TopoExplore: Homology as an Exploration Signal

Anonymous authors
Paper under double-blind review

Abstract

Exploration signals in reinforcement learning are currently computed from what an agent has seen: visitation counts, density estimates, or a model's prediction error at individual states. None of these report the holes in the visitation space. To capture them, we compute the persistent homology of the agent's own archive of visited states during exploration and turn each detected class—an enclosed region the archive surrounds but has not entered—into a selection bonus concentrated on the archived states from which entry is possible, gated by an attempt counter that retires candidate entrances and prunes sealed structures. The method, TopoExplore, is Go-Explore plus one additive term, so the comparison with Go-Explore isolates that term. On the 189 held-out worlds of the open-source TopoGym benchmark of topologically varied hard-exploration environments, TopoExplore finds the goal in 167 worlds against 152 for Go-Explore and, where both find it, with a median 126k steps against 186k. On an environment that stress-tests far-apart chambers that create holes in the archive space, TopoExplore enters all six or eight chambers across 5 seeds, against 1 and 2 of 5 seeds for Go-Explore. On Montezuma's Revenge, built over the room graph, the term helps whenever the archive surrounds a room it has not entered, puts 33–45% of the selections on the surrounding rooms while it is active, and enters the room sooner in five of six cases; such enclosures are rare on this game, so the score is unchanged. Beyond discrete, restorable state spaces, the same signal drives an embodied explorer that walks to every target it selects and explores from depth alone; a vision-language-action navigation policy fine-tuned on its trajectories succeeds 58% more often on unseen scenes than one fine-tuned on shortest-path oracle demonstrations.

1 Introduction

Hard exploration became an important problem in deep reinforcement learning when DQN, after succeeding on many Atari games (Bellemare et al., 2013) where exploration is not the bottleneck, scored nothing on Montezuma's Revenge (Mnih et al., 2015), and on Pitfall!, under sticky-action evaluation, algorithms without demonstrations scored at most zero until Go-Explore (Ecoffet et al., 2019; 2021). The decade since has mostly asked what an agent should be rewarded for seeing: rare states, by counts and their density surrogates (Bellemare et al., 2016; Ostrovski et al., 2017; Tang et al., 2017); badly predicted states, by curiosity (Pathak et al., 2017; Burda et al., 2019b); both at once with episodic novelty and tens of billions of frames (Badia et al., 2020b;a). All are statistics of the state the agent is in — a count, a density, a prediction error — and all have the weakness of the signal being local in state and in time. A model that predicts a corridor well says nothing about the room it encloses, and an agent driven by a bonus such as RND's can suffer detachment, drifting away from a frontier whose bonus it has partly consumed and never finding its way back (Ecoffet et al., 2019; 2021), so the entrance to a sealed region earns no more bonus than any wall, with nothing in the signal proportional to how many states lie behind it.

Go-Explore (Ecoffet et al., 2019; 2021) changed how the field approaches exploration by introducing a simple archive of visited states on which an exploration signal can act; selecting cells by visit counts, missing neighbours and domain-knowledge cells rather than a learned bonus, then robustifying the trajectories it found, it surpassed human performance on both games. Intelligent Go-Explore (Lu et al., 2025) replaces these heuristics with a foundation model that reads the archive and chooses which state to return to, which actions to explore from it, and which new states are interesting enough to keep, a model call at every one of these decisions, which is computationally expensive. The archive

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.

remains a useful place for an exploration signal; what it lacks is a direct, global signal of the shape of the environment seen so far.

Persistent homology provides such a signal: the first homology of the visited set counts exactly the regions the agent has walked around but not into, and in higher dimensions the cavities in its coverage. Each such region is a class, and it comes paired with a persistence, a magnitude that grows with the region's radius and measures how robust the region is to perturbation; computing it needs neither a model nor a gradient that might vanish. This paper uses it as one of the main exploration signals by treating each detected class as a hypothesis, *there is an enclosed region here that is perhaps enterable*, and giving it a selection bonus, its *mass*, spread over the archived cells best placed to test it. Each candidate cell's share shrinks with every failed attempt; once all are exhausted, the class is pruned and marked sealed. We use Go-Explore because its archive is exactly what the homology signal would be computed on anyway, and because it is a proven method whose score we can change by one term.

Primary contributions.

1. **The first use of persistent homology as an exploration signal in reinforcement learning.** The H_1 classes of the archive of visited states decide where the agent explores from, with no map of the environment and no planner (Section 3); prior uses of homology in robotics and learning are reviewed in Section 2.
2. **Evidence that it explores enclosed regions faster when it is there.** On 189 held-out TopoGym worlds, where both find the goal, TopoExplore finds it faster than Go-Explore (median speed-up $1.33\times$, $1.97\times$ with the curvature term), and it finds it in more worlds (Section 5.1); on Montezuma's Revenge, built over the room graph, it entered an encircled room 186k–853k frames sooner than the same seed without it in five of six paired cases, and on Pitfall!, which has no enclosures, the term costs little (Section 5.2, Appendix C.2).
3. **The signal as a data generator for embodied policies.** Walking to each target instead of restoring states, the same explorer collects navigation trajectories from depth alone on which a vision-language-action policy succeeds 58% more often on unseen scenes than on oracle demonstrations (Section 5.3).

2 Background and Related Work

We outline prior exploration signals in more detail, give the background on persistent homology the method rests on, and then survey how topological data analysis has been used in the field.

Counts and archives. Count-based bonuses and their density-model surrogates (Bellemare et al., 2016; Ostrovski et al., 2017; Tang et al., 2017) score states individually and are blind to the shape of the visited region. Go-Explore (Ecoffet et al., 2019; 2021) keeps the visited cells in an archive and alternates returning to a selected cell with exploring from it, the cell drawn with probability proportional to a score over per-cell visitation attributes plus a neighbour term; our method builds directly on it and changes only its score, whereas Intelligent Go-Explore (Lu et al., 2025) hands selection, exploration and archiving alike to a foundation model's judgement over the archive. Frontier exploration (Yamauchi, 1997) drives to the boundary between known-free and unknown space. Novelty search (Lehman & Stanley, 2011) shares the archive but scores members by behavioural distance, and MAP-Elites (Mouret & Clune, 2015) bins them by behaviour and keeps the best performer per bin; neither scores by the geometry the archive collectively describes.

Distribution objectives and curiosity. Maximum-entropy exploration (Hazan et al., 2019) and skill discovery (Eysenbach et al., 2019) optimise the distribution of visited states and ignore whether its support has holes. Prediction-error curiosity (Schmidhuber, 1991), ICM (Pathak et al., 2017), RND (Burda et al., 2019b), disagreement (Pathak et al., 2019) and information gain (Houthooft et al., 2016) respond only to the present: high where the model is currently wrong, and carrying nothing about a region the agent has surrounded. Agent57's budgets (Badia et al., 2020a) are three orders of magnitude beyond ours; we use RND and ICM as references at a matched step budget, though at a far larger computational budget (Appendix C.5).

Complexes, H_1, and persistence. A complex glues vertices, edges, faces and higher-dimensional cells, the last of which we will not need, along their boundaries; we use only simplicial and cubical ones. Over $\mathbb{F}_2$ the boundary map ∂_n sends each n-cell (n-dimensional cell) to the sum of its $(n-1)$-faces, $\partial_n \partial_{n+1} = 0$, and the first homology group $H_1(K) = \ker \partial_1 / \operatorname{im} \partial_2$ is, informally, closed loops modulo those that bound filled regions (Edelsbrunner & Harer, 2010). We use two standard constructions, chosen according to the structure of the state space. For a grid or lattice, we use a cubical complex. Given a set of lattice cells $A \subset \mathbb{Z}^2$, let $K(A)$ contain the unit squares indexed by A together with all of their edges and vertices. In the planar case, each independent class in $H_1(K(A))$ corresponds to a bounded connected component of the complement, i.e., an enclosed pocket of cells not in A. For a finite point set P in a general metric space, the standard construction is the Vietoris–Rips complex $\mathrm{VR}_\varepsilon(P)$, which fills every subset of points whose pairwise distances are at most ε. In Euclidean space, write P_r for the union of the closed balls of radius r about the points of P, and $\check{C}_r(P)$ for the Čech complex, which fills every subset of points whose balls share a common point. By the nerve lemma (Borsuk, 1948), $\check{C}_r(P)$ has the homotopy type of P_r, and hence its homology, whereas Vietoris–Rips only approximates it, sitting between two Čech complexes, $\check{C}_{\varepsilon/2}(P) \subseteq \mathrm{VR}_\varepsilon(P) \subseteq \check{C}_\varepsilon(P)$. We use the alpha complex, which has the same homotopy type as $\check{C}_r(P)$ (Edelsbrunner, 1995, Thm. 3.2) but in two and three dimensions is much faster to build than either. Increasing the radius gives a nested family of complexes, a filtration. Grid experiments use the cubical complex; the embodied explorer uses the alpha complex, built with GUDHI (Maria et al., 2014). Persistent homology removes the need to rely on a single scale by tracking homology classes throughout a filtration. A class that first appears at scale ε_b and becomes filled at ε_d defines an interval $[\varepsilon_\mathrm{b}, \varepsilon_\mathrm{d})$ with persistence $\mathrm{pers} = \varepsilon_\mathrm{d} - \varepsilon_\mathrm{b}$. Hence persistence measures how long a hole survives as the complex is gradually thickened. In our method, H_1 identifies where unexplored regions occur, while persistence provides a scale-dependent measure of how substantial those structures are, and how robust they are to perturbations of the space. Carlsson (2009) provides a survey of the subject and Edelsbrunner & Harer (2010) is a full introduction for more details.

Topological data analysis. The closest lineage is robotic coverage and exploration: de Silva & Ghrist (2007) certify, from sensor connectivity alone and without coordinates, that a static sensor deployment leaves no hole; Kim et al. (2013) send robots along distinct homology classes of paths around obstacles toward the unexplored region, and Bhattacharya et al. (2015) plan paths through such a map by their homology classes. The robotic works assume a map with coordinates, so that any point can be addressed, and a planner that can reach any point of it; ours has neither. In learning, persistence has entered as an input or a regulariser (Hofer et al., 2017; Carrière et al., 2020; Moor et al., 2020) and as a complexity measure of the weights that decides when training stops (Rieck et al., 2019), never as a signal for where an agent explores, and in RL "topological" has meant a graph over past experience (Savinov et al., 2018; Hong et al., 2022), which captures connectivity but not whether the visited set encloses anything.

3 Method

TopoExplore is Go-Explore plus one additive term in the cell score. The archive, the return mechanism, the exploration rollout, and the base score are inherited unchanged, so every comparison with Go-Explore in Section 5 isolates that one term.

Setup. An agent maintains an archive A_t of discretised cells, each storing the best trajectory that reached it and per-cell counters. At each iteration it draws a cell c with probability $\mathrm{Score}(c) / \sum_{c' \in A_t} \mathrm{Score}(c')$, returns to it, explores for a fixed number of steps, and adds every visited cell to the archive. Go-Explore's score is

$$\mathrm{CellScore}(c) \;=\; M(c)\Big[\sum_{a \in \mathrm{attrs}} \Big(w_a \Big(\tfrac{1}{v(c,a)+\epsilon_1}\Big)^{p_a} + \epsilon_2\Big) \;+\; \sum_{n} w_n \big(1 - \mathrm{HasNeighbor}(c,n)\big) \;+\; 1\Big], \tag{1}$$

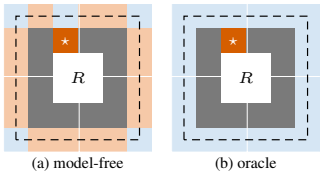
with $\mathrm{attrs} = \{\text{times chosen},\ \text{times seen},\ \text{times chosen since new}\}$ the visitation attributes of Ecoffet et al. (2019), the last being the number of times the cell was chosen since exploration from it last found a new or better cell, the neighbour term counting lattice neighbours absent from the archive (on Montezuma's Revenge also the cell at the same position holding one more key), and $M(c)$ Go-Explore's level multiplier, 1 outside Montezuma's Revenge.

3.1 The archive complex and its H_1 classes

Let K_t be the complex of Section 2 over A_t: cubical on a lattice, where its H_1 classes are exactly the components of unvisited space enclosed by visited space, and the alpha complex of the surface points the embodied explorer has seen (Appendix B.2). Each live class γ has $\mathrm{rim}(\gamma)$, the archived cells that are 4-neighbours of the enclosed unarchived region: precisely the archived cells from which entry may be possible. Its magnitude $b(\gamma)$ is the enclosed region's inradius, which is also the persistence of the class under thickening.

3.2 Gating, attempt counts, and retirement

A rim cell is useful only if some action from it enters the enclosed region. Define $g(c,\gamma) = 1$ if it does and 0 otherwise, and $\mathrm{gated}(\gamma) = \{c \in \mathrm{rim}(\gamma) : g(c,\gamma) = 1\}$ for the true entrances. If g were known the bonus would go to $\mathrm{gated}(\gamma)$ directly, the oracle mode of Appendix A; a partial version of this is what the texture arm supplies when an observed entrance affordance narrows a class's support (Section 3.3). The agent in general has no model of the environment's dynamics and cannot evaluate g; it can only attempt entry and count the attempts. We call this *model-free*, as opposed to an oracle that knows g (Appendix A); every run here is model-free: observations may narrow the rim (Appendix B.1), but entrances are found by trial. It keeps an attempt count $u(c,\gamma)$, incremented on every selection of c for γ regardless of outcome, and weights the cell by



(a) model-free (b) oracle

Figure 1: Where the bonus goes. (a) Model-free: the whole rim (orange), the visited cells bordering R; attempts find the door ($\star$). (b) Oracle, g known: the door alone. Dashed: γ.

$$\hat{g}(c,\gamma) \;=\; \big(1 + u(c,\gamma)\big)^{-p_u}, \qquad \mathrm{active}(\gamma) \;=\; \{\, c \in \mathrm{rim}(\gamma) : u(c,\gamma) < u_{\max} \,\}. \tag{2}$$

A cell leaves the active set after $u_{\max}$ attempts, and a class whose active set is empty is pruned and reported as sealed. Note that a decoy, a class around a solid and so with no true entrance (Appendix A), absorbs at most $|\mathrm{rim}(\gamma)|\, u_{\max}$ bonus draws before it is pruned (Appendix A).

3.3 The topological term

With $\psi(c,\gamma) = \hat{g}(c,\gamma) / \sum_{c' \in \mathrm{active}(\gamma)} \hat{g}(c',\gamma)$ the share of a class's *mass*, its total bonus $w_{\mathrm{topo}}(t)\, b(\gamma)^{p_{\mathrm{topo}}}$, that rim cell c receives,

$$S_{\mathrm{hom}}(c) \;=\; w_{\mathrm{topo}}(t) \sum_{\gamma \,\in\, \mathrm{live}(t)\,:\, c \in \mathrm{active}(\gamma)} b(\gamma)^{p_{\mathrm{topo}}}\, \psi(c,\gamma), \tag{3}$$

and the cell score becomes

$$\mathrm{Score}_{\mathrm{TE}}(c) = M(c)\big[\mathrm{CellScore}(c) + S_{\mathrm{hom}}(c) + S_{\mathrm{loc}}(c)\big], \tag{4}$$

where $M(c)$ is the environment's optional cell multiplier (Go-Explore's level weight on Montezuma's Revenge) and S_{loc} an optional local term, zero in TopoExplore proper. We use $w_{\mathrm{topo}}(t) = \alpha |A_t|$ so that the topological term's share of the total score, $\delta_t = w_{\mathrm{topo}} \sum_{\gamma\ \mathrm{live}} b(\gamma) \big/ \big(\sum_{c \in A_t} \mathrm{CellScore}(c) + w_{\mathrm{topo}} \sum_{\gamma\ \mathrm{live}} b(\gamma) \big)$ stays bounded away from zero as the archive grows (Appendix A).

The TopoExplore algorithm. Algorithm 1 (Appendix B) states the exploration phase in full; its starred lines are the only changes from Go-Explore, and with $\alpha = 0$ the two are identical. Phase 2 is the Backward Algorithm of Salimans & Chen (2018) as Go-Explore uses it, unchanged; TopoExplore changes only which trajectories it is given.

What TopoExplore guarantees. Chamber i has enclosing class γ_i, magnitude b_i and door-entry probability q_i; $s_{\min} = 1$ and $s_{\max}$ bound CellScore, $B_t^{\mathrm{live}} = \sum_{\gamma\ \mathrm{live}} b(\gamma)$, $\delta_{\min} \le \delta_t$ for all t, and T is the episode horizon. Precise statements and proofs are in Appendix A.

Theorem 1 (Dilution versus anytime entry, informal)**.** *Before γ_i is detected, any method whose per-cell score is bounded within a fixed ratio $s_{\max}/s_{\min}$, Go-Explore included, enters chamber i on the next episode with probability at most $C(T)/|A_t|$, with $C(T)$ fixed by T and the score bounds. Once γ_i is detected and every non-entrance rim cell has been retired, while the class is still live, TopoExplore enters with probability at least $\delta_t\,(b_i/B_t^{\text{live}})\,(q_i/|\text{gated}(\gamma_i)|)$, which $w_{\text{topo}}(t) = \alpha|A_t|$ holds constant in t.*

Theorem 2 (Fixed-budget separation, informal)**.** *Let k chambers remain in an otherwise explored world of m cells. TopoExplore enters all of them in at most $\delta_{\min}^{-1}\sum_i \beta_i$ expected iterations, with β_i counting chamber i's rim and interior: linear in what is left, independent of m. At twice that budget, N^*, TopoExplore enters all k with probability at least $1/2$, while Go-Explore enters all k with probability at most $(eN^*\lambda_0)^k$ for $\lambda_0 = s_{\max}\bar{n}/(s_{\min}m)$, $\bar{n}$ the largest number of archived cells within one rollout of a door: exponentially small in k whenever the explored world is much larger than what remains, $m \gg \sum_i \beta_i$.*

Theorem 3 (Certification, informal)**.** *On a finite grid, model-free TopoExplore almost surely reaches a stopping time that is observable — no frontier cell and no live class remain — at which every chamber has been entered and filled and every surviving class is a decoy pruned by gate exhaustion. It then reports H_1 of the environment with a certificate: the exhausted gate log for each surviving class, the filled interior for each class whose region was filled.*

Informally: while no hole is detected TopoExplore is Go-Explore; once one is, scaling w_{topo} with the archive keeps a constant fraction of selections on the hole's rim, so the time to enter k remaining chambers grows linearly in what is left while Go-Explore's chance of entering all k decays exponentially in k. Eventual coverage of a finite grid holds for either method (Lemma 5); the separation is in the rate, and the certificate records why each surviving hole was declared unenterable.

