torriti, M., Toro, R., Cheein, F.A. and Araya, M., (2026) 'LEAN: Local Energy-Aware Navigation for Skid-Steer Robots Via Deep Reinforcement Learning', *IEEE Robotics and Automation Letters*.

21 October 2026

LEAN: Local Energy-Aware Navigation for Skid-Steer Robots via Deep Reinforcement Learning

Octavio Aguila-Rigordi[1], Miguel Torres–Torriti[1,4], Rodrigo Toro[1], Fernando Auat Cheein[2,4], and Mauricio Araya[3,4]

***Abstract*—Energy-efficient local navigation of skid-steer mobile robots (SSMRs) in dense environments is constrained by a structural trade-off: the low-curvature, low-power maneuvers that minimize locomotion energy conflict with the high-curvature, high-yaw-rate corrections required to weave through dense obstacle fields without colliding. Model-based planners do not resolve this tension, but expose it as a cost knob, since weighting energy more heavily cuts consumption only by degrading performance as the scenes grow harder. We present the LEAN approach, a local planner based on deep reinforcement learning that learns locomotion-efficient navigation among obstacles from a calibrated energy signal while being model-free at inference, mapping observations directly to commands without solving any closed-form dynamic or power model online. The energy signal enters the policy only as raw motor currents, and in place of typical goal-progress dense reward shaping, the policy is guided by hindsight relabeling that keeps the reward path-shape-agnostic. Benchmarked using a 140-kg tracked SSMR against NMPC and EA-DWA variants that share the simulator's identified dynamics and power model, raising the energy weight of either baseline lowers consumption only by sacrificing success, leaving the most aggressive energy-aware baselines up to 14.2% success points below the proposed approach at the hard tier. LEAN instead reduces energy by 33.2±25.4% to 39.1±30.5% across training seeds in simulation and 11.9±4.1% in real-world experiments over the Vanilla-DRL (energy-blind) ablation at unchanged success rate, and selects actions faster than the model-based controllers, with the simulation ranking preserved when transferred to cluttered outdoor environments. Compared to the energy-aware model-based NMPC-0.05, LEAN reduces energy by 6.3±4.9% in easy low-clutter environments up to 22.3±1.1% in harder high-clutter environments across training seeds in simulation and by 21.8±21.6% in real-world experiments.**

***Index Terms*—Energy-aware planning, local navigation, mobile robots, motion planning.**

I. Introduction

THE electrification of high-payload off-road platforms in construction [1], agriculture [2], and mining [3] makes energy efficiency a critical design constraint on autonomy, battery sizing, and operating cost [4]. Skid-steer mobile robots (SSMRs) are valued for their simplicity and traction, but their track slippage and skidding make energy-efficient motion control especially demanding [5].

Classical model-based planners embed an explicit robot model in the planner cost. Effati et al. [6] derive a closed-form energy-optimal SSMR solution, but only for obstacle-free flat ground with no online obstacle reaction. NMPC lifts this by re-solving a finite-horizon problem online: Santos et al. [7] penalize unknown obstacles but include no energy term, and Prado et al. [8] track a pre-planned reference on a skid-steer loader with no online avoidance; energy-aware car-like variants [9], [10] inherit pure-rolling assumptions that do not transfer to SSMRs.

The reactive Dynamic Window Approach (DWA) [11] scores one-step velocities by heading, clearance, and speed but ignores energy; energy-aware DWA (EA-DWA) adds a model-predicted energy term, reporting up to 9.79% savings on a Mecanum platform [12], [13], and learned variants regress energy from payload for a 13% warehouse reduction [14]. Yet all model-based approaches evaluate a predictive model online and drift over the horizon, and SSMRs are especially affected, since skidding violates the pure-rolling and no-side-slip constraints of standard wheeled-vehicle models [15].

Deep reinforcement learning (DRL) differs structurally: identification is amortized offline and the deployed policy is model-free at inference. DRL is now a common mapless local planner, mapping range data and a goal vector to velocity commands [16]–[18], yet its reward pervasively pairs sparse terminal bonuses with dense goal-progress shaping [16], [18] and hand-tuned behavior terms [19], [20]; as shown in [21], this greedy goal-pursuit induces steering oscillations and torque chatter that raise energy on heavy SSMRs. Energy-aware DRL for SSMRs stays rare: Salvi and Krovi [22] penalize action magnitude as a smoothing surrogate rather than true power, and [23] targets control effort, not motor energy. A gap thus remains for energy-aware DRL local planners that learn efficient behavior in clutter without an online power model, fixed minimum-turn-radius, or dense goal-progress shaping.

We propose LEAN, an energy-aware, model-free-at-inference local planner for SSMRs, compared against an "energy-blind" ablation (Vanilla-DRL), three NMPC variants, and the EA-DWA family, in simulation and on a real SSMR. The main contributions of this work are summarized in: (i) *energy reduction without sacrificing success rate or mission time compared to NMPC or EA-DWA*, (ii) *an energy-aware*

Manuscript received: June, 23, 2026; Revised September, 15, 2026; Accepted September, 29, 2026.

This paper was recommended for publication by Editor Aniket Bera upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by the National Agency of Research and Development (ANID) of Chile under grants Fondecyt 1220140, ANID CIA250006.

[1]School of Engineering, Pontificia Universidad Católica de Chile, Chile oeaguila@uc.cl, mtorreto@uc.cl, rntoro@uc.cl

[2]Dept. of Engineering, Harper Adams University, United Kingdom fauat@harper-adams.ac.uk

[3]Dept. of Electronics, Universidad Tecnica Federico Santa Maria, Chile mauricio.araya@usm.c

[4]Advanced Center for Electronics and Electrical Engineering, Chile

Digital Object Identifier (DOI): see top of this page.

0000–0000 © 2026 IEEE

local planner that requires no dynamic or power model at inference, (iii) *the use of an unbiased, path-shape-free reward for energy-aware learning*. Hindsight relabeling [24] densifies a sparse goal signal in place of the dense goal-progress shaping pervasive in DRL navigation approaches, freeing the policy to discover energy-efficient trajectories that deviate from the straight-line goal direction instead of penalizing them. This sparse-reward/HER backbone is adopted from our prior work [21]; the new element is showing it is precisely what lets a calibrated energy signal reshape trajectories, which dense goal-progress shaping would forbid. Our preliminary work [21] studied reward sparsity and HER as regularizers for control smoothness, in simulation only and with no energy objective. Everything concerning energy is new: the calibrated electrical-power reward term and two raw motor-current observation channels (Section III); calibrated NMPC and EA-DWA baselines that optimize an identical energy criterion, each swept over the energy weight; a hierarchical benchmark on a shared RRT* global path; and sim-to-real validation of every controller on a 140 kg tracked platform (Sections V–VI).

