REACHABILITY, NOT RESEMBLANCE: LEARNING THE PLANNING GEOMETRY OF LATENT WORLD MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

World models (WMs) let an agent act by imagining the consequences of candidate actions (Craik, 1943; Ha & Schmidhuber, 2018). Latent WMs (LeCun, 2022) roll candidate action sequences forward in latent space and keep the one whose predicted final latent lies nearest the goal, so the distance behind "nearest", the planning geometry, is what planning optimizes. Current models, LeWorldModel (LeWM) (Maes et al., 2026) among them, use a plain Euclidean norm in an encoder trained only to predict and to avoid collapse, a geometry with no reason to reflect temporal reachability, and whether the mismatch matters has never been measured. Here we show that the mismatch is benign for near goals and dominant for far ones, and that it can be repaired without retraining: Geometry for World Models (GeoWM), a head with 0.6% of the parameters fitted to the frozen encoder so that consecutive states lie within one unit and random pairs far apart, approximates temporal distance and, read as the terminal cost, restores long-horizon planning at no added planning cost. On four benchmarks the head matches or exceeds the baseline at the standard goal offset and separates from it as the goal recedes, raising success from 16% to 38% on Push-T and from 38% to 94% on Two-Room at four times that offset. The gain is in the cost, not the representation, and the head transfers unchanged to a WM of another family. The planning geometry is thus the lever for long-horizon planning, the one component prior work has held fixed while improving the rollout and the search around it. More broadly, what a planner is asked to measure matters more than the machinery around it: plan by reachability, not resemblance.

1 INTRODUCTION

That an agent might carry a small-scale model of the world and act by running it forward was proposed before the field existed (Craik, 1943); the world model (WM) (Ha & Schmidhuber, 2018) is its modern form. A reward-free, reconstruction-free instantiation has recently gained traction: Joint-Embedding Predictive Architectures (JEPAs) encode observations into a low-dimensional latent space and predict future latents instead of pixels (LeCun, 2022), then plan with sampling-based Model Predictive Control (MPC) in that space, from offline trajectories alone.

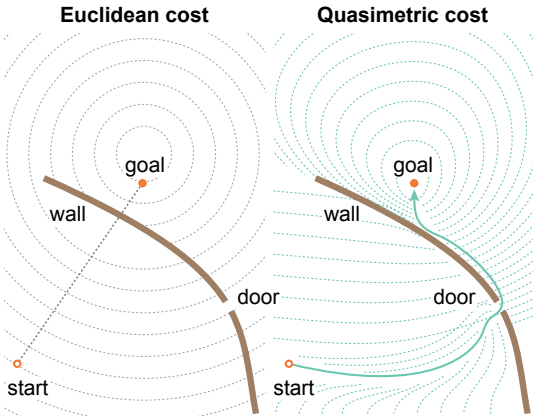

Figure 1: **Euclidean and temporal planning geometries around an obstacle.** A cost whose level sets ignore reachability misranks states however well the planner searches: Euclidean level sets (left) are concentric about the goal and cross the wall, so the descending path runs into the barrier, whereas a cost trained to approximate temporal distance (right) is blocked by the wall and emanates from the door, so the path detours through it. Our cost is a quasimetric, but Section N shows the asymmetry is not what produces this: its symmetrization gives the same field.

Planning in such a model is, concretely, distance minimization: a planner such as the Cross-Entropy Method (CEM) (Rubinstein & Kroese, 2004) rolls many action sequences through the learned dynamics and keeps the one whose predicted terminal latent lies nearest the goal, under $\mathcal{C}(\hat{z}_H) = \|\hat{z}_H - z_g\|_2^2$. The distance behind "nearest" is therefore the *planning geometry*, what planning optimizes. In LeWorldModel

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.

(LeWM) (Maes et al., 2026), the model we build on, and its predecessors, that geometry is Euclidean distance in an encoder trained for next-step prediction and collapse avoidance: it measures *resemblance*, how alike two states are to an encoder trained to predict, not *reachability*, how many actions separate them. An agent separated from its goal by a wall (Tolman, 1948), or a manipulator that must back away before it can push, is a case where the two come apart (Fig. 1).

Whether the mismatch matters depends on how far the goal is. Within one rollout horizon a Euclidean cost is good enough: the candidates already bracket the goal and replanning corrects the residue. As the goal recedes the cost must instead order candidates that all fall short, exactly where resemblance and reachability disagree; its misorderings are no longer absorbed, and planning degrades steadily.

Work on latent WMs has improved the rollout and the search around this cost, or sidestepped the model with an amortized goal-reaching policy that never rolls the dynamics out (Ghosh et al., 2021); the cost itself has stayed fixed (Section 2). We change only the cost, and find that doing so needs no retraining. Our method, Geometry for World Models (GeoWM), fits a small quasimetric head (Wang et al., 2023) that shapes latent distance toward temporal distance, not by regressing step counts but by holding consecutive states within one unit and pushing random pairs apart. Fitted to a *frozen* encoder, it leaves representation and dynamics bit-identical to the baseline and replaces only the terminal cost at test time. Fitting the head jointly with the encoder buys a median of 0.7 points for 40% more training time; we report it as an upper bound (Section 4.4).

Throughout, *long-horizon* refers to the temporal distance from start to goal, not to the number of steps the planner imagines ahead, which is held fixed. The result is a long-horizon *separation*: at the protocol offset of 25 steps the head is never behind the baseline and on two of four benchmarks only level, and as the goal recedes the two separate on every benchmark; at four times that offset LeWM has fallen to 16% on Push-T and 38% on Two-Room while the head holds 38% and 94%, over three seeds. Because the head is fitted to a frozen encoder it costs nothing at short horizon, so the curves separate rather than cross, and the separation is larger on episodes no arm trained on.

Contributions (i) We identify a long-horizon failure mode of Euclidean-cost latent planning and trace it to a planning geometry that does not encode temporal reachability, measured directly rather than inferred from task success (Sections 3 and 4.2). (ii) We propose GeoWM, a head fitted to a frozen WM that turns latent distance into a planning geometry for 0.6% more parameters, no retraining, and no added planning cost (Section 3); its quasimetric structure is not what carries the effect, since symmetrizing it costs 8.0 points on Push-T and nothing elsewhere (Section N). (iii) All four benchmarks, over three seeds and swept to the longest horizon each admits, show the predicted separation; the head repairs the measured geometry on all four without the encoder changing, to within 0.031 of joint training, and that repair does not predict the planning gain (Section 4.3). (iv) The effect lives in the cost: a representation shaped by the same objective but read by the Euclidean cost recovers between -4% and $+27\%$ of the gain, depending on the benchmark (Section 4.4). (v) The same head, unchanged, attached to a WM of another family recovers a comparable multiplier on Push-T from one frozen checkpoint, so the effect is not tied to one training recipe (Section 4.3).

2 Background

Latent planning and its fixed cost JEPAs predict future embeddings rather than pixels and differ mainly in how they prevent collapse: DINO-WM freezes a pretrained encoder (Zhou et al., 2025), PLDM regularizes variance and covariance across five loss terms (Sobal et al., 2025), and LeWM pairs a prediction loss with a Gaussian-embedding regularizer that provably prevents collapse (Maes et al., 2026). Control in all of them is CEM (Rubinstein & Kroese, 2004) under receding-horizon MPC, minimizing the squared Euclidean distance between the predicted terminal latent and the goal embedding. The effort since has gone to the rollout and the search, through parallel action-prefix prediction (Gao & Xu, 2026), hierarchical planning (Zhang et al., 2026), and test-time adaptation (Wang et al., 2026), or to sidestepping the model with an amortized policy (Ghosh et al., 2021); the cost itself has stayed fixed.

Learned planning costs Outside this setting the cost a planner minimizes has been learned before, as a bootstrapped terminal value (Hansen et al., 2022; 2024), as a value read as a distance for graph search (Eysenbach et al., 2019), and as a temporal-distance representation descended by a policy

(Park et al., 2023; Wang et al., 2023; Myers et al., 2025); all learn a *value*, and none rolls out a WM at test time. Quasimetric reinforcement learning (RL) establishes that the optimal goal-reaching value is a quasimetric (Wang et al., 2023; Myers et al., 2025). We import that structure into a WM and read it as the terminal cost, which raises a question value learning does not face: a distance accurate on encoder outputs need not stay accurate on the latents a rollout drifts into, the concern of value-equivalent model learning (Grimm et al., 2020; Lambert et al., 2020). We build on the LeWM encoder so that the backbone, data, budget, and planner are fixed, and the planning geometry is the only variable.

Concurrent work Recent preprints add temporal or reachability structure to models of this family, most on the same backbone (Destrade et al., 2026; Li et al., 2026b; Thil et al., 2026; Li et al., 2026a; Bai & Xiong, 2026). TRM (Li et al., 2026a) also keeps the WM frozen and reads a reachability head as the terminal cost, but regresses $|t_i - t_j|$ with an explicitly symmetric head, where ours is constrained rather than regressed, and asymmetric. TD-JEPA (Bai & Xiong, 2026) fits a quasimetric head jointly with the model, evaluates at the protocol offset alone, finds the temporal cost *losing* there on Push-T, and reverts to Euclidean, exactly the regime in which Section 3.2 predicts the metric is not binding. None sweeps the goal distance, so none reports the separation or the attribution between cost, representation, and search; Section A discusses each.

3 Method

3.1 Setup and the Latent World Model

The setting is reward-free goal-conditioned control from pixels. Training data is an offline dataset of trajectories $\{(o_{1:T}, a_{1:T})\}$ from unspecified behavior policies, without reward labels; at test time the agent receives a current observation o_1 and a goal observation o_g, and succeeds if an environment-specific predicate is met within a step budget. An encoder f_θ maps an observation to a latent $z_t = f_\theta(o_t)$, and a predictor g_ϕ models action-conditioned dynamics $\hat{z}_{t+1} = g_\phi(z_t, a_t)$. Following LeWM, both are trained end-to-end with a next-latent prediction loss and the Sketched-Isotropic-Gaussian Regularizer (SIGReg) weighted by λ,

$$\mathcal{L}_{\text{LeWM}} = \|\hat{z}_{t+1} - \text{sg}(z_{t+1})\|_2^2 + \lambda \,\text{SIGReg}(Z), \tag{1}$$

where sg is the stop-gradient and Z collects the latents in a batch. Planning then solves

$$a^\star_{1:H} = \arg\min_{a_{1:H}} \mathcal{C}(\hat{z}_H), \qquad \mathcal{C}(\hat{z}_H) = \|\hat{z}_H - z_g\|_2^2, \tag{2}$$

by CEM, executing the first K actions before replanning from the new observation ($K = H$ in the LeWM recipe).

3.2 Why a Euclidean Planning Geometry Fails at Long Horizons

Nothing in Eq. (1) constrains the *multi-step* structure of the latent space: the prediction term is a one-step constraint, and the regularizer shapes the marginal rather than the temporal relation between distant states. The Euclidean distance minimized by Eq. (2) therefore measures resemblance, predictive or perceptual similarity, rather than the number of actions separating two states, and the two diverge whenever an obstacle makes nearby-looking states far apart in time, or progress is non-monotone in appearance.

Why, then, does Euclidean-cost planning work at all? Because at short horizons the planner absorbs the error: the candidates CEM considers already bracket the goal, so the cost only breaks ties among nearby options, and the receding-horizon replan corrects the residue before it compounds. At long horizons the same cost must instead order candidates that all fall short, supplying a descent direction across a large region of latent space, exactly where a geometry that misrepresents reachability provides no usable gradient and where no amount of sampling within a fixed horizon repairs it. The prediction is sharp and testable: the two costs should differ little at short horizons compared with the gap that opens as the goal recedes, and the shaped geometry should never be the weaker of the two.

3.3 A Quasimetric Planning Geometry

We want a learned $d_\psi(z_s, z_g)$ that approximates the number of actions needed to reach g from s. Such a function cannot in general be a metric, because reachability is directional: a state at the edge of a ledge is one action from the bottom, and the bottom cannot reach it back. The structure that fits is a *quasimetric*, which keeps non-negativity and the triangle inequality but drops symmetry, and we parameterize d_ψ as a metric residual network (MRN) head (Liu et al., 2023) on encoder outputs. A projection $\psi : \mathbb{R}^{192} \to \mathbb{R}^{K \times M}$ maps a latent into K groups of M components, and the distance is the group-averaged, component-wise maximum of the positive part of the difference,

$$d_\psi(x, y) \;=\; \frac{1}{K} \sum_{k=1}^{K} \max_{m \leqslant M} \operatorname{relu}\big(\psi(x)_{k,m} - \psi(y)_{k,m}\big). \tag{3}$$

This construction is asymmetric by design and satisfies the triangle inequality for every choice of ψ, so the structural properties hold throughout training rather than being approached asymptotically (Section C). The motivation above is an argument, not a result: Section N tests it by symmetrizing this same head, replacing $\operatorname{relu}(\psi(x) - \psi(y))$ with $|\psi(x) - \psi(y)|$ at identical parameters and supervision, and the asymmetry earns its place on one benchmark of four. What the structure does earn is extrapolation. A clip is four frames, so no head here is ever shown a separation beyond three action blocks; an unrestricted multilayer perceptron (MLP) of matched capacity regressed on that separation plans *below* the Euclidean baseline at long range, whereas the quasimetric extrapolates from the same window (Section M).

3.4 The Quasimetric Objective and How It Is Fitted

The head is fitted by adding one term to Eq. (1),

$$\mathcal{L} = \mathcal{L}_{\text{LeWM}} + \beta\, \mathcal{L}_{\text{qm}}, \qquad \mathcal{L}_{\text{qm}} = \underbrace{\mathbb{E}_{s,s'}\, e^{-d_\psi(z_s, z_{s'})}}_{\text{push apart random pairs}} + \mu\, \underbrace{\mathbb{E}_{s \to s'}\big[\operatorname{relu}\big(d_\psi(z_s, z_{s'}) - 1\big)^2\big]}_{\text{consecutive states} \leqslant 1 \text{ step}}. \tag{4}$$

The two terms encode the definition of temporal distance from opposite sides. The local term states that states one environment transition apart are at most one unit apart under d_ψ, which fixes the unit of the metric and anchors it to the action scale; the push term states that randomly paired states should not be close, which excludes the trivial solution $d_\psi \equiv 0$. Together they aim at the shortest-path structure of the transition graph: the local constraint propagates along trajectories, and the push term supplies the pressure that makes the propagation tight. Wang et al. (2023) show that this pair imposed exactly, maximizing $\mathbb{E}[d_\psi]$ subject to $d_\psi(z_s, z_{s'}) \leqslant 1$ on every transition, recovers $-d_\psi = V^\star$ for deterministic dynamics under sufficient coverage, since the shortest-path distance is the largest quasimetric dominated by the unit step. Equation (4) departs from that program in two respects: the weight μ of the local term is a fixed hyperparameter rather than a learned dual variable, and the push is e^{-d_ψ}, whose gradient vanishes on pairs already far apart. We therefore take the recovery result as the reason these are the right two terms to write, not as a property of our fixed point, and measure the geometry that results (Section 4.2). The three terms of the joint objective constrain different objects: the prediction loss the dynamics, as the only action-conditioned term; the regularizer the embedding distribution; the quasimetric term the distances. Section C shows that none can stand in for another.