Local-signal arms. S_{loc} carries no part of the theory; two arms test what happens when we look at local topological properties and when we give semantic meaning to the unvisited holes. `ricci` gives a bonus to cells whose mean incident Ollivier–Ricci curvature (Ollivier, 2009) on the archive graph is negative (bottlenecks such as doorways and corridors), to test whether a local signal speeds up the coverage outside the holes that Theorem 2 assumes (an otherwise explored world), so that its separation takes hold sooner. `texture` records features the agent observed at each cell (door sprites, key possession, skull proximity on Montezuma's Revenge) and, when a live class's active rim contains cells showing an entrance feature, puts the class's whole mass on them, so it also acts on S_{hom} with a smaller, better-placed support.

4 Experimental Setup

TopoGym (chamber worlds). TopoGym is a library that contains a benchmark of 2D worlds whose topology is controlled by construction (Anonymous, 2026). The hold-out benchmark is TopoGym-v1's frozen `test` split, 189 worlds (63 units $\times$ 3 layout seeds) spanning chambers, decoys, bottlenecks, mazes, textured scenarios and non-trivial topologies, given 1M steps per world; TopoGym provides a fixed `tune` split, disjoint from the hold-out split, on which every method was tuned. Three further chamber-count families, not in the benchmark, satisfy the hypotheses of Theorem 2 (doors more than one rollout away, an otherwise explored world) to different degrees, OpenFieldChamberCount most closely; each places k chambers so that reaching the next costs more than one episode.

Montezuma's Revenge (Atari). The canonical hard-exploration game (Bellemare et al., 2016) of the Arcade Learning Environment (Bellemare et al., 2013), with 24 rooms in level 1, several gated behind keys, and the domain-knowledge cells of Ecoffet et al. (2019). Phase 1 runs deterministically with snapshot restore; archival arms get 15M agent steps (60M frames) and the intrinsic-motivation baselines 25M (100M frames), the sum of both TopoExplore phases. The complex is built at two scales, run as separate arms on the same seeds: within each room, on the cell lattice, and over the rooms themselves, on the room graph the agent learns from the passages it crosses — the latter so that a room the archive surrounds forms an H_1 class (Section 5.2). Pitfall!, a chain of 255 thin rooms with little enclosed structure or reward, is run with the same campaign and reported in Appendix C.2.

Embodied navigation (MultiON on HSSD). The embodied experiment tests whether the trajectories a topological explorer collects are good training data for a navigation policy. The task is MultiON

Table 1: Hold-out benchmark: 189 worlds, 1M steps each, methods tuned on the disjoint `tune` split. Goal found: worlds in which any training episode reached the goal — a treasure chest; each TopoExplore arm is paired with Go-Explore world by world, exact sign test. Steps to first goal: over the worlds where both found it, medians, the median per-world ratio Go-Explore/TopoExplore, and worlds where TopoExplore was faster / slower (sign test $p < 10^{-3}$ for every arm). PPO+ICM's hyperparameters were set to standard values rather than tuned (Appendix B.4), and 46 of its worlds were lost to withdrawn cloud machines, not to failure.

	Goal found in training		Steps to first goal (both found)	
Method	worlds	only TE / GE (p)	median TE vs GE	speed-up (faster/slower)
Go-Explore	152	—	—	—
TopoExplore (none)	167	16 / 1 (0.0003)	126k vs 186k	1.33× (97 / 54)
TopoExplore + ricci	168	19 / 3 (0.0009)	90k vs 180k	1.97× (118 / 31)
TopoExplore + texture	168	17 / 1 (0.0001)	123k vs 186k	1.50× (101 / 50)
TopoExplore + both	169	19 / 2 (0.0002)	85k vs 183k	1.97× (119 / 31)
PPO	3		—	
PPO+RND	4		—	
PPO+ICM	49		143 of 189 worlds	

(Wani et al., 2020) in the HSSD version of the 2024 MultiON challenge, in its single-goal projection on Habitat (Savva et al., 2019) with HSSD scenes (Khanna et al., 2024): 42 training scenes, 20 unseen evaluation scenes, 12 goal categories held out of training. The explorer is TopoExplore's selection rule and gate over a complex of seen surface points built from depth alone, with no archive to restore: it walks to every candidate it selects and keeps a per-episode map (Appendix B.2). The policy is InternVLA-N1-System2 (InternNav Team, 2025; Wei et al., 2026) fine-tuned with LoRA (Hu et al., 2022) under one recipe on one of six training sets: oracle shortest-path demonstrations, topological-explorer episodes, two other explorers on the same scenes and starts, and two mixtures (Appendix B.2). The headline metric is success, stopping within 1 m of the goal, over ∼325 evaluation episodes per seed and split, five seeds per arm with seeds and episodes shared across arms.

Baselines and statistics. Go-Explore is identical code with $w_{\text{topo}} = 0$, so the difference from TopoExplore is exactly S_{hom}; PPO+RND (Burda et al., 2019b) follows its reference implementation and PPO+ICM pairs ICM's losses (Pathak et al., 2017) with PPO as in Burda et al. (2019a); frontier, random and shortest-path planners are the embodied baselines. Aggregates are medians or interquartile means over seeds with 95% bootstrap intervals (Agarwal et al., 2021), seeds being the unit of replication. Every hyperparameter, its value and its provenance is in Appendix B.4; TopoExplore's were selected once on TopoGym and frozen before any Atari run, while RND uses its authors' values tuned on Montezuma's Revenge (at 32 rather than 128 parallel environments), so the in-domain tuning advantage lies with it. ICM, never run on Atari by its authors, takes RND's PPO settings and a TopoGym-searched intrinsic scale, and its inverse and forward losses were weighted $0.2/0.8$ rather than the published $0.8/0.2$ (Appendix B.4).

5 Results

5.1 Topologically varied worlds

The hold-out benchmark. Table 1 is the cleanest TopoExplore-over-Go-Explore result in the paper. On the roughly 150 worlds where both find the goal, TopoExplore finds it first on two thirds to four fifths of them, with a median speed-up of $1.33\times$ for the plain term and $1.97\times$ with the curvature term (faster on 118 of 149 worlds, slower on 31), the medians of the raw step counts falling from 180–186k to 85–126k. The distribution is right-skewed, so the ratio of means, $2.05\times$, sits above the median. TopoExplore also finds the goal in more worlds: Go-Explore in 152 of 189, every TopoExplore arm in 167 to 169, and the pairing is one-sided: plain TopoExplore gains 16 worlds and loses one ($p = 3 \times 10^{-4}$), and every sign test is below 10^{-3} (all 63 units in Figure 14). PPO and PPO+RND find the goal in 3 and 4 worlds and PPO+ICM in 49 of the 143 it completed (34%, against 80–89% for the archive methods), every run at the full 1M-step budget. The archive methods

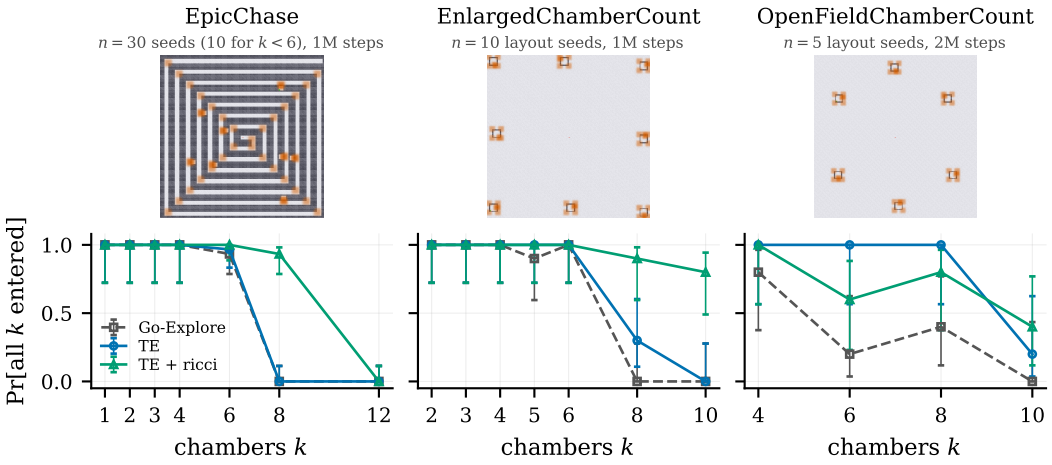


Figure 2: Probability of entering all k chambers within budget, Wilson 95% CIs, with one world of each family above its curve; vermillion marks cells of negative Ollivier–Ricci curvature (doorways and bottlenecks), which the ricci arm rewards. Left to right, the walls become less consequential: a spiral whose corridors dictate the walk, chambers on the perimeter of an open square, and the same chambers with no reachable boundary, where nothing but the archive's own holes marks a chamber and the theory's geometric hypotheses hold. Goal, cost and survival curves and chamber grids in Appendix C.4.

Table 2: Montezuma's Revenge, Phase 1, 60M frames. Best archived trajectory return: median, interquartile mean (IQM) [stratified-bootstrap 95% CI], paired two-sided Wilcoxon against Go-Explore on shared seeds.

Method	n	Median	IQM [95% CI]	paired p vs. Go-Explore
Go-Explore	200	31,400	31,629 [28,456, 34,382]	reference
TopoExplore (plain)	50	27,500	28,173 [23,308, 32,765]	$p = 0.699$ ($n = 50$)
TopoExplore + texture	200	**36,000**	**35,057** [31,125, 38,653]	$p = 0.032$ ($n = 200$)

explore with uniformly random actions from the selected cell, so this compares structured selection with random exploration against a learned policy without selection; learning the explorer used after selection is left to future work.

The k-sweeps. The three chamber-count families test Theorem 2 where its hypotheses hold and show what the local arms add where they hold least. Each places k chambers so that a method that cannot keep returning to distant frontiers enters all k with a probability that drops sharply in k, which Figure 2 shows as a step. On EpicChase the bottleneck is reaching the chambers at all, through a spiral of tight corridors, and there the curvature arm helps: at $k = 8$ Go-Explore and plain TopoExplore enter every chamber in none of 30 seeds and TopoExplore+ricci in 28, since negative curvature marks the doorways along the wall. EnlargedChamberCount, ten random layouts of the same design, gives the same ordering. OpenFieldChamberCount is the family most faithful to the theorem's assumptions: with the wall removed nothing local remains to follow, and plain TopoExplore enters every chamber in every world up to $k = 8$ and finds the goal in 5 of 5 worlds where Go-Explore falls to 0.20–0.40 and the curvature arm to intermediate. Per-k numbers, goal-found and cost curves are in Appendix C.4.

5.2 Montezuma's Revenge: the enclosures that matter are rooms

Figure 3 shows where homology can help on this game. The complex built within a room encloses only platforms and walls, decoys in the sense of Appendix A, since doors and ladders lie on the room boundary, so nothing behind them is enclosed (Appendix C.1); built over the room graph the agent learns from the passages it crosses, it sometimes encloses a room the archive has surrounded. This

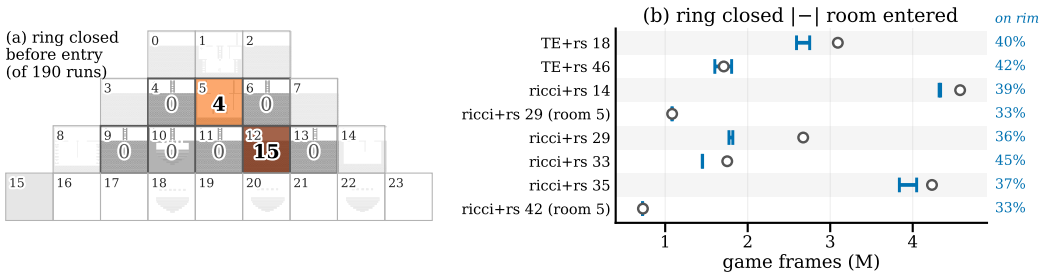


Figure 3: Room-scale enclosures in Montezuma's Revenge. (a) Level 1 on the layout learned from crossed passages; each interior room carries the number of runs, of 190, in which the ring of visited rooms closed around it before it was entered, over a faint screenshot of the room. Faded rooms lie on the pyramid's edge and cannot be enclosed. (b) Each class the room-scale term (rs) detected before its room was entered: the interval from the frame at which the ring closed to the frame at which the room was entered, with the share of selections in that interval that landed on the ring; the open circle is when the same seed entered the room without the term.

Table 3: Explorer trajectories against oracle shortest-path demonstrations as training data for InternVLA-N1-System2 on MultiON (HSSD). Success at 1 m, mean over five training seeds [95% t-interval], 325–327 unseen-scene and 327–329 held-out-category episodes per seed and arm; the best trained arm per column in bold. The oracle+explorer mixture and every paired comparison are in Appendix B.2.

Training data	n	unseen scenes	held-out categories
base model (untrained)	—	0.01	0.01
oracle demonstrations	5	0.11 [0.10, 0.12]	0.12 [0.10, 0.14]
frontier explorer data	5	0.15 [0.12, 0.17]	0.13 [0.10, 0.15]
random-over-candidates explorer data	5	0.13 [0.11, 0.15]	0.12 [0.09, 0.16]
topological explorer data	5	**0.18 [0.15, 0.20]**	0.13 [0.11, 0.15]
top-two-band explorer mixture	5	0.16 [0.13, 0.18]	**0.16 [0.13, 0.18]**

happens mostly for room 12, occasionally for room 5 (Figure 3a). In 7 of the 100 runs with the room-scale term, the agent visited all four rooms around room 12 (or room 5) before entering it, giving the term an enclosed room to act on; 33 to 45% of all selections then landed on the ring and, in five of the six room-12 cases, entered the room sooner than the same seed without it, by 186k to 341k frames in four and by 853k in the fifth, whose two runs had already diverged at room 5 (Figure 3b, Appendix C.1). Since surrounding the two rooms is rare, every arm is even with Go-Explore w.r.t. score at 50 seeds (Table 12).

The texture arm adds features homology cannot see (an adjacent door sprite, key possession) and puts a live class's mass on them (Section 3). Over 200 paired seeds TopoExplore+texture scores higher than Go-Explore (Table 2: paired two-sided Wilcoxon $p = 0.032$, median 36,000 against 31,400, ahead on 113 pairs to 84, paired $d = 0.18$), with two caveats: (1) the selection logs show the gain came mostly through prioritising doors and keys (Appendix C.1), and (2) a Go-Explore+texture control, which would make the attribution exact, has not been run.

Plain TopoExplore is faster early and slightly slower by the end. The term accounts for 21% of selections in the first 200k frames, and at one million frames TopoExplore leads Go-Explore in rooms found by a paired median of two rooms ($p = 0.01$, 28 seeds ahead to 15), reaching the fifth room at 0.69M frames against 0.99M ($p = 0.03$); by one million frames the term's share is about one percent and by two million the lead has faded (Figure 8). Over all 24 rooms of level 1 the term costs little: on the 50 shared seeds plain TopoExplore reaches the last room in 60% of runs against Go-Explore's 62%, at a median 43M frames against 42M where it does, and scores 27,500 against 30,000 ($p = 0.70$, Table 2).

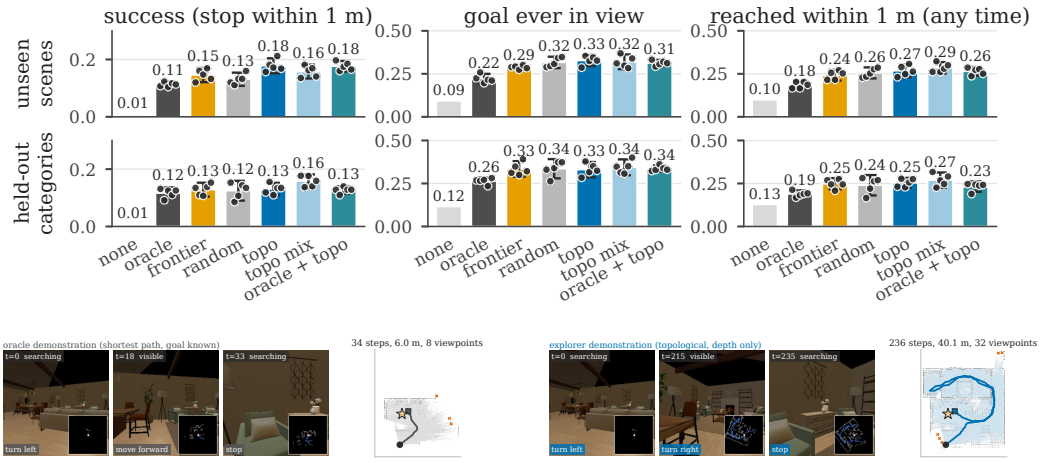


Figure 4: Top: Table 3 with two companion statistics (goal ever in view, within 1 m at any time), rows unseen scenes and held-out categories; bars the seed mean with a 95% t-interval, dots the seeds. Explorer data, alone or mixed, generalises best to unseen scenes; on held-out categories the explorers' data do not separate. Bottom: one episode ("find the flower in vase") demonstrated by the oracle (goal known) and by the explorer (depth only): three frames each with the egocentric map the policy sees and the imitated action, and the top-down path each built (dots landmarks, crosses frontier, star goal).