II. Problem Formulation

Consider a mobile robot operating in a bounded planar workspace $\mathcal{W} \subset \mathbb{R}^2$ populated by static obstacles $\mathcal{O} \subset \mathcal{W}$, with free space $\mathcal{F} = \mathcal{W} \backslash \mathcal{O}$. Control is exerted through a body-frame velocity command (action) $\mathbf{a} = (v_c, \omega_c) \in \mathcal{A} \subset \mathbb{R}^2$, computed from a partial observation

$$\mathbf{o}_t = (\mathcal{L}_t,\ v_t,\ \omega_t,\ i_{R,t},\ i_{L,t},\ \mathbf{g}_t), \tag{1}$$

where $\mathcal{L}_t \in \mathbb{R}^{N_L \times 3}$ encodes a 2D LiDAR scan of N_L beams as triples $(\sin\alpha_k, \cos\alpha_k, d_k)$, with α_k the azimuth angle and d_k the range of the k-th beam, $i_{R,t}, i_{L,t}$ are the right and left motor currents, and $\mathbf{g}_t = (\|\Delta\mathbf{g}\|, \sin\psi, \cos\psi)$ is the body-frame goal vector encoding the distance $\|\Delta\mathbf{g}\|$ and bearing ψ to the goal.

Local navigation. Given an initial pose $\mathbf{s}_0 \in \mathcal{F}$ and a goal $\mathbf{g} \in \mathcal{F}$, find a policy $\pi : \mathbf{o}_t \mapsto \mathbf{a}_t$ that drives the robot to within $\|\mathbf{p}_{xy} - \mathbf{g}_{xy}\| \le \epsilon_g$ of the goal, without collision, within a horizon of T control steps. Each step consumes energy

$$\Delta E_t = \big(P_{\text{mech}}(\mathbf{s}_t, \boldsymbol{\tau}_t) + P_{\text{ohm}}(\boldsymbol{\tau}_t)\big)\,\Delta t, \tag{2}$$

where $\boldsymbol{\tau}_t = (\tau_R, \tau_L)$ are the right and left motor torques, P_{mech} denotes mechanical output, and P_{ohm} denotes ohmic losses in the motor windings. The objective is to find π minimizing $\mathbb{E}\big[\sum_{t=0}^{T_f} \Delta E_t\big]$ subject to the goal-reaching and collision-avoidance constraints, where $T_f \le T$ is the terminal step.

Hierarchical navigation. Given additionally a map of $\mathcal{W}$, a global planner produces a collision-free reference path $\gamma : [0, L] \to \mathcal{F}$ from $\mathbf{s}_0$. At each control step, an arc-length lookahead operator selects an intermediate subgoal $\mathbf{g}_t^{\text{sub}} = \gamma(s_t + \ell)$, where s_t is the robot's current projection onto γ and $\ell > 0$ is a fixed lookahead distance. The local policy π then acts on $\mathbf{g}_t^{\text{sub}}$ in place of $\mathbf{g}$; the objective and constraints are inherited from the local case.

Evaluation metrics. An episode is a *success* when it ends without collision, without timeout, and satisfies $\|\mathbf{p}_{xy} - \mathbf{g}_{xy}\| \le \epsilon_g$; collisions correspond to non-floor contacts and timeouts to episodes that exhaust the horizon T without meeting either prior condition. Path efficiency follows the success weighted by path length (SPL) metric [25]

$$\text{SPL} = \mathbf{1}_{\text{succ}}\,\frac{d_0}{\max(d_0, L_{\text{ex}})}, \tag{3}$$

where $d_0 = \|\mathbf{g}_{xy} - \mathbf{s}_{0,xy}\|$ and L_{ex} is the executed path length. Energy use is summarized by the mean episode energy (J) and the energy per meter (J/m),

$$E_{\text{avg}} = \mathbb{E}\left[\sum_t \Delta E_t\right], \qquad \text{E/m} = \frac{E_{\text{avg}}}{L_{\text{ex}}}. \tag{4}$$

Navigation metrics E_{avg}, E/m, and episode duration are computed exclusively over successful episodes, so that efficiency reflects execution quality rather than failure rate.

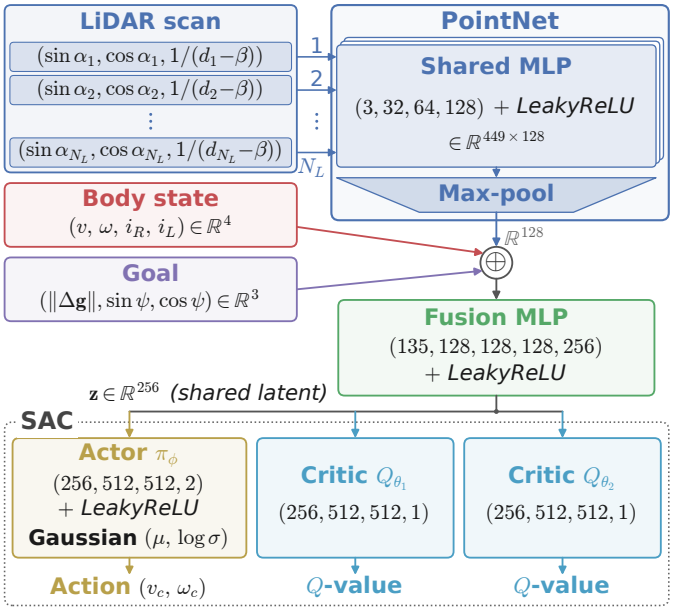
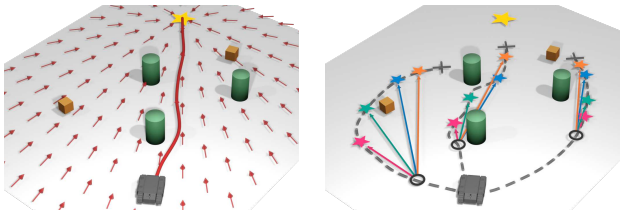


Fig. 1. DRL architecture of the LEAN approach. All inputs are normalized to $[-1, 1]$.

III. LEAN Approach Based on DRL

Architecture. A soft actor-critic (SAC) agent [26] is adopted for its off-policy sample efficiency in continuous control; the full pipeline is illustrated in Fig. 1. The LiDAR set is encoded with a PointNet-style module [27], following the support-point LiDAR encoding in [17]: a shared per-beam MLP and a symmetric max-pool aggregate the N_L beams into a permutation-invariant scene embedding. Following the architecture in [18], each beam is represented as the triple $(\sin\alpha_k, \cos\alpha_k, 1/(d_k - \beta))$, where the inverse-distance reparameterization sharpens sensitivity to nearby obstacles while compressing far-field returns, and the resulting embedding is fused with (v_t, ω_t), the currents $(i_{R,t}, i_{L,t})$, and a body-frame goal vector $(\|\Delta\mathbf{g}\|, \sin\psi, \cos\psi)$ by an MLP into a 256-dimensional latent shared by the actor and twin critics.



(a) Dense shaping (b) HER ($k=4$, future)

Fig. 2. (a) Dense shaping produces a goal-directed gradient (red arrows) that locks trajectories onto a single path. (b) Three failed episodes (dashed); each ring marks a sampled transition t; the four colored arrows show the $k=4$ future states relabeled as virtual goals (stars), supplying dense reward signal without constraining path geometry.

