ECOLATENT: A PHYSICS-GUIDED LATENT INNER WORLD MODEL FOR VEHICLES AND ROBOTS

Anonymous authors
Paper under double-blind review

ABSTRACT

World models provide a basis for embodied decision-making, yet modeling the surrounding environment alone leaves an essential question unanswered: how will the agent's own internal world evolve under its actions? We introduce ECOLATENT, a physics-guided latent inner world model for vehicles and robots, and instantiate it on vehicle longitudinal and powertrain dynamics. The model combines explicit physical state with a recurrent latent operating context inferred from causal sensor history. Physical equations and learned residual dynamics jointly advance the internal state under candidate actions, enabling imagined trajectories to support downstream decisions. We use vehicle energy management as a concrete case study, with energy represented as one component of the internal world. On 842 h of operational vehicle logs, an 85k-parameter model achieves 0.653 m/s speed RMSE and 1.61% cumulative-fuel normalized MAE over 60 s, matching a 272k-parameter GRU on speed while preserving a structural fuel-cut rule. A policy is optimized through the learned inner world; cross-model evaluation reveals that its estimated benefit depends strongly on the evaluation dynamics. These results demonstrate a compact inner-world representation and an imagination-based decision pipeline in vehicles, while motivating extensions to other internal subsystems and robotic embodiments.

1 INTRODUCTION

Vehicles and robots act in two coupled worlds: the external environment around them and the internal world of their own physical operation. World-model approaches have made it possible to learn representations of observed environments and imagine future interactions for control (Ha & Schmidhuber, 2018; Hafner et al., 2019; 2020; 2025). However, an environment-centered description does not by itself make the agent's internal operating state explicit. The same command can have different consequences depending on actuator behavior, mechanical state, and recent operating history. Understanding the world around an agent therefore needs a complementary capability: modeling the world within it.

We call this capability an *inner world model*: an action-conditioned model of the embodied system's own dynamics, operating context, and resource evolution. For vehicles and robots, this perspective connects internal state estimation to decisions about how the system should act. Its central challenge is to combine partially observed, history-dependent behavior with known physical structure. A purely learned representation can capture operating history but need not preserve physical constraints; a fixed physical model makes those constraints explicit but can miss changing context and unmodeled dynamics. An inner world model should retain both interpretable physical state and a latent representation of what recent observations reveal about the system.

We introduce ECOLATENT, a physics-guided latent inner world model that combines these two representations (Fig. 1). A causal history encoder initializes the latent operating context, and a recurrent transition couples physical equations with learned residual dynamics. Candidate actuator actions advance the resulting state, producing imagined trajectories for downstream decision-making. We instantiate the framework on vehicles: the modeled state includes motion, powertrain operation, and accumulated resource use, while a road profile supplies external conditions along the imagined trajectory. Energy is thus one coordinate of the internal world and one possible decision objective, rather than the definition of the modeling task.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

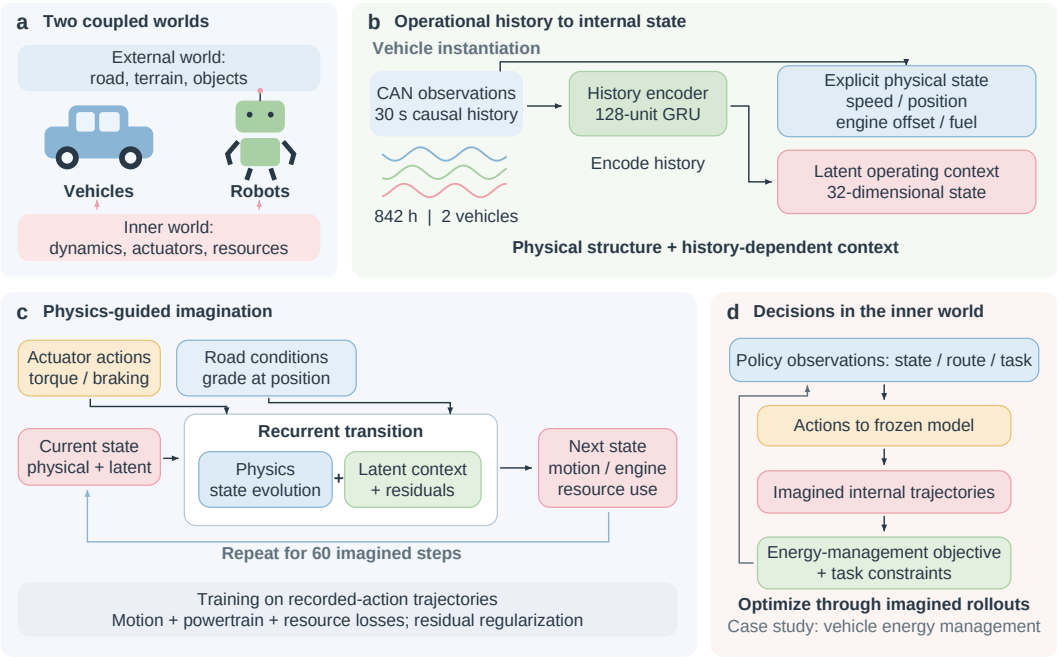

Figure 1: **EcoLatent: from internal observations to imagined decisions.** **(a)** Inner world modeling complements the external-world perspective for vehicles and robots. **(b)** In the vehicle instantiation, causal CAN history initializes a physical state and learned operating context. **(c)** Candidate actions and road conditions drive a physics-guided recurrent transition to generate internal trajectories. **(d)** A downstream policy is optimized through these trajectories, with vehicle energy management as the implemented case study. Vehicles provide the experimental case; robots illustrate the broader scope. The policy does not observe latent context directly.

The decision interface is central to this formulation. A separate policy is optimized through differentiable imagined rollouts, using internal trajectories to evaluate an objective together with progress, terminal-state, drivability, and empirical action-support terms. We study energy management as one concrete application of that interface, building on vehicle look-ahead control (Hellström et al., 2009; Sciarretta et al., 2015). Other internal decisions, including dynamics control and actuator or resource management, motivate the broader framework. Our experiments evaluate the vehicle instantiation on later trips, across vehicles, and under component ablations, then examine how policy estimates change across evaluation worlds. This last step exposes the distinction between fitting recorded trajectories and establishing the consequences of alternative actions (Lin et al., 2026; Qiao et al., 2026; Chen et al., 2026a).

Our contributions are threefold: (i) a physics-guided latent inner-world formulation that couples explicit vehicle dynamics and resource state with history-conditioned operating context; (ii) an empirical evaluation of its rollout fidelity, physical consistency, compactness, and transfer using operational vehicle logs; and (iii) an imagination-based downstream decision pipeline, instantiated for energy management and assessed across learned and calibrated evaluation worlds. The resulting 85k-parameter representation matches a 272k-parameter GRU on speed and enforces fuel-cut structure. Cross-world policy disagreement identifies a remaining challenge for learning decision-relevant internal dynamics from observational logs.

2 Related Work

World models and embodied self-modeling. Latent world models learn compact representations and dynamics for imagined control (Ha & Schmidhuber, 2018; Hafner et al., 2020). PlaNet plans in learned latent dynamics (Hafner et al., 2019), DreamerV3 learns across diverse control tasks (Hafner et al., 2025), and DayDreamer applies world models to physical robots (Wu et al., 2023). Our focus

is the agent's internal physical operation: actuator responses, motion, latent operating context, and resource evolution. Robot self-modeling provides a related perspective, from task-agnostic self-models (Kwiatkowski & Lipson, 2019) to learned body morphology and kinematics for planning (Chen et al., 2022). ECOLATENT develops the inner-world perspective around recurrent operational dynamics and physical resource accounting, with a vehicle instantiation. It complements external-environment modeling rather than requiring reconstruction of the surrounding scene.