5.3 Embodied navigation: explorer trajectories as training data

Table 3 and Figure 4 report the pilot. On unseen scenes topological-explorer data beats oracle demonstrations by 6.5 points (0.18 [0.15, 0.20] against 0.11 [0.10, 0.12], $p = 0.001$, ahead on five seeds of five), beats the main comparator, a nearest-frontier explorer on the same scenes and starts, by 3.2 points (0.15, $p = 0.001$, five of five), and beats a uniform draw over its own candidate set by 4.7 points (0.13, $p = 0.02$, five of five), so the gain comes from the mass the homology term puts on the rim targets, not from the targets alone. It also brings the goal into view more often (0.33 against 0.22 for oracle data, 3.9 points over frontier data, $p = 0.02$) and stops within a metre more often per sighting. On held-out categories the explorer arms and the oracle overlap (0.12–0.16), and the 50/50 oracle-explorer mixture matches the explorer arm on both splits. The bottom of Figure 4 shows the mechanism: an oracle demonstration walks straight to a goal the policy cannot yet see, so a policy imitating it never sees a search, while the explorer's demonstration of the same episode sweeps the apartment until the goal appears. Absolute success is low for every trained arm (0.11–0.18 at 1 m with a small vision-language-action model at our compute budget, about ten times the untrained base), so the comparison between arms is the claim; arms share training seeds and evaluation episodes, and every comparison is a two-sided paired t over five seeds (Appendix B.2).

6 Conclusion

An exploration archive has a geometry of its own, and its homology says where the archive has closed around unvisited space without vanishing like a gradient. Used as a selection signal, it reaches the goal first on TopoGym's 189 held-out worlds (median 126k steps against 186k for Go-Explore), and with the curvature arm enters all eight adversarial chambers in 28 of 30 seeds where Go-Explore enters them in none. Where nothing is enclosed it is a bounded cost: on Pitfall!, and on Montezuma's Revenge unless the complex is built over rooms, where it enters encircled rooms sooner but leaves the score unchanged (Appendix C.1). Beyond restorable state spaces, its trajectories train a vision-language-action navigation policy that succeeds 58% more often on unseen scenes than oracle demonstrations do. A natural next step is to combine this global signal with local ones (curiosity, visit counts, Ollivier–Ricci curvature) and to explore from the selected cell with a learned policy rather than random actions.

Reproducibility statement

Every number in the main text is regenerated by one command from the committed summaries in the supplementary code, and every figure from the run trees the same code fetches. Appendix D gives the commands, the data each one reads, where the hyperparameters, proofs, 200-seed test and per-seed rows live, and the cross-architecture determinism check that lets seeds run on different machines be combined.

References

Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron C. Courville, and Marc G. Bellemare. Deep reinforcement learning at the edge of the statistical precipice. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 29304–29320, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/f514cec81cb148559cf475e7426eed5e-Abstract.html.

Anonymous. TopoGym: Environments and benchmarks for topological exploration in reinforcement learning, 2026. Anonymised for review; included in the supplementary material.

Adrià Puigdomènech Badia, Bilal Piot, Steven Kapturowski, Pablo Sprechmann, Alex Vitvitskyi, Zhaohan Daniel Guo, and Charles Blundell. Agent57: Outperforming the Atari human benchmark. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 507–517. PMLR, 2020a. URL http://proceedings.mlr.press/v119/badia20a.html.

Adrià Puigdomènech Badia, Pablo Sprechmann, Alex Vitvitskyi, Zhaohan Daniel Guo, Bilal Piot, Steven Kapturowski, Olivier Tieleman, Martín Arjovsky, Alexander Pritzel, Andrew Bolt, and Charles Blundell. Never give up: Learning directed exploration strategies. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020b. URL https://openreview.net/forum?id=Sye57xStvB.

Marc G. Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The Arcade Learning Environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47:253–279, 2013. doi: 10.1613/JAIR.3912. URL https://doi.org/10.1613/jair.3912.

Marc G. Bellemare, Sriram Srinivasan, Georg Ostrovski, Tom Schaul, David Saxton, and Rémi Munos. Unifying count-based exploration and intrinsic motivation. In Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pp. 1471–1479, 2016. URL https://proceedings.neurips.cc/paper/2016/hash/afda332245e2af431fb7b672a68b659d-Abstract.html.

Subhrajit Bhattacharya, Robert Ghrist, and Vijay Kumar. Persistent homology for path planning in uncertain environments. *IEEE Transactions on Robotics*, 31(3):578–590, 2015. doi: 10.1109/TRO.2015.2412051. URL https://doi.org/10.1109/TRO.2015.2412051.

Karol Borsuk. On the imbedding of systems of compacta in simplicial complexes. *Fundamenta Mathematicae*, 35:217–234, 1948. doi: 10.4064/fm-35-1-217-234.

Yuri Burda, Harri Edwards, Deepak Pathak, Amos J. Storkey, Trevor Darrell, and Alexei A. Efros. Large-scale study of curiosity-driven learning. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019a. URL https://openreview.net/forum?id=rJNwDjAqYX.

Yuri Burda, Harrison Edwards, Amos J. Storkey, and Oleg Klimov. Exploration by random network distillation. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019b. URL https://openreview.net/forum?id=H1lJJnR5Ym.

Gunnar Carlsson. Topology and data. *Bulletin of the American Mathematical Society*, 46(2):255–308, 2009. doi: 10.1090/S0273-0979-09-01249-X.

Mathieu Carrière, Frédéric Chazal, Yuichi Ike, Théo Lacombe, Martin Royer, and Yuhei Umeda. PersLay: A neural network layer for persistence diagrams and new graph topological signatures. In *The 23rd International Conference on Artificial Intelligence and Statistics, AISTATS 2020*, volume 108 of *Proceedings of Machine Learning Research*, pp. 2786–2796. PMLR, 2020.

Vin de Silva and Robert Ghrist. Coverage in sensor networks via persistent homology. *Algebraic & Geometric Topology*, 7(1):339–358, 2007. doi: 10.2140/agt.2007.7.339.

Adrien Ecoffet, Joost Huizinga, Joel Lehman, Kenneth O. Stanley, and Jeff Clune. Go-Explore: a new approach for hard-exploration problems. *CoRR*, abs/1901.10995, 2019. URL http://arxiv.org/abs/1901.10995.

Adrien Ecoffet, Joost Huizinga, Joel Lehman, Kenneth O. Stanley, and Jeff Clune. First return, then explore. *Nature*, 590(7847):580–586, 2021. doi: 10.1038/S41586-020-03157-9. URL https://doi.org/10.1038/s41586-020-03157-9.

Herbert Edelsbrunner. The union of balls and its dual shape. *Discrete & Computational Geometry*, 13(3-4):415–440, 1995. doi: 10.1007/BF02574053. URL https://doi.org/10.1007/BF02574053.

Herbert Edelsbrunner and John L. Harer. *Computational Topology: An Introduction*. American Mathematical Society, Providence, RI, 2010. ISBN 978-0-8218-4925-5.

Benjamin Eysenbach, Abhishek Gupta, Julian Ibarz, and Sergey Levine. Diversity is all you need: Learning skills without a reward function. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. URL https://openreview.net/forum?id=SJx63jRqFm.

Elad Hazan, Sham M. Kakade, Karan Singh, and Abby Van Soest. Provably efficient maximum entropy exploration. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2681–2691. PMLR, 2019. URL http://proceedings.mlr.press/v97/hazan19a.html.

Christoph Hofer, Roland Kwitt, Marc Niethammer, and Andreas Uhl. Deep learning with topological signatures. In *Advances in Neural Information Processing Systems 30, NIPS 2017*, pp. 1634–1644, 2017.

Zhang-Wei Hong, Tao Chen, Yen-Chen Lin, Joni Pajarinen, and Pulkit Agrawal. Topological experience replay. In *The Tenth International Conference on Learning Representations, ICLR 2022*. OpenReview.net, 2022.

Rein Houthooft, Xi Chen, Yan Duan, John Schulman, Filip De Turck, and Pieter Abbeel. VIME: variational information maximizing exploration. In Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pp. 1109–1117, 2016. URL https://proceedings.neurips.cc/paper/2016/hash/abd815286ba1007abfbb8415b83ae2cf-Abstract.html.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022.

InternNav Team. InternVLA-N1: An open dual-system vision-language navigation foundation model with learned latent plans. https://github.com/InternRobotics/InternNav, 2025.

Mukul Khanna, Yongsen Mao, Hanxiao Jiang, Sanjay Haresh, Brennan Shacklett, Dhruv Batra, Alexander Clegg, Eric Undersander, Angel X. Chang, and Manolis Savva. Habitat synthetic scenes dataset (HSSD-200): An analysis of 3D scene scale and realism tradeoffs for ObjectGoal navigation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Soonkyum Kim, Subhrajit Bhattacharya, Robert Ghrist, and Vijay Kumar. Topological exploration of unknown and partially known environments. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2013, Tokyo, Japan, November 3-7, 2013*, pp. 3851–3858. IEEE, 2013. doi: 10.1109/IROS.2013.6696907. URL https://doi.org/10.1109/IROS.2013.6696907.

Joel Lehman and Kenneth O. Stanley. Abandoning objectives: Evolution through the search for novelty alone. *Evolutionary Computation*, 19(2):189–223, 2011. doi: 10.1162/EVCO_A_00025. URL https://doi.org/10.1162/EVCO_a_00025.

Cong Lu, Shengran Hu, and Jeff Clune. Intelligent Go-Explore: Standing on the shoulders of giant foundation models. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=apErWGzCAA. arXiv:2405.15143.

Clément Maria, Jean-Daniel Boissonnat, Marc Glisse, and Mariette Yvinec. The Gudhi library: Simplicial complexes and persistent homology. In Hoon Hong and Chee Yap (eds.), *Mathematical Software - ICMS 2014 - 4th International Congress, Seoul, South Korea, August 5-9, 2014. Proceedings*, volume 8592 of *Lecture Notes in Computer Science*, pp. 167–174. Springer, 2014. doi: 10.1007/978-3-662-44199-2_28. URL https://doi.org/10.1007/978-3-662-44199-2_28.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin A. Riedmiller, Andreas Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015. doi: 10.1038/NATURE14236.

Michael Moor, Max Horn, Bastian Rieck, and Karsten M. Borgwardt. Topological autoencoders. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020*, volume 119 of *Proceedings of Machine Learning Research*, pp. 7045–7054. PMLR, 2020.

Jean-Baptiste Mouret and Jeff Clune. Illuminating search spaces by mapping elites. *CoRR*, abs/1504.04909, 2015. URL http://arxiv.org/abs/1504.04909.

Yann Ollivier. Ricci curvature of Markov chains on metric spaces. *Journal of Functional Analysis*, 256(3):810–864, 2009. doi: 10.1016/j.jfa.2008.11.001.

Georg Ostrovski, Marc G. Bellemare, Aäron van den Oord, and Rémi Munos. Count-based exploration with neural density models. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pp. 2721–2730. PMLR, 2017. URL http://proceedings.mlr.press/v70/ostrovski17a.html.

Deepak Pathak, Pulkit Agrawal, Alexei A. Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pp. 2778–2787. PMLR, 2017. URL http://proceedings.mlr.press/v70/pathak17a.html.

Deepak Pathak, Dhiraj Gandhi, and Abhinav Gupta. Self-supervised exploration via disagreement. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pp. 5062–5071. PMLR, 2019. URL http://proceedings.mlr.press/v97/pathak19a.html.

Bastian Rieck, Matteo Togninalli, Christian Bock, Michael Moor, Max Horn, Thomas Gumbsch, and Karsten M. Borgwardt. Neural persistence: A complexity measure for deep neural networks using algebraic topology. In *7th International Conference on Learning Representations, ICLR 2019*. OpenReview.net, 2019.

Tim Salimans and Richard Chen. Learning Montezuma's Revenge from a single demonstration. *CoRR*, abs/1812.03381, 2018. URL http://arxiv.org/abs/1812.03381.

Nikolay Savinov, Alexey Dosovitskiy, and Vladlen Koltun. Semi-parametric topological memory for navigation. In *6th International Conference on Learning Representations, ICLR 2018*. OpenReview.net, 2018.

Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, Devi Parikh, and Dhruv Batra. Habitat: A platform for embodied AI research. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pp. 9338–9346. IEEE, 2019. doi: 10.1109/ICCV.2019.00943. URL https://doi.org/10.1109/ICCV.2019.00943.

Jürgen Schmidhuber. A possibility for implementing curiosity and boredom in model-building neural controllers. In Jean-Arcady Meyer and Stewart W. Wilson (eds.), *From Animals to Animats: Proceedings of the First International Conference on Simulation of Adaptive Behavior*, pp. 222–227. MIT Press, 1991.

Haoran Tang, Rein Houthooft, Davis Foote, Adam Stooke, Xi Chen, Yan Duan, John Schulman, Filip De Turck, and Pieter Abbeel. #exploration: A study of count-based exploration for deep reinforcement learning. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 2753–2762, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/3a20f62a0af1aa152670bab3c602feed-Abstract.html.

Saim Wani, Shivansh Patel, Unnat Jain, Angel X. Chang, and Manolis Savva. MultiON: Benchmarking semantic map memory using multi-object navigation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Meng Wei, Chenyang Wan, Jiaqi Peng, Xiqian Yu, Yuqiang Yang, Delin Feng, Wenzhe Cai, Chenming Zhu, Tai Wang, Jiangmiao Pang, and Xihui Liu. Ground slow, move fast: A dual-system foundation model for generalizable vision-and-language navigation. In *The Fourteenth International Conference on Learning Representations, ICLR 2026*, 2026. URL https://openreview.net/forum?id=GK4rznYwhn. arXiv:2512.08186.

David Williams. *Probability with Martingales*. Cambridge University Press, Cambridge, 1991. doi: 10.1017/CBO9780511813658. Lévy's extension of the Borel–Cantelli lemmas, Section 12.15.

Brian Yamauchi. A frontier-based approach for autonomous exploration. In *Proceedings 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation CIRA'97 - Towards New Computational Principles for Robotics and Automation, July 10-11, 1997, Monterey, California, USA*, pp. 146–151. IEEE Computer Society, 1997. doi: 10.1109/CIRA.1997.613851. URL https://doi.org/10.1109/CIRA.1997.613851.

A Theory

Appendix A.1 fixes the grid-world model and the objects the analysis is about; Appendix A.2 writes down the two selection rules being compared; Appendix A.3 collects the lemmas; Appendices A.4 to A.6 restate Theorems 1 to 3 from the main text and prove all three; Appendix A.7 says what is not claimed. Everything here concerns the exploration phase alone. Go-Explore explores and then robustifies, and TopoExplore changes only cell selection, so the two methods can differ only in how they explore.

A.1 The model

Grid world. The world is a rectangle $G \subset \mathbb{Z}^2$, finite unless stated otherwise, partitioned into free cells F and obstacle cells O; the agent occupies one free cell at a time. F is 4-connected, and obstacle components are the 4-connected components of O. Dynamics are deterministic with $\mathcal{A} = \{\text{up}, \text{down}, \text{left}, \text{right}\}$: action a in cell c moves to $c + a$ if that cell is free and leaves the agent at c otherwise. A *door* is a free cell with obstacle cells on two opposite sides and free cells on the other two, so every door has width one and passing through one is ordinary movement. We assume O is well-composed, meaning it contains no diagonal pinch: there is no pair of free cells meeting only at a corner point with obstacles on the other diagonal. Well-composedness is what makes the discrete Jordan curve theorem unambiguous and, at the Rips scale fixed below, prevents an edge from leaking through a corner.

Features. Let $W \subseteq O$ be a union of obstacle components, none of which touches the outer boundary of G. For a set of cells S, write $\text{enc}(S)$ for the union of bounded complementary components of S in $\mathbb{Z}^2$. Let $D(W)$ be the set of doors whose adjacent obstacle cells lie in W, and require $W \cup D(W)$ to be a simple ring: 4-connected, with $\text{enc}(W \cup D(W))$ connected. Define the enclosed free region

$$I(W) \;:=\; \big(\text{enc}(W \cup D(W)) \cap F\big) \setminus D(W). \tag{5}$$

W is a *chamber* if $I(W) \neq \emptyset$ and a *decoy* if $I(W) = \emptyset$. A chamber necessarily has $D(W) \neq \emptyset$: W alone enclosing free cells would make them unreachable and contradict 4-connectivity of F, so every enclosure of reachable free space is door-mediated. A decoy is solid, or its enclosed region is entirely obstacle; nothing is inside it and the feature is purely topological. A *pocket* is a transient object of exploration rather than of the environment: a set of unvisited free cells enclosed by visited cells, whose perimeter includes non-door free cells. Decoys are born when circumnavigated and never die; chambers are born when circumnavigated and die when their interior is filled; pockets are born and die during ordinary exploration. The main text calls the region a class encloses, for a chamber, decoy or pocket alike, its enclosed region.

Exploration primitive. Both algorithms share one explorer: from the selected archived cell, take $L \geq 1$ i.i.d. uniform random actions (π_{explore} of Algorithm 1 is uniform in every run of this paper). Write $q_0 := 1/|\mathcal{A}| = 1/4$ for the probability that one step equals a prescribed action. Because the actions are i.i.d. uniform, for any 4-connected path $(c_0, \ldots, c_n)$ of free cells starting at the launch cell, with realising actions $a_1, \ldots, a_n$,

$$\Pr\big[\text{step } k{+}1 \text{ is } a_{k+1} \;\big|\; \text{the first } k \text{ steps traversed } c_0, \ldots, c_k\big] \;=\; q_0 \qquad (0 \leq k < n), \tag{6}$$

the conditioning being vacuous.

Notation. $A_t \subseteq F$ is the archive at iteration t, each cell stored with a returning trajectory rooted at the origin x_0, and $A_0 = \{x_0\}$. $N = |F|$ on a finite grid. $s_{\min}$ and $s_{\max}$ are uniform bounds on the Go-Explore cell score, fixed in Appendix A.2. A *frontier cell* at iteration t is an archived cell with a free neighbour outside A_t. For a cycle γ, a closed 4-connected path of archived cells, $R(\gamma)$ is the bounded region of $\mathbb{Z}^2$ it encloses, well defined by the Jordan curve theorem and unambiguous under well-composedness. The graph distance $d_G(x, y)$ is the number of moves in a shortest 4-connected free path, so $d_G(x, y) \geq \|x - y\|_1 \geq \|x - y\|_2$, with equality on the left exactly when no obstacle forces a detour.