Frozen and joint The two forms differ only in whether the gradient of $\mathcal{L}_{\text{qm}}$ reaches the encoder. In the *frozen* form, GeoWM, the encoder and the predictor are a trained LeWM model held fixed and only ψ receives gradient, so representation and dynamics are bit-identical to the baseline and the terminal cost is the sole difference between the arms; we verified this by replanning the fitted checkpoint with the head disabled and reproducing the baseline row cell for cell. The head has 115,200 parameters, 0.6% of the model, and fitting it costs 6.3 ± 0.2 hours on one H100 against a checkpoint that already exists. In the *joint* form, GeoWM (joint), the same term enters the WM objective with weight β and its gradient reaches f_θ as well, which on Push-T raises training time by 40% (Section B). We expected the joint form to be the stronger, since planning scores *rollout* latents $\hat{z}_H$ that drift from the encoder manifold as H grows; measurement bears that out only in part (Section 4.4). We lead with the frozen form because it isolates the terminal cost as the cause, is a plug-in for a model that has already been trained, and is what can be attached to a trained model of a different family (Section 4.3).

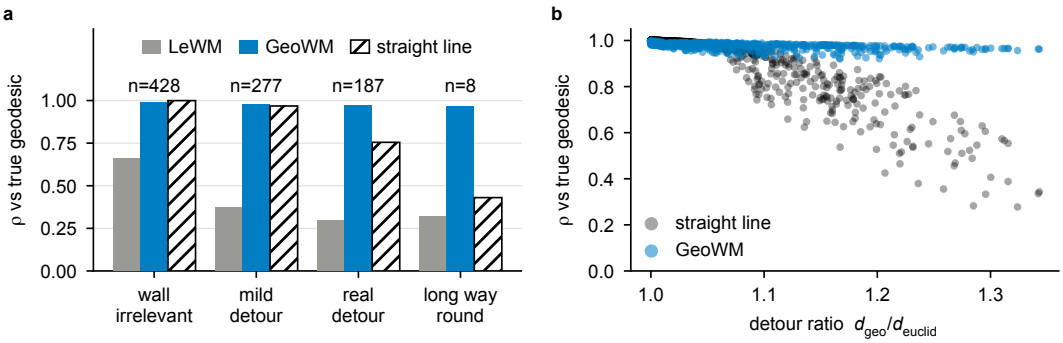

Figure 2: **The learned cost recovers reachability, not distance.** Both panels are measured against *exact* geodesics on Two-Room with no trajectory in the loop. **(a)** Spearman ρ against the true shortest-path distance, averaged over goals and stratified by the detour ratio $d_{\text{geo}}/d_{\text{euclid}}$. The hatched series is the null, the straight-line distance itself, and therefore what a cost that had learned only distance could score: it starts at 1.000 where the wall bends no path and collapses to 0.430 where it does, while the head holds near 0.97 throughout. **(b)** The same, ungrouped, over all 900 held-out goals. Section H gives the numbers.

Planning and configuration At test time the LeWM planner is unchanged in every respect (candidate count, iteration count, elite fraction, horizon, receding-horizon schedule), and only the terminal cost of Eq. (2) is replaced by $d_\psi(\hat{z}_H, z_g)$, one projection whose cost is negligible against the rollout; Algorithm 1 states both steps. One head configuration is held constant everywhere: an MRN head of 16 groups over a 256-dimensional projection with $\mu = 5$ and $\beta = 1$, the first values tried and never swept, so no number here rests on a hyperparameter chosen against the evaluation it is reported on, and the LeWM regularizer at its published $\lambda = 0.1$. Four arms share architecture, data, optimizer, epoch budget, and planner: **LeWM**, the unmodified baseline; **GeoWM**, the head fitted to that frozen baseline; **GeoWM (joint)**, the two trained together; and **shaped-enc.**, the trained GeoWM (joint) encoder scored by the Euclidean cost of Eq. (2), which separates the shaped representation from the metric that shaped it (Section 4.4).

4 Experiments

4.1 Benchmarks, Protocol, and Arms

We evaluate on the four benchmarks used by LeWM, spanning three control modalities: Push-T (2D manipulation), OGBench-Cube (3D manipulation), Two-Room (2D navigation), and Reacher (DeepMind Control). Two-Room and Push-T are the settings in which reachability and resemblance most obviously come apart; Cube and Reacher provide a manipulation and a control contrast. Every arm follows the LeWM recipe, a Vision Transformer (ViT)-Tiny encoder, a ViT-S predictor, frame-skip 5, 10 epochs, and CEM with 300 candidates and 30 iterations on Push-T and 10 elsewhere, as specified per environment by Maes et al. (2026). The published protocol samples a start state from a dataset trajectory, takes the goal to be the state 25 steps later in the same trajectory, and allows 50 environment steps. Section B gives every architectural and optimization detail and Section D the verified dataset statistics.

Data, splitting, and fairness We train on the released archives, split $90\%/10\%$ with the published seed but at *episode* rather than clip granularity, which withholds whole trajectories and costs the baseline nothing (Sections D and E). Within every arm pair the training split and the 50 evaluation $(\text{start}, \text{goal})$ pairs are identical by construction. We report two protocols, *in-distribution*, which reproduces LeWM by sampling pairs from all episodes, and *held-out*, which samples only from episodes no arm trained on; the two are not comparable to each other, but each is internally paired. Mechanism questions are settled on Push-T, where the diagnostic arms separate at all, since every GeoWM configuration saturates Two-Room.

Reproduction and the random floor On Push-T our LeWM reaches 94.0 ± 4.0 against the published 96.0 ± 2.83, and elsewhere differs by at most ten points with the sign mixed, the signature of reproduction noise rather than a weakened baseline (Table A1). A random-action baseline calibrates every other number: it satisfies the success predicate 45.3% of the time on Cube and 5.3% on Push-T at offset 25, mostly from start states that already satisfy it, and falls to 0–8% at the longest offset; the comparisons that carry our claim sit on a floor of essentially zero (Section F).

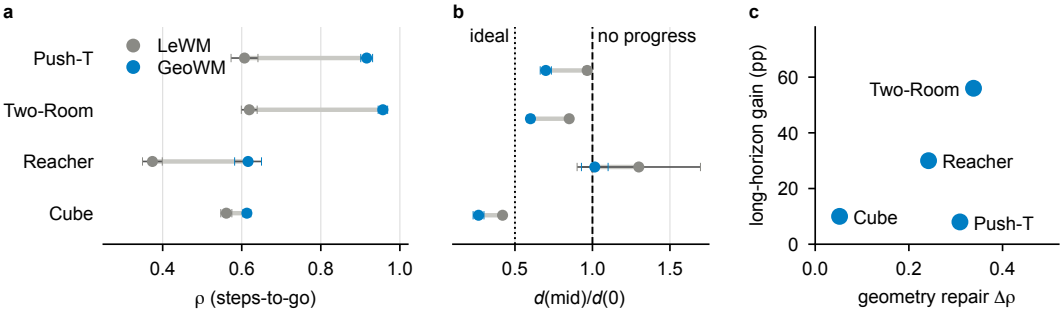

Figure 3: **The head repairs the global ordering of distances on a representation it never altered.** All quantities are measured directly on held-out episodes, not inferred from task success; each row joins the baseline (grey) to the head (blue), so the connector is the repair. **(a)** Rank correlation between latent distance and true steps-to-go: the head reaches 0.92 on Push-T and 0.96 on Two-Room and halves the across-seed spread; the repair on Reacher and Cube is real but smaller, and joint fitting moves any value by at most 0.031. **(b)** The cost halfway along a trajectory relative to its start, for which 0.5 is ideal and 1.0 registers no progress; the Reacher baseline exceeds 1 in all three seeds, and the head reduces it to 1.015 without removing it. **(c)** The repair against the long-horizon planning gain at the largest offset each benchmark admits: positively associated but not proportional. Error bars are the spread across three seeds; Table A3 gives every number.

4.2 The Planning Geometry, Measured Directly

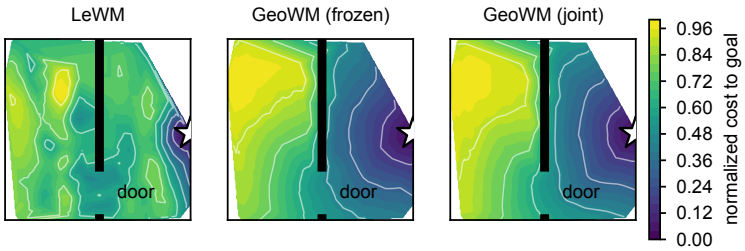

Figure 4: **The same latent space, read as a spatial map.** Held-out Two-Room states are placed at their true agent position and colored by the planning cost each model assigns to one fixed goal (star); white curves are level sets, and the black bar is the wall and its door. The Euclidean cost (left) is patchy, with spurious interior extrema and closed level sets, local minima a descending planner can be trapped in, whereas the quasimetric fitted to the frozen encoder (center) or jointly with it (right) decreases smoothly toward the goal and funnels through the door. This is what Table A3 measures numerically, shown spatially.

Our claim is about geometry, so we measure it directly rather than inferring it from task success. For held-out trajectory segments we compute the rank correlation ρ between the latent distance from a state to a later goal and the number of steps separating them *along the recorded trajectory*, and the ratio of the cost at a trajectory midpoint to the cost at its start, for which 0.5 is ideal and 1.0 means the cost registers no progress. Steps along a recorded trajectory are an upper bound on reachability, not reachability itself, since a behavior policy dawdles and detours; Fig. 2 closes that gap on Two-Room, whose obstacle geometry is known, by correlating each cost against the exact geodesic instead. Table A3 reports both statistics for all four benchmarks over three seeds, Fig. 3 plots them, and Fig. 4 shows the same quantity spatially on Two-Room.

The head does what it is designed to do, and does it without the encoder changing: on Push-T it lifts ρ from 0.607 to 0.916 and on Two-Room from 0.619 to 0.957, roughly halving the across-seed spread, with Reacher and Cube receiving a smaller repair. Since the encoder it reads is the trained baseline, bit-identical to the arm it is compared against, that difference is the work of d_ψ alone; fitting the head jointly with the encoder adds at most 0.031 on any benchmark. The midpoint statistic shows how severe the Euclidean pathology is: a Push-T trajectory halfway to its goal still sits at 97% of its starting cost under LeWM, leaving a planner almost no gradient to descend, against 0.699 once the head supplies the cost. Against exact geodesics on Two-Room, the straight-line null collapses from 1.000 to 0.430 as the wall bends the path, while LeWM sits near 0.3 and the head holds near 0.97 (Fig. 2), so what the cost recovers is reachability rather than distance.

4.3 A Long-Horizon Separation

Figure 5 plots the sweep, Table A5 gives the numbers with per-seed values in Table A6, and Section I shows what the failures look like on the ground. We sweep the goal offset D, the temporal

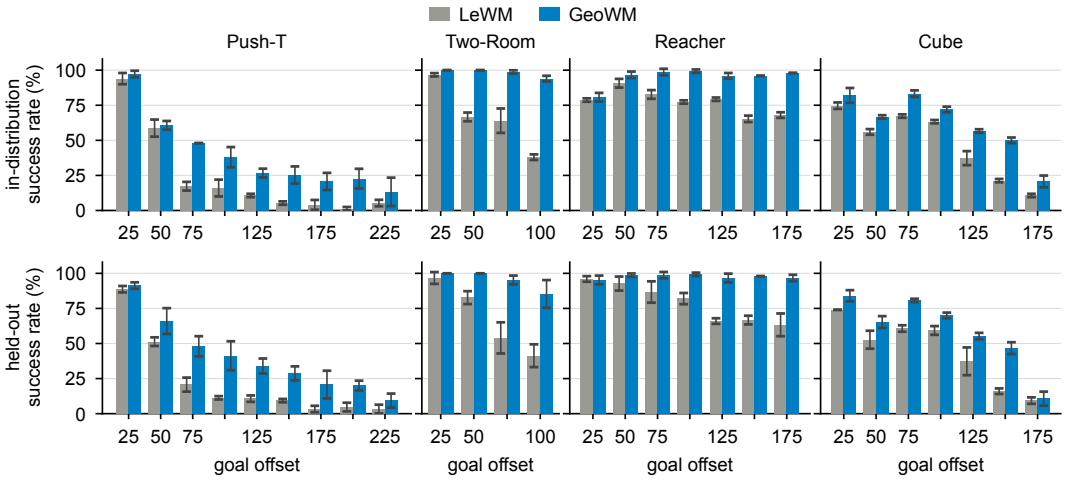

Figure 5: **The two costs are close when the goal is near and separate as it recedes.** Each panel pairs the two arms at every goal offset of one benchmark, trained on identical data with identical splits and scored on identical (start, goal) pairs, so a pair of bars differs only in the terminal cost. Bars are the mean over three seeds and whiskers one standard deviation across them; the GeoWM whisker vanishes where its three seeds agree exactly. The Two-Room sweep stops at offset 100 because its trajectories are at most 101 steps long.

separation between start and goal along the recorded trajectory: the *task* horizon, distinct from the *rollout* horizon H the planner imagines over, which is fixed at 5 action blocks in every cell and swept separately in Section L. The budget is offset + 50, except that the offset-25 column keeps the published budget of 50; both arms always share whatever budget a cell uses.

The shape of the effect At the protocol offset the head is never behind. Push-T and Reacher are level there against the across-seed spread, which is what Section 3.2 predicts; Two-Room and Cube are already ahead, by 3.3 and 7.3 points in distribution and 3.3 and 10.0 held out, in every seed, so the horizon at which the Euclidean cost begins to bind is benchmark-dependent. As the goal recedes the arms separate on every benchmark. On Push-T, by offset 100 LeWM has collapsed to 16.0 ± 6.0 against 38.0 ± 7.2, and at offset 225, nine times the protocol horizon, it sits at 5.3 ± 2.3 against 13.3 ± 10.1. On Two-Room LeWM falls to 38.0 ± 2.0 while the head holds 94.0 ± 2.0. On Reacher the head stays horizon-independent while the baseline falls to around 65, the sharpest form of the prediction, since the metric is what stops being adequate as the horizon grows and nothing else about the model has changed. Cube marks where the ceiling belongs to the task rather than the metric: the margin is widest at offset 150 (50.0 ± 2.0 against 21.3 ± 1.2), but by 175 both arms are near failure; Cube also carries two observations we cannot explain, recorded in Section R. The pattern is not that the head improves everywhere by a constant amount but that the Euclidean cost degrades with horizon and the learned one does not. The arms do not cross, since the frozen head costs nothing at short horizon, and the separation is *larger* on episodes the model never trained on, the opposite of what memorization would produce.

A second backbone To ask whether the effect belongs to latent planning or to LeWM in particular, we attach the same head, unchanged, to PLDM (Sobal et al., 2025), a model of a different family. Both arms load one trained PLDM checkpoint and differ only in the terminal cost. The separation reappears in the same form (Table A7, Section J): on Push-T held out, over three seeds, the head leads at every offset, its weakest seed beats the strongest Euclidean seed in every column, and the multiplier it supplies at the long offsets is close to the one on LeWM. Our PLDM is trained for a tenth of its recipe, so the absolute level is lower and the comparison is internal to one checkpoint rather than a claim about PLDM as published. The terminal cost, not the LeWM recipe, is what the effect attaches to.

4.4 Which Component Carries the Effect

By construction nothing but the terminal cost differs between the arms of Section 4.3; this subsection asks whether anything else could carry the effect, and what the parts of the head are worth.