Physics-guided internal dynamics. Hybrid models such as APHYNITY separate known physics from learned residuals (Yin et al., 2021). ECOLATENT combines that structural principle with recurrent operating context and explicit internal-state accumulation. Vehicle look-ahead control uses topographic maps, physical dynamics, and dynamic programming (Hellström et al., 2009; Sciarretta et al., 2015); repeated vehicle passes can improve road-grade maps (Sahlholm & Johansson, 2010). In our instantiation, an effective training-trip force-balance map supplies external conditions to the inner world. It is retrospective preview rather than independently measured terrain ground truth.

Decisions through learned dynamics. Probabilistic ensembles such as PETS propagate dynamics uncertainty during planning (Chua et al., 2018). Offline model-based control faces distribution shift and model exploitation (Yu et al., 2020; Lin et al., 2026; Qiao et al., 2026); support-constrained offline learning motivates our behavior-density penalty (Kumar et al., 2019). A Bayes-adaptive treatment emphasizes that multiple dynamics can fit the same offline data (Chen et al., 2026a). Our cross-world policy evaluation examines this issue for internal dynamics. We demonstrate policy optimization through the inner world without assuming calibrated counterfactual uncertainty or measured on-road benefit.

3 DATA COLLECTION AND VEHICLE INSTANTIATION

3.1 ON-ROAD DATA COLLECTION

We follow the real-vehicle acquisition procedure described by Chen et al. (2026b), recording vehicle operation together with localization and surrounding-traffic information. The platform combines onboard CAN signals with GPS/RTK-GNSS, inertial sensing, forward perception, and V2X communication (Fig. 2). During road operation, the acquisition system records motion and trajectory information, powertrain and actuator signals, and available traffic-interaction information.

For ECOLATENT, we use the provider's timestamped, 1 Hz operational exports from two vehicles. The modeling pipeline retains CAN-derived motion and powertrain variables, GPS information for speed calibration and road mapping, and camera-derived lead information where usable. Raw images, point clouds, and V2X messages are not model inputs. Appendix A details the acquisition procedure, vehicle photographs, signal selection, temporal alignment, and preprocessing. Our experiments use the real-vehicle logs; the reference article's SUMO–MATLAB simulation dataset is not used.

3.2 MODELING VARIABLES AND DATASET PREPARATION

Problem formulation. Given a causal observation history h_t, an inner world model initializes a state z_0 and recursively simulates $z_{k+1} = f_\phi(z_k, a_k, e_k)$ under candidate actions a_k and external conditions e_k. The state combines observed physical variables with latent operating context; task-relevant trajectories can then be evaluated by a downstream decision objective. Here e_k includes grade at the simulated position, and the modeled internal dynamics cover longitudinal motion and powertrain operation. We use resource accounting and energy management to instantiate the model outputs and downstream objective. The experimental embodiment is a vehicle; adapting the formulation to a robot requires corresponding observations, actuator variables, and physical priors.

Logs and split. The data comprise 3,934 CSV fragments from two vehicles over April–May 2025, totaling 842 h at a provider-downsampled 1 Hz. Fragments are chained into contiguous segments and grouped into 276 trips. Trips beginning through May 8 form training (522 h, 167 trips); May 9–13 form validation (120 h, 35 trips); later trips form test (200 h, 74 trips). Windows require at least 15 km/h, valid fuel labels, and at least 90% map coverage. The main test set contains 29,808



Figure 2: **Onboard data-collection platform.** Sensor and communication overview for the real-vehicle acquisition procedure described by Chen et al. (2026b). The schematic shows the broader instrumentation; EcoLatent uses the exported CAN, localization, and selected perception-derived channels described in Section 3.1, rather than all pictured sensors as model inputs.

stride-10 s windows from 31 eligible trips. The effective sample size for uncertainty is therefore trips rather than overlapping windows.

Signals and actions. Accelerator pedal is not a complete action: during 49.8% of moving time it is zero while indicated engine torque averages 24%, consistent with unlogged cruise operation. We instead use the ECU-reported indicated torque fraction τ_t, brake fraction b_t, and retarder fraction r_t as the action $a_t = (\tau_t, b_t, r_t)$. Torque is an observed actuator-level quantity, not evidence that the dataset contains randomized control interventions. The measured dashboard speed is corrected using a vehicle-specific factor fitted on training GPS rows. Provider mass and gear are undocumented estimates. Radar lead distance fails a kinematic check; camera-derived lead distance is used in the exploratory policy simulation.

Internal resource labels. The ECU instantaneous fuel-economy signal lags torque by one sample. For moving rows we construct a fuel-rate proxy $f_t = v_{t+1}^{\mathrm{dash}}/\mathrm{FE}_{t+1}$ in L/h, with zero assigned when the aligned signal and torque indicate fuel cut. This supplies labels on 99.5% of moving rows. It is an ECU-derived proxy, not metered fuel flow. Fuel power is $P_{f,t} = f_t L/3.6$, with $L = 35.8$ MJ/L and P_f in kW. The kinetic-energy buffer is $E_{\mathrm{kin}} = 10^{-6} m v^2/2$ in MJ, using estimated mass. For policy comparisons we use $E_{\mathrm{eq}} = E_{\mathrm{fuel}} - \Delta E_{\mathrm{kin}}/\eta_m$, with calibrated $\eta_m = 0.430$; potential energy is omitted because the prescribed road is common to the compared controllers.

Road preview. Per-trip GPS altitude and provider grade are noisy. An effective grade is estimated per approximately 50 m geographic cell and 45-degree heading sector using training-trip traction-phase force balance, shrunk toward heavily smoothed fleet GPS altitude. Training-row lookup excludes that trip's direct cell contributions; validation and test rows only query the training map. The resulting grade may also absorb location-specific unmodeled resistance. On held-out traction rows, its fixed-physics acceleration R^2 is 0.69, versus -0.64 for provider grade (Appendix B, Fig. 8). Ninety-four percent of moving test samples lie in cells traversed by at least three training trips. The route and its map are retrospectively available; deployment would require comparable preview.

4 ECOLATENT: A LATENT INNER WORLD MODEL

Physical state and latent operating context. Let h_t be the preceding 30 s of causal CAN and map features, m the available mass estimate, and $g(s)$ the distance-indexed road profile. A 128-unit GRU encodes h_t into $c_0 \in \mathbb{R}^{32}$. We initialize

$$z_0 = (v_0, s_0, F_0, d_{\omega,0}, c_0), \qquad s_0 = F_0 = 0, \tag{1}$$

where F is cumulative fuel energy in MJ and $d_\omega = \log[\omega/\bar{\omega}(v)]$ is the dimensionless log offset from a calibrated speed-to-engine-speed map $\bar{\omega}(v)$. Thus $\omega = \bar{\omega}(v)e^{d_\omega}$. Mechanical energy follows from m and v; F is directly accumulated rather than decoded from an unconstrained latent vector.

Physics-guided state evolution. The vehicle instantiation couples longitudinal motion and powertrain resource use. Its nominal acceleration and fuel equations are

$$m(1+\delta)\dot{v} = \eta[\tau T_{\mathrm{ref}} - T_{\mathrm{loss}}(\tau)]\frac{\omega}{v} - (k_b b + k_r r)mg - c_r mg\cos\theta - k_a v^2 - mg\sin\theta, \tag{2}$$

$$P_f^{\mathrm{nom}} = \left(1 - e^{-\tau/0.005}\right)\max\{0, w_a(\tau T_{\mathrm{ref}}\omega/1000) + w_b\omega + w_c\}, \quad \theta = \arctan g(s), \tag{3}$$

with $T_{\mathrm{loss}}(\tau) = T_f + (T_{\mathrm{mot}} - T_f)e^{-\tau/0.02}$. These are calibrated effective equations: fitted $\eta \simeq 1$ should not be interpreted as a physical driveline efficiency. Here T_{ref} and T_{loss} are in N m, ω is in rad/s, and fuel power is in kW; the factor $1/1000$ converts indicated power to kW. The calibrated marginal fuel-to-wheel factor is $\eta/w_a = 0.430$. Appendix C.1 defines the remaining constants and numerical guards.