The complex. For a finite cell set P and a scale ε^*, the Vietoris–Rips complex $\text{Cl}_{\varepsilon^*}(P)$ has vertex set P, an edge $\{p, q\}$ whenever the centre distance $d(p, q) \leq \varepsilon^*$, and a filled triangle on every

triple that is pairwise joined. Centre distances near a cell take the values 1, $\sqrt{2}$ and 2, which makes $\varepsilon^* \in [\sqrt{2}, 2)$ the valid range: at $\varepsilon^* \geq \sqrt{2}$ a 2×2 block of free cells is filled by triangles rather than left as a spurious hole, at $\varepsilon^* < 2$ no edge crosses a wall of thickness one, and well-composedness rules out the remaining leak through diagonal corners. We fix $\varepsilon^* \in [\sqrt{2}, 2)$ throughout and write $K_t := \mathrm{Cl}_{\varepsilon^*}(A_t)$. At this scale the clique complex of a well-composed cell set is the filled-squares complex, so the cubical complex the lattice backend builds (Appendix B) has the same H_1 and the same enclosed regions, and the analysis applies to either. The clique complex is here to help with the analysis, not for the actual implementation, which computes the cubical H_1 directly by flood fill.

At this scale $H_1(K_t)$ is computable by flood fill: each enclosed region containing no cell of A_t is one class. Its representative γ is a shortest enclosing loop of cells and its rim is

$$\mathrm{rim}(\gamma) \;:=\; \{\, p \in A_t : p \text{ is 4-adjacent to a cell of } R(\gamma) \setminus A_t \,\}. \tag{7}$$

Rim and representative differ where it matters. Once a door cell is visited but the interior is not, the tightest enclosing cycle passes in front of the door while the true entrance is the visited door cell itself, one step inward: on the rim but on no simple enclosing cycle. Placing the bonus on γ rather than $\mathrm{rim}(\gamma)$ would leave such a chamber with no entrance and prune it while it is still live. We write Γ_t for the set of representatives of $H_1(K_t)$, one per class. As A_t grows, a class is born when its region is first enclosed and dies when the region fills; the interval between the two is its bar.

Return modes. Canonical Go-Explore returns to the selected archived cell either by restoring simulator state or, the dynamics being deterministic, by replaying its stored trajectory. Under *reset-return* (RR) the environment forces a reset to x_0 every T steps and return costs $\ell(c, A_t)$ replay steps, leaving $T - \ell(c, A_t)$ for exploration; every archived cell then lies within graph distance T of the origin, so the effective universe is finite with $|U_T| \leq 2T^2 + 2T + 1$. Under *teleport-return* (TR) the selected cell's state is restored directly, no replay cost is paid, and reach is unbounded, but at most T cells are archived per episode, so deterministically

$$|A_t| \;\leq\; 1 + tT. \tag{8}$$

The default elsewhere is the base model above, used by both methods: replay return with a full L-step rollout and no episode structure, which coincides with (TR) at $T = L$. Each result states the mode it assumes.

A.2 The two selection rules

Go-Explore. The score is Equation (1), with count attributes $\mathrm{attrs} = \{\text{times chosen}, \text{times seen}, \text{times chosen since a new cell}\}$ and neighbour slots $\mathrm{nbrs} = \{\text{left}, \text{right}, \text{up}, \text{down}\}$, where $\mathrm{HasNeighbor}(c, n) = 1$ iff the position $c + n$ is archived. Selection is proportional: $\mathrm{CellProb}(c) = \mathrm{CellScore}(c) / \sum_{c' \in A_{t-1}} \mathrm{CellScore}(c')$. With $n_{\max} := 2w_h + 2w_v$ the maximum total neighbour score, every cell and every time satisfy $1 \leq \mathrm{CellScore}(c) \leq s_{\max}$ with

$$s_{\min} := 1, \qquad s_{\max} := \sum_a \left(w_a \epsilon_1^{-p_a} + \epsilon_2 \right) + n_{\max} + 1, \tag{9}$$

and therefore the *selection floor*

$$P(\text{select } c \mid A_{t-1}) \;\geq\; \frac{s_{\min}}{s_{\max}\, |A_{t-1}|}. \tag{10}$$

Equation (10) is the single fact behind every eventual-coverage claim below.

Gates. The gate of a class γ asks of each rim cell whether some action from it still reaches unexplored interior:

$$g(c, \gamma) := \begin{cases} 1 & \text{if some action takes } c \text{ to an unvisited free cell of } R(\gamma), \\ 0 & \text{otherwise,} \end{cases} \tag{11}$$

$$\mathrm{gated}(\gamma) := \{\, c \in \mathrm{rim}(\gamma) : g(c, \gamma) = 1 \,\}.$$

In *oracle mode* the algorithm selects from $\mathrm{gated}(\gamma)$ directly. In *model-free mode* it cannot evaluate g and instead keeps the attempt count $u(c, \gamma)$, the number of times c has been selected for γ whatever the outcome, weights rim cells by the surrogate $\hat{g}(c, \gamma) = (1 + u(c, \gamma))^{-p_u}$, and retires a cell at $u_{\max} \geq 1$, giving $\mathrm{active}(\gamma)$ as in Equation (2). A class is *live* if $\mathrm{gated}(\gamma) \neq \emptyset$ (oracle) or $\mathrm{active}(\gamma) \neq \emptyset$ (model-free), and *pruned* (reported as sealed in the main text) otherwise. Analyses of either mode may refer to $\mathrm{gated}(\gamma)$, which is ground truth; only the selection rule is mode-specific.

Cycle weights and the TopoExplore score. $b(\gamma) > 0$ is the magnitude of a live class; the default is the inradius of the unvisited part of $R(\gamma)$, equivalently the persistence of the class under the thickening filtration of the archive, since thickening by that radius is exactly what fills the hole. Any positive weight preserves correctness and the choice affects only rates. The score is Equation (4) with S_{hom} as in Equation (3). Throughout the analysis $M(c) \equiv 1$; with a multiplier bounded in $[M_{\min}, M_{\max}]$ every bound below survives with $s_{\min} \to M_{\min} s_{\min}$ and $s_{\max} \to M_{\max} s_{\max}$. We take $p_{\text{topo}} = 1$ throughout, the value used in every experiment (Table 8); for general p_{topo} replace $b(\gamma)$ by $b(\gamma)^{p_{\text{topo}}}$ below. The index set of the sum in Equation (3) depends on the mode: in oracle mode it runs over $c \in \text{gated}(\gamma)$ with $\psi(c,\gamma) = 1/|\text{gated}(\gamma)|$, in model-free mode over $c \in \text{active}(\gamma)$ with $\psi(c,\gamma) = \hat{g}(c,\gamma) / \sum_{c' \in \text{active}(\gamma)} \hat{g}(c',\gamma)$, so in either mode each live class contributes total mass $w_{\text{topo}}\, b(\gamma)$ and $S_{\text{hom}} \equiv 0$ off live classes. Write

$$B_t^{\text{live}} := \sum_{\gamma \text{ live}} b(\gamma), \qquad \Sigma_t^{\text{GE}} := \sum_{c \in A_t} \text{CellScore}(c), \qquad \delta_t := \frac{w_{\text{topo}} B_t^{\text{live}}}{\Sigma_t^{\text{GE}} + w_{\text{topo}} B_t^{\text{live}}}. \tag{12}$$

δ_t is the *realised topological share*. The proportional draw over Score_{TE} can be read as a draw over $2|A_t|$ entries, each cell contributing a Go-Explore entry of mass $\text{CellScore}(c)$ and a topological entry of mass $S_{\text{hom}}(c)$; δ_t is exactly the probability that the draw lands on a topological entry, and the mass of a set of cells is the sum of their masses.

The local term S_{loc} carries no part of the theory. The results below do not require $S_{\text{loc}} \equiv 0$, only a uniform per-cell bound $S_{\text{loc}}(c) \le s_{\text{loc}} < \infty$: the total mass it contributes is then at most $s_{\text{loc}}|A_t|$, the same order as $\Sigma_t^{\text{GE}} \in [\,|A_t|, s_{\max}|A_t|\,]$, so every appearance of $s_{\max}$ below absorbs it through $s_{\max} \mapsto s_{\max} + s_{\text{loc}}$. Floors and ceilings survive with changed constants.

One structural remark on Equation (3). Even with shortest representatives the sum can have more than one term in two dimensions. In a theta configuration, an outer visited ring with a visited strip through the middle enclosing two unvisited pockets, the two classes' representatives are the left and right loops, the strip cells lie on both, and a strip cell with an unvisited neighbour in each pocket is gated for both: it is a doorway to two distinct unexplored regions and receives both contributions. The bounds below depend only on the total live mass $w_{\text{topo}} B_t^{\text{live}}$ and are insensitive to the choice of representatives.

A.3 Machinery

Lemma 1 (Wall-blindness, Go-Explore and TopoExplore)**.** $\text{HasNeighbor}(c, n)$ *tests archive membership of the position* $c + n$*, so a neighbour absent because it is a wall and one absent because it is unexplored free space are indistinguishable. Consequently (i) every wall-adjacent cell retains a neighbour score forever, since wall positions are never archived, and along the rim of a pocket or chamber that score is carried by all* $|\text{rim}(\gamma)|$ *cells alike; and (ii) the neighbour score raises any single cell's score by at most* $n_{\max}$*, a factor* $(1 + n_{\max})$ *over the floor* $s_{\min} = 1$*, on a lattice without levels or keys (*$M \equiv 1$*), as in TopoGym.*

Proof. Both claims use the definition of HasNeighbor in Equation (1). For (i), a wall position is not a free cell and so is never added to the archive, hence $\text{HasNeighbor}(c, n) = 0$ for that slot at every t, and the test returns the same value for an unexplored free neighbour. For (ii), with $M \equiv 1$ and only lattice neighbours, the neighbour sum is at most $\sum_n w_n = 2w_h + 2w_v = n_{\max}$, and $\text{CellScore}(c) \ge 1$ by the $+1$ in Equation (1). □

The implication of Lemma 1 is that nothing in Go-Explore's score singles out a door front. A cell in front of a door and a cell against a blank wall carry the same neighbour score for the same reason.

Lemma 2 (Floor and share)**.** $\text{Score}_{\text{TE}}(c) \ge 1$ *always, and conditional on any history the TopoExplore kernel satisfies*

$$P(\textit{select } c) \;\ge\; \frac{s_{\min}}{s_{\max}^{\text{TE}}(t)\,|A_t|}, \qquad P(\textit{select } c) \;\ge\; (1 - \delta_t)\,\frac{\text{CellScore}(c)}{\Sigma_t^{\text{GE}}}, \tag{13}$$

where $s_{\max}^{\text{TE}}(t) := s_{\max} + \max_c S_{\text{hom}}(c) \le s_{\max} + w_{\text{topo}} B_t^{\text{live}}$.

Proof. $\text{Score}_{\text{TE}}(c) = \text{CellScore}(c) + S_{\text{hom}}(c) + S_{\text{loc}}(c) \geq \text{CellScore}(c) \geq 1$ since both added terms are non-negative. For the first inequality of Equation (13), the numerator is at least $s_{\min} = 1$ and each of the $|A_t|$ denominator terms is at most $s^{\text{TE}}_{\max}(t)$. For the second, split the normaliser as $\Sigma^{\text{GE}}_t + w_{\text{topo}} B^{\text{live}}_t$, so that the Go-Explore entry of cell c alone gives

$$P(\text{select } c) \;\geq\; \frac{\text{CellScore}(c)}{\Sigma^{\text{GE}}_t + w_{\text{topo}} B^{\text{live}}_t} \;=\; \frac{\text{CellScore}(c)}{\Sigma^{\text{GE}}_t} \cdot \frac{\Sigma^{\text{GE}}_t}{\Sigma^{\text{GE}}_t + w_{\text{topo}} B^{\text{live}}_t} \;=\; (1-\delta_t)\,\frac{\text{CellScore}(c)}{\Sigma^{\text{GE}}_t},$$

using the definition of δ_t in Equation (12). □

The second inequality says TopoExplore behaves like Go-Explore damped by $(1-\delta_t)$, and exactly like Go-Explore conditional on the draw landing on non-topological mass.

Lemma 3 (Immediate decoy pruning, oracle mode)**.** *In oracle mode a decoy's class is pruned as soon as it is detected: every rim cell has all interior-side neighbours obstructed and, once the surrounding corridor is visited, no unvisited free neighbour. Decoys therefore absorb zero bonus mass.*

Proof. Detection of a decoy W means the archive encloses W, so every cell of $\text{rim}(\gamma)$ is archived and every cell of $R(\gamma)$ is an obstacle by $I(W) = \emptyset$ in Equation (5). No action from a rim cell reaches an unvisited free cell of $R(\gamma)$, so $g(c, \gamma) = 0$ for every c by Equation (11), $\text{gated}(\gamma) = \emptyset$, the class is pruned, and S_{hom} is zero on it by the convention that S_{hom} vanishes off live classes. □

Lemma 4 (Crossover)**.** *Since* $\Sigma^{\text{GE}}_t \in [\,|A_t|,\ s_{\max}|A_t|\,]$,

$$\delta_t \;\geq\; \frac{w_{\text{topo}} B^{\text{live}}_t}{s_{\max}|A_t| + w_{\text{topo}} B^{\text{live}}_t}, \tag{14}$$

so $\delta_t = \Theta(1)$ *while* $s_{\max}|A_t| \lesssim w_{\text{topo}} B^{\text{live}}_t$ *and* $\delta_t = \Theta\big(w_{\text{topo}} B^{\text{live}}_t / (s_{\max}|A_t|)\big)$ *beyond.*

Proof. Σ^{GE}_t sums $|A_t|$ terms each in $[1, s_{\max}]$ by Equation (9), giving the stated range; substituting the upper end into Equation (12) gives Equation (14), which is decreasing in $s_{\max}|A_t|$ and takes the two stated regimes according to which term dominates the denominator. □

Lemma 4 is the dilution phenomenon on the TopoExplore side. CellScore sums per-attribute terms over the whole archive, so its normaliser grows like $|A_t|$; against a fixed w_{topo} the topological share goes to zero and TopoExplore degenerates to Go-Explore. Past the crossover TopoExplore's advantage at a detected entrance is the multiplicative factor $\Theta(w_{\text{topo}} b_i / s_{\max})$ rather than unbounded, unless $w_{\text{topo}} = \Omega(|A_t|)$. This is why the schedule $w_{\text{topo}}(t) = \alpha|A_t|$, and not a constant, is the default.

Lemma 5 (Eventual full exploration)**.** *In the finite grid of Appendix A.1, both Go-Explore and TopoExplore visit every cell of F in almost surely finite time, and*

$$\mathbb{E}[\tau_{\text{cover}}(\text{GE})] \leq \frac{s_{\max}}{s_{\min}} \cdot \frac{N^2}{q_0}, \qquad \mathbb{E}[\tau_{\text{cover}}(\text{TE})] \leq \frac{s^{\text{TE}}_{\max}}{s_{\min}} \cdot \frac{N^2}{q_0}, \tag{15}$$

with $s^{\text{TE}}_{\max} = s_{\max} + w_{\text{topo}} \sup_t B^{\text{live}}_t$.

Proof. At any iteration with $A_t \neq F$, connectivity of F supplies a frontier cell $c^* \in A_t$. By Equation (10) it is selected with probability at least $s_{\min}/(s_{\max} N)$, and by Equation (13) the same holds for TopoExplore with $s_{\max} \mapsto s^{\text{TE}}_{\max}$. Conditional on selecting c^*, the first exploration step moves into an unvisited neighbour with probability at least q_0 by Equation (6). Hence every iteration with an incomplete archive discovers at least one new cell with probability at least

$$p_0 \;:=\; \frac{q_0\, s_{\min}}{s_{\max} N}, \tag{16}$$

and the same with $s^{\text{TE}}_{\max}$ for TopoExplore.

Let $\mathcal{F}_t$ be the σ-field generated by the first t iterations. Call iteration t a discovery if at least one new cell is archived during it, let $T_0 := 0 < T_1 < \cdots < T_K$ be the discovery iterations, and let

$D_j := T_j - T_{j-1}$. Every discovery adds at least one cell and $|A_0| = 1$, so coverage completes at the K-th discovery with $K \le N - 1$.

Fix j and condition on $\mathcal{F}_{T_{j-1}}$ restricted to $\{j \le K\}$, on which the archive is incomplete throughout epoch j and Equation (16) applies at every step of it. We claim

$$P\big(D_j > m \mid \mathcal{F}_{T_{j-1}}\big) \;\le\; (1-p_0)^m \qquad (m \ge 0), \tag{17}$$

by induction on m. The case $m = 0$ is trivial. For $m \ge 1$, $\{D_j > m\}$ is $\{D_j > m-1\}$ intersected with the event that iteration $T_{j-1} + m$ is not a discovery, and $\{D_j > m-1\} \in \mathcal{F}_{T_{j-1}+m-1}$, so by the tower property

$$\begin{aligned} P\big(D_j > m \mid \mathcal{F}_{T_{j-1}}\big) &= \mathbb{E}\Big[\mathbf{1}\{D_j > m-1\}\, P\big(\text{no discovery at } T_{j-1}{+}m \mid \mathcal{F}_{T_{j-1}+m-1}\big) \;\Big|\; \mathcal{F}_{T_{j-1}}\Big] \\ &\le (1-p_0)\, P\big(D_j > m-1 \mid \mathcal{F}_{T_{j-1}}\big), \end{aligned}$$

the inequality using Equation (16), which is uniform over histories. That uniformity is why no independence between epochs is needed. Summing Equation (17) gives $\mathbb{E}[D_j \mid \mathcal{F}_{T_{j-1}}] = \sum_{m \ge 0} P(D_j > m \mid \mathcal{F}_{T_{j-1}}) \le 1/p_0$ on $\{j \le K\}$.