The cost, not the representation The method changes only the cost; the joint form changes the representation too, which leaves one question open: does a shaped representation carry any of the

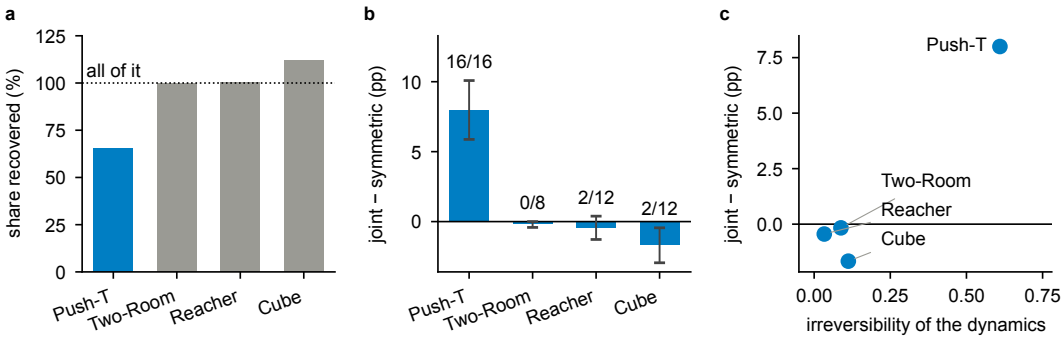

Figure 6: **What the asymmetry of the quasimetric is worth.** The ablation replaces $\mathrm{relu}(\psi(x) - \psi(y))$ by $|\psi(x) - \psi(y)|$, the symmetrization of the same head at identical parameters, supervision, and triangle inequality, so one thing changes. It is the one comparison reported on the jointly trained arm alone, since the symmetric variant is a retrained head and a frozen counterpart would need three more training runs; Table A11 asks the same question of the frozen arm without retraining by reading its head backwards. **(a)** What the symmetric arm recovers of the LeWM-to-GeoWM (joint) gain: all of it on three benchmarks. **(b)** The paired difference over the 150 problems behind each cell, with a bootstrap interval and the number of cells favoring GeoWM (joint); on Push-T no single cell resolves eight points, but all sixteen point the same way. **(c)** That difference against how irreversible the recorded dynamics of each benchmark are, measured from logged states without the encoder. Asymmetry pays on the one benchmark whose motion does not run backwards, one positive case out of four rather than a demonstrated mechanism (Section N).

effect on its own? Table A5 answers it with an arm that changes only that, replanning the trained joint checkpoint with its head disabled. The representation alone recovers -4% and $+6\%$ of the gain on Push-T and Reacher, where the shaped encoder tracks the baseline at every offset, and $+16\%$ and $+27\%$ on Cube and Two-Room (Section K). The split falls this way because the cost is what the search consumes: CEM ranks candidates by the terminal cost alone, so a representation in which reachability is legible does not help if the function reading it still measures appearance.

Not the rollout, and not the search The rollout is not what the cost replaces: cutting it to a single action block costs the head nearly half its success where the arms separate, 38.0 to 19.3 at offset 100, and the head leads at every rollout horizon once the goal is far (Section L). Nor does more search close the gap: three times the CEM iterations moves every cell by at most four points.

What joint training adds Fitting the head jointly with the encoder is the upper bound on what the terminal cost can deliver, and it is a modest one: over the held-out cells both arms cover it buys a median of 0.7 points for 40% more training time (Table A5, Section K). The gain is concentrated rather than spread. On Reacher and Cube the two forms are level at every offset, so there the frozen head has taken the entire gain; what joint training buys sits at the longest offsets of Push-T and Two-Room, where the rollout is longest and the baseline weakest, and on Push-T it pays for that at the protocol offset. Coupling the metric to the representation therefore helps only where a frozen encoder does not already support the metric. Nor does it degrade the representation: linear probes for physical quantities on all four benchmarks over three seeds move in favor of the joint form or stay level, and none moves against it (Section O). We report the joint form throughout as an upper bound, not as the configuration we recommend.

What the asymmetry is worth The head is a quasimetric, and Fig. 6 asks what its asymmetry contributes by symmetrizing the same head at identical parameters and supervision. The symmetric arm recovers the entire gain on Two-Room, Reacher, and Cube, and asymmetry is worth 8.0 points on Push-T, the one benchmark whose recorded motion does not run backwards (Section N). What repairs long-horizon planning is a cost trained to approximate temporal distance; whether that cost is a quasimetric or a metric changes the outcome on one benchmark of four.

Does the planning gain track the geometry gain? In direction only. Every benchmark whose geometry improves also plans better, but the magnitude does not scale: Cube receives the smallest repair we measure ($\Delta\rho = +0.05$) and gains at every offset, while Push-T receives a repair six times larger ($+0.31$) and the smallest gain at its longest offset (Fig. 3c). Rank correlation is an incomplete summary of a geometry: the Euclidean cost on Reacher is *larger* at the midpoint of a trajectory than at its start in all three seeds, a sign error no search can work around, and repairing it is not visible in $\Delta\rho$ at all. We claim the association and not a functional correspondence.

Cheaper explanations, closed Six alternatives would account for the separation without temporal geometry, and none survives. Holding the execution budget at twice the goal distance throughout, rather than at offset $+\,50$, leaves the separation at every offset and moves its size by a median of 0.7 points (Table A15). Whitening the latent recovers -3% of the gap at offset 100, and accumulating the Euclidean stage cost along the rollout reaches 9.3 against a baseline 16.0, so it is the geometry rather than the norm or the decision to read it once (Table A9). Reading the head backwards, $d_\psi(z_g, \hat{z}_H)$ for $d_\psi(\hat{z}_H, z_g)$, returns it to the baseline, so the direction is used and not incidental (Table A11). And two learned costs fail where this one does not: an unrestricted MLP regressed on true step separations reaches 4.0, worse than Euclidean, and a head of matched capacity fitted by the *same* objective on the *same* pairs with only the metric structure removed fits that objective *better* and still plans at or below the baseline at every offset, because it collapses to a near-constant distance that gives the search nothing to rank (Table A10). The converse, that a metric fitted on encoder outputs fails on rollout latents as they drift off the encoder manifold, is what we expected and did not find; why the drift is benign is the open question this section leaves.

4.5 What the Evidence Supports, and How Far

The claim is narrow: within a fixed architecture, dataset, budget, and planner, replacing a Euclidean terminal cost by one approximating temporal distance converts a long-horizon collapse into stable performance, on benchmarks whose geometry is independently measured to be pathological. We claim no state of the art, since every arm is held at the LeWM recipe so that the geometry is the only variable, nor that quasimetric structure is novel (Wang et al., 2023); what is new is that the geometry of the terminal cost, rather than the rollout or the search, is the binding constraint at long horizons.

No single cell should be read to better than roughly ten points: a 50-episode evaluation carries a binomial standard error of 6–7 points, and re-evaluating one checkpoint on identical pairs while changing only the device moved a cell by 20, so three seeds and paired reading are what make cells usable. Because both arms of a cell solve identical problems, Table A2 reports the per-problem difference: the head leads in 21 of 24 seed-cell pairs, and the problems only the head solves rise from 12 to 49 with distance while those only the baseline solves stay flat (Sections F and G).

The method has boundaries. The gain is horizon-driven rather than an artifact of which episodes support a distant goal, since it survives a sweep on a cohort fixed at the longest offset, but on that cohort, restricted to the episodes that support the longest goals, the head is 10.7 to 11.3 points behind at the protocol offset (Section Q). Repairing the goal metric addresses only the metric: on a substantially harder image-based manipulation suite, planning failed for reasons unrelated to geometry and no metric intervention moved the outcome (Section R). And Cube marks a task ceiling, with both arms near failure by offset 175.

5 Conclusion

Planning in a latent WM is distance minimization, so the planning geometry is what planning optimizes. We showed that the Euclidean geometry supplied by a prediction-trained encoder fails at long horizons, diagnosed the failure by measuring the geometry directly, and repaired it without retraining the WM, by fitting a small head to the frozen encoder under local temporal constraints and reading it as the terminal cost. The repair produces the predicted separation on all four benchmarks, holds on episodes no model trained on, and transfers unchanged to a WM of another family.

The effect lives in the cost: a reshaped representation read by the Euclidean cost recovers little, coupling the head to the encoder buys a median of 0.7 points for 40% more training, and an unstructured head fitted by the same loss collapses to a near-constant distance and plans no better than the baseline. Two questions remain open: why a metric fitted on encoder outputs stays accurate on drifted rollout latents, and how far the repair reaches, since it cannot help where planning fails for reasons other than geometry (Section R). Within those bounds the lesson holds: in long-horizon latent planning the planning geometry, not the search or the rollout machinery around it, is the lever. Plan by reachability, not resemblance.

AI USE STATEMENT

Large language models (LLMs) were used during this project as coding and writing assistants, including to help draft and refactor experiment and analysis scripts, revise prose, and check the consistency of the manuscript with its tables and figures. The empirical results reported in this paper are based on experiment runs conducted and reviewed by the authors, with tables and figures generated primarily from recorded experimental outputs. Citations and LLM-assisted content were reviewed by the authors. The research questions, experimental design, evaluation choices, and interpretation of the results were determined by the authors, who take responsibility for the final content of the paper.

ETHICS STATEMENT

All authors have read and adhere to the ICLR Code of Ethics. This work trains and evaluates goal-reaching controllers in simulation only, on the four public benchmark archives released with Maes et al. (2026), and involves no human subjects, no personal data, and no deployment on physical hardware. The contribution is a planning cost, a small head fitted to a frozen encoder, and it carries no application-specific risk beyond that already present in simulated robot learning. We state our deviations from the published protocol in Section E and report the failure cases and the boundaries of the claim in Section 4.5 and Section R, including a setting in which the method does not help, so that the scope is not overstated.

REPRODUCIBILITY STATEMENT

All experiments use the four public benchmark archives released with Maes et al. (2026), and every LeWM arm follows the published training recipe. Section B specifies the architecture, optimizer, planner, compute, and every hyperparameter of the added head; Section D specifies the datasets and their verified sizes; Section E specifies the split and evaluation seeds and the audit confirming that paired arms see identical training data and identical evaluation problems; and Section F compares our reproduction of the baseline with the published numbers. The method adds one small head to a trained checkpoint and changes only the terminal cost of the planner, leaving the WM and the search unchanged. All source code, including the training and evaluation scripts and the scripts that regenerate the tables and figures from the logged runs, will be released upon acceptance.

REFERENCES

Jiaxin Bai and Jiaxuan Xiong. Temporal-distance JEPA: Plan-aware representation learning for latent world model predictive control. *arXiv preprint arXiv:2607.25337*, 2026. 3, A1

Kenneth James Williams Craik. *The Nature of Explanation*. Cambridge University Press, Cambridge, UK, 1943. 1

Matthieu Destrade, Oumayma Bounou, Quentin Le Lidec, Jean Ponce, and Yann LeCun. Value-guided action planning with JEPA world models. *arXiv preprint arXiv:2601.00844*, 2026. 3, A1

Benjamin Eysenbach, Ruslan Salakhutdinov, and Sergey Levine. Search on the replay buffer: Bridging planning and reinforcement learning. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2019. 2, A1

Yuntian Gao and Xiangyu Xu. Fast LeWorldModel. *arXiv preprint arXiv:2606.26217*, 2026. 2

Dibya Ghosh, Abhishek Gupta, Ashwin Reddy, Justin Fu, Coline Manon Devin, Benjamin Eysenbach, and Sergey Levine. Learning to reach goals via iterated supervised learning. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2021. 2

Christopher Grimm, André Barreto, Satinder Singh, and David Silver. The value equivalence principle for model-based reinforcement learning. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 3, A1

David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2018. 1

Nicklas Hansen, Xiaolong Wang, and Hao Su. Temporal difference learning for model predictive control. In *Proceedings of International Conference on Machine Learning (ICML)*, 2022. 2, A1

Nicklas Hansen, Hao Su, and Xiaolong Wang. TD-MPC2: Scalable, robust world models for continuous control. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2024. 2, A1

Nathan O. Lambert, Brandon Amos, Omry Yadan, and Roberto Calandra. Objective mismatch in model-based reinforcement learning. In *Learning for Dynamics and Control (L4DC)*, 2020. 3, A1

Yann LeCun. A path towards autonomous machine intelligence. *OpenReview*, 2022. Version 0.9.2. 1

Liangyu Li, Shengzhi Wang, Libin Qiu, Mingliang Xiong, and Qingwen Liu. World model control by trajectory reachability metrics. *arXiv preprint arXiv:2605.22164*, 2026a. 3, A1

Wenyuan Li, Guang Li, Keisuke Maeda, Takahiro Ogawa, and Miki Haseyama. Predictive but not plannable: RC-aux for latent world models. *arXiv preprint arXiv:2605.07278*, 2026b. 3, A1

Bo Liu, Yihao Feng, Qiang Liu, and Peter Stone. Metric residual network for sample efficient goal-conditioned reinforcement learning. In *Proceedings of AAAI Conference on Artificial Intelligence (AAAI)*, 2023. 4

Lucas Maes, Quentin Le Lidec, Damien Scieur, Yann LeCun, and Randall Balestriero. LeWorldModel: Stable end-to-end joint-embedding predictive architecture from pixels. *arXiv preprint arXiv:2603.19312*, 2026. 1, 2, 5, 10, A3, A4, A15

Vivek Myers, Bill Chunyuan Zheng, Benjamin Eysenbach, and Sergey Levine. Offline goal-conditioned reinforcement learning with quasimetric representations. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2025. 3, A1, A17

Seohong Park, Dibya Ghosh, Benjamin Eysenbach, and Sergey Levine. HIQL: Offline goal-conditioned RL with latent states as actions. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2023. 3, A1

Reuven Y. Rubinstein and Dirk P. Kroese. *The Cross-Entropy Method: A Unified Approach to Combinatorial Optimization, Monte-Carlo Simulation, and Machine Learning*. Springer, New York, NY, 2004. 1, 2

Vlad Sobal, Wancong Zhang, Kyunghyun Cho, Randall Balestriero, Tim G. J. Rudner, and Yann LeCun. Learning from reward-free offline data: A case for planning with latent dynamics models. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2025. 2, 7

Lucas Thil, Jesse Read, Rim Kaddah, and Guillaume Doquet. Subspace-decomposed JEPAs: Disentangling progression and content in latent world models. *arXiv preprint arXiv:2605.31111*, 2026. 3, A1

Edward C. Tolman. Cognitive maps in rats and men. *Psychological Review*, 55(4):189–208, 1948. 2

Tongzhou Wang, Antonio Torralba, Phillip Isola, and Amy Zhang. Optimal goal-reaching reinforcement learning via quasimetric learning. In *Proceedings of International Conference on Machine Learning (ICML)*, 2023. 2, 3, 4, 9, A1, A2, A17

Ying Wang, Oumayma Bounou, Yann LeCun, and Mengye Ren. AdaJEPA: An adaptive latent world model. *arXiv preprint arXiv:2606.32026*, 2026. 2

Wancong Zhang, Basile Terver, Artem Zholus, Soham Chitnis, Harsh Sutaria, Mido Assran, Randall Balestriero, Amir Bar, Adrien Bardes, Yann LeCun, and Nicolas Ballas. Hierarchical planning with latent world models. *arXiv preprint arXiv:2604.03208*, 2026. 2

Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. DINO-WM: World models on pre-trained visual features enable zero-shot planning. In *Proceedings of International Conference on Machine Learning (ICML)*, 2025. 2, A3

Algorithm 1 GeoWM (frozen). The head is fitted to a trained WM that receives no gradient, and then read as the terminal cost of an otherwise unchanged planner. The joint variant differs in the one marked line.