At each one-second step, a GRU cell updates context from $(z_k, a_k, g(s_k), m)$; a 128-unit residual MLP outputs $(\Delta a_k, \Delta P_k, \Delta d_{\omega,k})$. The model uses $v_{k+1} = \max\{0, v_k + \Delta t(\dot{v}_k^{\mathrm{nom}} + \Delta a_k)\}$, trapezoidal position integration, $F_{k+1} = F_k + 10^{-3}P_{f,k}\Delta t$, and $d_{\omega,k+1} = \lambda d_{\omega,k} + \Delta d_{\omega,k}$ with $\Delta t = 1$ s and $\lambda = 0.95$. Fuel power applies the same fuel-cut gate to a nonnegative transform of nominal power plus its residual. Road grade is looked up at the model's own s_k, including under alternative actions.

Learning from internal trajectories. We train 60-step open-loop rollouts with scaled squared errors on speed, position, fuel power, cumulative fuel, and engine speed, plus a residual-acceleration penalty of weight 0.01. The resulting model has 85,443 trainable parameters.

From imagination to decisions. Once fitted, the inner world generates trajectories under actions proposed by a policy. The policy observes the specified motion, powertrain, route, and task features; the recurrent context remains internal to the world model. Differentiating a trajectory objective through the frozen transition trains the policy to select actions. In the implemented case, the objective combines resource use with progress, terminal-speed, drivability, and action-support terms (Section 6). This separates the internal dynamics model from the downstream objective and makes energy management a use of the model rather than its sole output.

5 INNER WORLD MODEL EVALUATION

Protocol. All predictors use identical test windows, observed 30 s history where applicable, future logged actuator actions, mass estimates, and the same distance-indexed road information. Baselines are persistence, a history-and-preview ridge regressor, calibrated physics, a 272k-parameter recurrent GRU, and an 18k-parameter Markov physics-plus-residual model (PhysRes). Targeted variants remove nominal physics/fuel-cut gating or road preview. Learned models use three seeds and validation-selected checkpoints after 12,000 optimizer steps. Errors below are on recorded-action open-loop rollouts, not alternative-action outcomes. We report seed variation and paired bootstrap intervals over 31 test trips (2,000 resamples).

Internal-state rollout fidelity. Table 1 and Fig. 3 show that ECOLATENT and the larger GRU have essentially the same 60 s speed error: paired difference -0.003 m/s, 95% trip-bootstrap interval $[-0.009, 0.008]$. The cumulative-fuel MAE difference is -0.0047 MJ, interval $[-0.0110, -0.0014]$, about 2.4% lower relative to the GRU. Equivalent-energy MAE differs by -0.010 MJ, with an

Table 1: Internal-state rollout fidelity at 60 s. Values are mean ± seed standard deviation where applicable. Fuel nMAE divides cumulative-fuel MAE by mean logged cumulative fuel.

Model	Params	Speed RMSE (m/s)	Fuel nMAE (%)	E_{eq} MAE (MJ)
Persistence	0	1.82	50.0	7.47
Ridge	49k	$0.945 \pm .002$	$2.87 \pm .002$	$1.20 \pm .001$
Physics only	0	0.880	2.97	1.19
PhysRes	18k	$0.755 \pm .002$	$1.85 \pm .03$	$0.970 \pm .005$
GRU	272k	$0.656 \pm .006$	$1.65 \pm .02$	$0.831 \pm .015$
EcoLatent	**85k**	$\mathbf{0.653 \pm .009}$	$\mathbf{1.61 \pm .009}$	$\mathbf{0.821 \pm .008}$

interval spanning zero. Against PhysRes, ECOLATENT reduces speed RMSE by 0.102 m/s, interval $[-0.121, -0.071]$, and equivalent-energy MAE by 0.149 MJ. The result supports history-conditioned latent context and compactness, while offering little speed-accuracy advantage over a strong unstructured predictor.

Physical structure, memory, and external conditions. Removing road preview raises speed RMSE from 0.653 to 1.20 m/s; removing latent context (PhysRes) raises it to 0.755; removing physics raises it to 0.674. The same ordering holds for equivalent-energy error (0.821, 1.49, 0.970, and 0.859 MJ, respectively). Road information is therefore the dominant predictor of this task, rather than a small auxiliary input. The physics prior has a modest recorded-action accuracy effect but enforces zero predicted fuel when logged torque is zero. Without it, the no-physics model predicts more than 5 kW on 12.3% of fuel-cut steps; the GRU does so on 15.1%. The labels themselves do so on 7.8%, which limits how literally this check can be interpreted.

Transfer, regimes, and data efficiency. Training on vehicle A alone and testing on vehicle B gives ECOLATENT 0.927 ± 0.042 m/s speed RMSE and 2.06 ± 0.04% fuel nMAE, versus 0.970 ± 0.026 m/s and 2.70 ± 0.94% for the GRU. This is one held-out vehicle, so it does not establish fleet-wide transfer. On hilly windows ($n = 701$), ECOLATENT is worse than the GRU (1.33 versus 1.27 m/s); it also loses on long-uphill fuel error (2.7% versus 2.1%). At 5% of training trips ECOLATENT is least data-efficient (1.38 m/s), while PhysRes attains 1.12 m/s; the shared road map remains fitted on all training trips in this experiment. Figure 4 shows the crossover as more trips become available; Appendix B provides the full regime breakdown (Fig. 10).

Action sensitivity and counterfactual limits. Under a uniform $+0.05$ torque perturbation, ECOLATENT predicts higher speed in 97.6% of tested windows; the calibrated physics model does so in 100%. Yet the median implied marginal fuel-to-wheel efficiency is 0.309 for ECOLATENT and 0.416 for calibrated physics; other learned models imply 0.26–0.30. Observational actions are confounded by grade, load, and driver intent. Sign consistency checks are useful diagnostics, but they cannot tell which efficiency response is correct.

6 DOWNSTREAM DECISIONS: ENERGY MANAGEMENT

We now use the inner world as a differentiable simulator for a downstream decision problem. Energy management provides a concrete objective over internal motion and resource trajectories; the same modeling interface motivates other internal-control tasks, although those tasks are not evaluated here.

A 128-unit policy network is initialized by behavior cloning and optimized on training-trip scenarios by gradients through frozen, 60-step ECOLATENT rollouts, with validation-based model selection. Evaluation uses 2,743 non-overlapping held-out highway scenarios. The principal comparison uses a speed-band half-width of $\Delta v = 2$ m/s; Appendix D reports wider sensitivity checks. Its observation includes speed, progress, engine-speed offset, mass, lead-gap features, terminal-speed target, and grade samples up to 1 km ahead; it does not observe the world model's latent context. The objective combines E_{eq}, speed-band, headway, action-change, progress, and symmetric terminal-speed terms. Distance must match the logged 60 s distance and terminal speed is driven toward the

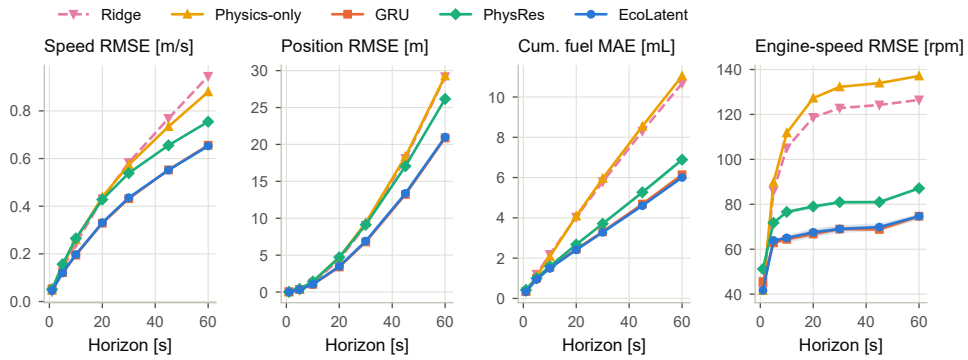

Figure 3: Internal-state rollout error versus horizon on later trips, with persistence omitted for scale. Future logged actions are provided to every predictor.