Now $\tau_{\text{cover}} = \sum_{j=1}^{K} D_j = \sum_{j=1}^{N-1} D_j \mathbf{1}\{j \le K\}$ and $\{j \le K\} = \{A_{T_{j-1}} \ne F\} \in \mathcal{F}_{T_{j-1}}$, so

$$\mathbb{E}[\tau_{\text{cover}}] = \sum_{j=1}^{N-1} \mathbb{E}\Big[\mathbf{1}\{j \le K\}\, \mathbb{E}\big[D_j \mid \mathcal{F}_{T_{j-1}}\big]\Big] \;\le\; \sum_{j=1}^{N-1} \frac{P(j \le K)}{p_0} \;\le\; \frac{N}{p_0} \;=\; \frac{s_{\max}}{s_{\min}} \cdot \frac{N^2}{q_0},$$

which is Equation (15), and identically for TopoExplore. Finiteness of the expectation forces $\tau_{\text{cover}} < \infty$ almost surely. □

The guarantee in Lemma 5 comes from the $+1$ floor and connectivity alone. Every form of shaping, whether counts, neighbours or topology, moves rates and never the guarantee. The N^2 comes from crediting a single frontier cell per iteration at the floor rate $\sim 1/N$ for each of the $\sim N$ discoveries needed; in open regions many cells are frontier, so the bound is loose.

Lemma 6 (Bounded per-class cost, model-free)**.** *Each class γ absorbs at most $|\text{rim}(\gamma)|\, u_{\max}$ bonus selections before it is pruned or its region is entered; for a decoy, pruning is certain.*

Proof. $u(c, \gamma)$ increments on every selection of c for γ regardless of outcome. After $u_{\max}$ selections of a given cell that cell leaves active(γ) by Equation (2). There are $|\text{rim}(\gamma)|$ such cells, so unless the region is entered first, after at most $|\text{rim}(\gamma)|\, u_{\max}$ selections active(γ) is empty and the class is pruned, after which S_{hom} vanishes on it. For a decoy every rim cell fails deterministically, since its interior-side neighbours are obstacles and its exterior-side neighbours are archived, so entry never occurs and pruning is certain. □

Lemma 6 is the model-free analogue of Lemma 3 and is what makes the term safe to leave on: the budget sunk into sealed structure is finite and independent of run length. The Go-Explore component of the score still selects such cells at its usual floor rate; what is bounded is the bonus mass a decoy can absorb.

Lemma 7 (Almost-sure completeness under (TR))**.** *Under teleport-return, for any fixed cell x, in particular any chamber-interior cell, both Go-Explore and TopoExplore archive x in almost surely finite time; consequently $|A_t| \to \infty$ almost surely.*

Proof. Fix a free path $x_0 = y_0, y_1, \ldots, y_k = x$ and let y_i be the last archived cell on it at the current time. Selecting y_i has probability at least $s_{\min}/(s^{\text{TE}}_{\max}|A_t|)$ by Equation (13), and stepping to y_{i+1} has probability at least q_0 by Equation (6). With Equation (8) the conditional probability of progress at episode t is therefore at least

$$\frac{q_0\, s_{\min}}{s^{\text{TE}}_{\max}} \cdot \frac{1}{1+tT}, \qquad \sum_{t \ge 1} \frac{q_0\, s_{\min}}{s^{\text{TE}}_{\max}\,(1+tT)} = \infty.$$

By the second Borel–Cantelli lemma in Lévy's conditional extension (Williams, 1991), which states that $\sum_t P(E_t \mid \mathcal{F}_{t-1}) = \infty$ forces infinitely many E_t to occur almost surely, progress events occur infinitely often, and k of them reach x. □

A.4 Theorem 1: dilution versus anytime entry

Chamber i has enclosing class γ_i, magnitude b_i, and door-entry probability $q_i \geq q_0$ from a gated cell.

Theorem 1 (dilution versus anytime entry, restated). *Conditional on any history $\mathcal{F}_t$: (i) Ceiling, either return mode, and TopoExplore before detection. If chamber i is undiscovered and $M := \max_{c \in A_t} d_G(c, x_0)$, then*

$$P(\textit{enter chamber } i \textit{ next episode} \mid \mathcal{F}_t) \;\leq\; \frac{s_{\max}\, n_D\, h_T}{s_{\min}\, |A_t|} \;\leq\; \frac{C(T)}{M}, \tag{18}$$

where n_D is the number of cells from which the door is reachable within the exploration budget, $h_T \leq 1$ bounds the probability that a rollout from such a cell reaches the door before the budget runs out, and $C(T) := s_{\max} n_D h_T / s_{\min}$. Under space-filling exploration, $|A_t| \geq \kappa M^2$ for a covered fraction $\kappa \in (0, 1]$, the bound improves to $C(T)/(\kappa M^2)$. (ii) Floor, either return mode. If γ_i is detected and its interior unvisited, then

$$\begin{aligned} P(\textit{enter chamber } i \textit{ next episode} \mid \mathcal{F}_t) \;&\geq\; \frac{w_{\text{topo}}\, b_i}{s_{\max}|A_t| + w_{\text{topo}} B_t^{\text{live}}} \cdot \frac{q_i}{|\text{gated}(\gamma_i)|} \\ &=\; \delta_t \cdot \frac{b_i}{B_t^{\text{live}}} \cdot \frac{q_i}{|\text{gated}(\gamma_i)|}, \end{aligned} \tag{19}$$

which is constant in t under $w_{\text{topo}}(t) = \Omega(|A_t|)$. In model-free mode the gated factor is reached by exhaustion rather than by an oracle: each non-gated rim cell leaves active(γ_i) *after $u_{\max}$ draws, so* active$(\gamma_i) \subseteq$ gated(γ_i) *after at most* $|\text{rim}(\gamma_i)|\, u_{\max}$ *rim draws.*

Proof. (i) Entering chamber i during one episode requires selecting one of the n_D cells from which the door is reachable within the budget, and the rollout then reaching the door, which has probability at most h_T. A union bound over those cells gives

$$P(\text{enter chamber } i \mid \mathcal{F}_t) \;\leq\; n_D\, h_T \cdot \max_{c \in A_t} P(\text{select } c),$$

$$P(\text{select } c) = \frac{\text{CellScore}(c)}{\Sigma_t^{\text{GE}}} \;\leq\; \frac{s_{\max}}{s_{\min}\, |A_t|},$$

using CellScore$(c) \leq s_{\max}$ in the numerator and CellScore$(c') \geq s_{\min} = 1$ for each of the $|A_t|$ denominator terms, both from Equation (9). Chaining the two gives the first inequality of Equation (18). For the second, $|A_t| \geq M$: the stored trajectory of the farthest archived cell c visits only archived cells and, being a walk from x_0 to a cell with $d_G(c, x_0) = M$, visits at least $M + 1$ distinct ones. Substituting and writing $C(T) := s_{\max} n_D h_T / s_{\min}$ gives $C(T)/M$, and substituting $|A_t| \geq \kappa M^2$ instead gives $C(T)/(\kappa M^2)$.

For TopoExplore before detection, no live class encloses the chamber, so $S_{\text{hom}} = 0$ on its door front while the normaliser $\Sigma_t^{\text{GE}} + w_{\text{topo}} B_t^{\text{live}}$ only grows, and the same ceiling holds.

(ii) The selection rule factors the entry event into three conditional probabilities. By Equation (12) and $\Sigma_t^{\text{GE}} \leq s_{\max}|A_t|$,

$$P(\text{draw lands on } \gamma_i\text{'s mass} \mid \mathcal{F}_t) = \frac{w_{\text{topo}}\, b_i}{\Sigma_t^{\text{GE}} + w_{\text{topo}} B_t^{\text{live}}} \;\geq\; \frac{w_{\text{topo}}\, b_i}{s_{\max}|A_t| + w_{\text{topo}} B_t^{\text{live}}},$$

$$P(\text{select } c \mid \text{draw on } \gamma_i) = \psi(c, \gamma_i) = \frac{1}{|\text{gated}(\gamma_i)|} \quad \text{(oracle mode)},$$

$$P(\text{enter chamber } i \mid \text{select } c) \geq q_i,$$

and multiplying the three gives Equation (19). For the bound to stay away from zero the factor b_i / B_t^{live} must stay bounded below, which it does: by Lemma 3 a decoy contributes zero to B_t^{live}, and a pocket contributes $b(\gamma)$ only until it is entered, so B_t^{live} is bounded by the total magnitude of the genuine unentered chambers plus transient pocket terms.

In model-free mode the oracle share is replaced by $\psi(c, \gamma_i) = \hat{g}(c, \gamma_i) / \sum_{c' \in \text{active}(\gamma_i)} \hat{g}(c', \gamma_i)$. Before any attempts $\hat{g} \equiv 1$, so $\psi(c, \gamma_i) = 1/|\text{rim}(\gamma_i)|$, and cells with equal counts keep equal weights,

so the allocation over active(γ_i) stays near-uniform as counters climb. Every cell, gated or not, leaves active(γ_i) after exactly $u_{\max}$ selections, so the rim absorbs at most $|\mathrm{rim}(\gamma_i)|\, u_{\max}$ draws in total by Lemma 6; the non-gated cells exhaust after at most $|\mathrm{rim}(\gamma_i) \setminus \mathrm{gated}(\gamma_i)|\, u_{\max}$ of them, and from that point active$(\gamma_i) \subseteq$ gated(γ_i), so the gated share of the normalised weights is exactly 1. Each gated selection enters with probability at least q_i and the door front receives up to $|\mathrm{gated}(\gamma_i)|\, u_{\max}$ selections before it too exhausts, so the class is pruned before first entry with probability at most

$$(1 - q_i)^{u_{\max}} \;\le\; e^{-q_i u_{\max}}. \tag{20}$$

We call this the model-free exhaustion sweep and reuse it in Appendix A.5.

For the behaviour in t: Equation (14) is $\Theta(1)$ while $s_{\max}|A_t| \lesssim w_{\text{topo}} B_t^{\text{live}}$ and decays like $w_{\text{topo}} B_t^{\text{live}}/(s_{\max}|A_t|)$ beyond, and Lemma 7 gives $|A_t| \to \infty$ almost surely. So for fixed w_{topo} the condition eventually fails and Equation (19) becomes $\Theta\big(w_{\text{topo}} b_i q_i/(s_{\max}|A_t|\,|\mathrm{gated}(\gamma_i)|)\big)$, while under $w_{\text{topo}}(t) = \Omega(|A_t|)$ it holds for all t, $\delta_t = \Theta(1)$ uniformly, and Equation (19) is a feature-dependent constant. □

For Go-Explore, and for TopoExplore before detection, no selection mass sits on the hole: by Lemma 1 the door front is just another archived cell among $|A_t|$, so the more the archive grows the less likely any single episode is to enter the chamber. For TopoExplore after detection the class carries mass $w_{\text{topo}} b(\gamma_i)$ concentrated on its $O(1)$ gated cells no matter how large the archive has grown. The only thing that erodes that is the crossover of Lemma 4, which the schedule removes.

A.5 Theorem 2: all-but-k holes

Setup. Finite grid, teleport-return with episode horizon $T = L$. There are k chambers with pairwise-disjoint walls $W_1, \dots, W_k$ and doors $d_1, \dots, d_k$, and the doors are at pairwise graph distance greater than L, so a single rollout enters at most one previously-unentered chamber. Condition both algorithms on the same late-exploration event E_m: every free cell outside $\bigcup_i I(W_i)$ is archived, $m := |A_0| = N - \sum_i |I(W_i)|$, every γ_i has been circumnavigated and is therefore detected and live, and every interior is unvisited. No nesting is assumed.

Theorem 2 (fixed-budget separation, restated). *Model-free TopoExplore with a schedule maintaining $\delta_t \ge \delta_{\min} > 0$, and a gate budget $u_{\max} \ge q_i^{-1} \log(s_{\max}^{\text{TE}} N / s_{\min})$ for every i. Conditional on E_m, with the per-chamber budget*

$$\beta_i := |\mathrm{rim}(\gamma_i)|\, u_{\max} + 2q_i^{-1} + |I(W_i)|\,(1 + u_{\max}), \tag{21}$$

(i) $\mathbb{E}[\tau_{\text{all}}(\text{TE})] \le \delta_{\min}^{-1} \sum_{i=1}^{k} \beta_i$, linear in the total remaining rim size and interior volume and independent of m; (ii) for Go-Explore, every iteration and every unentered chamber satisfy, conditional on the history, $P(\text{enter } i) \le \lambda_0 := s_{\max}\bar{n}/(s_{\min} m)$ with $\bar{n} := \max_i n_i$ and n_i the number of cells from which door d_i is reachable within one rollout, so the number of distinct chambers entered within N iterations is stochastically dominated by Binomial$(N, k\lambda_0)$ *and $P(\text{GE enters all } k \text{ within } N) \le (eN\lambda_0)^k$; (iii) at $N^* := \lceil (2/\delta_{\min}) \sum_i \beta_i \rceil$, TopoExplore enters all k with probability at least $1/2$ while Go-Explore's probability is at most $(eN^*\lambda_0)^k$, exponentially small in k whenever $m \gg \sum_i \beta_i$.*

Proof. (i) Under E_m every γ_i is detected and live at $t = 0$, so $B_t^{\text{live}} = \sum_{i=1}^{k} b_i$ and each draw lands on topological mass with probability $\delta_t \ge \delta_{\min}$ by the schedule assumption. A chamber's bar dies when its interior is filled and not at first entry, so an entered chamber retains positive live mass until its interior is archived. Count the topological draws until all k chambers are entered and classify each into exactly one of three types: X draws select a non-gated rim cell of a still-unentered class, Y draws select a gated cell of a still-unentered class, and Z draws land on the live structure of an already-entered chamber, meaning its rim or the classes its partially-filled interior forms.

For X: every non-door rim cell fails deterministically, since its wall-side neighbours are obstacles and its corridor-side neighbours are archived under E_m, so each such draw increments $u(c, \gamma_i)$ and after $u_{\max}$ failures that cell leaves active(γ_i) by Lemma 6. Class i has $|\mathrm{rim}(\gamma_i)|$ rim cells, so

$$\mathbb{E}[X] \;\le\; \sum_{i=1}^{k} |\mathrm{rim}(\gamma_i)|\, u_{\max}. \tag{22}$$

For Y: a gated draw on class i enters chamber i with probability at least q_i, and Y counts draws on class i only until first entry, so Wald's identity applied per class gives

$$\mathbb{E}[Y] \;\le\; \sum_{i=1}^{k} \frac{1}{q_i}. \tag{23}$$

This accounting presumes class i stays live until first entry. Gated cells also accumulate failures, so the door front can exhaust $u_{\max}$ failures before a success and prune the class; by Equation (20) and the $u_{\max}$ hypothesis this has probability at most $e^{-q_i u_{\max}} \le s_{\min}/(s_{\max}^{\mathrm{TE}} N)$. On that event entry still occurs through the selection floor of Equation (13): a door-front cell is selected with probability at least $s_{\min}/(s_{\max}^{\mathrm{TE}} N)$ per iteration and enters with probability at least q_i, so in expected time at most $s_{\max}^{\mathrm{TE}} N/(s_{\min} q_i)$. The product of the two is at most q_i^{-1} per chamber, the second q_i^{-1} in Equation (21).

For Z: under E_m every free cell outside the interiors is archived, so a type-Z draw on entered chamber i either archives a new interior cell, possible at most $|I(W_i)|$ times, or fails on a cell of an interior pocket rim and increments that cell's counter; those rim cells are interior cells, so at most $|I(W_i)|$ of them absorb at most $u_{\max}$ failures each, giving

$$\mathbb{E}[Z] \;\le\; \sum_{i=1}^{k} |I(W_i)|\,(1 + u_{\max}). \tag{24}$$

Each iteration is a topological draw with probability at least $\delta_{\min}$ conditional on the history, so by Wald's identity again, and combining Equations (22) to (24) with the pruning term,

$$\mathbb{E}[\tau_{\mathrm{all}}(\mathrm{TE})] \;\le\; \frac{\mathbb{E}[X] + \mathbb{E}[Y] + \mathbb{E}[Z]}{\delta_{\min}} + \sum_{i=1}^{k} e^{-q_i u_{\max}} \frac{s_{\max}^{\mathrm{TE}} N}{s_{\min} q_i} \;\le\; \frac{1}{\delta_{\min}} \sum_{i=1}^{k} \beta_i,$$

and no m dependence survives. The gate-budget hypothesis is a convenience that makes β_i clean. The runs use $u_{\max} = 8$, below it; the middle expression then still bounds $\mathbb{E}[\tau_{\mathrm{all}}(\mathrm{TE})]$, with the pruning sum in place of the absorption it bounds, and it is that sum, not m, that the smaller gate budget enlarges.

(ii) This is Equation (18) conditional on E_m, with two additions. The archive only grows, so $|A_t| \ge m$ for all t and the ceiling is uniform in t:

$$P(\text{enter chamber } i \text{ this iteration}) \;\le\; \frac{s_{\max}\, n_i\, h_T}{s_{\min}\, |A_t|} \;\le\; \frac{s_{\max}\, \bar{n}}{s_{\min}\, m} \;=\; \lambda_0,$$

using $h_T \le 1$ and $n_i \le \bar{n}$. And by Lemma 1, under E_m chamber i's door front carries the same neighbour score as every other wall-adjacent cell, so nothing in the score singles it out. A union bound over the at most k unentered chambers gives $P(\text{enter some new chamber this iteration}) \le k\lambda_0$, and the separation assumption caps entries at one per iteration, which gives the binomial domination. For the tail bound, if $\mathrm{Bin}(N, p) \ge k$ then some specific set of k trials all succeeded, so a union bound over the $\binom{N}{k}$ such sets gives

$$P(\mathrm{Bin}(N, p) \ge k) \;\le\; \binom{N}{k} p^k \;\le\; \frac{N^k}{k!} p^k \;\le\; \left(\frac{eNp}{k}\right)^k,$$

where the last step uses $k! \ge (k/e)^k$, itself from $e^k = \sum_{j \ge 0} k^j/j! \ge k^k/k!$. Substituting $p = k\lambda_0$ cancels the k and leaves $(eN\lambda_0)^k$.