Training. for each batch of sub-trajectories $(o_{1:T}, a_{1:T})$
  $z_t \leftarrow f_\theta(o_t)$; $\hat{z}_{t+1} \leftarrow g_\phi(z_t, a_t)$
  $\mathcal{L}_{\text{pred}} \leftarrow \|\hat{z}_{t+1} - \text{sg}(z_{t+1})\|_2^2$; $\mathcal{L}_{\text{reg}} \leftarrow \text{SIGReg}(Z)$
  $\mathcal{L}_{\text{qm}} \leftarrow \mathbb{E}\, e^{-d_\psi(z, z_{\text{perm}})} + \mu\, \mathbb{E}\, \text{relu}(d_\psi(z_t, z_{t+1}) - 1)^2$ ▷ random-pair push; consecutive-pair unit
  update ψ only on $\mathcal{L}_{\text{qm}}$; θ, ϕ are the trained baseline and receive no gradient ▷ joint variant: update θ, ϕ, ψ on $\mathcal{L}_{\text{pred}} + \lambda\mathcal{L}_{\text{reg}} + \beta\mathcal{L}_{\text{qm}}$
Planning. given o_1, goal o_g: $z_1 \leftarrow f_\theta(o_1)$, $z_g \leftarrow f_\theta(o_g)$
  repeat until budget exhausted
    $a^\star_{1:H} \leftarrow \text{CEM}\big(\arg\min_{a_{1:H}} d_\psi(\hat{z}_H, z_g)\big)$ ▷ cost is the only change
    execute $a^\star_{1:K}$, re-observe, re-encode

A Extended Related and Concurrent Work

Section 2 condenses the account given here. The framing we adopt is that of objective mismatch (Lambert et al., 2020) and value-equivalent model learning (Grimm et al., 2020): a model should be correct about the quantity used for decisions, which for goal reaching is a distance to the goal. Learned planning costs have precedents outside latent world model (WM) planning, as a bootstrapped terminal value in TD-MPC (Hansen et al., 2022; 2024), as a goal-conditioned value read as a distance for graph search in SoRB (Eysenbach et al., 2019), and as temporal-distance representations descended by a policy (Park et al., 2023; Wang et al., 2023; Myers et al., 2025); all learn a value, and none rolls out a WM at test time.

Concurrent work Five recent preprints add temporal or reachability structure to models of this family, most on the same backbone. Destrade et al. (2026) plan against a quasi-distance that approximates a learned value, Li et al. (2026b) add budget-conditioned reachability supervision, and Thil et al. (2026) reserve a progression subspace of the latent. TRM (Li et al., 2026a) keeps the WM fixed as we do, fitting a reachability head to a frozen model and reading it as the terminal cost, but regresses $|t_i - t_j|$ with an explicitly symmetric head, where ours is constrained rather than regressed and asymmetric; we report no number for their design. The difference is not cosmetic: our head never sees a temporal label beyond three action blocks, and a head of matched capacity fitted by the same local objective without the metric structure collapses to a constant distance (Section M). Closest is TD-JEPA (Bai & Xiong, 2026), which fits a quasimetric head jointly with the model, evaluates at the protocol offset alone, finds the temporal cost *losing* there on Push-T, and reverts to Euclidean, exactly the regime in which Section 3.2 predicts the metric is not binding and in which a single offset cannot see what Section 4.3 measures. None of the five sweeps the goal distance, so none reports the separation or the attribution between cost, representation, and search.

B Implementation Details

Algorithm 1 states both steps of the method. The remainder of this section gives every architectural and optimization detail; all settings are shared by all arms.

Encoder and predictor The encoder is a Vision Transformer (ViT)-Tiny with patch size 14, 12 layers, 3 attention heads, and hidden dimension 192, roughly 5M parameters. The observation embedding is the class-token embedding of the last layer followed by a one-layer projection with batch normalization, which is necessary because the final ViT layer applies layer normalization and would otherwise render the anti-collapse objective vacuous. The predictor is a ViT-S backbone with 6 layers, 16 attention heads, and dropout 0.1, roughly 10M parameters, with learned positional embeddings and causal masking over the observation history; actions enter through adaptive layer normalization whose parameters are initialized to zero. Images are 224×224.

Optimization The batch size is 128 over sub-trajectories of 4 frames with 4 action blocks of 5 primitive actions each, at frame-skip 5, for 10 epochs over the training split, matching the epoch budget of the published recipe. The Gaussian-embedding regularizer uses 1024 random projections with weight $\lambda = 0.1$.

The quasimetric head The head is a metric residual network (MRN) on the 192-dimensional latent: a two-layer projection with GELU activation to 256 dimensions, reshaped into $K = 16$ groups of $M = 16$ components and combined as in Eq. (3). The local-term weight is $\mu = 5$ and the quasimetric weight $\beta = 1$ in every reported result, so all numbers in the paper come from a single configuration. Random pairs for the push term are formed by a random permutation within the flattened batch, which costs one permutation and no additional forward passes. The head adds 115,200 parameters and no test-time cost beyond one projection of the terminal and goal latents.

Planner Cross-Entropy Method (CEM) uses 300 candidate action sequences, 30 optimization iterations on Push-T and 10 on the other environments, an elite set of the top 30 trajectories, initial sampling variance 1, and a rollout horizon H of 5 action blocks, corresponding to 25 environment steps at frame-skip 5. The receding-horizon schedule executes the entire optimized sequence before replanning.

Compute All training runs use a single H100 GPU. The added term is not free at training time: on Push-T the baseline takes 7 h 06 and Geometry for World Models (GeoWM) (joint) 9 h 57 for the same ten epochs, about 40% more, which is the cost of the extra head and the pairwise distances it computes on every batch; fitting the frozen head to an existing checkpoint takes 6.3 ± 0.2 hours. At planning time the head is one projection of the terminal and goal latents and its cost is negligible against the rollout, so the separation is bought with training compute and nothing else. Wall-clock cost is dominated by data loading rather than by the model: Two-Room and Push-T complete 10 epochs in a few hours, whereas Reacher, whose image dataset is the largest, requires approximately one hour per epoch.

C The Quasimetric Head and Its Objective

A quasimetric on a set X is a function $d : X \times X \to \mathbb{R}_{\geqslant 0}$ with $d(x, x) = 0$ and $d(x, z) \leqslant d(x, y) + d(y, z)$, without the requirement $d(x, y) = d(y, x)$. Asymmetry is essential for control: the number of actions needed to go from x to y is in general not the number needed to return, and irreversible transitions make the distinction stark. The construction in Eq. (3) satisfies both axioms for any choice of the projection ψ, because the group-wise maximum of the positive part of a difference is a quasimetric on each group and the average of quasimetrics is a quasimetric. The structural properties are therefore architectural rather than learned and hold at every point of training, which is what makes the head safe to fit jointly with a representation that is itself moving.

Why the two terms With only the local term of Eq. (4), the objective is minimized by the degenerate solution $d_\psi \equiv 0$, in which all states coincide. With only the push term, distances grow without bound and carry no relation to the transition structure. Together the terms pin the scale to one unit per environment transition and propagate that scale along trajectories. The object they aim at is the shortest-path quasimetric of the transition graph, whose negation is the optimal goal-reaching value (Wang et al., 2023). That identification is a theorem about the constrained program of Wang et al. (2023) and not about Eq. (4), which relaxes it in the two respects listed in Section 3. We do not know in which direction our fixed point deviates, and it need not be the same direction on every benchmark: the midpoint statistic of Table A3 is above the ideal 0.5 on Reacher, indicating a cost that stays flat far from the goal, and below it on Cube, indicating the opposite. This is why the geometry is reported as a measurement in Section 4.2 rather than argued from the objective.

Division of labor among the three terms The joint objective has three components acting on the same latent space, and none can stand in for another, because each constrains a different object. Only the prediction loss trains the dynamics: it is the sole action-conditioned term, and in our implementation the only term evaluated on the predictor output, since the quasimetric term is computed entirely on encoder embeddings. Its local component, $d_\psi(z_t, z_{t+1}) \leqslant 1$, superficially resembles a prediction constraint, but it is an inequality rather than a target, it lives in the projected space of the head rather than in the latent, and it carries no action argument, so it can state that consecutive states are close without stating where a particular action leads. Removing the prediction loss would leave the predictor at initialization while planning continues to roll out through it, a structural rather than empirical failure, which is why we do not report it as a benchmark row. Only the regularizer constrains the distribution: the push term separates the pairs drawn in a batch, whereas Sketched-Isotropic-Gaussian Regularizer (SIGReg) constrains the embedding distribution as a whole, a strictly stronger requirement.

D BENCHMARKS AND DATASETS

Every dataset is the archive released with Maes et al. (2026), re-downloaded, checked against its published byte length, and counted before use. Two-Room holds 10,000 episodes over 920,809 frames of mean length 92.1; Cube, 10,000 episodes over 2,010,000 frames of uniform length 201; Reacher, 10,000 episodes over 2,010,000 frames, again of length 201; and Push-T, 18,685 episodes over 2,336,736 frames of mean length 125.1, ranging from 49 to 246. The Push-T count disagrees with the published description, which gives 20,000 episodes of average length 196 and attributes the dataset to Zhou et al. (2025); the originating paper lists Push-T at 18,500 trajectories of length 100 to 300 and lists 20,000 against a different environment in the same table, so the archive is the correct dataset and the later figure was carried across environments. We report the archive we train on.

Push-T is a 2D manipulation task in which an agent represented as a disc must push a T-shaped block into a target configuration, with interaction restricted to pushing (Zhou et al., 2025). Two-Room is a 2D navigation task in which an agent must cross from one room to another through a single door, with trajectories generated by a noisy heuristic policy that steers toward the door and then toward the target. OGBench-Cube is a 3D manipulation task in which an arm must pick up a cube and place it at a target location; we use the single-cube variant, whose archive matches the published 10,000 episodes of 200 steps up to an off-by-one in length. Reacher is a DeepMind Control task in which a two-joint arm must reach a target configuration, with success defined by alignment of the joints with the target; its trajectories are collected by a trained policy, and its archive carries the same off-by-one as Cube. In all four cases we train for 10 epochs on the 90% episode-level training split, matching the epoch budget of the published recipe but not its data volume: the released configuration splits at clip granularity and therefore exposes clips from every episode, whereas we withhold whole episodes so that a genuinely unseen evaluation pool exists (Section E). This deviation costs us data and can only lower our numbers. Pixel observations are the only model input.

E EVALUATION PROTOCOL AND FAIRNESS AUDIT

Protocol Following LeWorldModel (LeWM), a start state is sampled uniformly from the valid start positions of dataset trajectories, and the goal is the state occurring a fixed offset later in the *same* trajectory, which guarantees that the goal is reachable and consistent with the data-generating dynamics. Success is the environment-specific predicate at any point within the budget, and every cell uses 50 evaluation episodes. The two protocols we report differ only in which trajectories those pairs are drawn from; the model is the same in both, trained once on the 90% split. In-distribution sampling draws from every episode, so most pairs come from trajectories the model trained on, which is the setting Maes et al. (2026) report; the held-out protocol restricts sampling to the episodes withheld from that same split, recomputed from the identical fraction and seed, so no pair comes from a trajectory the model has seen. A number is therefore never a comparison between two differently trained models, and the two protocols are not comparable to each other; only the two arms within a protocol are.

Splits Every result uses an episode-level split with seed 3072 and a training fraction of 0.9, derived from a single deterministic function of the episode count, the fraction, and the seed. Two arms trained with the same configuration therefore receive identical training and held-out episodes. We verified that no arm overrides the seed and that the permutation depends only on the episode count and the seed, so a smaller training fraction yields a strict superset of the held-out episodes rather than a different set.

Evaluation pairs The evaluation sampler draws its $(\text{start}, \text{goal})$ pairs from a generator seeded at 42 in the evaluation configuration, shared by all arms of a benchmark. Paired arms therefore face the identical 50 problems, and both arms are checked at run time to have loaded the intended objective, so the comparison is paired rather than merely distributional.

Deviations from the published protocol Three, stated plainly. First, the split is at episode rather than clip granularity (Section 4.1); this is deliberate, and it means our models see 9000 of the 10,000 Two-Room episodes where the released configuration would expose clips from all of them. Second, the goal-offset sweep raises the budget to offset + 50 beyond the protocol offset, since a longer offset needs a larger budget for the goal to be reachable at all; the offset-25 column keeps the published budget of 50 so that it remains directly comparable to Table A1, and both arms always share

Table A1: **Published baselines at the LeWM protocol (offset 25, budget 50), with our arms added.** Published rows are read from the comparison of Maes et al. (2026); dashes are benchmarks a method was not run on. Δ is our LeWM against the published one, and the deviations are mixed in sign, which is reproduction noise rather than a systematically weakened baseline. Only Push-T has a published variance to compare against, and we land inside it; the other three are read from an across-environment comparison that states none, so those deviations are approximate. The random floor is far from zero on Cube.

	Push-T	Two-Room	Reacher	Cube
Random (published)	2	0	10	48
Random (ours)	5.3	28.0	14.7	45.3
GCIQL	20	100	–	64
GCIVL	33	100	–	56
GCBC	75	100	–	84
DINO-WM	74	100	79	**86**
DINO-WM + proprioception	92	100	–	–
PLDM	78	97	78	65
LeWM (published)	96.0 ± 2.83	87	86	74
LeWM (our reproduction)	94.0 ± 4.0	96.7 ± 1.2	78.7 ± 1.2	74.7 ± 2.3
Δ	-2.0	$+9.7$	-7.3	$+0.7$
GeoWM (frozen)	97.3 ± 2.3	$\mathbf{100.0 \pm 0.0}$	80.7 ± 3.1	82.0 ± 5.3
GeoWM (joint)	88.0 ± 2.0	$\mathbf{100.0 \pm 0.0}$	80.0 ± 5.3	84.7 ± 1.2

whatever budget a cell uses. Third, an unresolved discrepancy: our Two-Room baseline reaches 96 where Maes et al. (2026) report 87. We tested the two candidates we could control and eliminated both: correcting the CEM iteration count to the per-environment value moves every cell by at most four points, inside the sampling error of a 50-episode evaluation, and retraining under the released clip-level split also lands at 96. What remains is the selection of evaluation trajectories, which the published description does not pin down precisely enough to match, and the error bar on the published figure, which we cannot read numerically. We report the gap rather than tuning it away; a baseline stronger than the published one is the direction that disfavors our own method.

F Reproduction, the Random Floor, and Evaluation Noise

Reproduction On Push-T our LeWM reaches 94.0 ± 4.0 against the published 96.0 ± 2.83; elsewhere it differs by at most ten points with the sign mixed, reproduction noise rather than a systematically weakened baseline (Table A1). Because the protocol matches, the baselines Maes et al. (2026) evaluate can be read alongside ours without re-running them. At offset 25 GeoWM (joint) matches the best published entry on Two-Room, sits a point below it on Cube and six to eight below on Reacher and Push-T, which is what Section 3.2 predicts, since that offset is exactly where the metric is not the binding constraint. The comparison is confined to offset 25, the only setting with published numbers.

The random floor A random action sequence is not hopeless at offset 25: over three runs it satisfies the success predicate 45.3% of the time on Cube, 28.0% on Two-Room, 14.7% on Reacher, and 5.3% on Push-T, and the published Cube figure of 48% sits inside our run-to-run spread, so that floor is a property of the benchmark rather than of our setup. Most of it is start states that already satisfy the predicate: the same policy restricted to five steps still scores 44% on Cube and 14% on Two-Room. Both effects fall away with horizon. At the longest offset of each benchmark the floor is 0% on Push-T and Two-Room and 7–8% on Reacher and Cube, with already-solved starts at 0% everywhere but Cube. The comparisons that carry our claim therefore sit on a floor of essentially zero: at offset 225 the baseline scores 5.3% against a random 0% while the head completes 13.3%.