Reward. A sparse terminal signal is combined with a dense per-step penalty:

$$r_t = \begin{cases} r_s, & \text{if success,} \\ r_c, & \text{if collision,} \\ -w_s - w_e\,\Delta E_t, & \text{otherwise.} \end{cases} \tag{5}$$

with $r_s = 50$, $r_c = -200$, and $w_s = 1$. The proposed LEAN approach sets $w_e = 0.004$. This value was selected on the validation split as the largest energy weight that lowered energy per meter while keeping the success rate within the 95% confidence interval of the energy-blind ablation ($w_e = 0$); its sensitivity, and the frontier obtained by sweeping w_e as done for the baselines, are analyzed in Section V.

Hindsight experience replay. Sparse terminal rewards suffer from severe exploration gaps, and the classical remedy of a dense goal-progress reward introduces a per-state gradient pointing toward the goal that biases training trajectories toward a preferred shape (Fig. 2a). This path-shaping bias is particularly detrimental to energy-aware learning, as energy-efficient maneuvers frequently deviate from the straight-line goal direction and would therefore be penalized by dense shaping. We instead densify the sparse signal through HER [24] with the *future* strategy and $k = 4$ relabelled goals per transition, treating each visited state as a virtual success (Fig. 2, right); HER supplies learning signal without prescribing a path, leaving multiple feasible trajectories admissible. This choice follows our prior study [21], which did not include an energy objective in the reward nor observation space. The results in [21] (Table 2) show that densifying the sparse goal reward through hindsight relabeling matches or exceeds dense distance-progress shaping in success rate and SPL across all difficulties while cutting the RMS angular acceleration (α_{rms}) by an order of magnitude, i.e. sparse training learns smoother and better policies, with the advantage becoming more evident as the environment becomes harder.

Training. The actor and twin critics are implemented as $[512, 512]$ MLPs with LeakyReLU activations operating on the shared 256-d embedding of Fig. 1, with the entropy temperature tuned automatically against a target entropy of $-\dim(\mathcal{A})$. Remaining hyperparameters are listed in Table I. Training fits comfortably on commodity hardware: each policy completes in 19.5 hours on a single NVIDIA RTX 3050 (6 GB), with system RAM as the dominant resource cost at approximately 20 GB.

TABLE I
DRL TRAINING HYPERPARAMETERS.

Parameter	**Value**
Optimizer / learning rate	Adam, 3×10^{-4}
Batch size	512
Discount γ	0.99
Polyak coefficient τ	0.005
Entropy coefficient	auto, target $-\dim(\mathcal{A})$
Replay buffer size	10^6 transitions
HER (k, strategy)	(4, future)
Update ratio	1 gradient step per vec. env. step
Target update interval	every 2 gradient steps
Total environment steps	5×10^6



Fig. 3. Real-world experiment with the 140 kg tracked skid-steer robot in the open grass circuit used for the hierarchical navigation benchmark.

IV. Experimental Protocol

Because long-range goals in cluttered environments exceed the horizon of any purely reactive local controller, navigation systems conventionally pair one with a global planner. We therefore evaluate every controller within this hierarchical framework as formalized in Section II, in both simulation and the real world. In each benchmark a global planner produces a single path per start-goal pair, and all controllers execute on that exact frozen path from the same initial pose, so the only variable across controllers is how each one tracks the shared reference.

The platform is a tracked AgileX Bunker SSMR (Fig. 3), with dimensions of $1.023 \times 0.778 \times 0.400$ m, a mass of approximately 140 kg, and powered by a 48 V, 30 Ah lithium battery. Perception is provided by a VLP16 LiDAR mounted on the chassis, configured as a 2D 449-beam circular array spanning 360° with a 20 m range cutoff. The onboard controller reports the two motor currents and the battery voltage, from which we obtain the instantaneous electrical power $P_t = V_t\,(|i_{R,t}| + |i_{L,t}|)$ used as the per-step hardware energy signal $\Delta E_t = P_t\,\Delta t$ in Eq. 2. This power is the energy metric for every controller and the reference calibrating the baseline's power model, whereas the LEAN approach observes the motor currents $i_{R,t}, i_{L,t}$ directly.

Simulator. Training and evaluation run in MuJoCo [28], whose rigid-body contact solver accurately captures the slip and resistance forces that govern tracked non-holonomic platforms. The SSMR model is pre-calibrated from real grass-terrain trials (steady linear and in-place spin runs) by joint constrained least squares over the motor torque constants, viscous damping, and Coulomb friction and scrubbing torque; it reproduces held-out navigation trajectories with RMSE 0.09 m/s in v and 0.35 rad/s in ω. The power model is calibrated against the real robot's battery-power measurements.

Training and evaluation environments. Worlds are procedurally generated within a $14 \times 14\,\mathrm{m}^2$ arena populated with randomly placed cylinders and oriented boxes at three difficulty levels: easy, medium, and hard (Fig. 4). Each level is characterized by five BARN-inspired geometric metrics [29]: distance to closest obstacle, average visibility, dispersion, characteristic dimension, and path tortuosity. Three worlds per difficulty are generated for each of the train, validation, and test splits, yielding 27 worlds in total. Generation is steered by density profiles targeting each class, and every candidate world is accepted only if its metrics fall within pre-defined thresholds. All 9 training worlds (3 of each difficulty level) run simultaneously as independent parallel environments, so the replay buffer is filled concurrently with observations spanning all difficulty levels and obstacle layouts, discouraging overfitting to any single map. At the start of each episode, both the initial pose $\mathbf{s}_0$ and the goal $\mathbf{g}$ are sampled uniformly at random from $\mathcal{F}$, with $\|\mathbf{s}_{0,xy} - \mathbf{g}_{xy}\| \in [1.0, 4.0]\,\mathrm{m}$.

Global planner and path tracking. For hierarchical experiments the global planner is RRT* [30], whose asymptotic-optimality property drives the sampled tree toward near-shortest collision-free routes as the sample budget grows, with a 0.1 m obstacle inflation radius and an arc-length lookahead distance $\ell = 2\,\mathrm{m}$ for path tracking.