(iii) For TopoExplore, part (i) gives $\mathbb{E}[\tau_{\mathrm{all}}(\mathrm{TE})] \le \delta_{\min}^{-1} \sum_i \beta_i \le N^*/2$, so Markov's inequality gives $P(\tau_{\mathrm{all}}(\mathrm{TE}) > N^*) \le \mathbb{E}[\tau_{\mathrm{all}}(\mathrm{TE})]/N^* \le 1/2$. For Go-Explore, substitute N^* into the tail bound of (ii). □

TopoExplore wastes at most $|\mathrm{rim}(\gamma_i)|\, u_{\max}$ failed tries per class, because a constant share $\delta_{\min}$ of all draws lands on the rim and the attempt counters retire each rim cell after $u_{\max}$ draws, concentrating that share onto the door front while it remains active. Entering a chamber begins to reduce its bar, and the remaining interior absorbs at most $|I(W_i)|(1 + u_{\max})$ further draws before the bar dies and its share passes to the classes that are left. Go-Explore has no concentration mechanism: by

Equation (18) each remaining door front holds a $\sim 1/m$ share of attention forever, so at a budget sized to TopoExplore's sweeps Go-Explore must win k independent $1/m$-lotteries. Nesting is unnecessary; dilution alone gives the result.

This is the statement Figure 2 tests. The chamber-count families were built to instantiate its hypotheses: k chambers whose doors sit beyond one episode horizon in a world the agent has otherwise explored, so that m grows while $\sum_i \beta_i$ does not. The predicted shape is a step in k for Go-Explore and a flat curve for an arm that keeps δ_t from decaying, which is what the left and middle panels of Figure 2 show. The right panel is the control: the open-field family removes the wall a curvature signal can follow, and the ordering inverts.

A.6 Theorem 3: certification

Theorem 3 (certification, restated). *Finite grid, model-free TopoExplore. Almost surely there is a finite τ_{cert} at which (i) every chamber and pocket bar has died because its interior is visited, (ii) every decoy class is pruned by gate exhaustion, and (iii) Γ_t consists of exactly the k_d decoy classes. τ_{cert} is observable: stop when no frontier cell and no live class remains. At τ_{cert} the algorithm outputs $H_1 = \mathbb{F}_2^{k_d}$ with a certificate, namely the exhausted gate log for each reported class and the explicit interior filling for each killed one.*

Proof. Lemma 5 gives full coverage almost surely.

(i) The archive only grows, so the complexes form a filtration $K_{t_1} \subseteq K_{t_2} \subseteq \cdots$ and persistent H_1 assigns each class a bar $[t_b, t_d)$, with t_b the first time the class exists and t_d the first time it maps to zero under the inclusions, $t_d = \infty$ if never (Carlsson, 2009). For a pocket, at its birth time t_b the archived cells first contain a closed chain of Rips edges

$$z = \{c_1c_2\} + \{c_2c_3\} + \cdots + \{c_rc_1\}, \qquad [z] \neq 0 \text{ in } H_1(K_{t_b}),$$

winding once around the enclosed unarchived cells. The class is non-zero because no 2-chain of K_{t_b} has boundary z: at $\varepsilon^* < 2$ no edge crosses a wall of thickness one, so every simplex realises inside the union of archived cells and $|K_{t_b}|$ misses the centre p of each enclosed unarchived cell; a filling 2-chain would contract z in $\mathbb{R}^2 \setminus \{p\}$, contradicting winding number 1 about p. At the fill time t_d the enclosed cells are archived and the Rips triangles on them, which at $\varepsilon^* < 2$ are exactly the triples that are mutually 8-adjacent, assemble into a 2-chain c with $\partial c = z$, so $[z] = 0$ in $H_1(K_{t_d})$ and the bar is $[t_b, t_d)$. Which triangulation is immaterial; only that one exists. A chamber is the same computation with enclosed region $W_i \cup I(W_i)$ and t_b the circumnavigation time, with archiving $I(W_i)$ supplying the filling 2-chain. At full coverage every interior is archived, so every chamber and pocket bar has died and no frontier cell remains. Both conditions are functions of the archive and are therefore observable.

(ii) A decoy's enclosed region is obstacle by $I(W) = \emptyset$, so its cells are never archived, the winding obstruction above is permanent, no filling 2-chain ever exists, and the bar is $[t_b, \infty)$. Its class is pruned rather than killed: each rim cell keeps selection probability at least $s_{\min}/(s^{\text{TE}}_{\max}|A_t|) > 0$ at every iteration by Equation (13), so the conditional Borel–Cantelli lemma (Williams, 1991), as in Lemma 7, gives infinitely many selections almost surely. Every one fails, so the counters exhaust after at most $|\text{rim}(\gamma)|\, u_{\max}$ selections of the class by Lemma 6 and the class prunes.

(iii) At the almost surely finite time τ_{cert} when both (i) and (ii) have occurred, Γ_t consists of exactly the k_d decoy classes. The stopping rule is sound: with no frontier cell, no future archive growth is possible, so no new bar can be born and no pruned bar can revive. □

Go-Explore has no analogue. Counts never certify that a rim is exhausted rather than unlucky, because a count that has not produced a discovery is indistinguishable from one that cannot.

A.7 What is not claimed

These are statements about the selection distribution and about the budget a decoy can absorb, conditional on detection and on the gate budget $u_{\max} \geq q_i^{-1} \log(s^{\text{TE}}_{\max} N / s_{\min})$. They do not assert a sample-complexity separation from Go-Explore in general environments, and the empirical sections are not

presented as verifying one. The chamber-count families of Section 5.1 were constructed to satisfy the geometric hypotheses (Section 4). Montezuma's Revenge and the embodied task do not satisfy them: rooms are entered through doors the agent has often already passed, detection is partial, and the episode horizon is not matched to the door geometry. The results there are reported as evidence about the signal in practice, not about the theorems, though they lend the theorems some colour.

B Implementation details

On grids, H_1 of the cubical complex is computed exactly by flood fill, each bounded component of unvisited cells being one class, and a class's persistence under thickening is its inradius, computed by breadth-first search; the embodied explorer's alpha complex is built with GUDHI (Maria et al., 2014). Algorithm 1 states TopoExplore's exploration phase in full.

Algorithm 1 TopoExplore, exploration phase. Starred lines ($\star$) are the additions to Go-Explore.

Input: start cell x_0; rollout length L; Go-Explore score constants (Equation (1)); topological weight α, exponents p_{topo}, p_u, retirement threshold $u_{\max}$, rebuild cadence τ

$A_0 \leftarrow \{x_0\}$ with the empty trajectory; visitation counts and attempt counts $u \leftarrow 0$
for $t = 1, 2, \ldots$ until the step budget is spent **do**
  $\star$ every τ iterations: rebuild K_t on A_{t-1}; read its live classes γ, rims $\mathrm{rim}(\gamma)$, magnitudes $b(\gamma)$ ▷ §3.1
  $\star$ $\mathrm{active}(\gamma) \leftarrow \{c \in \mathrm{rim}(\gamma) : u(c,\gamma) < u_{\max}\}$; prune (seal) γ if $\mathrm{active}(\gamma) = \emptyset$ ▷ Eq. (2)
  $\star$ $w_{\text{topo}} \leftarrow \alpha\,|A_{t-1}|$; compute $S_{\text{hom}}(c)$ for $c \in A_{t-1}$, zero off active rims ▷ Eq. (3)
  select $c \in A_{t-1}$ with probability $\mathrm{Score}_{\text{TE}}(c) \big/ \sum_{c'} \mathrm{Score}_{\text{TE}}(c')$ ▷ Eq. (4)
  return to c: restore its stored state, or walk its trajectory
  for $\ell = 1, \ldots, L$ **do** ▷ explore
    take $a_\ell \sim \pi_{\text{explore}}$; observe cell x; **if** $x \notin A$ **then** add x with its trajectory
  end for
  update visitation counts; replace stored trajectories by shorter or higher-scoring ones
  $\star$ $u(c,\gamma)$ += 1 for every live γ with $c \in \mathrm{active}(\gamma)$ ▷ regardless of outcome
end for
Phase 2: robustify the best archived trajectories into a policy (Backward Algorithm) ▷ not analysed here

B.1 Backends

Lattice (Atari). Cells carry an integer site within a room (one lattice per level–room pair). Per room we build the cubical complex of visited sites over a walled base and read H_1 classes as {cycle, rim, pocket} triples; class magnitude is the enclosed region's inradius, computed as the maximum BFS depth from its boundary. Class identity across passes is (level–room pair, minimum pocket site), which is stable while any of the pocket survives, so gate counters persist across recomputation.

The rim in each backend. Appendix A defines $\mathrm{rim}(\gamma)$ as the archived cells 4-adjacent to an unvisited cell of $R(\gamma)$, walls included. The TopoGym runs narrow it to the archived cells 4-adjacent to unvisited *free* cells the agent has already seen, for every arm: a wall-facing cell carries no bonus, a chamber has no rim until some of its interior has been seen, so the entry guarantees of Appendix A apply from that point on, and a decoy, which has no free interior, never receives the bonus rather than absorbing up to $|\mathrm{rim}(\gamma)|\,u_{\max}$ draws before it is pruned. The lattice backend on Atari widens it to every archived cell 8-adjacent to the enclosed region, adding the cells that touch it only at a corner; the counting bounds of Appendix A then hold with $|\mathrm{rim}(\gamma)|$ replaced by the size of that shell, which still contains every true entrance. The embodied backend places the bonus on the landmarks within the rim support radius of Table 8.

B.2 Embodied adaptation: walking instead of restoring

The embodied explorer of Section 5.3 keeps Equation (4)'s selection rule and the gate of Section 3.2 and changes what they operate on. The Atari and TopoGym versions rest on an archive of restorable states: a selected cell is returned to by snapshot restore, so travel is free and the archive persists for the whole run. A navigation policy has neither. The adaptation below is what the pilot ran; its reference planners (nearest-frontier, random walk, and the shortest-path oracle that produced the demonstration arm) share the same episode loop and action space, and Figure 6 shows all four on one



Figure 5: What the embodied explorer sees at its own decisions, five HSSD scenes. Top: the frame at which it arrives at a hole-rim probe, the topological term's target, with the 64-px egocentric map the policy is later trained on in the corner. Bottom: the first frame of the same episode in which the goal object is in view.

episode. Evaluation counts behind Table 3: 325–327 unseen-scene and 327–329 held-out-category episodes per seed for every trained arm; the untrained base on one evaluation of 325 and 328.

The map replaces the archive. Each episode starts a fresh map and nothing persists across episodes. Every second step the depth frame (256^2, 79° HFOV, range 5 m, hits above 2.2 m discarded) is unprojected into surface hits, and the map is a δ-net of those hits with $\delta = 0.2$ m, each landmark carrying its sensor-facing normal and its acquisition time. Rays that returned nothing within range contribute their far ends to a *frontier*: a 0.5 m net of open-ray ends that must snap to the navmesh and retire once a sweep passes within 1 m of them.

The complex and its classes. The coverage complex is the subcomplex of the alpha complex (Edelsbrunner, 1995) of the landmarks at radius 2δ, built with GUDHI, that keeps only simplices whose vertex normals agree, so that the two sides of a wall never merge; the balls it is built from cover the witnessed surface at the map's resolution. Filtration is by acquisition time. The classes the planner acts on are the *rims* of unacquired regions: boundary landmarks whose tangent-plane neighbourhood has an unseen side, clustered one rim per region and given a stable identity across passes by a registry. Indoors a rim is a doorway into an unseen room, the edge of an occlusion shadow behind furniture, or the edge of what the depth range reached. A pass runs every 20 steps and on every arrival.

Candidates and supports. A candidate is a live frontier point, a rim's *probe* (a navigable point at its mouth, snapped to the navmesh), or, failing both, a pose the rim was first seen from (a re-sweep view). A rim's support is its frontier points within 1.5 m, its probe, or its re-sweep views, in that order. The score is Equation (4) with the count term $w_a(1+u)^{-p_a}$ ($w_a = 0.3$, $p_a = 0.5$) over how often the candidate was chosen, a frontier term of constant weight 1, and the homology term $w_{\text{topo}} \sum_{\gamma\,:\,c\in\text{active}(\gamma)} (\text{perimeter}(\gamma)/\delta)\,\psi(c,\gamma)$ with ψ the gate's normalised share over the rim's active support. w_{topo} is the constant 1 rather than the archive-size schedule of the grid version, because candidate counts indoors are small. The gate is unchanged: u is incremented on every selection of the pair, the share is weighted by $(1+u)^{-p_u}$ with $p_u = 1$, a candidate retires at $u_{\max} = 6$, a rim whose support is fully retired is pruned and never re-nominated, and rim progress resets its counters.

Return is walking. The draw is proportional to score times $\exp(-\lambda d)$ with d the geodesic distance to the candidate; the pilot's arm used $\lambda = 0$, so selection is the proportional rule of the grid version and walking distance is not penalised. The selected candidate is reached by a greedy geodesic follower in at most 150 primitive steps (0.25 m forward, 30° turns), every one of which is executed and recorded in the policy's own action space, so the trajectory the policy later imitates is exactly what the explorer did. On arrival the explorer takes four random moves, senses, and runs a pass; a frontier

candidate retires once visited and a probe once reached. When no candidate exists it takes a burst of ten random primitives so that the episode can still end. Episodes are capped at 500 steps.

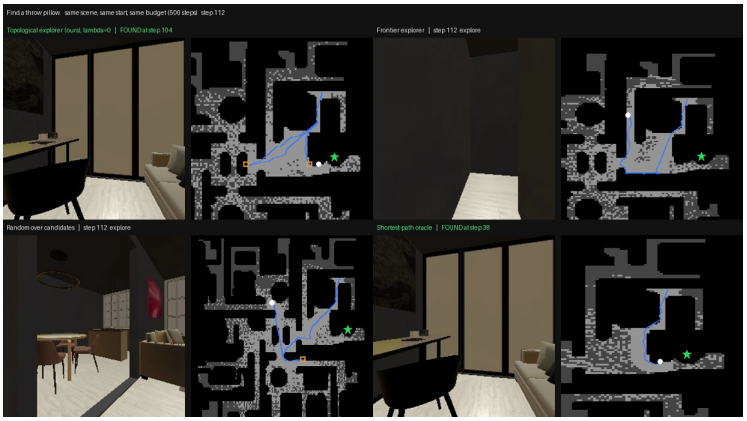


Figure 6: The four explorers that generated the training pools on one episode of the fixed simulator, same scene, start, and 500-step budget: camera and top-down map per explorer, goal starred. Find a throw pillow: the topological explorer (top left) has it in view at step 104 and the shortest-path oracle (bottom right) at step 38; the frontier and random explorers have not found it by the end of the budget.

Approach and stop. Once the goal is visible within 4 m the episode switches to an approach phase: the follower walks to the reachable goal view point nearest the object and calls stop. This phase uses the episode's goal view points, which the oracle demonstrations use throughout; the exploration phase, which is the part the topological signal controls and the part that differs between arms, uses depth alone. Success is the ObjectNav rule (stop within 1 m geodesic of a goal view point) and is what Table 3 reports for the fine-tuned policies, which drive the same primitive interface at evaluation.

Comparator arms. The six training sets, all trained with the recipe of Table 8 and a stop cue in the answer format, are: 455 oracle shortest-path demonstrations; 352 topological-explorer episodes; the two comparison explorers below; a 953-episode mixture of the top two quality bands over eleven explorer variants; and a 642-episode 50/50 mixture of oracle and explorer episodes, the mixtures budgeted to the topological arm's imitated steps. The arms are the successful non-holdout episodes of two other explorers from the same trajectory pool as the topological arm, same scenes, starts, explorer seed and 500-step budget: the nearest-frontier explorer (292 episodes, 54.8k imitated steps) and a uniform draw over the topological planner's candidate set of frontier points and rim probes (327 episodes, 45.5k steps), against the topological arm's 352 episodes and 54.7k steps; training recipe, seeds 0–4 and evaluation sets as for every other arm. Primary test: success at 1 m on unseen scenes, topological minus frontier, paired by training seed, two-sided paired t over five seeds at $\alpha = 0.05$. Secondary, same rule: topological minus random, the held-out-category split, and goal ever in view. Decision rule: both rows enter the main table whatever the sign, and the 50/50 oracle+explorer mixture moves to this appendix with its evaluation extended to the same counts, which it now has (Table 4). Table 5 lists every paired comparison; the per-seed values behind them are in `figures/embodied/pilot_per_seed.csv` and the tests in `pilot_paired_tests.csv`, both written by `analysis/embodied/pilot_table.py`.

Table 4: Every training set of the embodied pilot, including the oracle+explorer mixture. Success at 1 m, mean over five training seeds [95% t-interval], 325–327 unseen-scene and 327–329 held-out-category episodes per seed and arm; the best trained arm per column in bold.

Training data	n	unseen scenes	held-out categories
base model (untrained)	—	0.01	0.01
oracle demonstrations	5	0.11 [0.10, 0.12]	0.12 [0.10, 0.14]
frontier explorer data	5	0.15 [0.12, 0.17]	0.13 [0.10, 0.15]
random-over-candidates explorer data	5	0.13 [0.11, 0.15]	0.12 [0.09, 0.16]
topological explorer data	5	**0.18 [0.15, 0.20]**	0.13 [0.11, 0.15]
top-two-band explorer mixture	5	0.16 [0.13, 0.18]	**0.16 [0.13, 0.18]**
oracle + explorer, 50/50	5	0.18 [0.16, 0.19]	0.13 [0.11, 0.14]

Table 5: Paired-by-seed comparisons between training sets on success at 1 m (first named minus second): mean difference over the five shared training seeds with its 95% t-interval, the two-sided paired t-test p-value, and the number of seeds on which the first arm is ahead. The first row is the primary comparison.