How precisely a single cell can be read A 50-episode evaluation carries a binomial standard error of 6–7 points, and the planner adds more: re-evaluating one checkpoint on identical pairs, changing only the device, moved a single cell by 20 points, because CEM amplifies a floating-point difference through 30 iterations into a different selected sequence. Neither arm was favored and the separation did not move, but no single cell should be read to better than roughly ten points, and every benchmark is therefore evaluated on GPU. Three seeds are what make cells usable: single-seed readings twice pointed the opposite way from the three-seed mean, once by eleven points on

Table A2: **The paired analysis, run on the method.** Push-T, three seeds. The *only* columns count problems one arm solves and the other does not, summed over three seeds, and the last column marks cells where all three seeds point the same way. The head leads in 21 of 24 seed-cell pairs, three seeds agree in six of eight cells, and the problems only the head solves rise from 12 to 49 as the goal recedes while the problems only the baseline solves stay flat, so the head is not trading one set of solved problems for another.

Protocol	Offset	LeWM	GeoWM	paired Δ	only LeWM	only GeoWM	3/3
IN-DISTRIBUTION	25	94.0	97.3	**+3.3**	2	7	✓
	50	58.7	60.7	**+2.0**	18	21	–
	75	17.3	48.0	**+30.7**	9	55	✓
	100	16.0	38.0	**+22.0**	9	42	✓
HELD-OUT	25	88.7	92.0	**+3.3**	7	12	–
	50	51.3	66.0	**+14.7**	13	35	✓
	75	20.7	48.0	**+27.3**	7	48	✓
	100	11.3	41.3	**+30.0**	4	49	✓

a headline cell. Because both arms of a cell solve identical pairs, Section G reports per-problem differences, which are what a paired comparison rests on.

G Paired Intervals over Evaluation Problems

Three training seeds give three numbers per cell, and a standard deviation over three numbers is a thin summary of an evaluation whose own noise we measured directly (Section F): that noise lives in the planner, not in the training seed, so adding seeds is the wrong lever for it. The right lever is already in the logs. The evaluator records an outcome per evaluation episode, and paired arms are handed the identical (start, goal) pairs, so episode i under one arm is the same problem as episode i under the other; the quantity the paper claims is a per-problem advantage. For every cell we compute the mean paired difference over 150 problems (3 seeds $\times$ 50 episodes), with a 95% percentile bootstrap interval that resamples episodes within each seed and averages the three seed means, so the interval covers evaluation and planning stochasticity with the training seeds held fixed. The across-seed spread is a different quantity, and neither subsumes the other. We also count the McNemar pair, problems only the baseline solved and problems only the head solved, which is what a paired binary comparison rests on, since ties say nothing about which arm is better.

For the joint arm, 43 of the 48 cells favor GeoWM (joint) with an interval excluding zero. The five that do not are the ones the argument predicts: at offset 25 Push-T favors the baseline (-7.3, interval $[-13.3, -1.3]$) and Reacher is level under both protocols, the short-horizon end of the sweep rather than an absence of evidence. The interval also settles cells whose margins are too small to read from a mean alone: Cube held out at offset 175 is $+5.3$ with interval $[+1.3, +9.3]$ and McNemar counts of 1 against 9. Table A2 tabulates the same analysis for the method, whose per-episode outcomes sit inside the raw logs of the arm sweep. The two runs are separate executions, so pairing them is only valid if both drew the same problems, which the evaluator verifies by comparing the chosen episode indices before pairing a cell and which held in every cell; the paired runs agree with Table A5 within a point in half the cells and differ by up to six in the rest, the run-to-run spread of Section F.

H Measuring the Planning Geometry

The steps-to-go probe To measure geometry independently of task success, we sample held-out trajectory segments and, for each, compute the distance from every state in the segment to a designated later goal state under the geometry being probed, together with the true number of steps remaining. The statistic is the Spearman rank correlation between the two over all sampled pairs: planning consumes only the ordering of candidates and is invariant to any monotone rescaling of the cost, so rank correlation is the appropriate summary. As a secondary statistic we track the ratio of the distance at the midpoint of a segment to the distance at its start, which should be near one half for a well-behaved geometry and near one for a geometry that fails to register progress. Table A3 reports both for all four benchmarks under one protocol (60 held-out episodes per seed, goal the final frame, per-episode ρ then averaged).

What the probe can and cannot establish The method is the GeoWM (frozen) column of Table A3, whose encoder is the baseline encoder unchanged, so every difference from the LeWM

Table A3: **The planning geometry measured directly, all benchmarks.** Mean ± std over three seeds. ρ is the rank correlation between latent distance and true steps-to-go; $d(\text{mid})/d(0)$ is the cost halfway along a trajectory relative to its start, for which 0.5 is ideal and 1.0 means the cost does not register progress at all. The GeoWM (frozen) column is the method, whose encoder is the baseline encoder unchanged; bold marks it where it improves on the baseline.

	ρ (steps-to-go)			$d(\text{mid})/d(0)$		
Benchmark	LeWM	GeoWM (frozen)	GeoWM (joint)	LeWM	GeoWM (frozen)	GeoWM (joint)
Push-T	0.607 ± 0.034	$\mathbf{0.916 \pm 0.015}$	0.947 ± 0.008	0.966 ± 0.023	$\mathbf{0.699 \pm 0.036}$	0.577 ± 0.042
Two-Room	0.619 ± 0.020	$\mathbf{0.957 \pm 0.012}$	0.958 ± 0.010	0.850 ± 0.007	$\mathbf{0.600 \pm 0.022}$	0.594 ± 0.007
Reacher	0.374 ± 0.025	$\mathbf{0.616 \pm 0.034}$	0.614 ± 0.031	1.299 ± 0.398	$\mathbf{1.015 \pm 0.086}$	1.007 ± 0.077
Cube	0.561 ± 0.014	$\mathbf{0.613 \pm 0.010}$	0.629 ± 0.013	0.420 ± 0.015	0.265 ± 0.035	0.219 ± 0.015

column is the work of d_ψ alone; the joint form adds at most 0.03 to ρ on any benchmark and 0.001 on Two-Room. The head raises ρ on every benchmark, most on Push-T and Two-Room and least on Cube, and its across-seed spread is a third to a quarter of the baseline's. The two statistics do not agree about Cube: its midpoint ratio moves from 0.420 to 0.219, further from the ideal 0.5 than the baseline was, in the opposite direction to the Reacher baseline's 1.299. Neither deviation stops a planner, which consumes only the ordering, but the objective plainly does not drive every benchmark to the same fixed point and we do not claim it does. The label is steps along a *recorded* trajectory, and the local term of Eq. (4) propagates adjacency along exactly those trajectories, so a rise in ρ is close to what training this objective must produce, and the baseline receives no temporal supervision at all, which makes the comparison of the two ρ values weak evidence on its own. The ground-truth measurement below answers that circularity, and it covers Two-Room only, where the obstacle geometry is known well enough to compute exact geodesics; on the other three benchmarks ρ should be read as confirming that the objective did what it was written to do, not as independent evidence that reachability was learned.

Ground-truth reachability on Two-Room A behavior policy dawdles and detours, so k steps from s_t to s_{t+k} is an upper bound on how far apart they are, not the distance itself; a cost could agree perfectly with temporal progress through the data and still misrank states by reachability. Two-Room settles this without any trajectory in the loop. The environment is a plane divided by a vertical wall with a single gap, and the true agent position is recorded for every frame, so the geodesic between two points is exact in closed form: a straight line when both lie on the same side of the wall, and otherwise the shortest path through the door. We correlate each planning cost, squared Euclidean latent distance for LeWM and d_ψ for the two GeoWM arms, against that ground truth over the same held-out episodes the rest of the appendix uses; the wall geometry is read from the environment rather than estimated, and the door discretization is fine enough that the geodesic is exact to well under a pixel.

One number does not settle it on its own. For a goal that the wall barely separates from the states being ranked, the geodesic is nearly the straight-line distance, and a cost that has learned only distance scores as well as one that has learned reachability; such goals cannot distinguish the two hypotheses. We therefore use every goal in the held-out pool in turn, 900 in all, and report ρ stratified by the *detour ratio*, the mean of $d_{\text{geo}}/d_{\text{euclid}}$ over the states being ranked (Table A4). The bands are unequal by construction: the environment has one wall and one door, so most goals are barely separated from most states and only 195 goals bend a path by more than a tenth, and those are the ones that carry the result. In the two bands where the straight line still explains the geodesic, the head and the straight line agree and nothing is learned; where it stops explaining it, at 0.755 and then 0.430, the frozen head holds at 0.968 and 0.964, the joint form at 0.970 and 0.967, and LeWM sits near 0.3. The learned cost is therefore tracking reachability around the obstacle, not distance to the goal, measured against ground truth with no trajectory in the loop.

I Full Results

The main text reports these results in figures and prose; the tables give them in full. Table A5 is the complete sweep of all four arms on all four benchmarks, Table A6 gives the individual seeds behind its Push-T cells, and Fig. A1 shows what one cell of the separation looks like on the ground.

Per-seed results Every cell of Table A5 is a mean over three seeds. Table A6 gives the individual seeds behind the Push-T cells, the longest sweep and the one the mechanism sections rest on, so

Table A4: **Each cost against the exact geodesic on Two-Room.** Spearman ρ over 900 held-out goals, each ranking the 1380 held-out states, averaged within each stratum and stratified by how much the wall bends the path. The last column is the null this experiment exists to defeat, what the straight-line distance itself achieves and therefore what a cost that had learned only distance could score: it collapses from 1.000 to 0.430 as the detour grows, and the head does not move.

Stratum	detour ratio	goals	LeWM	GeoWM (frozen)	GeoWM (joint)	straight line
wall irrelevant	< 1.02	428	0.662	**0.988**	0.990	1.000
mild detour	1.02–1.10	277	0.374	**0.978**	0.981	0.968
real detour	1.10–1.30	187	0.294	**0.968**	0.970	0.755
long way round	$\geqslant 1.30$	8	0.322	**0.964**	0.967	0.430
all goals	–	900	0.494	**0.981**	0.983	0.934

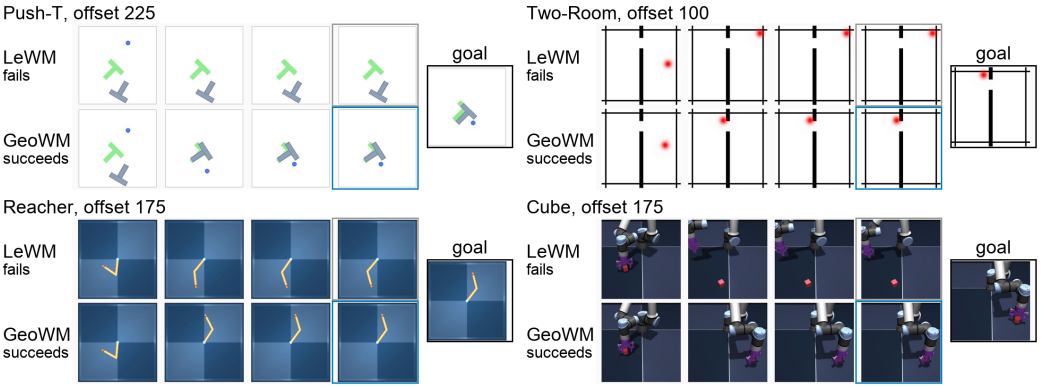

Figure A1: **Rollouts at the longest offset of each benchmark.** Each row shows four frames spanning the episode, with the goal in the last column; the upper row is LeWM and the lower row GeoWM. Both arms are given the identical start and goal, and every row is the first episode in which the Euclidean baseline fails and the head succeeds. The failures are not near-misses: on Two-Room the baseline agent (red) stalls against the wall instead of routing to the door, and on Reacher and Cube it settles into a configuration the cost scores as close but that is not the goal. Rollouts are from the seed-3072 checkpoint of each arm over 20 episodes and are illustrations rather than evidence, which Table A5 carries.

that a reader can check whether any comparison rests on a single favorable run rather than on the average; Section G gives the per-problem view. Each seed moves both the episode split and the initialization, so the spread covers split variance as well as optimization variance; within a seed the two arms share a split and are scored on the same $(\text{start}, \text{goal})$ pairs. The spread is widest at the longest offsets, where few pairs remain feasible: at offset 225 only about 2500 in-distribution pairs and about 1000 held-out pairs exist, and a single episode is worth two percentage points. The direction of every comparison beyond offset 50 is the same in all three seeds.

Rollouts at the longest offset Both arms in Fig. A1 are given the identical start and goal, so a row is a paired comparison and not a pair of runs that happened to differ. The success counts behind the figure, on the 20-episode sample of the seed-3072 checkpoints, are 9 against 12 on Two-Room, 11 against 20 on Reacher, 3 against 3 on Cube, and 1 against 3 on Push-T, so the Cube and Push-T strips are drawn from a handful of successes.

J The Second Backbone in Full

Table A7 gives the PLDM sweep of Section 4.3 beside the LeWM cells at the same offsets. Within each backbone the two columns are one checkpoint scored two ways, so each pair is an intervention on the terminal cost alone. On Push-T held out, over three seeds, the Euclidean cost scores 78.7, 38.7, 6.7, and 4.7 at offsets 25, 50, 75, and 100, against 88.7, 58.0, 27.3, and 12.7 for the same checkpoint scored by d_ψ; the arms do not overlap at any offset, the weakest of the three seeds under d_ψ beating the strongest Euclidean seed in every column. The multiplier the head supplies, 4.1 and 2.7 at offsets 75 and 100, is close to the 2.3 to 3.7 it supplies on the corresponding LeWM cells, while the absolute level is far lower, which is what a backbone trained for a tenth of its recipe predicts. The sweep stops at offset 100 because the Euclidean arm has reached the random-policy

Table A5: **Full results: all four arms on all four benchmarks at every offset of the sweep, three seeds.** Success rate (%); ID and HO are the in-distribution and held-out protocols; every entry is a three-seed mean $\pm$ the spread across seeds. GeoWM (frozen) fits the head to a frozen LeWM encoder of the same seed, so only the terminal cost differs from LeWM; shaped-enc. replans the trained GeoWM (joint) checkpoint with its head disabled, so only the cost reverts; dashes mark offsets at which that arm was not evaluated. The last two columns are what each arm recovers of the LeWM-to-GeoWM (joint) gap, shown where both diagnostic arms were evaluated and the gap exceeds three points in favor of GeoWM (joint). The cost alone recovers a median of 80 to 100%, the representation alone -4 to $+27$%.