Baselines. We compare against two model-based families, a predictive NMPC and the reactive energy-aware DWA (EA-DWA), both *physics-grounded* in the same motor and terrain calibration as the simulator and both optimizing the identical energy term of (2), so their energy objectives are directly comparable to the RL reward. With state $\mathbf{x} = [x, y, \theta, v, \omega]^\top$ and torque input (τ_R, τ_L), the NMPC continuous-time dynamics are

$$\dot{\mathbf{x}} = \begin{bmatrix} v\cos\theta \\ v\sin\theta \\ \omega \\ \dfrac{\tau_R + \tau_L}{mr} - \dfrac{c}{m}\,v \\ \dfrac{(\tau_R - \tau_L)\,W}{2Jr} - \dfrac{b}{J}\,\omega - \dfrac{\tau_c}{J}\tanh(\omega/\epsilon) \end{bmatrix}, \tag{6}$$

where (m, J, r, W) are the mass, rotational inertia, wheel radius, and track separation width, and (c, b, τ_c) are viscous translational, viscous rotational, and Coulomb friction/scrubbing torque coefficients. Discretizing the dynamics with forward Euler yields $\mathbf{x}_{k+1} = \mathbf{F}(\mathbf{x}_k, \mathbf{u}_k)$, which is employed to solve the NMPC

$$\min_{\{\mathbf{x}_k, \mathbf{u}_k, s_k\}} \sum_{k=0}^{N-1} \big[q\|\mathbf{p}_k - \mathbf{g}\|^2 + w_e\,\Delta E(\mathbf{x}_k, \mathbf{u}_k)\big] + q_N\|\mathbf{p}_N - \mathbf{g}\|^2 + \lambda_1 \mathbf{1}^\top \mathbf{s} + \lambda_2\|\mathbf{s}\|^2, \tag{7}$$

subject to the discretized dynamics $\mathbf{x}_{k+1} = \mathbf{F}(\mathbf{x}_k, \mathbf{u}_k)$, soft collision constraints $\|\mathbf{p}_{k+1} - \mathbf{o}_j\|^2 + s_k \geq d_{\text{safe}}^2$ against the M nearest LiDAR returns $\{\mathbf{o}_j\}_{j=1}^{M}$ with slack $s_k \geq 0$, and box limits $|v_k| \leq v_{\max}$, $|\omega_k| \leq \omega_{\max}$, and $|\tau_{R,k}|, |\tau_{L,k}| \leq \tau_{\max}$. Here $\mathbf{p}_k$ is the planar position extracted from $\mathbf{x}_k$ and $\Delta E(\mathbf{x}_k, \mathbf{u}_k) = (P_{\text{mech}} + P_{\text{ohm}})\,\Delta t$ reuses the identified power model of (2), so DRL and NMPC optimize an identical physical energy criterion. The three variants NMPC-w_e for $w_e \in \{0.0,\, 0.05,\, 0.1\}$ span the robustness-energy frontier. The problem is solved in receding horizon at each control step, warm-started by shifting the previous solution forward. Although the decision variables are torques (τ_R, τ_L), the robot's low-level interface accepts body-velocity references, so we command the one-step-ahead predicted $(v_1^\star, \omega_1^\star)$ extracted from the optimal state $\mathbf{x}_1^\star$. This substitution neither advantages nor disadvantages the NMPC: LEAN and every baseline emit the same body-velocity command (v_c, ω_c) to the identical low-level tracker, so the reported energy gap in the field experiments section reflects the commanded trajectory, not the actuation interface, and over a single $\Delta t = 0.1\,\mathrm{s}$ step the tracked velocity closely follows the torque-optimal prediction. The NMPC is solved with IPOPT [31] in CasADi [32] over a horizon of $N = 20$ steps at $\Delta t = 0.1\,\mathrm{s}$, with $M = 20$ obstacle points and $d_{\text{safe}} = 0.75\,\mathrm{m}$. Its per-step solve time is reported with the other controllers in the simulation results (Sec. V).

The reactive EA-DWA baseline scores admissible one-step velocities with the standard DWA objective augmented by an energy penalty

$$\max_{(v,\omega)\,\in\,V_r} \sigma\big(\alpha\,\mathrm{head} + \beta\,\mathrm{clear} + \gamma\,\mathrm{vel} - w_e\,\Delta E(v,\omega)\big), \tag{8}$$

where head, clear, and vel are the heading, clearance, and forward-velocity terms, σ normalizes their weighted sum, and $\Delta E(v, \omega)$ again reuses the power model of (2). The search set $V_r = V_s \cap V_d \cap V_a$ intersects the velocity limits V_s, the dynamic window V_d of velocities reachable from the current state under the acceleration bounds, and the admissible set V_a that permits a safe stop before the nearest obstacle, and is evaluated one step ahead by discretizing the velocity space. Its variants EA-DWA-w_e for $w_e \in \{0.0,\, 0.0003,\, 0.0005\}$ span the same robustness-energy frontier. The two ranges differ by two orders of magnitude because the energy term is scaled differently in each objective: in the NMPC cost ΔE (Joules) is summed over the N-step horizon against a quadratic position error, whereas in the EA-DWA score a single one-step ΔE competes against normalized, order-unity heading, clearance, and velocity terms. The weights are therefore not directly comparable; each family was swept from energy-blind ($w_e = 0$) to the largest weight before success degraded sharply on the validation split, so both frontiers cover the same energy-blind-to-energy-biased regime.

In simulation, both start and goal poses were randomly sampled from collision-free configurations and accepted only when their straight-line distance fell between 8.0 and 20.0 m. All these pairs were fixed in advance, with RRT* run once per pair to plan the global path. Each of the three difficulty levels comprises 3 worlds of 100 episodes each, yielding 300 episodes per difficulty that are shared across all controllers. Each world is split into 5 batches, producing 15 batch means. Confidence intervals are reported at the 95% level. Unless stated otherwise, these intervals are *within-policy* and across three different SAC training seeds: they capture episode-level variability across the 15 batch means of 3 trained policies per configuration, and including training-seed variability, which is usually the dominant source of spread in SAC.

In the real world, we evaluate the four highest-success controllers from simulation on a rectangular outdoor grass