Comparison	split	difference [95% CI]	p (seeds ahead)
topological − frontier	unseen scenes	+0.032 [+0.021, +0.044]	0.001 (5/5)
	held-out categories	+0.002 [-0.030, +0.034]	0.866 (2/5)
topological − random	unseen scenes	+0.047 [+0.011, +0.083]	0.022 (5/5)
	held-out categories	+0.006 [-0.042, +0.054]	0.757 (2/5)
topological − oracle	unseen scenes	+0.065 [+0.043, +0.088]	0.001 (5/5)
	held-out categories	+0.014 [-0.026, +0.054]	0.386 (3/5)
frontier − oracle	unseen scenes	+0.033 [+0.012, +0.054]	0.012 (5/5)
	held-out categories	+0.012 [-0.012, +0.036]	0.237 (4/5)
frontier − random	unseen scenes	+0.015 [-0.023, +0.053]	0.339 (3/5)
	held-out categories	+0.004 [-0.018, +0.026]	0.666 (2/5)
oracle + explorer 50/50 − topological	unseen scenes	-0.002 [-0.035, +0.031]	0.884 (2/5)
	held-out categories	-0.004 [-0.038, +0.031]	0.782 (2/5)

Table 6: The same paired comparisons on whether the goal ever came into view.

Comparison	split	difference [95% CI]	p (seeds ahead)
topological − frontier	unseen scenes	+0.039 [+0.009, +0.070]	0.024 (4/5)
	held-out categories	-0.004 [-0.041, +0.033]	0.775 (3/5)
topological − random	unseen scenes	+0.011 [-0.026, +0.048]	0.452 (3/5)
	held-out categories	-0.005 [-0.090, +0.080]	0.871 (1/5)
topological − oracle	unseen scenes	+0.106 [+0.071, +0.142]	0.001 (5/5)
	held-out categories	+0.066 [+0.010, +0.123]	0.030 (5/5)
frontier − oracle	unseen scenes	+0.067 [+0.042, +0.092]	0.002 (5/5)
	held-out categories	+0.071 [+0.006, +0.136]	0.039 (5/5)
frontier − random	unseen scenes	-0.028 [-0.062, +0.005]	0.079 (0/5)
	held-out categories	-0.001 [-0.070, +0.068]	0.963 (2/5)
oracle + explorer 50/50 − topological	unseen scenes	-0.015 [-0.063, +0.033]	0.443 (1/5)
	held-out categories	+0.006 [-0.042, +0.054]	0.744 (3/5)

Holes in the viewpoint sets of the training trajectories. The viewpoint set of a trajectory is the set of positions it sensed from (every second step and every arrival, one point per 0.25 m cell). Its holes are loops of viewpoints that enclose floor the trajectory never stood on: the H_1 classes of the alpha filtration over the two-dimensional viewpoint set, a class born at radius b and dying at radius d being a hole whose loop closes once its viewpoints are within $2b$ of one another and which is filled by discs of radius d, so d is the radius of the largest empty disc it encloses. Classes with $d - b < 0.1$ m, the gap between two consecutive poses, are not counted. Figure 7 and Table 7 compare the goal-reaching trajectories of the topological and frontier explorers, the trajectories the two arms were

trained on, on rliable's interquartile mean over trajectories with a 95% stratified-bootstrap interval (Agarwal et al., 2021). Topological trajectories leave fewer and smaller holes on every statistic: the probability that a frontier trajectory has more than a topological one is 0.54 to 0.55 with intervals above 0.5 throughout, holes of radius at least 1 m occur in 22% of topological trajectories against 29% of frontier ones (interquartile means 0.00 and 0.08), and the enclosed hole area is 2% of the viewpoint set's hull for topological trajectories against 5% for frontier. The difference is not one of range: frontier trajectories reach farther (extent radius 3.5 m against 3.1) and take longer (173 steps against 117), and the gap persists per 100 steps. On trajectories that failed to reach the goal, all at the 500-step cap, the three explorers do not separate (2 to 3 holes each), so the difference is in how the goal is reached, not in how much ground is covered.

The random-over-candidates explorer shows as few holes as the topological one (0.33 against 0.37, hole area 2% of the hull) and yet its data trains the weakest explorer policy (Table 4), for a reason the pool makes visible. Random succeeds on 73% of pool episodes against 79% for the topological explorer, the gap being on far goals (0.52 against 0.66), and the episodes it does finish are the shorter and more direct ones (105 steps, fewest decisions): conditioning on success selects the trajectories in which the uniform draw happened to send it toward the goal early. That tidiness is luck rather than a strategy, since nothing in the observation the policy conditions on produced it, and unconditionally random's trajectories have the same number of holes as topological ones (0.72 against 0.70 over all episodes). A rim probe, by contrast, is a target the map itself defines, so the same egocentric map tends to produce the same move and the policy sees a consistent mapping from map to action. The evaluation rows agree: policies trained on random data sight the goal and come within a metre of it as often as topologically trained ones (0.23 to 0.29 against 0.23 to 0.31 on unseen scenes) and then fail to stop (stop rate 0.60 to 0.70 against 0.70 to 0.82). Table 7 is computed by `analysis/embodied/viewpoint_set_figure.py` from the per-trajectory CSV `figures/embodied/viewpoint_set_holes.csv`, which the source repository's `scripts/viewpoint_set_holes.py` writes from the recorded poses of every pool episode.

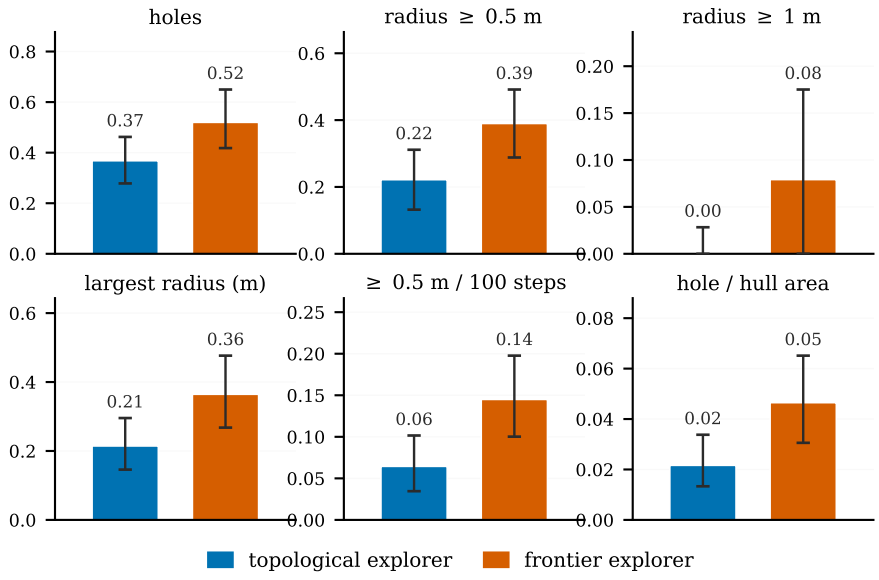


Figure 7: Holes in the viewpoint sets of goal-reaching training trajectories, topological against frontier explorer: interquartile mean over trajectories with a 95% stratified-bootstrap interval (422 and 353 trajectories). Holes are H_1 classes of the alpha filtration over the viewpoint set with persistence at least 0.1 m; a hole's radius is that of the largest empty disc it encloses; hole area is the sum of those discs over the area of the viewpoint set's convex hull.

Table 7: The statistics of Figure 7 for all four training sets (goal-reaching trajectories, interquartile mean [95% stratified-bootstrap interval]), and the size of the trajectories.

per goal-reaching trajectory	topological (422)	frontier (353)	random (391)	oracle (536)
holes	0.37 [0.28, 0.46]	0.52 [0.42, 0.65]	0.33 [0.23, 0.43]	0.00
holes of radius ≥ 0.5 m	0.22 [0.13, 0.31]	0.39 [0.29, 0.49]	0.17 [0.08, 0.27]	0.00
holes of radius ≥ 1 m	0.00 [0.00, 0.03]	0.08 [0.00, 0.18]	0.00 [0.00, 0.02]	0.00
largest hole radius (m)	0.21 [0.15, 0.30]	0.36 [0.27, 0.48]	0.19 [0.11, 0.27]	0.00
holes ≥ 0.5 m per 100 steps	0.06 [0.03, 0.10]	0.14 [0.10, 0.20]	0.06 [0.02, 0.10]	0.00
hole area / hull area	0.02 [0.01, 0.03]	0.05 [0.03, 0.07]	0.02 [0.01, 0.03]	0.00
viewpoints	42	48	38	18
steps	117	173	105	39
extent radius (m)	3.05	3.50	3.07	1.94

B.3 Per-game specifics

Montezuma's Revenge: cells $(\text{level}, \text{room}, \text{keys}, x/8, y/16)$ from RAM, latched across the black transition frames where room and position update at different times; termination on life loss; Go-Explore's level multiplier and more-keys neighbour term. Pitfall!: cells $(\text{room}, x/8, y/16)$ with room from the byte of the game's linear-feedback shift register (LFSR) that generates the world; *no* termination on life loss (death is a usable left-edge teleport); rewards scaled $\times 10^{-3}$ and never clipped in the learned baselines.

B.4 Hyperparameters

Table 8 lists every hyperparameter behind a reported number, its final value, and its provenance. Three sources recur. *Search*: a staged Optuna search of TopoExplore's knobs on TopoGym's `tune` split, run once and frozen before any Atari run; the same winners were selected on both TopoGym suites, which is what licensed freezing them. *Go-Explore*: the per-game domain-knowledge cell-score weights of Ecoffet et al. (2019), Appendix A.6, Table 2, used verbatim by both Go-Explore and TopoExplore so both methods run with identical Atari settings. *Reference*: a baseline's own paper. Values marked *default* are implementation defaults never tuned; *hand-set* values are ours, with the reason given.

Table 8: Hyperparameters, final values, and provenance.

Hyperparameter	Value	Source
Archive methods on Atari (Go-Explore and every TopoExplore arm)		
Budget per run	15M agent steps (60M frames)	matched to Go-Explore's level-1 solve with domain knowledge (57.6M frames)
Frameskip; sticky actions; episode cap	4; off (deterministic); 100k steps	Go-Explore Phase 1; cap hand-set (Go-Explore ends episodes on life loss or game over)
Explore rollout L	100 steps	Go-Explore
Cell (Montezuma)	(level, room, keys, $x/8$, $y/16$)	Go-Explore domain-knowledge cells
Cell (Pitfall!)	(room byte, $x/8$, $y/16$)	Go-Explore, room from the LFSR byte
Count exponent p_a; ϵ_1, ϵ_2	0.5; 10^{-3}, 10^{-5}	Go-Explore
Montezuma count / neighbour weights	0, 0, 0 / 0.3 (horiz.), 0.1 (vert.), 10 (more keys)	Go-Explore App. A.6, Table 2
Montezuma level weight $M(c)$	$0.1^{\,\text{max level} - \text{level}(c)}$	Go-Explore
Pitfall! count / neighbour weights	1.0, 0, 0.5 / 1.0, 0	Go-Explore App. A.6, Table 2
The topological term (all TopoExplore arms, all domains)		
Schedule gain α in $w_{\text{topo}}(t) = \alpha \lvert A_t \rvert$	4.597	search

continued on the next page

Hyperparameter	Value	Source
Magnitude exponent p_{topo}	1	default
Gate decay p_u; retirement $u_{\max}$	0.293; 8	search
Class magnitude $b(\gamma)$	inradius (lattice); persistence (point cloud)	Section 3
Rebuild cadence τ; region	every iteration; one room	default; Appendix B
Local-signal arms		
w_{ricci}	115.98	search
Texture slot weights (TopoGym-canonical)	door 19.91, hallway 0.559, wormhole 1.851, drop-adjacent 1.851, water 0.368, ladder 0.057, bridge 0.056	search
Montezuma slots	`has_key` 1.0; `door_adjacent`, `openable_door` = door; `skull_adjacent` = drop-adjacent	adaptor default; mapped to the canonical slots
Montezuma door detector	colour (232,204,99), ≥ 12 px/column, ≥ 3 columns; near = 14 px; skull near = 24 px	hand-set from the sprite
Pitfall! slot	`underground` 0.5	adaptor default, untuned
TopoGym hold-out and chamber sweeps (archive arms)		
Budget per world	1M steps (2M on OpenFieldChamberCount)	benchmark
GE base w_a, p_a, w_n	0.3, 0.5, 3.0	search (inherited and confirmed)
TopoExplore knobs and arm weights	as above	search
TopoGym gradient baselines (RLlib PPO)		
Budget; early stopping	1M steps per world; none	benchmark; Appendix D
Train batch; env runners	4,000; 2×2 envs	default
PPO: lr; entropy coefficient	3×10^{-4}; 0.01	grid on `tune` split: lr $\{10^{-4}, 3 \times 10^{-4}\}$ $\times$ entropy $\{10^{-3}, 10^{-2}\}$
PPO+RND: lr; intrinsic scale	3×10^{-4}; 0.05	grid on `tune` split: lr $\times$ scale $\{0.05, 0.5\}$
PPO+ICM: lr; η; β; feature dim	3×10^{-4}; 0.5; 0.2; 64	standard values, not searched: lr as PPO/RND chose; η unstated in Pathak et al. (2017), 0.5 beat 0.05 on one single-layout search; β reference; dim default
Atari gradient baselines (PPO+RND, PPO+ICM)		
Budget	25M agent steps (100M frames)	sum of both TopoExplore phases
Observation	84×84 grey, 4-frame stack, sticky $p = 0.25$, no no-ops	reference (Burda et al., 2019b)
Envs; rollout; epochs; minibatches	32; 128; 4; 4	reference, except envs: RND's preliminary 32 (its default is 128)
lr; clip; entropy; value coef.	10^{-4}; 0.1; 0.001; 0.5	reference; value coef. as the reference code's 0.5·MSE
γ_{ext}, γ_{int}, λ; advantage coefs.	0.999, 0.99, 0.95; 2.0 ext, 1.0 int	reference
RND predictor update proportion	0.25	RND's value for 128 envs; its code uses 1.0 at 32 (deviation)
ICM β; η	0.2; 0.5	β of Pathak et al. (2017), but applied to the inverse loss (losses weighted 0.2/0.8, not 0.8/0.2); η as TopoGym
Observation-normalisation warm-up	50 steps	hand-set (RND's code: 6,400)
Embodied explorer (MultiON on HSSD)		

continued on the next page

Hyperparameter	Value	Source
Depth sensor	256^2, 79° HFOV, 5 m range, hits above 2.2 m dropped	simulator config
Sensing / topology cadence	every 2 / every 20 steps	hand-set
Landmark net δ; frontier net	0.2 m; 0.5 m	hand-set
Rim support radius; walk cap; episode cap	1.5 m; 150 steps; 500 steps	hand-set
Count term w_a, p_a; frontier weight	0.3, 0.5; 1	as TopoGym base
w_{topo}; p_u; $u_{\max}$	1 (constant); 1; 6	hand-set (few candidates indoors)
Distance discount λ	0	pilot arm
VLA fine-tuning		
Base model	InternVLA-N1-System2 (Qwen2.5-VL-7B)	InternNav Team (2025)
LoRA rank; α; dropout	16; 32; 0.05	default
Optimiser; lr; schedule; weight decay	AdamW; 10^{-4}; cosine to zero after 100 linear warm-up steps; 0	hand-set
Steps; batch	3,000; 4	hand-set (48 GB card)
History; image size; action chunk	4 frames; 224; 4	hand-set

C Extended results

C.1 Montezuma's Revenge: the 200-seed test and what each arm does

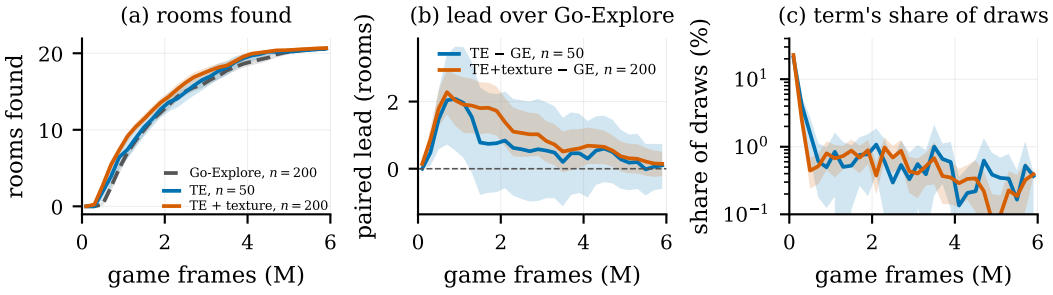


Figure 8: Where the term acts on Montezuma's Revenge, all Phase-1 seeds. (a) Rooms found against game frames, IQM with a 95% bootstrap band. (b) Paired lead over Go-Explore in rooms found, mean with a 95% band. (c) Share of draws the homology term carries, per 200k-frame bin, log scale.

The 200-seed test. *Hypothesis:* TopoExplore+texture achieves a higher best-trajectory score than Go-Explore at matched budget (15M steps, 60M frames, Phase 1). *Design:* arms Go-Explore and TopoExplore+texture at $n = 200$ seeds each, the same tooling, budget and stride throughout. *Metric:* best archived trajectory return. *Test:* paired Wilcoxon signed-rank on the 200 shared seeds, two-sided, $\alpha = 0.05$; secondary, rliable IQM with a stratified bootstrap 95% CI and Cohen's d on the paired differences. Power is about 95% at $d = 0.26$ and 80% at $d = 0.20$. *Outcome:* $W = 8029$, $p = 0.032$, paired $d = 0.18$, texture higher on 113 pairs, Go-Explore on 84, 3 ties; IQMs in Table 2. Plain TopoExplore, ricci and both were not taken past 50 seeds: they are null there (Table 9) and nothing in their mechanism predicts otherwise on this game.