Benchmark		Offset	LeWM	GeoWM (frozen)	shaped-enc.	GeoWM (joint)	rec. fr	rec. se
Push-T	ID	25	94.0 ± 4.0	97.3 ± 2.3	94.7 ± 1.2	88.0 ± 2.0	–	–
	ID	50	58.7 ± 6.1	60.7 ± 3.1	56.7 ± 4.2	63.3 ± 5.0	$+43\%$	-43%
	ID	75	17.3 ± 3.1	48.0 ± 0.0	21.3 ± 2.3	45.3 ± 5.0	$+110\%$	$+14\%$
	ID	100	16.0 ± 6.0	38.0 ± 7.2	14.0 ± 4.0	42.0 ± 2.0	$+85\%$	-8%
	ID	125	10.7 ± 1.2	26.7 ± 3.1	10.0 ± 4.0	36.0 ± 5.3	$+63\%$	-3%
	ID	150	5.3 ± 1.2	25.3 ± 6.1	8.0 ± 3.5	39.3 ± 1.2	$+59\%$	$+8\%$
	ID	175	4.0 ± 3.5	20.7 ± 6.1	–	32.0 ± 3.5	–	–
	ID	200	1.3 ± 1.2	22.7 ± 7.0	7.3 ± 2.3	24.0 ± 2.0	$+94\%$	$+26\%$
	ID	225	5.3 ± 2.3	13.3 ± 10.1	9.3 ± 3.1	20.7 ± 7.6	$+52\%$	$+26\%$
	HO	25	88.7 ± 2.3	91.3 ± 2.3	91.3 ± 3.1	89.3 ± 3.1	–	–
	HO	50	51.3 ± 3.1	66.0 ± 9.2	42.7 ± 2.3	66.7 ± 4.6	$+95\%$	-56%
	HO	75	20.7 ± 5.0	48.0 ± 7.2	19.3 ± 4.2	41.3 ± 1.2	$+133\%$	-7%
	HO	100	11.3 ± 1.2	41.3 ± 10.3	11.3 ± 4.2	46.7 ± 3.1	$+85\%$	$+0\%$
	HO	125	10.7 ± 2.3	34.0 ± 5.3	7.3 ± 3.1	36.7 ± 5.8	$+90\%$	-13%
	HO	150	9.3 ± 1.2	28.7 ± 5.0	8.0 ± 2.0	37.3 ± 11.0	$+69\%$	-5%
	HO	175	3.3 ± 2.3	20.7 ± 9.9	–	32.0 ± 10.6	–	–
	HO	200	4.7 ± 3.1	20.0 ± 3.5	3.3 ± 3.1	25.3 ± 4.2	$+74\%$	-7%
	HO	225	3.3 ± 3.1	9.3 ± 5.0	2.7 ± 2.3	18.7 ± 8.3	$+39\%$	-4%
Two-Room	ID	25	96.7 ± 1.2	100.0 ± 0.0	97.3 ± 1.2	100.0 ± 0.0	$+100\%$	$+18\%$
	ID	50	66.7 ± 3.1	100.0 ± 0.0	64.7 ± 4.2	100.0 ± 0.0	$+100\%$	-6%
	ID	75	64.0 ± 8.7	98.7 ± 1.2	73.3 ± 1.2	100.0 ± 0.0	$+96\%$	$+26\%$
	ID	100	38.0 ± 2.0	94.0 ± 2.0	54.7 ± 9.9	98.7 ± 1.2	$+92\%$	$+28\%$
	HO	25	96.7 ± 4.2	100.0 ± 0.0	95.3 ± 1.2	100.0 ± 0.0	$+100\%$	-42%
	HO	50	82.7 ± 4.6	100.0 ± 0.0	88.0 ± 4.0	100.0 ± 0.0	$+100\%$	$+31\%$
	HO	75	54.0 ± 11.1	95.3 ± 3.1	74.7 ± 3.1	100.0 ± 0.0	$+90\%$	$+45\%$
	HO	100	41.3 ± 8.1	85.3 ± 9.9	58.0 ± 5.3	98.7 ± 1.2	$+77\%$	$+29\%$
Reacher	ID	25	78.7 ± 1.2	80.7 ± 3.1	73.3 ± 4.2	80.0 ± 5.3	–	–
	ID	50	90.7 ± 3.1	96.7 ± 2.3	89.3 ± 3.1	97.3 ± 3.1	$+91\%$	-21%
	ID	75	82.7 ± 3.1	98.7 ± 2.3	–	98.7 ± 1.2	–	–
	ID	100	77.3 ± 1.2	99.3 ± 1.2	82.0 ± 2.0	100.0 ± 0.0	$+97\%$	$+21\%$
	ID	125	79.3 ± 1.2	96.0 ± 2.0	76.0 ± 2.0	95.3 ± 1.2	$+104\%$	-21%
	ID	150	65.3 ± 2.3	96.0 ± 0.0	66.7 ± 1.2	96.0 ± 0.0	$+100\%$	$+5\%$
	ID	175	68.0 ± 2.0	98.0 ± 0.0	65.3 ± 1.2	97.3 ± 1.2	$+102\%$	-9%
	HO	25	96.0 ± 2.0	95.3 ± 3.1	96.7 ± 1.2	95.3 ± 2.3	–	–
	HO	50	92.7 ± 5.0	98.7 ± 1.2	94.7 ± 3.1	98.7 ± 1.2	$+100\%$	$+33\%$
	HO	75	86.7 ± 7.6	98.7 ± 2.3	–	100.0 ± 0.0	–	–
	HO	100	82.0 ± 4.0	99.3 ± 1.2	77.3 ± 6.4	97.3 ± 3.1	$+113\%$	-31%
	HO	125	66.0 ± 2.0	96.7 ± 3.1	69.3 ± 2.3	96.7 ± 3.1	$+100\%$	$+11\%$
	HO	150	66.7 ± 3.1	98.0 ± 0.0	71.3 ± 5.0	98.0 ± 0.0	$+100\%$	$+15\%$
	HO	175	63.3 ± 8.1	96.7 ± 2.3	65.3 ± 3.1	96.7 ± 2.3	$+100\%$	$+6\%$
Cube	ID	25	74.7 ± 2.3	82.0 ± 5.3	81.3 ± 2.3	84.7 ± 1.2	$+73\%$	$+66\%$
	ID	50	56.0 ± 2.0	66.7 ± 1.2	62.0 ± 0.0	69.3 ± 1.2	$+80\%$	$+45\%$
	ID	75	67.3 ± 1.2	83.3 ± 2.3	–	86.7 ± 3.1	–	–
	ID	100	63.3 ± 1.2	72.0 ± 2.0	68.7 ± 2.3	78.0 ± 2.0	$+59\%$	$+37\%$
	ID	125	37.3 ± 5.0	56.7 ± 1.2	32.7 ± 1.2	56.0 ± 2.0	$+104\%$	-25%
	ID	150	21.3 ± 1.2	50.0 ± 2.0	20.0 ± 4.0	52.0 ± 2.0	$+93\%$	-4%
	ID	175	10.7 ± 1.2	20.7 ± 4.2	13.3 ± 1.2	20.7 ± 4.6	$+100\%$	$+26\%$
	HO	25	74.0 ± 0.0	84.0 ± 4.0	77.3 ± 5.8	86.0 ± 0.0	$+83\%$	$+27\%$
	HO	50	52.7 ± 6.4	65.3 ± 4.2	54.7 ± 9.5	64.0 ± 4.0	$+112\%$	$+18\%$
	HO	75	60.7 ± 2.3	80.7 ± 1.2	–	81.3 ± 3.1	–	–
	HO	100	59.3 ± 3.1	70.0 ± 2.0	60.0 ± 5.3	72.7 ± 3.1	$+80\%$	$+5\%$
	HO	125	37.3 ± 9.9	55.3 ± 2.3	40.7 ± 1.2	60.7 ± 2.3	$+77\%$	$+15\%$
	HO	150	16.0 ± 2.0	46.7 ± 4.2	20.7 ± 3.1	46.7 ± 3.1	$+100\%$	$+15\%$
	HO	175	9.3 ± 2.3	10.7 ± 5.0	6.0 ± 4.0	13.3 ± 3.1	$+35\%$	-83%

Table A6: **Per-seed Push-T results behind the averaged rows of Table A5.** Success rate (%) at 50 evaluation episodes per cell, for the method and for both arms it is compared against. Seeds 1 and 2 are replicas of seed 3072 trained under the identical recipe.

Protocol	Arm	Offset	seed 3072	seed 1	seed 2	mean ± std
IN-DISTRIBUTION	LeWM	25	98	90	94	94.0 ± 4.0
		50	52	64	60	58.7 ± 6.1
		75	20	14	18	17.3 ± 3.1
		100	10	22	16	16.0 ± 6.0
		125	12	10	10	10.7 ± 1.2
		150	6	4	6	5.3 ± 1.2
		175	2	8	2	4.0 ± 3.5
		200	2	2	0	1.3 ± 1.2
		225	8	4	4	5.3 ± 2.3
	GeoWM (frozen)	25	100	96	96	97.3 ± 2.3
		50	60	64	58	60.7 ± 3.1
		75	48	48	48	48.0 ± 0.0
		100	32	46	36	38.0 ± 7.2
		125	26	24	30	26.7 ± 3.1
		150	24	32	20	25.3 ± 6.1
		175	22	26	14	20.7 ± 6.1
		200	30	16	22	22.7 ± 7.0
		225	12	4	24	13.3 ± 10.1
	GeoWM (joint)	25	86	90	88	88.0 ± 2.0
		50	64	68	58	63.3 ± 5.0
		75	40	50	46	45.3 ± 5.0
		100	42	40	44	42.0 ± 2.0
		125	32	42	34	36.0 ± 5.3
		150	38	40	40	39.3 ± 1.2
		175	28	34	34	32.0 ± 3.5
		200	22	24	26	24.0 ± 2.0
		225	24	12	26	20.7 ± 7.6
HELD-OUT	LeWM	25	90	90	86	88.7 ± 2.3
		50	52	48	54	51.3 ± 3.1
		75	26	20	16	20.7 ± 5.0
		100	12	10	12	11.3 ± 1.2
		125	12	12	8	10.7 ± 2.3
		150	8	10	10	9.3 ± 1.2
		175	6	2	2	3.3 ± 2.3
		200	4	8	2	4.7 ± 3.1
		225	4	0	6	3.3 ± 3.1
	GeoWM (frozen)	25	90	90	94	91.3 ± 2.3
		50	74	56	68	66.0 ± 9.2
		75	50	54	40	48.0 ± 7.2
		100	50	30	44	41.3 ± 10.3
		125	40	32	30	34.0 ± 5.3
		150	24	34	28	28.7 ± 5.0
		175	14	32	16	20.7 ± 9.9
		200	24	18	18	20.0 ± 3.5
		225	14	4	10	9.3 ± 5.0
	GeoWM (joint)	25	86	90	92	89.3 ± 3.1
		50	72	64	64	66.7 ± 4.6
		75	42	42	40	41.3 ± 1.2
		100	46	44	50	46.7 ± 3.1
		125	30	40	40	36.7 ± 5.8
		150	26	38	48	37.3 ± 11.0
		175	24	44	28	32.0 ± 10.6
		200	24	30	22	25.3 ± 4.2
		225	16	12	28	18.7 ± 8.3

Table A7: **The same head on a second WM family, against the same head on the first.** Push-T held out, three seeds. Within each backbone the two columns are one checkpoint scored two ways, so the pair is an intervention on the cost and not a comparison of models; across backbones only the multiplier is comparable, because this PLDM is trained for ten epochs against the hundred its own recipe specifies and its absolute scores sit correspondingly lower. The sweep stops at offset 100, by which point the Euclidean arm on PLDM has reached the random-policy floor of 5.3%.

	LeWM backbone			PLDM backbone		
Offset	Euclidean	GeoWM (frozen)	ratio	Euclidean	GeoWM (frozen)	ratio
25	88.7 ± 2.3	91.3 ± 2.3	$1.0\times$	78.7 ± 3.1	$\mathbf{88.7 \pm 2.3}$	$\mathbf{1.1}\times$
50	51.3 ± 3.1	66.0 ± 9.2	$1.3\times$	38.7 ± 4.2	$\mathbf{58.0 \pm 4.0}$	$\mathbf{1.5}\times$
75	20.7 ± 5.0	48.0 ± 7.2	$2.3\times$	6.7 ± 1.2	$\mathbf{27.3 \pm 6.1}$	$\mathbf{4.1}\times$
100	11.3 ± 1.2	41.3 ± 10.3	$3.7\times$	4.7 ± 1.2	$\mathbf{12.7 \pm 1.2}$	$\mathbf{2.7}\times$

floor of 5.3% there, scoring 4.7 while the head holds 12.7 on the same checkpoint, so the separation is not an artifact of both arms having failed.

K Both Mechanism Arms on All Four Benchmarks

GeoWM (joint) changes two things at once, so Table A5 changes them one at a time, and neither diagnostic arm requires training. GeoWM (frozen) fits the same quasimetric head, under the same loss, to a frozen LeWM encoder of the same seed (115,200 trainable parameters against 18,034,478), so representation and dynamics are bit-identical to the baseline and only the cost differs; the freeze was verified by replanning with the head disabled and reproducing the LeWM row cell for cell. Shaped-enc. replans the trained GeoWM (joint) checkpoint with the Euclidean cost forced back in, holding the representation fixed and reverting only the cost.

What each recovers is reported as a fraction of the LeWM-to-GeoWM (joint) gap, because that gap spans an order of magnitude across cells, and only where the gap exceeds three points, since below that the endpoints are level and the fraction divides one noise by another; where GeoWM (joint) is the weaker endpoint there is no gain to recover and no fraction is shown. Every entry is a three-seed mean, and a spread of zero means the arm solved the same number of problems in all three runs. Each benchmark is swept to the largest offset its trajectories admit, determined by counting valid (start, goal) pairs rather than assumed: Push-T to 225 (episodes reach 246 steps), Reacher and Cube to 175 (fixed 201-step episodes leave no valid start at 200), and Two-Room to 100 (episodes are at most 101 steps).

The cost, not the representation The dissociation is one-sided on all four benchmarks, and by a wide margin. Changing the cost alone recovers a median of 80% on Push-T, 82% on Cube, 98% on Two-Room, and 100% on Reacher; over the longest Push-T offsets, 125 to 225, the frozen head recovers a median of 67% while the shaped encoder recovers 0%. Changing the representation alone recovers -4% and $+6\%$ on Push-T and Reacher, where the shaped representation, read by the wrong cost, plans like the baseline, and $+16\%$ and $+27\%$ on Cube and Two-Room, so on those two a minority of the gain does survive in the representation. Cube is the one that most needed asking, since its geometry repair is the smallest we measure yet it improves at every offset, and the answer is that its improvement is not purely a cost effect. The split falls this way because the cost is what the search consumes: CEM ranks candidates by the terminal cost alone, so a representation in which reachability is legible does not help if the function reading it still measures appearance.

What joint training adds Read column against column, the table also measures what joint training adds over the frozen head. Over the 27 held-out cells both arms cover, GeoWM (joint) leads GeoWM (frozen) by a median of 0.7 points, for the 40% more Push-T training time recorded in Section B. The difference is concentrated: on Reacher the two are identical in five of seven cells and within two points elsewhere, on Cube the largest gap is 5.4 points, and the largest gaps sit at the longest offsets of the other two benchmarks, 5.3 to 11.3 points on Push-T from offset 150 and 13.4 on Two-Room at offset 100.

Table A8: **Success against planning horizon on Push-T, in distribution, three seeds.** H is in action blocks of five primitive actions, and $H = 5$, the protocol of this paper, is set in bold. The whole optimized sequence is executed before replanning, so the replan frequency co-varies with H: at $H = 1$ the planner replans five times as often as at $H = 5$. That direction strengthens the reading in the text, since $H = 1$ gets more replanning and still loses, and weakens the $H = 10$ column, where replanning is sparsest.