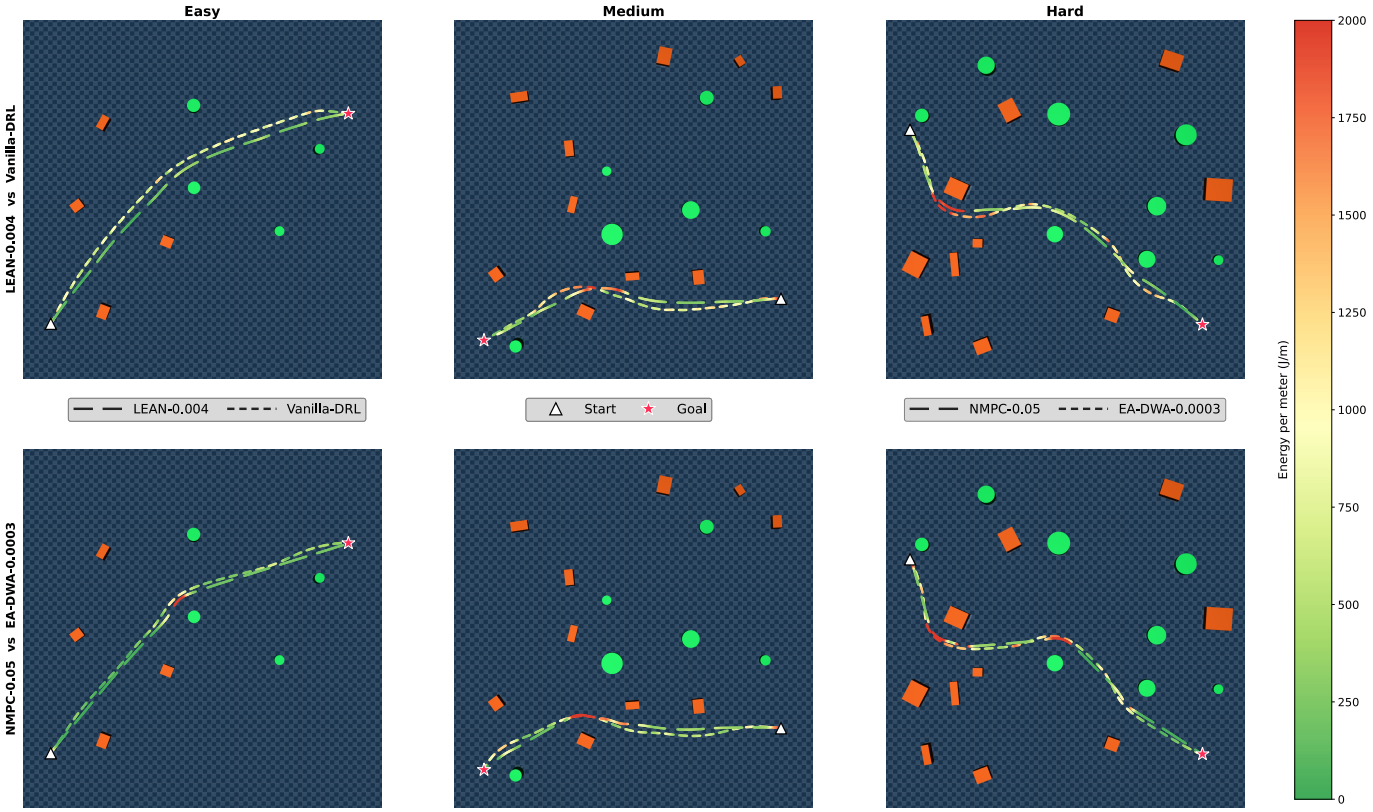


Fig. 4. Executed trajectories on the same global plans at easy (left), medium (center), and hard (right) difficulty, colored by energy per meter (J/m). The color scale is clipped at 2000 J/m to preserve contrast between controllers; segments exceeding this value, which occur mainly on the hardest scenes, are drawn saturated.

circuit of approximately 7×16 m, containing 17 obstacles whose density places the field in the hard difficulty class of the simulation benchmark. Three fixed poses are placed at each end of the circuit, and taking every crossing combination between the two ends yields $3\times3+3\times3 = 18$ routes spanning the roughly 15 m gap. As in simulation, each route's RRT* path is planned offline once and replayed identically for every controller, tracked with the same 2.0 m look-ahead, and each controller is run over all 18 routes. As in simulation, 95% confidence intervals are computed from a t-distribution, here directly over the 18 individual routes.

V. Simulation Results and Analysis

Table II reports success rate, energy per meter, and episode time for all eight controllers across the three difficulty levels, and Fig. 5 plots success against energy. The proposed LEAN approach reaches the same top-tier success rate as the strongest baselines: its success rate is not significantly different from the energy-blind NMPC-0.0 at any difficulty (two-sample Welch t-test on the 15 per-configuration batch means, $p > 0.05$), while consuming less than a third of its energy. NMPC-0.0 sustains that success rate only by spending above 3150 J/m at every difficulty, so ignoring energy in the objective forfeits large, achievable efficiency gains at no benefit to success. The energy-aware classical planners, NMPC-0.05, NMPC-0.1, and the EA-DWA family, cut energy sharply, but only by trading away success as their energy weight grows. At the hard level EA-DWA-0.0005 reaches 850±44 J/m while its success rate falls to 69.8±5.6%, against 82.4±2.6% for the LEAN-0.004 approach. The most aggressive variants, NMPC-0.1 and EA-DWA-0.0005, pay the steepest success penalty and are dominated on this trade-off, so the paired comparison and field trials below focus on the controllers that stay competitive on success, with every operating point retained here and in Fig. 5. The success rates of NMPC-0.0 and the LEAN approach with different energy weights coincide at all three difficulties because every controller tracks the same frozen RRT* plans from the same start poses over the same shared episodes, so that success is governed chiefly by plan feasibility; the two highest-success controllers consequently succeed on the same episode set.

The LEAN approach alone occupies the favorable upper-left corner of the trade-off Fig. 5 between energy per meter and success rate, holding the lowest energy among the high-success controllers while achieving the best success rates and improving on the "energy-blind" ablation Vanilla-DRL with no loss of success. Episode time is comparable across all controllers, so its energy savings carry no meaningful time penalty. The proposed LEAN approach also attains the highest SPL at every difficulty, with its path-efficiency lead over the model-based controllers widening on the hardest scenes, so the savings come from cleaner tracking of the shared plan rather than from cutting corners.

To test whether the learned policy traces its own energy-success frontier rather than occupying a single favorable point, w_e was swept over $\{0.001, 0.002, 0.004\}$ for LEAN,

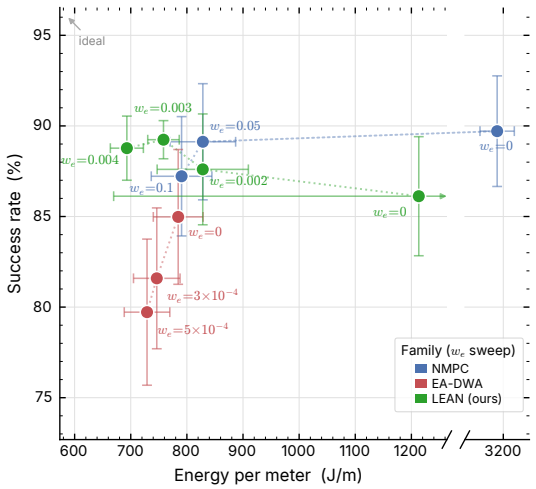


Fig. 5. Success rate versus energy per meter, averaged over all episodes across the three difficulty levels, with each controller family swept over w_e, including the LEAN frontier; the off-scale energy-blind NMPC-0.0 operating point is shown with a broken energy axis. LEAN confidence intervals is computed across episodes and three SAC training seeds.

mirroring the sweeps applied to the two baseline families, and the resulting operating points are added to Fig. 5 as the LEAN frontier. Across the sweep, raising w_e lowers energy per meter monotonically while success stays within its confidence interval up to $w_e = 0.004$ and drops only beyond it, placing 0.004 at the knee of the curve. Unlike the model-based families, whose success falls steeply as w_e grows, the learned frontier stays in the high-success region as energy decreases, indicating that LEAN shifts the attainable frontier rather than sliding along the baselines' one.

Fig. 4 colors each executed trajectory considering the instantaneous energy intensity and reveals the mechanism behind these numbers. Intensity peaks at the high-curvature corrections forced by obstacles, where the elevated yaw rate drives track slip and large motor currents, confirming that obstacle-induced turning, rather than forward translation, is the dominant cost. On the turns that cannot be avoided, the LEAN approach spreads the curvature over a smoother, wider arc and holds the coolest colors through the correction, committing to a sharp maneuver only when the geometry forces it, which is the trajectory-level origin of its energy advantage.