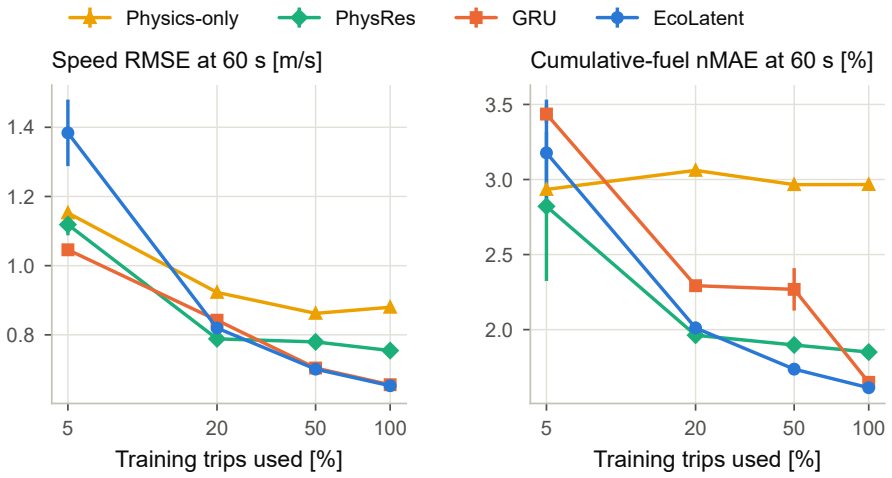

Figure 4: Internal-state rollout fidelity versus training-trip fraction. Physics is recalibrated on each subset, while the common fleet road map remains fixed; the figure measures predictor data efficiency conditional on that map. Error bars denote one seed standard deviation.

logged value. A behavior-density penalty discourages torque and brake commands outside the fitted training-action support. The lead trajectory is replayed without reacting to the ego policy.

In the optimization world, the policy's estimated improvement is 5.94% (trip-bootstrap 95% interval 4.99–7.01%); progress shortfall averages 2.22 m, terminal-speed deviation 0.77 km/h, and its speed-band RMS is 0.158 km/h. Removing the support penalty changes the estimated improvement only from 5.94% to 5.91%; the chosen support region contains 97% of held-out logged actions and the penalty is nearly inactive. The policy was revised twice after weaker endpoint formulations exploited the energy accounting; those failed formulations are documented in Appendix D.

Figure 5 shows substantial model dependence. Independent ECOLATENT seeds give similar answers but share architecture and data; PhysRes reduces the estimated benefit to 3.38%, and calibrated physics reduces it to 0.57%, where physics MPC ranks first. On the same scenarios, ECOLATENT's logged-action replay has 0.82 MJ mean absolute error per scenario, about 6% of mean log-derived equivalent energy (13.34 MJ), comparable to the claimed benefit. On a separate 60-scenario free-flow subset evaluated in calibrated physics, the policy uses 2.61% more equivalent energy than logged replay, while MPC uses 2.27% less. Full-preview, distance-grid dynamic programming gives a model-specific reference, not a bound on real driving; its objective is lower than MPC's, although their raw energy intensities are close. The different constraints and preview advantages preclude a universal performance ranking. Appendix D visualizes logged-action replay error, objective trade-offs, and a representative distance-domain rollout.

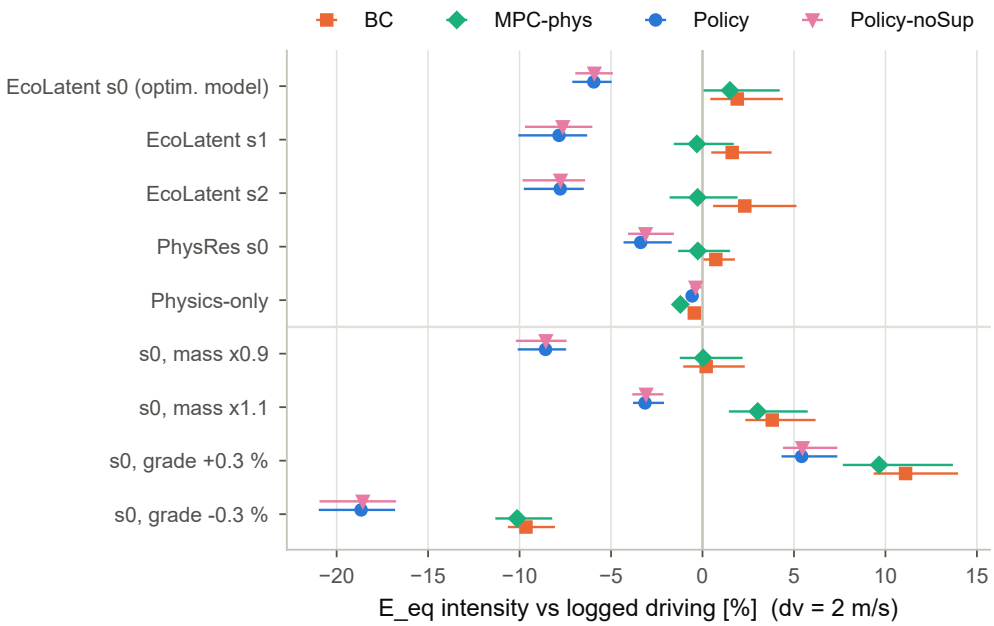

Figure 5: **Decisions depend on the evaluation world.** Equivalent-energy intensity change relative to logged-action replay within each world, at speed-band half-width $\Delta v = 2$ m/s. Negative values indicate lower modeled intensity. BC denotes behavior cloning, MPC-phys uses calibrated physics, and Policy-noSup removes the action-support penalty. Bars show 95% trip-bootstrap intervals (400 resamples). Lower rows perturb mass by $\pm 10\%$ or grade by ± 0.3 percentage points in EcoLatent seed 0. These are model estimates, not measured savings; exact unperturbed comparisons appear in Table 2.

7 Discussion and Limitations

The vehicle case study supports a compact, physics-guided inner-world representation: 85k parameters match a 272k GRU on 60 s speed, with modestly lower cumulative-fuel error and a structural fuel-cut rule. The ablations also narrow the claim. Most predictive value comes from a reliable road preview; the latent context needs substantial data and performs poorly on rare hilly and uphill regimes. The effective road map is trained on repeated routes, so performance may fall when routes are novel.

The control evidence remains observational and model-dependent. The fuel signal is an ECU proxy, mass and gear are estimations, and the fitted effective efficiency reflects scale uncertainty.

Broader inner-world scope. Internal trajectories could support powertrain coordination, actuator management, and robot motion/resource control. These extensions require embodiment-specific states, priors, and observations. The experiments establish the vehicle case, not a robot implementation or a complete model of every internal subsystem.

Conclusion. EcoLatent couples physical state with latent operating context to model the inner world of an embodied agent. Vehicle energy management demonstrates its action-conditioned imagination and decision interface, motivating complementary internal-world models for vehicles and robots.