Offset	Arm	$H = 1$	$H = 2$	$H = 5$	$H = 10$
25	LeWM	58.7 ± 1.2	79.3 ± 1.2	$\mathbf{94.7 \pm 3.1}$	45.3 ± 7.0
	GeoWM (frozen)	72.7 ± 7.6	83.3 ± 1.2	$\mathbf{96.0 \pm 3.5}$	40.7 ± 11.0
	GeoWM (joint)	70.0 ± 3.5	78.7 ± 3.1	$\mathbf{88.0 \pm 2.0}$	40.7 ± 3.1
100	LeWM	2.0 ± 2.0	2.7 ± 3.1	$\mathbf{16.7 \pm 4.2}$	13.3 ± 4.2
	GeoWM (frozen)	19.3 ± 6.1	16.7 ± 6.1	$\mathbf{38.0 \pm 5.3}$	29.3 ± 4.2
	GeoWM (joint)	21.3 ± 10.3	28.7 ± 6.1	$\mathbf{44.0 \pm 4.0}$	28.7 ± 1.2
225	LeWM	1.3 ± 1.2	4.0 ± 0.0	$\mathbf{4.7 \pm 1.2}$	2.7 ± 1.2
	GeoWM (frozen)	6.7 ± 6.4	6.0 ± 2.0	$\mathbf{9.3 \pm 5.8}$	9.3 ± 4.6
	GeoWM (joint)	8.0 ± 3.5	10.0 ± 3.5	$\mathbf{20.7 \pm 9.0}$	10.7 ± 3.1

L What the Rollout Still Contributes

The paper argues that the terminal cost, rather than the rollout or the search, is what binds at long horizons. Read carelessly that invites a stronger reading than we intend: that once d_ψ exists the WM is decoration, and the same result would follow from descending a learned temporal distance one step at a time. That reading is testable. At a rollout horizon of one action block the planner scores a single block against the goal and executes it, which is greedy descent on the metric with essentially no rollout. Table A8 gives the sweep; it is a separate execution of the protocol, so its $H = 5$ column reproduces Table A5 to within two points rather than exactly.

The rollout is not decoration At the offsets where the two arms separate, GeoWM (joint) at $H = 1$ reaches less than half of what it reaches at $H = 5$: 21.3 against 44.0 at offset 100, and 8.0 against 20.7 at offset 225. The metric and the rollout are complementary, not substitutes, and the confound runs the helpful way: $H = 1$ replans five times as often and still loses by half.

The gain is not bought by shortening the rollout GeoWM (joint) leads at every horizon, by 19.3, 26.0, 27.3, and 15.3 points at offset 100 for H of 1, 2, 5, and 10. There is no horizon at which the Euclidean cost catches up, so the effect reported in Section 4.3 is not an artifact of the particular H the protocol fixes.

Both arms fall at $H = 10$ Offset 25 drops from 94.7 to 45.3 for the baseline and from 88.0 to 40.7 for GeoWM (joint). Rolling ten blocks out compounds prediction error faster than the extra lookahead pays for, and it does so to both arms by almost the same factor, which places the drift in the model rather than in either cost. Sparser replanning contributes as well, and the two cannot be separated in this sweep.

Where the search is weakest, the cost matters most At offset 25 and $H = 5$ the baseline is the better arm, 94.7 against 88.0, which is the short-horizon end of the sweep. At the same offset and $H = 1$ the ordering reverses: 58.7 against 70.0. Cutting the horizon does to the search at a near goal what distance does to it at a far one, and the cost becomes the binding constraint in both cases. That is the same mechanism the paper argues for, arrived at by starving the search instead of by moving the goal.

M Alternative Terminal Costs

Replacing a squared Euclidean norm with d_ψ changes several things at once, and only one of them is the temporal supervision the paper argues for. This section rules out the cheaper explanations one at a time: three alternative costs (Table A9), an unstructured head at matched capacity fitted by the very same objective (Table A10), and two other ways of reading the fitted head at planning time (Table A11). All are on Push-T over three seeds, with the planner, the budget rule, and the evaluation pairs unchanged.

Not the conditioning of the latent The LeWM cost is a squared Euclidean norm on an unnormalized ViT-Tiny class token passed through a batch-normalized projection, and nothing in the objective makes its covariance isotropic: SIGReg constrains the marginal distribution, not the sec-

Table A9: **Three alternative terminal costs against both endpoints.** Push-T in distribution, three seeds. The first two are evaluation-only on the *baseline* checkpoint; the third is a separately trained arm. "recovered" is the fraction of the LeWM-to-GeoWM (frozen) gap the alternative reaches, shown only where that gap is positive, larger than three points, and larger than the two endpoints' combined across-seed spread; at offset 25 the gap is 3.3 against a spread of 4.6, at 50 it is 2.0, and at 225 it is 8.0 against a spread of 10.4, so only offset 100 admits a fraction.

Offset	LeWM	alternative cost	success	recovered	GeoWM (frozen)
25	94.0 ± 4.0	whitened Euclid.	85.3 ± 1.2	–	97.3 ± 2.3
		running Euclid.	81.3 ± 7.0	–	
		regression MLP	83.3 ± 6.4	–	
50	58.7 ± 6.1	whitened Euclid.	–	–	60.7 ± 3.1
		running Euclid.	–	–	
		regression MLP	38.0 ± 6.0	–	
100	16.0 ± 6.0	whitened Euclid.	15.3 ± 1.2	-3%	38.0 ± 7.2
		running Euclid.	9.3 ± 6.1	-30%	
		regression MLP	4.0 ± 0.0	-55%	
225	5.3 ± 2.3	whitened Euclid.	2.7 ± 3.1	–	13.3 ± 10.1
		running Euclid.	3.3 ± 1.2	–	
		regression MLP	0.7 ± 1.2	–	

ond moment, so a handful of high-variance directions could dominate the distance and the failure would be one of conditioning rather than of geometry. We estimate $\Sigma^{-1/2}$ from training-episode latents of the same checkpoint, flooring the eigenvalues at 10^{-5} of the largest since a 192-dimensional covariance from a finite sample has near-null directions whose inversion would amplify noise into exactly the directions the cost knows least about, and measure Euclidean distance in the whitened space. It recovers -3% of the gap at offset 100 and sits below the baseline at 225, 2.7 against 5.3; at offset 25 it is worse than the unwhitened cost, 85.3 against 94.0. Conditioning is not what the quasimetric is fixing.

Not the fact that only the endpoint is scored The comparison throughout is terminal-only Euclidean against terminal-only quasimetric, but most Model Predictive Control (MPC) implementations accumulate a stage cost along the rollout, and accumulating is the obvious remedy for a terminal cost that supplies no gradient far from the goal. It is worse, not better: 9.3 against 16.0 at offset 100 and 3.3 against 5.3 at 225. A stage cost that misorders states does not improve by being summed over more of them; the Euclidean *geometry* is the problem, not the decision to read it once.

Not any learned temporal cost Section N shows that the asymmetry is worth little, which leaves open whether the metric structure buys anything over a scorer trained on the same temporal signal. This control removes the structure and keeps the signal: an unrestricted multilayer perceptron (MLP) on the concatenated pair, 115,801 parameters against the head's 115,200, fitted by regressing the true step separation on within-clip pairs, with no triangle inequality, no inequality constraint, and no push term. Data, split, optimizer, epoch budget, and planner are the GeoWM (joint) recipe unchanged. It fits what it is shown and cannot plan on it. Validation error settles at 0.116 on a target of $\{0, 1, 2, 3\}$ blocks, a root mean squared error (RMSE) of about a third of a block, so the head has learned within-clip temporal distance accurately; planning with it then reaches 4.0 at offset 100 against a baseline at 16.0 and 0.7 at offset 225 against 5.3, not merely no better than Euclidean but substantially worse, with two of three seeds at zero. The failure is informative. A clip is four frames, so the regression head never sees a separation beyond three blocks, and without the push term nothing calibrates distances between states from different episodes; at planning time it is asked for the distance between a rolled-out endpoint and a goal a hundred to two hundred and twenty-five steps away, far outside anything it was fitted on. The quasimetric is trained on the same short window and does extrapolate, which Section 3 attributes to the triangle inequality propagating the local constraint along trajectories. This is the control for that argument: same supervision, same budget, same capacity, structure removed, extrapolation gone.

An unstructured head at matched capacity The regression baseline above changes the architecture, the objective, the supervision signal, and the pair distribution at once, so its failure cannot be attributed to any one of them. This control changes one thing. The projection is replaced by a linear map of the same width and the group-wise maximum of Eq. (3) by an MLP that scores the two projections jointly, giving 115,201 parameters against 115,200. Everything else is the configuration that produces the method: the same frozen encoder, the same $\mathcal{L}_{\text{qm}}$ evaluated by the same

Table A10: **An unstructured pairwise head at matched capacity plans worse than the baseline it was meant to improve.** The head is fitted to the same frozen encoder by the same objective on the same pairs. Push-T held out, three seeds.

Offset	LeWM	unstructured head	GeoWM (frozen)
25	88.7 ± 2.3	30.0 ± 5.3	$\mathbf{91.3 \pm 2.3}$
50	51.3 ± 3.1	33.3 ± 4.6	$\mathbf{66.0 \pm 9.2}$
75	20.7 ± 5.0	20.7 ± 8.1	$\mathbf{48.0 \pm 7.2}$
100	11.3 ± 1.2	7.3 ± 2.3	$\mathbf{41.3 \pm 10.3}$
225	3.3 ± 3.1	1.3 ± 2.3	$\mathbf{9.3 \pm 5.0}$

code path, the same pairs, the same schedule, and the same three seeds. Table A10 shows that it does not merely fail to match the method; at every offset it is at or below the Euclidean baseline. Two measurements over the same three seeds say why, and the first rules out the explanation that would void the control. The unstructured head does not underfit the shared objective, it fits it *better*: on held-out states $\mathcal{L}_{\text{qm}}$ is 0.57 ± 0.09 for it against 0.93 ± 0.03 for the quasimetric head, lower in every seed. What it has found is a degenerate solution the objective admits. The objective asks that random pairs be far apart and that consecutive frames lie within one unit, and both are satisfied by making *every* distance a constant below one: the push term is then as small as a constant can make it and the local term is exactly zero. The head does almost that. The ratio of its distance on consecutive frames to its distance on random pairs is 0.992 ± 0.004, against 0.19 ± 0.09 for the quasimetric head, and while its absolute scale drifts across seeds (0.59, 0.40, 0.71 on random pairs) that ratio does not, which is what a learned constant looks like. A cost that assigns a neighboring state and an arbitrary one the same value gives CEM nothing to rank, and that is what the table records.

The quasimetric form cannot take that solution. Its distance is the group-averaged maximum of positive differences between projections, so it is pinned to the geometry of ψ and vanishes identically when its two arguments agree; it cannot hold all pairs apart while holding consecutive pairs close, because both are read off the same projection. Structure here is not buying accuracy on the training objective, it costs accuracy on the training objective; it is excluding a solution that satisfies the objective without ordering anything. We had expected the triangle inequality to be the operative constraint and measured it directly. It is not what separates these two heads: neither violates it on our data, and for the unstructured head the test cannot bind at all. The tightness of the bound, the direct distance divided by the shortest detour through a sampled intermediate, is 0.95 ± 0.01 for the quasimetric head, whose largest values reach 1.000 in every seed, against 0.51 ± 0.00 for the unstructured head, whose every detour is twice the direct distance. A zero violation rate for the second is therefore uninformative, and we report the collapse above as the mechanism instead.

How the fitted head is read at planning time Two things about the terminal cost are choices rather than consequences of the objective, and neither needs retraining to test: which of its two arguments is the origin, and whether it is read once at the endpoint or accumulated along the rollout (Table A11). The head is asymmetric, and the planner scores $d_\psi(\hat{z}_H, z_g)$. Reading $d_\psi(z_g, \hat{z}_H)$ instead keeps the head, the checkpoint, the search, and the episodes identical and reverses only the direction, which asks whether the asymmetry is used at planning time rather than whether it can be trained around. It cannot be read backwards: the reversed cost tracks the Euclidean baseline at every offset and falls below it at the longest. This is a sharper statement than Section N makes, and the two are consistent rather than in tension: a symmetric head *retrained* from scratch recovers most of the gain on three benchmarks because it can learn a substitute, while the direction inside a head that was trained asymmetrically carries information the planner is already relying on. Accumulating the same cost along the rollout rather than reading it at the endpoint costs less but still costs. It sits below the baseline at the protocol offset, where the metric is not the binding constraint and averaging over a rollout only adds noise, and clears it from offset 50 outward by a widening margin, so the gain is not an artifact of reading the cost once; it stays below the endpoint form at every offset, so where the cost is read is a real if secondary choice. Together with the Euclidean stage-cost control above, this separates *which* cost is read from *how* it is read.

N What the Asymmetry Is Worth

The cost we train is a quasimetric, and the introduction motivates that choice by reachability being asymmetric; Fig. 6 summarizes this section. Comparing the head against a Euclidean

Table A11: **Two ways of reading the same fitted head, against the baseline and against the method.** Push-T held out, three seeds, no retraining anywhere in the table: every column uses the same checkpoints, and they differ only in the function the planner evaluates.

Offset	LeWM	reversed $d_\psi(z_g, \hat{z}_H)$	accumulated along rollout	GeoWM (frozen)
25	88.7 ± 2.3	80.0 ± 2.0	84.0 ± 0.0	$\mathbf{91.3 \pm 2.3}$
50	51.3 ± 3.1	45.3 ± 5.0	62.7 ± 7.6	$\mathbf{66.0 \pm 9.2}$
100	11.3 ± 1.2	12.7 ± 5.0	28.7 ± 6.1	$\mathbf{41.3 \pm 10.3}$
225	3.3 ± 3.1	1.3 ± 2.3	6.7 ± 4.2	$\mathbf{9.3 \pm 5.0}$

Table A12: **The symmetric arm against GeoWM (joint).** "recovers" is the median fraction of the LeWM-to-GeoWM (joint) gain it reaches, over cells where that gap exceeds three points in favor of GeoWM (joint), and "cells" counts those; the sign test uses every cell, which is why its denominator is larger. The paired difference is what asymmetry adds per problem, with a bootstrap interval over problems and a sign test over cells; on Push-T no single cell resolves eight points but all sixteen point the same way. Bold marks an interval excluding zero.

Benchmark	cells	recovers	GeoWM (joint) − sym	95% CI	sign	p
Push-T	14	62%	$\mathbf{+8.0}$	$[+5.9, +10.1]$	16/16	$2e-05$
Two-Room	8	100%	-0.2	$[-0.4, +0.0]$	0/8	1.00
Reacher	10	100%	-0.4	$[-1.3, +0.4]$	2/12	1.00
Cube	12	111%	$\mathbf{-1.7}$	$[-2.9, -0.4]$	2/12	1.00

cost cannot establish that the asymmetry is what helps, because a learned nonlinear comparator, temporal supervision, contrastive pressure on random pairs, extra capacity, and asymmetry all change together. This ablation moves one of them: the head becomes its own symmetrization. Equation (3) takes $\mathrm{relu}(\psi(x) - \psi(y))$ and we replace it by $|\psi(x) - \psi(y)|$, which equals $\max(\mathrm{relu}(\psi(x) - \psi(y)), \mathrm{relu}(\psi(y) - \psi(x)))$. The projection network, its parameter count, the training pairs, β, μ, the schedule, and the triangle inequality are all unchanged; the distance simply can no longer distinguish a direction from its reverse. Three seeds per benchmark, evaluated on the reported sweep.

Asymmetry earns its place on one benchmark of four (Table A12). On Push-T it is worth 8.0 points, an effect too small for any single cell of 150 problems to resolve but positive in all sixteen. On Two-Room and Reacher the symmetric arm recovers the entire gain, and those gains are large, sixty and thirty points, so this is a real null rather than two saturated benchmarks. On Cube the symmetric arm is slightly *better*. The conclusion the evidence supports is therefore narrower than the one a quasimetric framing would invite: what repairs long-horizon planning is a terminal cost trained to approximate temporal distance, and whether that cost is a quasimetric or a metric changes the outcome on one benchmark and not on the other three.