Comparing the energy column of Table II across controllers requires care, because they do not all solve the same episodes: each per-controller energy is averaged over a different subset, so a raw difference would conflate efficiency with which episodes each one solved. Table III resolves this by comparing the proposed LEAN-0.004 approach against each baseline only on the episodes both solve under the identical global plan. Aggregating the 205 to 283 shared episodes per row, it reports ΔE, the mean within-episode energy difference relative to the baseline where negative favors the LEAN method, "Win percentage", i.e. the share of those episodes on which the proposed LEAN approach consumes less energy, with a paired bootstrap 95% confidence interval on ΔE over the shared episodes and considering three-seed SAC training variability. Column "Sig." of Table III counts the seeds with a significant energy reduction. Within a seed, the paired bootstrap 95% interval of ΔE is at most ± 2.8 pp. Energy is tested per seed with a Wilcoxon signed-rank test on the paired episode energies, and success with an exact McNemar test on the paired outcomes. LEAN-0.004:

- uses less energy than Vanilla-DRL at every difficulty in all seeds ($p \leq 3.5 \times 10^{-20}$);
- uses less energy than NMPC-0.05 at medium and hard in all seeds ($p \leq 1.6 \times 10^{-19}$) and at easy in two seeds ($p \leq 7.8 \times 10^{-7}$);
- uses less energy than EA-DWA-0.0003 at medium and hard in all seeds ($p \leq 2.4 \times 10^{-12}$) and at easy in two seeds ($p \leq 9.5 \times 10^{-10}$);
- succeeds more often than EA-DWA-0.0003 at every difficulty in all seeds ($p \leq 0.014$);
- succeeds more often than Vanilla-DRL in one seed at medium ($p = 4.9 \times 10^{-4}$) and two at hard ($p \leq 1.1 \times 10^{-3}$);
- succeeds less often than NMPC-0.05 in one seed at medium ($p = 0.013$) and hard ($p = 0.041$).

The two comparisons that isolate the energy term are the most informative. Against its learned energy-blind ablation Vanilla-DRL, which differs from LEAN only in the energy reward, the LEAN-0.004 approach saves between $33.2 \pm 25.4\%$ and $39.1 \pm 30.5\%$ across all difficulties and across the three different training seeds; against NMPC-0.05, which isolates a learned policy against a model-based one at a matched energy objective, it saves between $6.3 \pm 4.9\%$ in easier difficulty environments and $22.3 \pm 1.1\%$ in harder difficulty cluttered environments. In both, the margin grows monotonically with difficulty, so the LEAN approach is strongest exactly where the energy-success tension is sharpest. Against the remaining energy-aware baselines the margin likewise widens from easy to hard. The comparison against the energy-blind NMPC-0.0, a controller no one would field in an energy study, is reported only as a sanity check: there LEAN uses 74.3 to 81.8% less energy and wins almost every episode. The LEAN approach does not dominate everywhere on easy: as Table II and Fig. 5 show, the most aggressive energy variants set aside above reach marginally lower energy there, but only at a lower success rate.

This comparison isolates the central result of the paper. The NMPC and EA-DWA baselines optimize the same calibrated power model online at every step, yet the proposed LEAN approach matches them on easy scenes and surpasses them on harder ones while never evaluating that model at inference, sensing energy only through the raw motor currents. A learned policy can thus recover, and on harder scenes exceed, the locomotion efficiency of planners built around an explicit power model, without paying the per-step cost of solving one. On the same hardware, LEAN selects an action in about 1 ms per step (1.08 ms, against 1.12 ms for Vanilla-DRL), more than an order of magnitude below the NMPC variants, whose solve time grows with the energy weight from 24 to 48 ms, while EA-DWA sits near 6 ms. All fit the 100 ms budget of the 10 Hz control loop. Because all four controllers already meet this budget, the order-of-magnitude timing advantage is not what enables real-time control at 10 Hz here; it is headroom that would let the learned policy run at substantially higher control

TABLE II
HIERARCHICAL NAVIGATION IN SIMULATION. RESULTS REPORTED FOR THE BEST AGENT OBTAINED FROM THREE DIFFERENT SAC TRAINING SEEDS SHOWING THE WITHIN-POLICY VARIABILITY. BEST VALUE PER COLUMN WITHIN EACH DIFFICULTY IN BOLD.

Method	Diff.	SR (%) ↑	E/m (J/m) ↓	SPL ↑	Time (s) ↓
NMPC-0.0	Easy	$\mathbf{97.6} \pm 1.8$	3250 ± 52	0.984 ± 0.006	$\mathbf{22.2} \pm 0.5$
	Medium	$\mathbf{87.6} \pm 5.7$	3154 ± 75	0.949 ± 0.010	$\mathbf{23.4} \pm 0.7$
	Hard	$\mathbf{84.0} \pm 5.6$	3157 ± 67	0.895 ± 0.021	$\mathbf{24.7} \pm 1.1$
NMPC-0.05	Easy	96.9 ± 2.4	618 ± 45	0.985 ± 0.005	22.5 ± 0.6
	Medium	87.2 ± 6.1	855 ± 74	0.952 ± 0.010	23.6 ± 0.7
	Hard	83.3 ± 5.9	1013 ± 55	0.897 ± 0.021	25.0 ± 1.1
NMPC-0.1	Easy	95.8 ± 3.5	600 ± 41	0.985 ± 0.005	22.5 ± 0.6
	Medium	84.7 ± 5.9	807 ± 68	0.953 ± 0.010	23.9 ± 0.9
	Hard	81.1 ± 5.4	965 ± 53	0.898 ± 0.021	25.0 ± 1.1
EA-DWA-0.0	Easy	95.5 ± 2.4	635 ± 48	0.980 ± 0.007	24.1 ± 0.5
	Medium	83.6 ± 6.4	806 ± 65	0.946 ± 0.010	25.4 ± 0.7
	Hard	75.8 ± 6.0	913 ± 39	0.894 ± 0.019	26.7 ± 1.1
EA-DWA-0.0003	Easy	93.8 ± 2.7	613 ± 44	0.980 ± 0.007	24.1 ± 0.5
	Medium	78.3 ± 6.7	754 ± 60	0.946 ± 0.011	25.5 ± 0.8
	Hard	72.6 ± 5.5	872 ± 41	0.893 ± 0.021	26.8 ± 1.2
EA-DWA-0.0005	Easy	92.8 ± 2.6	599 ± 42	0.979 ± 0.007	24.2 ± 0.5
	Medium	76.5 ± 6.6	738 ± 60	0.944 ± 0.012	25.4 ± 0.8
	Hard	69.8 ± 5.6	850 ± 44	0.891 ± 0.021	27.0 ± 1.2
Vanilla-DRL	Easy	96.6 ± 0.3	1140 ± 601	0.987 ± 0.002	23.7 ± 0.1
	Medium	83.8 ± 3.7	1217 ± 526	0.959 ± 0.005	24.9 ± 0.2
	Hard	77.7 ± 6.1	1300 ± 491	$\mathbf{0.921} \pm 0.012$	26.1 ± 0.3
LEAN-0.002	Easy	96.9 ± 0.6	717 ± 78	0.981 ± 0.008	24.0 ± 0.5
	Medium	85.1 ± 3.4	843 ± 87	0.954 ± 0.006	25.2 ± 0.5
	Hard	80.5 ± 5.2	952 ± 86	0.908 ± 0.007	26.3 ± 0.3
LEAN-0.003	Easy	97.4 ± 0.7	649 ± 41	0.988 ± 0.002	24.4 ± 1.7
	Medium	86.6 ± 1.1	769 ± 29	0.961 ± 0.002	25.4 ± 1.5
	Hard	83.5 ± 1.5	879 ± 25	0.916 ± 0.002	26.7 ± 1.7
LEAN-0.004	Easy	97.4 ± 0.5	$\mathbf{591} \pm 31$	$\mathbf{0.989} \pm 0.001$	23.9 ± 0.2
	Medium	86.3 ± 2.3	$\mathbf{700} \pm 36$	$\mathbf{0.963} \pm 0.003$	25.0 ± 0.3
	Hard	82.4 ± 2.6	$\mathbf{810} \pm 18$	0.920 ± 0.007	26.1 ± 0.4