REPRODUCIBILITY STATEMENT

Section 3 and Appendix A describe acquisition, preprocessing, and temporal splits. Section 4 and Appendix C.1 specify the model and fitting procedure; Appendices C and D document evaluation, uncertainty, and compute. The implementation retains preprocessing manifests, calibration settings, training logs, and per-window and per-scenario metrics for reproducing the reported analyses. The raw vehicle logs may require permission to redistribute; the numerical findings depend on their ECU-derived fields and the documented filtering choices.

AI USE STATEMENT

Generative AI tools assisted with manuscript drafting and editing. All AI-assisted text was verified against public sources. The authors remain responsible for all the content in the submission.

REFERENCES

Boyuan Chen, Robert Kwiatkowski, Carl Vondrick, and Hod Lipson. Full-body visual self-modeling of robot morphologies. *Science Robotics*, 7(68):eabn1944, 2022. doi: 10.1126/scirobotics.abn1944. URL https://robot-morphology.cs.columbia.edu/.

Jiayu Chen, Le Xu, Wen-Tse Chen, and Jeff Schneider. Bayes adaptive monte carlo tree search for offline model-based reinforcement learning. In *International Conference on Learning Representations*, 2026a. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/43f55776896a2e33239c2954519f605e-Abstract-Conference.html.

Zhengxian Chen, Qihang He, Chenggang Zhang, Han Hai, Yize Sun, Jieyu Wang, Chaosheng Huang, and Jun Li. Direct power prediction for energy management of ammonia–hydrogen hybrid heavy-duty vehicles. *Energy*, 360:141763, 2026b. doi: 10.1016/j.energy.2026.141763. URL https://doi.org/10.1016/j.energy.2026.141763.

Kurtland Chua, Roberto Calandra, Rowan McAllister, and Sergey Levine. Deep reinforcement learning in a handful of trials using probabilistic dynamics models. In *Advances in Neural Information Processing Systems*, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/3de568f8597b94bda53149c7d7f5958c-Abstract.html.

David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. In *Advances in Neural Information Processing Systems*, 2018. URL https://arxiv.org/abs/1809.01999.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pp. 2555–2565, 2019. URL https://proceedings.mlr.press/v97/hafner19a.html.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. In *International Conference on Learning Representations*, 2020. URL https://arxiv.org/abs/1912.01603.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, 640:647–653, 2025. doi: 10.1038/s41586-025-08744-2. URL https://www.nature.com/articles/s41586-025-08744-2.

Erik Hellström, Maria Ivarsson, Jan Åslund, and Lars Nielsen. Look-ahead control for heavy trucks to minimize trip time and fuel consumption. *Control Engineering Practice*, 17(2):245–254, 2009. doi: 10.1016/j.conengprac.2008.07.005. URL https://liu.diva-portal.org/smash/get/diva2:159541/FULLTEXT01.pdf.

Aviral Kumar, Justin Fu, Matthew Soh, George Tucker, and Sergey Levine. Stabilizing off-policy Q-learning via bootstrapping error reduction. In *Advances in Neural Information Processing Systems*, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/c2073ffa77b5357a498057413bb09d3a-Abstract.html.

Robert Kwiatkowski and Hod Lipson. Task-agnostic self-modeling machines. *Science Robotics*, 4(26):eaau9354, 2019. doi: 10.1126/scirobotics.aau9354. URL https://www.creativemachineslab.com/deep-self-modeling.html.

Haoxin Lin, Siyuan Xiao, Yi-Chen Li, Zhilong Zhang, Yihao Sun, Chengxing Jia, and Yang Yu. ADM-v2: Pursuing full-horizon roll-out in dynamics models for offline policy learning and evaluation. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/1651a9bfa452a13c9f26f71bfef1f68c-Abstract-Conference.html.

Zhongjian Qiao, Jiafei Lyu, Boxiang Lyu, Yao Shu, Siyang Gao, and Shuang Qiu. Model-based offline RL via robust value-aware model learning with implicitly differentiable adaptive weighting. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/7a209a1c557e5087272014e7dd37035f-Abstract-Conference.html.

Per Sahlholm and Karl Henrik Johansson. Road grade estimation for look-ahead vehicle control using multiple measurement runs. *Control Engineering Practice*, 18(11):1328–1341, 2010. doi: 10.1016/j.conengprac.2009.09.007. URL https://people.kth.se/~kallej/papers/vehicle_cep10.pdf.

Antonio Sciarretta, Giovanni De Nunzio, and Luis Leon Ojeda. Optimal ecodriving control: Energy-efficient driving of road vehicles as an optimal control problem. *IEEE Control Systems Magazine*, 35(5):71–90, 2015. doi: 10.1109/MCS.2015.2449688. URL https://hal.science/hal-01252197.

Philipp Wu, Alejandro Escontrela, Danijar Hafner, Pieter Abbeel, and Ken Goldberg. DayDreamer: World models for physical robot learning. In *Proceedings of the 6th Conference on Robot Learning*, volume 205, pp. 2226–2240, 2023. URL https://proceedings.mlr.press/v205/wu23c.html.

Yuan Yin, Vincent Le Guen, Jérémie Dona, Emmanuel de Bézenac, Ibrahim Ayed, Nicolas Thome, and Patrick Gallinari. Augmenting physical models with deep networks for complex dynamics forecasting. In *International Conference on Learning Representations*, 2021. URL https://arxiv.org/abs/2010.04456.

Tianhe Yu, Garrett Thomas, Lantao Yu, Stefano Ermon, James Zou, Sergey Levine, Chelsea Finn, and Tengyu Ma. MOPO: Model-based offline policy optimization. In *Advances in Neural Information Processing Systems*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/a322852ce0df73e204b7e67cbbef0d0a-Abstract.html.

A DATA COLLECTION AND PREPROCESSING PROTOCOL

A.1 REAL-VEHICLE ACQUISITION

The acquisition procedure follows the real-vehicle platform described by Chen et al. (2026b, Section 2.1.2 and Table 4). It combines vehicle-state sensing, localization, perception, and connected-traffic information. Figures 6 and 7 document the physical collection platform. The steps below distinguish acquisition from the subsequent processing of the operational exports used in this paper.

1. Instrument the vehicle and record internal signals. The onboard CAN interface provides motion and actuator-related channels, including vehicle speed, accelerator position, brake status, and engine speed. The operational exports additionally contain ECU-reported indicated torque, retarder level, brake pressure, and instantaneous fuel economy. These recorded channels preserve the relation between driver or onboard-controller actions and the vehicle's internal response. ECU-derived fuel economy is a proxy signal, not a separately metered fuel-flow measurement.

2. Acquire localization and traffic context. GPS/RTK-GNSS and inertial sensing provide localization and motion information. The reference platform uses a forward camera and millimeter-wave radar to observe surrounding traffic, while the connected-vehicle channel receives available roadside and nearby-vehicle information. The sensor overview in Fig. 2 includes the broader platform instrumentation. The exported dataset used here is a tabular signal product: raw imagery, LiDAR point clouds, ultrasonic data, dedicated torque-sensor streams, and V2X messages are not supplied to ECOLATENT.

3. Record road operation and export time series. Vehicle operation is recorded across acceleration, deceleration, cruising, and traffic interactions using the acquisition process above. The reference article illustrates this process on a 26 km route with suburban and expressway operation; that route length describes its reported test, not the total distance of our dataset. This paper analyzes 3,934 operational CSV fragments from two vehicles over April–May 2025, totaling 842 h after processing. The provider supplies downsampled 1 Hz exports with timestamps and channel values; native sensor sampling rates and the provider's clock-synchronization and downsampling algorithms are not available in these exports. We therefore describe alignment at the exported-signal level and do not assume hardware synchronization precision that the data cannot establish.