Is the one benchmark the irreversible one? The obvious guess is that asymmetry pays where the dynamics do not run backwards. That can be checked without the encoder, which matters because otherwise the model-level finding would be explained by the model. For a pair of states recorded 25 steps apart we ask whether the data ever makes that journey in reverse: is there a state within ϵ of the later one that reaches within ϵ of the earlier one inside 25 steps? The same question asked forward from an independent start controls for coverage, since a goal-directed behavior policy may simply never attempt to go back. Table A13 gives the rates. Push-T is the outlier: its reverse traversals resolve at 0.300 against a forward rate of 0.769, while the other three come within a tenth of their own forward rate. It is also the one benchmark where asymmetry helps. That is consistent with the motivation, but it is one positive case out of four and we do not present it as a demonstrated mechanism. Two limits are worth stating. The absolute figure moves with ϵ (Push-T reads 0.93 at the smallest tolerance and 0.61 at twice that), so only the ordering at comparable forward coverage carries meaning, and Cube read 0.46 at a tolerance where its forward rate was 0.248 and 0.11 once that rate reached 0.918. And a behavior policy that never tries to reverse would produce this signature without the dynamics forbidding it; the forward control bounds that confound but does not remove it.

Table A13: **Reversibility of the recorded dynamics.** 1000 state pairs per benchmark, computed from logged states only. ϵ is raised until forward traversal resolves often enough for the two directions to be compared, so the multiplier differs by benchmark and only readings at comparable forward rates should be compared.

Benchmark	state dim	ϵ	forward	reverse	$1 - \text{rev}/\text{fwd}$
Push-T	11	$\times 2$	0.769	0.300	$+0.61$
Two-Room	4	$\times 4$	0.546	0.498	$+0.09$
Reacher	2	$\times 4$	0.723	0.699	$+0.03$
Cube	21	$\times 2$	0.918	0.815	$+0.11$

O Probing the Shaped Representation

Reshaping a latent space to serve planning invites an obvious objection: perhaps the metric was bought by degrading the representation. We test this with the probing protocol of Maes et al. (2026), fitting probes on training episodes and scoring them on held-out ones so that probe capacity cannot be mistaken for representation quality. Table A14 reports the linear probe on all four benchmarks over three seeds; our nonlinear probe scores below the linear one in most cells, which indicates that it is underfitted rather than that the latent is linearly saturated, so we report the probe we can vouch for. The two arms compared are the baseline and the joint form, the only two whose representations differ; GeoWM (frozen) leaves the encoder untouched, so its probe values are the LeWM column by construction. GeoWM (joint) retains the physical content: every quantity moves in its favor or is level and none moves against it, the largest being the block orientation of Push-T at $0.934 \rightarrow 0.983$.

Probe sample size How many episodes the probe sees matters on one benchmark. Push-T episodes run from 49 to 246 steps, so the 40 training episodes the protocol defaults to do not define the scale the held-out episodes live on; the numbers climb monotonically with episodes and settle by 6000, where 6000 and 3000 agree to 0.03 on the worst quantity and 0.005 on the rest, and at that sample all three seeds agree with a spread of ± 0.011. The other three benchmarks are converged by 400 episodes. Coverage is bought with stride rather than with pixels, since the probe measures prediction across the state distribution and not within a trajectory: at 400 episodes, stride 3 and stride 10 agree within noise while stride 10 reads a third of the frames.

What neither arm carries The block yaw of Cube is the one quantity neither arm carries. Both sit at $r \approx 0$ with error 1.01, which is exactly what predicting the mean scores, and the reading does not move between 40 and 6000 episodes. The encoder is a ViT-Tiny reading a 224×224 render in which the cube is small, and neither objective gives it a reason to encode the orientation of the cube. Both arms are equally blind to it, so it bears on what these latents contain rather than on what shaping them costs.

Two negative measurements GeoWM (joint) does not improve the temporal path-straightening statistic of Maes et al. (2026), the mean cosine between consecutive latent velocity vectors: 0.685 ± 0.078 for LeWM against 0.654 ± 0.078 for the joint form, a difference well inside one standard deviation. Our objective constrains distances, not the curvature of latent paths, and the measurement confirms it leaves that property alone. Separately, a t-SNE embedding of the two latent spaces is essentially indistinguishable, so we do not present one as evidence: t-SNE preserves local neighborhood structure, which both arms get right because both are trained to predict the next latent, whereas the property at issue is global distance ordering, which t-SNE does not preserve and which Fig. 3 visualizes instead.

P A Constant Relative Execution Budget

Every other sweep in this paper sets the execution budget to offset $+ 50$. That rule keeps the absolute slack fixed and therefore lets the *relative* slack fall as the goal recedes: the budget is three times the goal distance at offset 25 and 1.22 times it at offset 225 under the held-out protocol. A reader could accordingly explain the separation without appealing to distance at all, by supposing that distant goals simply receive less excess budget relative to what they require and that a learned cost spends a tight budget better than a Euclidean one. The fixed-cohort sweep of Section Q does not address this, because it holds the problems fixed and not the budget ratio. This sweep holds the ratio at 2.0 at every offset, so relative slack is constant along it and distance is the only thing that varies; if the separation were an artifact of tightening slack, it would have to shrink here. It does not (Table A15).

Table A14: **Linear probes for physical quantities, all four benchmarks.** Mean ± std over three seeds, fitted on training episodes and scored on held-out ones; bold marks the better arm. Only quantities that are in the observation are probed: Two-Room and Reacher supply the goal as a separate image rather than rendering it into the frame, so goal position is not one. Angles are probed as $(\sin, \cos)$, since a linear probe on a raw angle is broken by wraparound. Every quantity moves in favor of GeoWM (joint) or is level and none moves against it, so shaping the representation does not lose physical content, and freezing it cannot. The block yaw of Cube is the one quantity neither arm carries.

		r (higher better)		error (lower better)	
Benchmark	Quantity	LeWM	GeoWM (joint)	LeWM	GeoWM (joint)
Push-T	agent location	0.954 ± 0.006	$\mathbf{0.967 \pm 0.002}$	0.091 ± 0.011	$\mathbf{0.067 \pm 0.005}$
	block location	0.984 ± 0.002	$\mathbf{0.990 \pm 0.001}$	0.034 ± 0.004	$\mathbf{0.020 \pm 0.001}$
	block angle	0.934 ± 0.011	$\mathbf{0.983 \pm 0.001}$	0.121 ± 0.020	$\mathbf{0.031 \pm 0.003}$
Two-Room	agent location	0.998 ± 0.000	$\mathbf{0.999 \pm 0.000}$	0.003 ± 0.000	$\mathbf{0.002 \pm 0.000}$
Reacher	agent location	0.999 ± 0.000	$\mathbf{0.999 \pm 0.000}$	0.003 ± 0.001	$\mathbf{0.001 \pm 0.000}$
	joint angles	1.000 ± 0.000	1.000 ± 0.000	0.001 ± 0.000	0.001 ± 0.000
Cube	agent location	0.988 ± 0.001	$\mathbf{0.996 \pm 0.000}$	0.023 ± 0.002	$\mathbf{0.009 \pm 0.000}$
	block location	0.994 ± 0.000	$\mathbf{0.997 \pm 0.000}$	0.012 ± 0.001	$\mathbf{0.007 \pm 0.000}$
	block angle	-0.013 ± 0.005	-0.011 ± 0.011	1.011 ± 0.001	1.010 ± 0.002

Table A15: **Push-T held out with the execution budget fixed at twice the goal offset.** Relative slack is the same at every distance. The last two columns give the separation the standard budget rule produces at the same offset and the difference between the two. The separation survives at every offset and its size barely moves: the two rules disagree by a median of 0.7 points and never by more than 6.0, inside the run-to-run spread of Section F. Distance, not the shrinking budget ratio, is what the sweep varies.

Offset D	Budget $2D$	LeWM	GeoWM (frozen)	Δ here	Δ standard	difference
25	50	86.7 ± 1.2	88.7 ± 5.0	$\mathbf{+2.0}$	$+2.6$	-0.6
50	100	50.7 ± 3.1	66.0 ± 8.7	$\mathbf{+15.3}$	$+14.7$	$+0.6$
75	150	21.3 ± 4.2	49.3 ± 6.4	$\mathbf{+28.0}$	$+27.3$	$+0.7$
100	200	10.7 ± 3.1	44.7 ± 9.0	$\mathbf{+34.0}$	$+30.0$	$+4.0$
125	250	11.3 ± 3.1	34.7 ± 5.0	$\mathbf{+23.4}$	$+23.3$	$+0.1$
150	300	9.3 ± 1.2	26.7 ± 2.3	$\mathbf{+17.4}$	$+19.4$	-2.0
175	350	4.0 ± 3.5	21.3 ± 12.7	$\mathbf{+17.3}$	$+17.4$	-0.1
200	400	2.7 ± 2.3	24.0 ± 5.3	$\mathbf{+21.3}$	$+15.3$	$+6.0$
225	450	4.7 ± 3.1	9.3 ± 4.2	$\mathbf{+4.6}$	$+6.0$	-1.4

Q A Horizon Sweep on a Fixed Cohort

A start state enters the reported sweep only if its own episode is long enough to place a goal at that offset, so the pool of episodes and starts shrinks as the goal recedes. The comparison within each cell is unaffected, since both arms always see the identical problems, but the sweep itself moves two things at once, and the reading that the gap opens *because* the horizon grows is not clean: it could in part be that the baseline is worse on the episodes long enough to support a distant goal. The control is to decide admission once, at the largest offset in the sweep, and then evaluate every offset on that same cohort. On Push-T that leaves 2,525 in-distribution starts and about 280 held-out ones, all of them near the beginning of unusually long episodes. Absolute rates are therefore not comparable to Table A5, since a long expert episode is a hard configuration and this cohort keeps only the hard ones; what is comparable is the shape.

The shape survives (Table A16). From offset 50 onward the restricted pool is harder for both arms, the baseline falling from 51.3 to 27.3 held out and the head from 66.0 to 26.7, and the head still leads from offset 75 outward, by 6.0, 5.3, 6.0, and 9.3 points, while GeoWM (joint) leads from offset 75 onward and reaches $+14.7$ and $+12.7$ at offset 225 under the two protocols. The separation is therefore a property of the horizon and not of which episodes are long enough to support it.

Two things this control surfaces do not flatter the method. First, the protocol offset is the exception, and not because the pool is harder: held out, the baseline scores 88.7 on it either way, exactly what it scores on the ordinary sweep, while the frozen head falls from 91.3 to 78.0 and the joint form from 89.3 to 60.0. These problems are no harder for a Euclidean cost; they are harder for a learned one, and the cost that helps at distance hurts more up close on hard configurations. The claim that the head costs nothing at short horizon therefore holds where starts are drawn as the protocol draws

Table A16: **Push-T with the cohort fixed at the offset-225 admission rule, three seeds.** Every offset is evaluated on the same starts, so only the goal distance varies. Bold marks a gap above five points.

Offset	LeWM	GeoWM (frozen)	GeoWM (joint)	Δ to frozen
IN-DISTRIBUTION				
25	89.3 ± 3.1	78.0 ± 8.0	59.3 ± 32.3	$\mathbf{-11.3}$
50	31.3 ± 5.0	30.0 ± 16.4	32.7 ± 12.2	-1.3
75	11.3 ± 5.8	20.7 ± 4.2	16.7 ± 10.1	$\mathbf{+9.3}$
100	11.3 ± 3.1	19.3 ± 11.4	16.0 ± 2.0	$\mathbf{+8.0}$
125	6.7 ± 4.6	8.7 ± 4.2	12.0 ± 4.0	$+2.0$
150	2.0 ± 2.0	8.0 ± 6.0	20.7 ± 10.3	$\mathbf{+6.0}$
200	8.0 ± 5.3	10.0 ± 3.5	20.0 ± 8.7	$+2.0$
225	4.0 ± 4.0	12.0 ± 7.2	18.7 ± 6.1	$\mathbf{+8.0}$
HELD-OUT				
25	88.7 ± 5.0	78.0 ± 10.6	60.0 ± 31.0	$\mathbf{-10.7}$
50	27.3 ± 11.7	26.7 ± 12.2	31.3 ± 8.1	-0.7
75	6.0 ± 4.0	12.0 ± 7.2	12.0 ± 7.2	$\mathbf{+6.0}$
100	6.0 ± 3.5	11.3 ± 12.1	10.0 ± 4.0	$\mathbf{+5.3}$
125	2.0 ± 2.0	8.0 ± 5.3	14.7 ± 16.8	$\mathbf{+6.0}$
150	2.7 ± 1.2	12.0 ± 7.2	25.3 ± 8.1	$\mathbf{+9.3}$
200	3.3 ± 3.1	6.7 ± 5.0	16.7 ± 8.3	$+3.3$
225	4.7 ± 5.0	8.7 ± 3.1	17.3 ± 9.2	$+4.0$

them and fails on a pool restricted to the episodes that support the longest goals. Second, that cell is unstable across training seeds on both GeoWM arms and on neither the baseline. Held out, the frozen head reads 87.3, 81.3, and 66.7 across seeds while the baseline on the same problems reads 94, 84, and 88; in distribution, the joint form reads 30, 54, and 94 against 92, 86, and 90. We re-measured the frozen cell three times per seed on the same checkpoints and the same admitted problems, and within a seed the readings move by two points, so the variation is between the heads that were trained, not in the evaluating of them, and the twenty-point device effect of Section F does not account for it. A learned cost has room to vary on a subpopulation where a fixed norm does not, and one head in three is materially worse here. We report the cohort sweep because it removes a confound, not because it is kind to the method, and the short-horizon cells are the part of the paper least supported by it.

R Negative Results and Boundaries of the Claim

We record four negative findings, because each delimits the claim.

A prediction that did not hold We expected a metric fitted on encoder outputs to fail when evaluated on rollout latents, which drift off the encoder manifold, and built the GeoWM (frozen) arm to demonstrate it. With the freeze asserted in code and verified end to end, the result was the opposite and held across three seeds (Section 4.4): separate fitting of the head is not the failure we predicted, and we do not have a mechanism for why the rollout drift is benign.

Geometry is not the only bottleneck In preliminary work on a substantially harder image-based manipulation suite, every arm we tried, including a strong search baseline, achieved a success rate of zero, and a diagnostic GeoWM (frozen) metric with excellent rank correlation on real latents did not change it. Systematic controls eliminated data scale, amortization, representation collapse, and training length as explanations, which implies that the errors there are systematically biased rather than merely noisy and that the binding constraint is something other than the goal metric. This bounds the scope of the present claim: repairing the geometry repairs the geometry, and a setting whose failure lies elsewhere will not be rescued by it.

Two Cube observations we cannot explain LeWM scores lower at offset 50 than at 100 (56.0 ± 2.0 against 63.3 ± 1.2), outside the seed spread and present in every seed; we have no account of it. And Cube carries the smallest geometry repair we measure ($\Delta\rho = 0.05$) alongside a planning gain at every offset, which is the strongest single piece of evidence that the two quantities do not track each other: the head can help without moving the diagnostic that motivated it.

No protocol-compatible external comparison Quasimetric reinforcement learning (RL) methods (Wang et al., 2023; Myers et al., 2025) learn a quasimetric value and extract a policy that descends

it, with no test-time rollout of a WM, and are evaluated under goal-conditioned protocols that differ from the in-distribution WM protocol used here. We considered porting them but concluded that a number-to-number comparison across protocols would be more misleading than informative, and that a faithful port is a separate undertaking. We state the gap rather than presenting an apples-to-oranges table.