TABLE III
PAIRED ENERGY COMPARISON IN SIMULATION OF LEAN-0.004 AGAINST EACH BASELINE. RESULTS CONSIDER JOINTLY THE WITHIN-POLICY AND THREE-SEED SAC TRAINING VARIABILITY.

Baseline	Diff.	ΔE (%) ↓	Win (%) ↑	Sig.
NMPC-0.05	Easy	-6.3 ± 4.9	57 ± 13	2/3
	Medium	-20.4 ± 2.9	76 ± 8	3/3
	Hard	-22.3 ± 1.1	85 ± 3	3/3
EA-DWA-0.0003	Easy	-6.3 ± 4.8	62 ± 14	2/3
	Medium	-11.9 ± 3.3	75 ± 7	3/3
	Hard	-12.5 ± 2.1	81 ± 4	3/3
Vanilla-DRL	Easy	-39.1 ± 30.5	89 ± 13	3/3
	Medium	-35.9 ± 27.4	89 ± 11	3/3
	Hard	-33.2 ± 25.4	90 ± 10	3/3

rates or on cheaper embedded compute, regimes in which the model-based solve times would exceed the available budget.

VI. REAL-WORLD RESULTS

We carry forward only the four highest-success controllers from the hard simulation tier: NMPC-0.0, NMPC-0.05, Vanilla-DRL, and LEAN-0.004. This set spans the two axes under test: model-based against learned control and energy-blind against energy-aware. We leave the EA-DWA family out because it recorded the lowest success rates of all controllers in simulation, the steepest price paid for its energy savings (Table II).

Table IV reports the outcome. The simulation ranking transfers intact on energy per meter, SPL, and time, where the margins are large: LEAN-0.004 reaches the lowest energy per meter and the best path efficiency, cutting energy per meter against its energy-blind ablation from 2586±86 to 2355±89 J/m, while lifting SPL from 0.935 to 0.962, and saving between 21.8±21.6% and 28.8±%5.9 per meter against the model-based NMPC controllers at higher path efficiency. The success rates, by contrast, rest on only 18 routes, so the 88.9% of LEAN and Vanilla-DRL $(16/18)$ against the 72.2% of NMPC-0.05 $(13/18)$ differs by three routes and are best read as comparable rather than as a ranking. Moreover, because energy is averaged only over successful runs, equal aggregate success does not guarantee that LEAN and Vanilla-DRL solved the same routes; on the subset of routes solved by both, LEAN uses 11.9±4.1% less energy per meter on average. It important to stress that in 3 out of 13 successful routes, NMPC-0.05 reached points in which the robot significantly slowed its speed trying to evade very close obstacles. This few cases that are less than 25% of the successful routes is one of the contributing factors that increased the energy consumption per meter compared to the energy-blind NMPC-0.0. If these cases were to be removed and considered unsuccessful with more strict time-out and minimum velocity constraints, NMPC-0.05 would have appeared as more energy efficient than NMPC-0.0. This observation reinforces the tension in NMPC-0.05 between selecting a traversable path and minimizing energy consumption, which is not faced by the LEAN approach. The absolute energy per meter is about three times larger on hardware than in simulation (2355 J/m against 591–810 J/m). This is expected rather than a modeling failure: the hardware signal $P_t = V_t\,(|i_{R,t}| + |i_{L,t}|)$ is the total battery draw, including fixed onboard-electronics and drivetrain losses, whereas the simulator energy of (2) accounts only for the motors' mechanical output and ohmic losses. What transfers is the relative ranking of controllers, not the absolute Joules-per-meter scale. The supplementary video shows the real-world trajectories of Vanilla-DRL, NMPC-0.05, and LEAN alongside their instantaneous energy usage along the route.

TABLE IV
REAL-WORLD RESULTS: HIERARCHICAL NAVIGATION. BEST VALUE PER COLUMN IN BOLD; SR IS A PROPORTION OVER 18 ROUTES AND CARRIES NO CONFIDENCE INTERVAL.

Method	SR (%) ↑	E/m (J/m) ↓	ΔE (%) ↓	SPL ↑	Time (s) ↓
NMPC-0.05	72.2	3610 ± 1112	-21.8 ± 21.6	0.908 ± 0.029	43.2 ± 11.9
NMPC-0.0	**88.9**	2917 ± 117	-28.8 ± 5.9	0.844 ± 0.021	36.4 ± 2.4
Vanilla-DRL	**88.9**	2586 ± 86	-11.9 ± 4.1	0.935 ± 0.008	$\mathbf{30.9} \pm 0.5$
LEAN-0.004	**88.9**	$\mathbf{2355} \pm 89$	–	$\mathbf{0.962} \pm 0.007$	31.5 ± 0.7

VII. CONCLUSION

We presented LEAN, a skid-steer local planner that learns energy-efficient navigation from a calibrated electrical-energy signal yet is model-free at inference, replacing dense goal-progress shaping by hindsight relabeling so the policy is free to discover low-energy trajectories. Against NMPC and EA-DWA baselines that share the same identified dynamics and power model, it reaches more goals and uses less energy than the energy-aware benchmarks, more so on harder scenes, with smoother, more path-efficient trajectories, in simulation and on a 140-kg robot on real terrain at no cost in mission time. It also selects actions more than an order of magnitude faster, because the modeling cost is amortized offline, which at the

present 10 Hz loop is headroom for higher control rates or cheaper compute rather than an enabling factor.

The approach requires a calibrated simulator and energy model at training time, and the evaluation covers a single platform on flat hard ground with fixed payload and static obstacles, without the formal safety guarantees the model-based benchmarks offer. Within that scope, a path-shape-free reward paired with a calibrated energy signal suffices for locomotion efficiency that transfers to the SSMR. Natural next steps are multi-platform transferability across robot dimensions, mass, and dynamics, alongside varied terrain, dynamic obstacles, and safety-constrained policies that pair the learned efficiency with formal guarantees.

ACKNOWLEDGMENTS

This project has been supported by the National Agency of Research and Development (ANID) of Chile under grants Fondecyt 1220140, ANID CIA250006.