A.2 FROM EXPORTED RECORDS TO MODEL-READY WINDOWS

4. Parse, order, and segment. Files are parsed into a common signal schema and sorted chronologically for each vehicle. A timestamp gap exceeding 3.5 s starts a new contiguous segment. Within a segment, timestamps are rounded to integer seconds relative to its start, duplicate second indices retain the first record, and missing seconds are inserted. Continuous channels are interpolated only inside gaps of at most three samples; discrete channels are forward-filled. The preprocessing produces 2,678 segments and 3,030,961 rows, including 25,697 inserted rows. Interpolation supplies a uniform grid, not new independent observations.

5. Group trips before splitting. Consecutive segments from the same vehicle belong to a common trip when their separation is at most 1,800 s. This gives 276 trips. The split is assigned by trip start date: through May 8 for training, May 9–13 for validation, and subsequent starts for testing. A trip crossing midnight remains in its original split. The resulting durations are 521.8, 120.5, and 199.7 h, respectively, with 167, 35, and 74 trips. Trip grouping is also the unit of bootstrap resampling.

6. Calibrate speed and normalize actuators. For each vehicle, the speedometer factor is the median dashboard/GPS-speed ratio over training rows for which both speeds exceed 30 km/h. Dividing dashboard speed by this factor gives corrected speed; the factors are 1.04068 and 1.04675 for vehicles A and B. Distance is integrated by the trapezoidal rule within each segment. Indicated torque and brake-pedal percentages are divided by 100, retarder level by 5, and engine speed is converted from rpm to rad/s. Estimated mass and gear retain their provider-derived status.



Figure 6: **Side view of the data-collection vehicle.** The photograph documents the tractor–trailer platform used for the acquisition procedure. The experimental analysis uses the measured and provider-estimated fields in the operational exports, rather than inferring fixed mass or geometric parameters from the photograph.



Figure 7: **Front view of the data-collection vehicle.** The photograph complements the instrumentation schematic in Fig. 2. The supplied photographs obscure registration identifiers; identifiable location traces are not reproduced.

7. Align resource labels and audit signals. To compensate for the one-sample delay of instantaneous fuel economy, the fuel-rate label at time t uses dashboard speed and fuel economy at $t + 1$. Fuel cut is assigned when the aligned fuel-economy signal and current indicated torque are both zero during motion; rates above 120 L/h are rejected. The history encoder receives the previous aligned fuel-power value, preserving causality despite label alignment. Camera-derived lead distance is accepted only in its valid range (0–150 m); the radar-derived lead channels fail the kinematic audit and are excluded. GPS speed is used for calibration rather than as the final state-speed channel.

8. Build road preview and select windows. Road-grade estimates use training-trip force balance and smoothed fleet GPS altitude, with each training trip's direct cell contribution excluded from its own lookup. Validation and test trips query the training map. Windows contain 30 s of history and a 60 s future rollout within one contiguous segment. They require speed of at least 15 km/h, valid mass and engine speed, valid fuel labels, and at least 90% future map coverage. The main evaluation uses a 10 s stride and 29,808 windows from 31 eligible test trips. Policy evaluation selects every other eligible non-overlapping 60 s highway scenario, yielding 2,743 scenarios. This selection is distinct from the training-trip windows used to optimize the policy.

Channel roles. The 13 history features are corrected speed, backward-difference acceleration, engine speed, indicated torque, service brake, retarder, accelerator pedal, previous aligned fuel power, effective grade, estimated mass, estimated gear, lead-validity flag, and lead distance. Future actions supply indicated torque, brake, and retarder values; future observations are rollout targets. Provider estimates and ECU-derived quantities are not treated as independent ground truth for their underlying physical variables.

B SUPPLEMENTARY INTERNAL-WORLD DIAGNOSTICS

The following plots expand the main-text rollout and robustness evaluation. Figure 8 compares road-preview sources, Fig. 9 shows the nominal fuel-power calibration, Fig. 10 resolves performance by operating regime, and Fig. 11 includes the persistence baseline omitted from the main-text horizon plot.

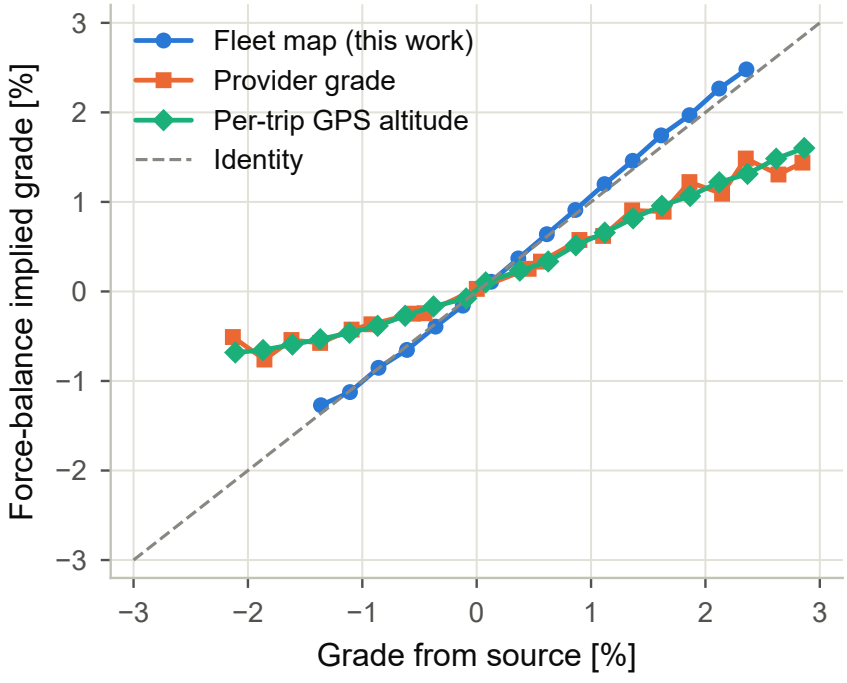

Figure 8: Held-out traction-phase validation of road-grade sources. The fleet map is inferred from training trips; its effective grade is not an independent terrain measurement.

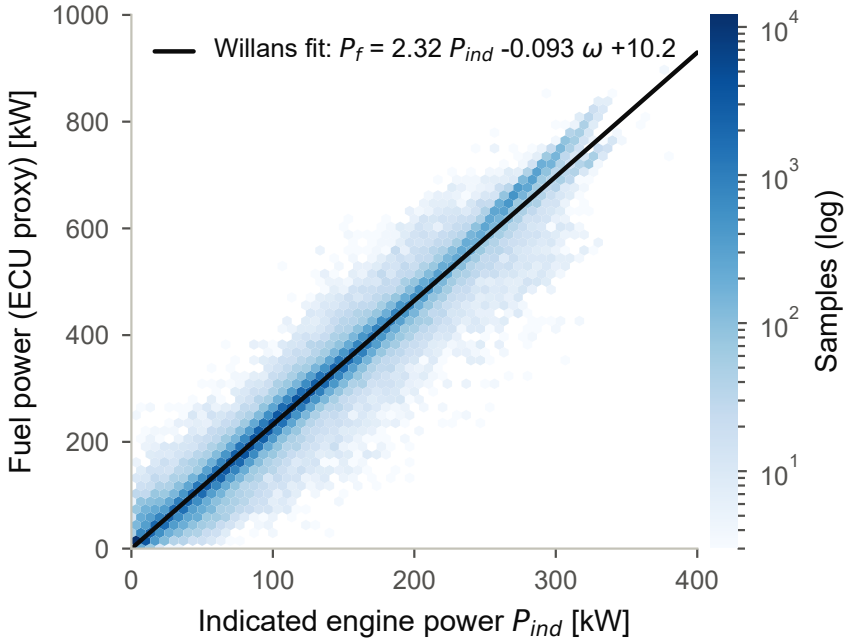

Figure 9: **Nominal powertrain calibration.** ECU-derived fuel power versus indicated engine power, with the fitted Willans relation. Color indicates sample counts on a logarithmic scale. The fit describes the observed operating data and does not establish a counterfactual fuel-to-wheel response.