REFERENCES

[1] X. Huang, W. Yan, H. Cao, S. Chen, G. Tao, and J. Zhang, "Prospects for purely electric construction machinery: Mechanical components, control strategies and typical machines," *Automation in Construction*, vol. 164, p. 105477, 2024.

[2] E. Scolaro, M. Beligoj, M. P. Estevez, L. Alberti, M. Renzi, and M. Mattetti, "Electrification of agricultural machinery: A review," *IEEE Access*, vol. 9, pp. 164 520–164 541, 2021.

[3] H. Du, L. Chan, J. Tong, R. Raad, F. Naghdy, Q. Guo, Y. Yu, M. R. Islam, F. Tubbal, M. Ros, Z. Li, and C. Ritz, "Industrial progress of robotic automation in mining applications: A survey," *Mining, Metallurgy & Exploration*, vol. 42, no. 2, pp. 537–556, 2025.

[4] H. Zhang, Y. Zhang, and T. Yang, "A survey of energy-efficient motion planning for wheeled mobile robots," *Industrial Robot*, vol. 47, no. 4, pp. 607–621, 2020.

[5] S. Dogru and L. Marques, "A physics-based power model for skid-steered wheeled mobile robots," *IEEE Transactions on Robotics*, vol. 34, no. 2, pp. 421–433, 2018.

[6] M. Effati and K. Skonieczny, "Energy-optimal trajectories for skid-steer rovers," *The International Journal of Robotics Research*, vol. 43, 12 2023.

[7] M. A. Santos, A. Ferramosca, and G. V. Raffo, "Nonlinear model predictive control schemes for obstacle avoidance," *Journal of Control, Automation and Electrical Systems*, vol. 34, no. 5, pp. 891–906, 2023.

[8] Á. J. Prado, M. Torres-Torriti, J. Yuz, and F. Auat Cheein, "Tube-based nonlinear model predictive control for autonomous skid-steer mobile robots with tire-terrain interactions," *Control Engineering Practice*, vol. 101, p. 104451, 2020.

[9] M. Salazar, A. Alessandretti, A. P. Aguiar, and C. N. Jones, "An energy efficient trajectory tracking controller for car-like vehicles using model predictive control," in *2015 54th IEEE Conference on Decision and Control (CDC)*, 2015, pp. 3675–3680.

[10] A. Parra, D. Tavernini, P. Gruber, A. Sorniotti, A. Zubizarreta, and J. Pérez, "On nonlinear model predictive control for energy-efficient torque-vectoring," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 1, pp. 173–188, 2021.

[11] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robotics & Automation Magazine*, vol. 4, no. 1, pp. 23–33, 1997.

[12] C. Henkel, A. Bubeck, and W. Xu, "Energy efficient dynamic window approach for local path planning in mobile service robotics," *IFAC-PapersOnLine*, vol. 49, no. 15, pp. 32–37, 2016, 9th IFAC Symposium on Intelligent Autonomous Vehicles IAV 2016.

[13] L. Xie, C. Henkel, K. Stol, and W. Xu, "Power-minimization and energy-reduction autonomous navigation of an omnidirectional mecanum robot via the dynamic window approach local trajectory planning," *International Journal of Advanced Robotic Systems*, vol. 15, no. 1, pp. 1–12, 2018.

[14] M. Mohammadpour, S. Kelouwani, M.-A. Gaudreau, B. Allani, L. Zeghmi, A. Amamou, and M. Graba, "Energy-efficient local path planning of a self-guided vehicle by considering the load position," *IEEE Access*, vol. 10, pp. 112 669–112 685, 2022.

[15] G. Campion, G. Bastin, and B. D'Andréa-Novel, "Structural properties and classification of kinematic and dynamic models of wheeled mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 1, pp. 47–62, 1996.

[16] L. Tai, G. Paolo, and M. Liu, "Virtual-to-real deep reinforcement learning: Continuous control of mobile robots for mapless navigation," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 31–36.

[17] W. Zhang, N. Liu, and Y. Zhang, "Learn to navigate maplessly with varied LiDAR configurations: A support point-based approach," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1918–1925, 2021.

[18] W. Zhang, S. Wang, M. Tan, Z. Yang, X. Wang, and X. Shen, "DRL-DCLP: A deep reinforcement learning-based dimension-configurable local planner for robot navigation," *IEEE Robotics and Automation Letters*, vol. 10, no. 4, pp. 3636–3643, 2025.

[19] J. Choi, K. Park, M. Kim, and S. Seok, "Deep reinforcement learning of navigation in a complex and crowded environment with a limited field of view," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 5993–6000.

[20] H.-C. Wang, S.-C. Huang, P.-J. Huang, K.-L. Wang, Y.-C. Teng, Y.-T. Ko, D. Jeon, and I.-C. Wu, "Curriculum reinforcement learning from avoiding collisions to navigating among movable obstacles in diverse environments," *IEEE Robotics and Automation Letters*, vol. 8, no. 5, pp. 2740–2747, 2023.

[21] Anonymous, "Reward sparsity as an implicit regularizer for effective skid-steer robot navigation in cluttered environments," in *Anonymous Conference*, 2026, reference details withheld for double-anonymous review; in press.

[22] A. Salvi and V. N. Krovi, "Experimental investigation of pose informed reinforcement learning for skid-steered visual navigation," *IEEE Transactions on Field Robotics*, vol. 2, pp. 690–710, 2025.

[23] G. Tang, N. Kumar, and K. P. Michmizos, "Reinforcement co-learning of deep and spiking neural networks for energy-efficient mapless navigation with neuromorphic hardware," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 6090–6097.

[24] M. Andrychowicz, F. Wolski, A. Ray, J. Schneider, R. Fong, P. Welinder, B. McGrew, J. Tobin, P. Abbeel, and W. Zaremba, "Hindsight experience replay," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017, pp. 5048–5058.

[25] P. Anderson, A. Chang, D. S. Chaplot, A. Dosovitskiy, S. Gupta, V. Koltun, J. Kosecka, J. Malik, R. Mottaghi, M. Savva, and A. R. Zamir, "On evaluation of embodied navigation agents," *arXiv preprint arXiv:1807.06757*, 2018.

[26] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *Proceedings of the 35th International Conference on Machine Learning (ICML)*, ser. Proceedings of Machine Learning Research, vol. 80, 2018, pp. 1861–1870.

[27] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 652–660.

[28] E. Todorov, "Convex and analytically-invertible dynamics with contacts and constraints: Theory and implementation in MuJoCo," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 6054–6061.

[29] D. Perille, A. Truong, X. Xiao, and P. Stone, "Benchmarking metric ground navigation," in *2020 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*. IEEE Press, 2020, pp. 116–121.

[30] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *The International Journal of Robotics Research*, vol. 30, no. 7, pp. 846–894, 2011.

[31] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical Programming*, vol. 106, no. 1, pp. 25–57, 2006.

[32] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "CasADi: A software framework for nonlinear optimization and optimal control," *Mathematical Programming Computation*, vol. 11, no. 1, pp. 1–36, 2019.