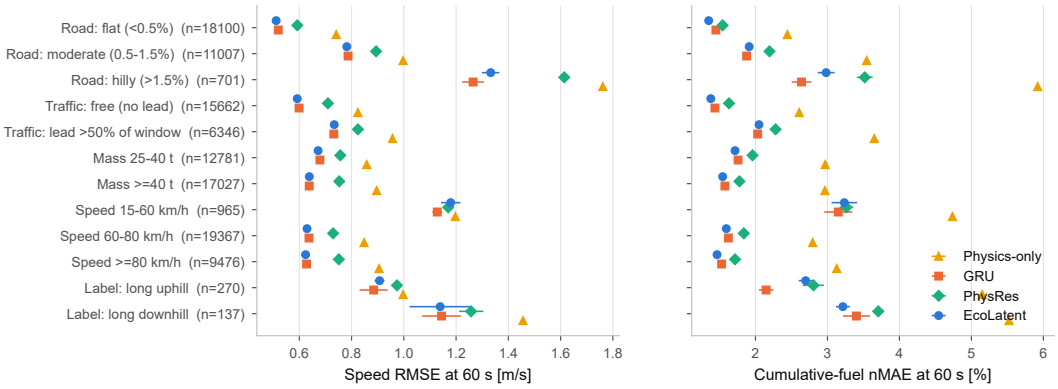

Figure 10: **Operating-regime breakdown.** Speed RMSE and cumulative-fuel nMAE at 60 s on recorded-action windows. Markers show seed means and bars one seed standard deviation. Each row defines a separate subset; subsets from different categories can overlap. Parenthesized counts refer to windows, not independent trips. The hilly and long-uphill subsets expose limitations masked by aggregate accuracy.

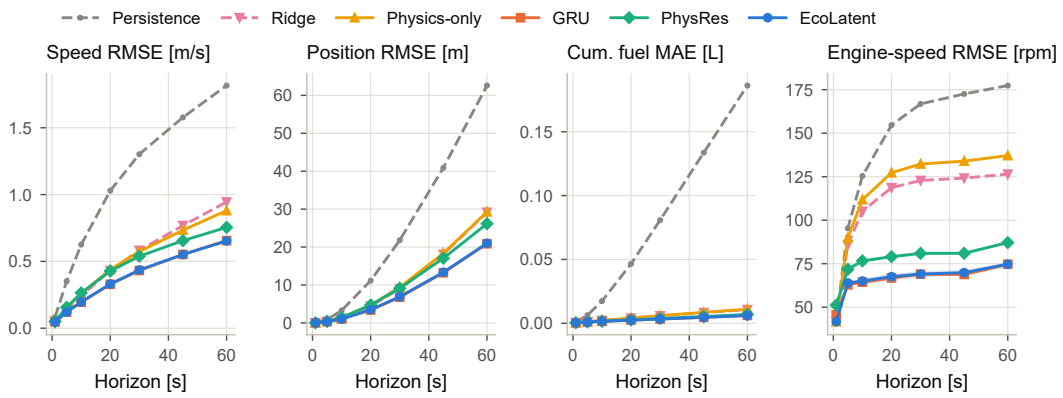

Figure 11: **Full-scale rollout errors.** Speed, position, cumulative fuel, and engine-speed errors versus horizon, including persistence. Cumulative-fuel error is shown in liters here and in milliliters in the zoomed main-text figure. All models receive the protocol's common action and road information where applicable.

C Additional Experimental Details

C.1 Model and Fitting Details

Calibration and fitting. The physics parameters are fitted on training rows. The effective driveline factor is 1.00, friction torque 152.6 Nm, motoring torque 319.5 Nm, rolling coefficient 0.00592, and Willans slope 2.32. The model-size pilot selected the 85k-parameter configuration trained for 12k steps; the 324k configuration at 4k steps had slightly better validation performance but did not justify scaling. Predictor training uses batch 512, AdamW at 2×10^{-3} with cosine decay, and validation selection every 500 steps. The main learned comparisons use three seeds. Computing consumed approximately 7.3 GPU-hours on one laptop RTX 4070, including about one hour of aborted or superseded runs; 6.34 hours are logged. A CUDA-graph training step matched eager execution bitwise in the recorded five-step check and was 7.3 times faster on that benchmark.

Units and transition details. Acceleration is in m/s^2, distance in m, engine angular speed in rad/s, fuel power in kW, and accumulated energy in MJ. The physics model uses $g = 9.81$ m/s^2, reference torque $T_{\mathrm{ref}} = 2573.81$ N m, and rotational-mass factor $\delta = 0.05$ as fixed constants; c_r, k_a, k_b, and k_r denote rolling, aerodynamic, service-brake, and retarder coefficients. The speed-to-engine-speed lookup is interpolated over its calibration grid, and the force calculation uses $\omega / \max(v, 0.5)$ to avoid division by zero. The context update receives scaled motion, action, grade, mass, nominal acceleration/power, and engine-offset features; accumulated distance and fuel do not directly enter that GRU cell. Residual outputs are scaled by 0.3 m/s^2, 50 kW, and 0.05 for acceleration, fuel power, and log engine-speed offset. The positive fuel transform is softplus with $\beta = 0.2$, followed by the fuel-cut gate. Squared rollout errors are normalized by 1 m/s, 20 m, 50 kW, 1 MJ, and 20 rad/s; the residual-acceleration penalty normalizes by 0.1 m/s^2 before applying weight 0.01.

Uncertainty and selection. Main prediction comparisons use paired cluster bootstrap over 31 test trips, 2,000 resamples. Test windows overlap at 10-s stride; treating them as independent would understate uncertainty. Cross-vehicle map coverage differs slightly across training settings, so the "trained on both" reference is not exactly paired with vehicle A-only training. Road-grade shrinkage was selected using validation data. Prediction errors are conditional on measured future actions and retrospectively available route preview.

D Policy Protocol and Negative Results

Endpoint accounting. A first policy with soft endpoint terms appeared to save about 20% equivalent energy by finishing 1.3 m/s slower and 20 m short. A one-sided terminal-speed requirement

Table 2: Estimated ΔE_{eq} intensity (%) versus logged-action replay on the same scenarios, in each evaluation world; lower is better. The policy is optimized in EcoLatent seed 0, while MPC optimizes calibrated physics. These are model outputs, not measured savings.

Controller	Eco s0	Eco s1	Eco s2	PhysRes	Physics
Behavior cloning	+1.90	+1.63	+2.31	+0.73	-0.45
Physics MPC	+1.50	-0.31	-0.27	-0.26	-1.21
EcoLatent policy	-5.94	-7.84	-7.78	-3.38	-0.57

instead encouraged overspeed and increased fuel. The final symmetric endpoint term and travel requirement were therefore applied to all learned controllers, with the terminal target included in their observations. This selection occurred during model development and should temper interpretation of the final estimate.

Support and traffic. The behavior model combines fuel-cut probability, Gaussian fueled torque, brake probability, and brake level. The support width $\kappa = 4$ was chosen on validation to flag 3.4% of held-out logged actions; a narrower width of 2 flagged 14%. An independent k-nearest-neighbor check flags 1.09% of actions for the regularized policy, 1.10% without the regularizer, and 1.04% for logged test actions. The headway term uses a replayed camera-derived lead position only when available; it is not a reactive traffic simulation.

Dynamic-programming reference. The offline reference uses calibrated physics, 10 m distance stages, a 0.02 m/s speed grid, inferred feasible controls, a hard terminal-speed tolerance of 0.25 m/s, and full road preview over the logged distance. On 60 selected free-flow scenarios its common objective is 13.41 MJ per scenario, versus 14.83 for 20 s physics MPC, 15.46 for the EcoLatent policy, and 16.25 for behavior cloning. Raw energy intensity can rank controllers differently because progress, endpoint, and speed-band penalties differ. The DP's grid and simplified physics prevent interpreting it as a real-world optimum.

Additional decision visualizations. Figure 12 compares model replay with log-derived equivalent energy; Fig. 13 varies the speed-band setting; and Fig. 14 illustrates a representative distance-domain trajectory. These plots provide diagnostics of the offline decision pipeline, not on-road controller validation.

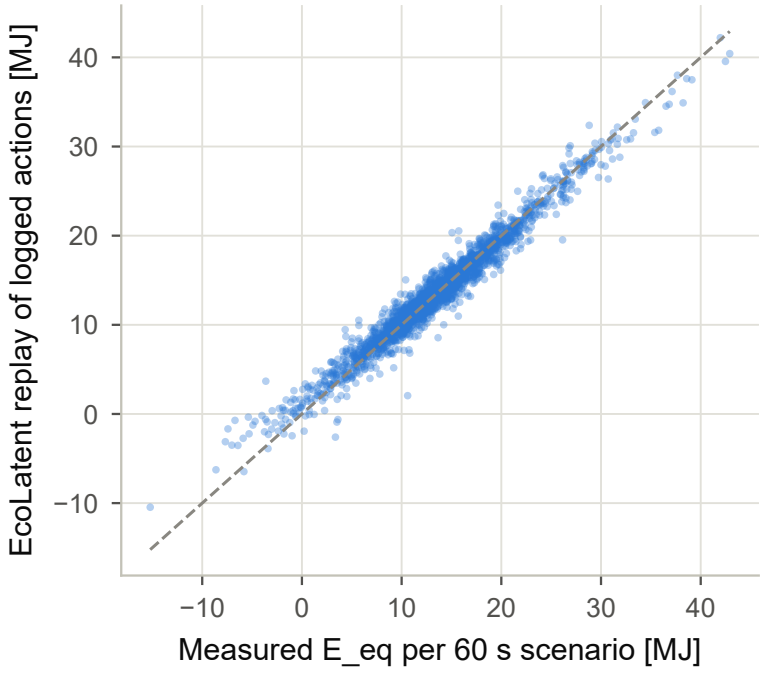

Figure 12: **Logged-action replay diagnostic.** Each point compares EcoLatent's 60 s replay with the equivalent energy constructed from the recorded signals for one held-out scenario. The dashed line is identity. "Measured" on the horizontal axis denotes the ECU-derived fuel proxy and estimated kinetic-energy adjustment, not independently metered fuel.

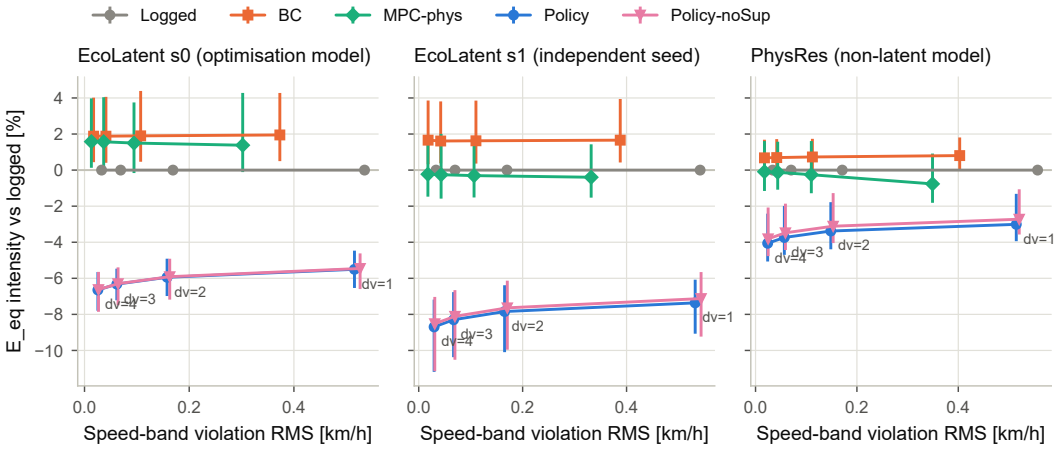

Figure 13: **Decision trade-offs across evaluation worlds.** Equivalent-energy intensity change versus RMS speed-band violation for half-widths $\Delta v \in \{1, 2, 3, 4\}$ m/s. Labels identify the policy's band settings; error bars are 95% trip-bootstrap intervals (400 resamples). Logged, BC, MPC-phys, Policy, and Policy-noSup denote logged replay, behavior cloning, physics MPC, the learned policy, and its version without the support penalty. The panels share scenarios but use different dynamics.

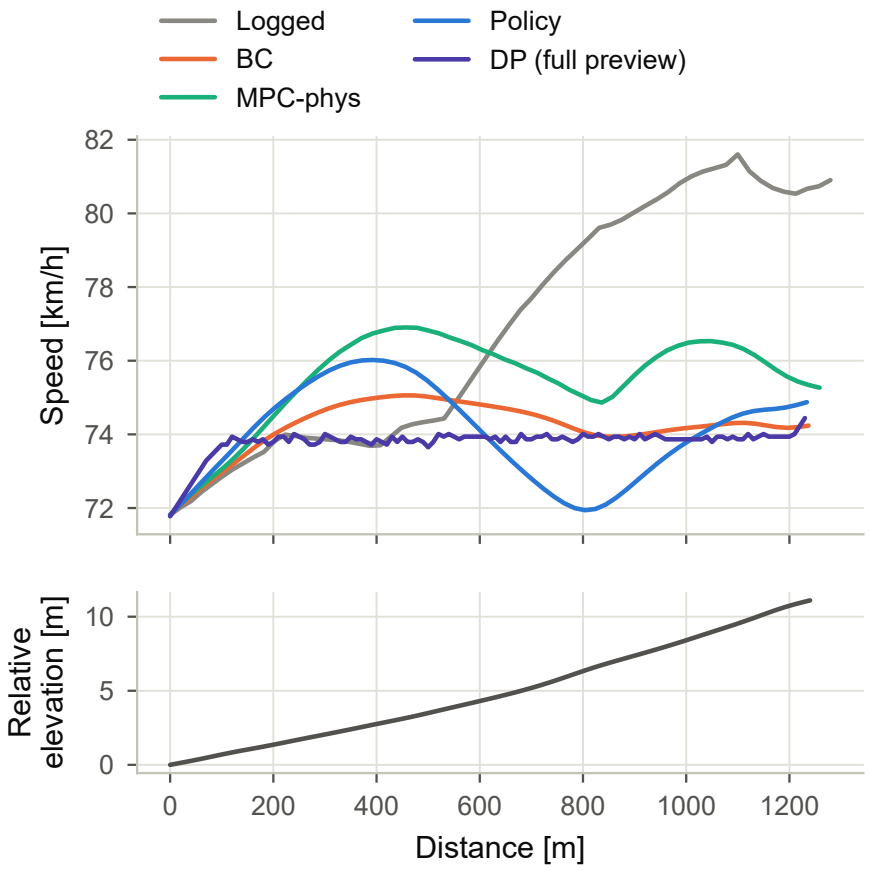

Figure 14: **Illustrative distance-domain decision rollout.** Speed trajectories for logged replay, behavior cloning (BC), physics MPC, the learned policy, and full-preview dynamic programming (DP), evaluated in calibrated physics on one selected free-flow scenario. The lower panel integrates the effective road-grade profile into relative elevation; it is not a surveyed terrain profile. Differences in horizon and constraints prevent interpreting DP as a real-world optimality bound.