Where the texture signal comes from. The adaptor emits per-cell slots from the live frame and RAM: key possession (`has_key`, from the inventory bits), and door adjacency, which is *seen* rather than read: the detector matches the door sprite's colour in the frame, requires at least 12 pixels in a column and three adjacent columns, which separates a door from a key and from the one-column rope, and since the sprite disappears when a door opens the same detector observes the door's state, so an opened door stops being an entrance without any rule saying so. `openable_door` is door adjacency while holding a key. The arm acts in two ways. Each slot adds a staleness-weighted bonus to the cell's score (Appendix B; the door slots carry the TopoGym-tuned weight of about 20, key possession 1).

Table 9: Every Montezuma arm, same statistics as Table 2; the local-signal arms were not extended past the 50-seed wave.

Method	n	Median	IQM [95% CI]	paired p vs. Go-Explore
Go-Explore	200	31,400	31,629 [28,456, 34,382]	reference
TopoExplore (plain)	50	27,500	28,173 [23,308, 32,765]	$p = 0.699$ ($n = 50$)
TopoExplore + texture	200	**36,000**	**35,057** [31,125, 38,653]	$p = 0.032$ ($n = 200$)
TopoExplore + ricci	49	28,500	26,892 [19,744, 34,820]	$p = 0.739$ ($n = 49$)
TopoExplore + both	44	27,000	26,141 [20,818, 32,946]	$p = 0.731$ ($n = 44$)

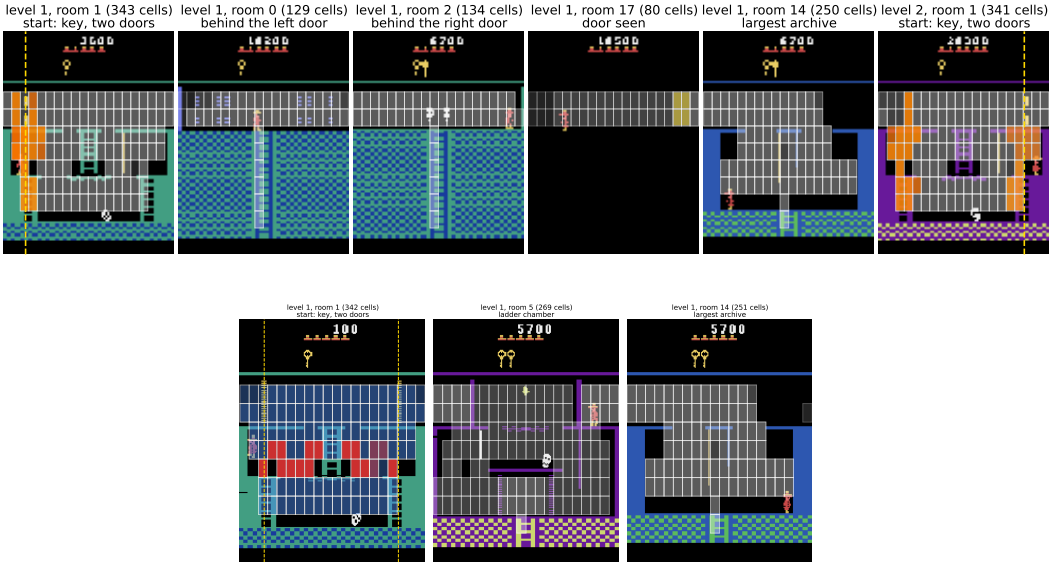


Figure 9: Inside the archive on Montezuma's Revenge. Each frame is an archived state re-rendered by replaying its stored action string; boxes are the archived cells of that room in screen coordinates (x = RAM[42], y from RAM[43]). Top, a TopoExplore+texture run: cells carrying `door_adjacent` (yellow) and `openable_door` (orange), dashed lines the door columns the detector found in the live frame; the rooms behind the doors and the level-2 copy of the start room with its own doors. Bottom, a TopoExplore+ricci run: cells with negative Ollivier–Ricci curvature on the archive graph (red) against the rest (blue); the negative cells sit at the room's pinches.

And when a live class's active rim contains a cell carrying `openable_door` or `door_adjacent`, the class's whole mass goes to those cells, turning "somewhere on this rim" into "this door, and only while you hold the key" (Figure 9, top). Where no door is in view the arm is inert and the term is plain TopoExplore.

Who carried the door draws. The two ways can be told apart from the selection logs. Every draw records the selected cell, its total score S, and the homological mass H on it; the persisted archive records each cell's slots. On Montezuma's Revenge $S = M(\text{base} + H' + T')$ with M the level multiplier, base the neighbour term (in $\{1.1, 11.1\}$ up to 0.8, the large value while the "same cell, one more key" neighbour is missing, which is almost always) and T' the texture bonus. H/S is exact. T' is not logged, so a door draw with $H/S \leq 1/2$ is binned by $S/M - H/M = \text{base} + T'$: above 22.9 the texture bonus exceeds the base whichever base applied, below 2.2 there is no room for a bonus, and between the two the answer depends on which base applied. Go-Explore has $T' = 0$, so its texture bin must be empty, and it is. Table 10 gives the result over every Phase-1 seed. Narrowing is engaged (the door cell carries homological mass, and that mass is the majority of its score) on 1.2% of the texture arm's door draws, against 2.3% of plain TopoExplore's, where a door sits on a rim only by chance; the arm does not raise the count of homology-carried door draws. What it raises is the door-draw rate, from 1.9% and 1.8% of all draws in Go-Explore and plain TopoExplore to

2.9%, and every one of those extra draws is carried by the slot bonus: 35% of its door draws are unambiguously texture-carried where Go-Explore has none. The two kinds of door draw lead to different outcomes. Homology-carried door draws enter the class they were drawn for (23 entries per 1,000) but seldom collect reward (22 reward-bearing episodes per 1,000); slot-carried door draws enter nothing (0.4 per 1,000) and collect reward at 83 per 1,000. Across all draws the arm also shifts about four points of selection mass from unslotted cells to key-holding and skull-adjacent ones. Read together, the Montezuma gain is what the door and key cues buy when added to Go-Explore's score inside the TopoExplore loop, and the logs does not attribute it to the homological term; a Go-Explore+texture arm with the same slot bonus and no topological pass is the control that would settle the attribution, and it has not been run. The table is recomputed from the episode and archive logs by `analysis/montezuma_door_draws.py`.

Table 10: Door draws on Montezuma's Revenge, Phase 1, all seeds. A door draw is a selection landing on a cell whose slots include `door_adjacent`. Carried by: homology when the cell's homological mass is the majority of its score; texture when the non-homological part of the score exceeds 22.9 (the texture bonus exceeds the largest possible base); either when it lies between 2.2 and 22.9; base below 2.2. Yield: episodes launched from the draw that collected reward, and class entries, per 1,000 door draws in each bin.

		Door draws	carried by (% of door draws)				reward episodes / class entries per 1,000 door draws		
Method	n	(% of draws)	homology	texture	either	base	homology	texture	either
Go-Explore	200	1.88	0	0	89.6	10.4	0 / 0	0 / 0.0	55 / 0.00
TopoExplore	50	1.85	2.3	0	87.5	10.1	22 / 21	0 / 0.0	60 / 0.01
TopoExplore + texture	200	2.86	1.2	34.6	63.9	0.4	22 / 23	83 / 0.4	84 / 0.01

Where the term's weight went. Figure 10 lays out the 24 rooms of level 1 in the arrangement learned from the passages the agent crossed (a room change inside one explore phase at the left or right edge, or through the floor or ceiling; `analysis/montezuma_room_layout.py`) and tints every lattice cell by the homological share of the draws that landed on it, summed over the 50 plain TopoExplore seeds. The weight sits on four solids: the central platform of the start room (room 1), the block of room 5, the wall of room 8 and the block of room 14, with traces on the lava-bridge rooms 10, 18, 20 and 22. None of these is a pocket of free space. Of the 83,819 homology-carried draws on level 1, 1.9% launched an episode that entered its class and 9.6% ended in a retirement; room 5's block, which holds the skull, took the most draws and entered 1.5% of the time. This is the bounded cost of decoys of Appendix A, incurred on the only structures a per-room complex can enclose, and it is the mechanism of the term's neutrality here: the doors are room exits on the perimeter and never lie on a rim, and the corridor rooms enclose nothing at all. The enclosure that would matter is one level up, and on this level it is rare. Treating rooms as cells of a cubical complex on the learned map, an unvisited room is a pocket when the four rooms sharing an edge with it have been visited; only rooms 4, 5, 6 and 9 to 13 have four such neighbours inside the pyramid, and over the 190 runs with room-entry logs the ring closed before entry around room 12 in 15 and around room 5 in 4 (Figure 3a). Room 14, the room most often found last, is a dead end on the perimeter, not an enclosure.

The room-scale complex, run. We ran that complex as an arm: rooms as the cells of a cubical complex on the learned map, a pocket's support the archived cells of each neighbouring room nearest to it, the term otherwise unchanged; alone (TopoExplore + room scale) and on top of the curvature arm, 50 seeds each, paired against the campaign's Go-Explore and TopoExplore seeds (Tables 11 and 12). The ring closed before entry in 7 of the 100 runs carrying the term, 8 classes in all (Tables 13 and 14): six around room 12 and two around room 5, each with 56 to 86 cells of support, drawing between 1 and 448 times before the pocket was entered. While a class was live the term ran the selection: 33 to 45% of all draws in that window landed on its rim, and on those draws the homology term was 72 to 82% of the selected cell's score (0.01 for the one-draw room 5 class of seed 29). The two room 5 pockets were entered within two thousand frames of closing. Of the six room 12 pockets, four were entered by an episode launched from the rim, and five were entered sooner than the same seed reached the room without the term, by 186k to 341k frames in four and by 853k in the fifth (seed 29, whose two runs had already parted at its room 5 pocket), and one later, by 96k (Figure 3). None of it reaches the score: TopoExplore + room scale equals plain TopoExplore on 49 of 50 seeds, and every arm is even with Go-Explore (24 to 26 wins of 50, $p \geq 0.6$). The curvature arm here

runs on the passage-aware graph, where a doorway or ladder is a one-cell-wide edge between rooms (Figure 12); at 50 seeds it is even with Go-Explore and with TopoExplore as well. The event the theory needs is real on this game and the term does what it should when it occurs; it is too rare, and its pocket too cheap, to move a 60M-frame score.

Table 11: The room-scale wave: best archived trajectory return at 60M frames, IQM [percentile-bootstrap 95% CI] over seeds 0–49 (`analysis/montezuma_rooms_wave.py`).

Arm	n	IQM score [95% CI]
Go-Explore	50	31,138 [25,800, 35,581]
TopoExplore	50	28,173 [23,511, 33,073]
TopoExplore + room scale	50	27,800 [22,950, 33,116]
TopoExplore + ricci	50	26,265 [20,976, 32,512]
TopoExplore + ricci + room scale	50	26,069 [19,942, 33,078]

Table 12: The room-scale wave, paired: wins/losses/ties and the two-sided Wilcoxon p of each arm against Go-Explore and against TopoExplore on the same seeds.

Arm	n	vs Go-Explore (W/L/T, p)	vs TopoExplore (W/L/T, p)
Go-Explore	50	–	24/26/0, 0.70
TopoExplore	50	26/24/0, 0.70	–
TopoExplore + room scale	50	26/24/0, 0.69	0/1/49, –
TopoExplore + ricci	50	24/24/2, 0.70	24/24/2, 0.87
TopoExplore + ricci + room scale	50	24/24/2, 0.62	23/25/2, 0.98

Table 13: Every room-scale class born before its pocket was entered, over the 100 runs carrying the term: the enclosed room, the frame at which the ring closed (millions), the support in cells, the draws the class received before entry, and the share of all selections in that window that landed on the rim.

Arm	Seed	Room	Closed (M)	Support	Draws	On rim (%)
TopoExplore + room scale	18	12	2.59	83	331	40
TopoExplore + room scale	46	12	1.60	61	448	42
TopoExplore + ricci + room scale	14	12	4.32	59	30	39
TopoExplore + ricci + room scale	29	5	1.08	82	1	33
TopoExplore + ricci + room scale	29	12	1.77	75	80	36
TopoExplore + ricci + room scale	33	12	1.45	56	10	45
TopoExplore + ricci + room scale	35	12	3.84	86	416	37
TopoExplore + ricci + room scale	42	5	0.72	79	5	33

What curvature sees. The ricci arm computes Ollivier–Ricci curvature on the graph the archive induces (the lattice relation restricted to archived cells, never the unexplored world), with exact per-edge W_1 by assignment on unit-expanded measures, and gives a bonus proportional to how far below zero a cell's mean incident curvature sits. Negative curvature marks pinches. In the starting room of a `te-ricci` run (342 archived cells, curvature defined on all) 13% of cells are negative, with the two most negative at the symmetric left and right descents, exactly where the room pinches (Figure 9, bottom). That is the signal the chamber worlds reward (Section 5.1); on this game a pinch is rarely where the key is, and the arm is null. Figures 11 and 12 show the field over whole worlds. On TopoGym the negative cells are exactly the chamber doorways, corridor corners and maze junctions: at most 2.3% of the free cells in the chamber families, 14–19% in the maze and the Ladders scenario. On Montezuma's Revenge, over a full level-1 archive with the crossed passages added as edges of the archive graph (`analysis/montezuma_ricci_map.py`), 6.7% of sites are negative and they sit where a ladder meets a corridor; the passages between rooms join one corridor to another and are as flat as the corridors themselves (mean curvature $+0.11$ at passage sites against $+0.13$ elsewhere). At the archive's 8×16 binning the game is a graph of thin corridors, which is why a pinch detector finds little to reward there and much in a world of rooms.

Table 14: The same classes: the homological share of the selected cell's score on the rim draws, and the frame at which the room was entered in this run (starred where the entering episode was launched from the rim) and in the same seed run without the term.

Arm	Seed	Room	Hom. share	Entered (M)	Without term (M)
TopoExplore + room scale	18	12	0.82	2.75*	3.09
TopoExplore + room scale	46	12	0.79	1.81	1.71
TopoExplore + ricci + room scale	14	12	0.76	4.34	4.57
TopoExplore + ricci + room scale	29	5	0.01	1.08*	1.08
TopoExplore + ricci + room scale	29	12	0.76	1.82*	2.67
TopoExplore + ricci + room scale	33	12	0.75	1.45*	1.75
TopoExplore + ricci + room scale	35	12	0.80	4.05*	4.23
TopoExplore + ricci + room scale	42	5	0.72	0.73	0.73

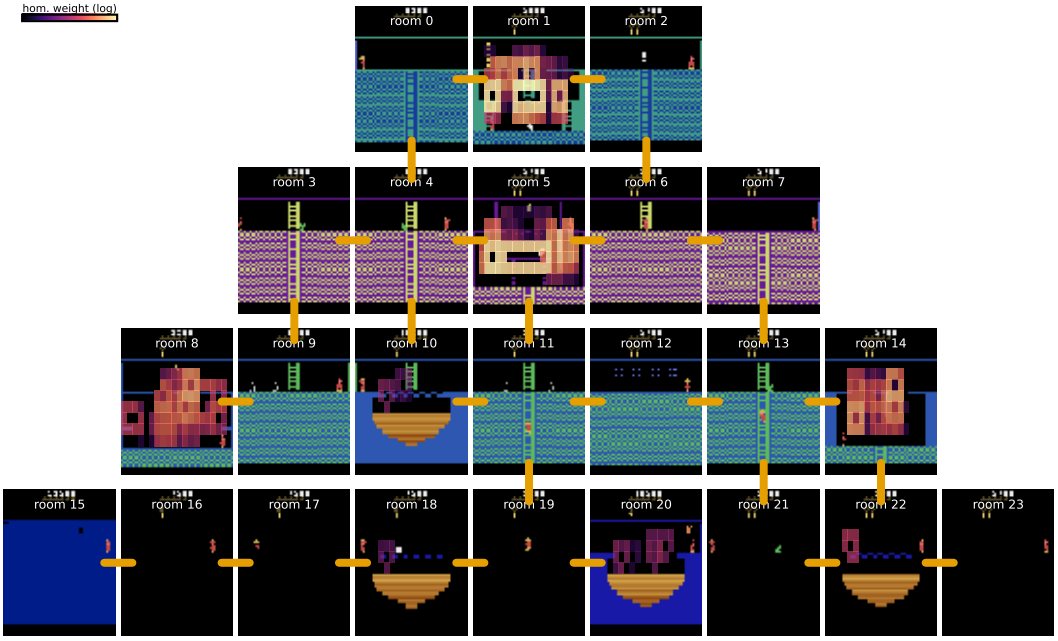


Figure 10: Level 1 of Montezuma's Revenge in the layout learned from the passages the agent crossed (bars across shared edges), each room re-rendered from an archived state, with every lattice cell tinted by the homological weight the term drew through it over the 50 plain TopoExplore seeds (log scale; untinted where the archive never surrounded anything). Every tinted structure is a solid: the map shows where the decoy cost was incurred.

C.2 Pitfall!: more rooms, no more score

Pitfall! is a chain of 255 thin, nearly linear rooms with sparse positive reward and dense small negative rewards; cells are $(\text{room}, x_{\text{bin}}, y_{\text{bin}})$ with the room taken from the world-generating LFSR byte, which disambiguates the visually aliased rooms that defeat downscaled-frame cells. It is the regime with the least for the homology term to find, since the environment has almost no enclosed structure, and most rooms carry no reward, so even rooms found need not raise the score. The campaign was run at the flagship budget on every arm so that the result is a measurement rather than an assumption.

The plain term neither helps nor hurts. At five seeds plain TopoExplore trailed Go-Explore (median 56 rooms against 67); at ten the deficit is gone (67 against 67, overlapping intervals) and frames to every milestone coincide. The mechanism logs (Section 5.2) says why nothing happens in either direction: Pitfall!'s rooms are thin horizontal strips — the archive occupies a median of 5 of 16 vertical bins per room — so sparse visits create one-cell "holes" that are coverage artefacts, not chambers. About 100 such classes are born per run (mean over ten seeds), 82 of which die naturally

Table 15: Pitfall!, Phase 1, 60M game frames, matched budgets, $n = 10$ seeds per arm. Frames to k rooms (median, 95% bootstrap CI, in millions) and rooms discovered; the score is in Table 16. Frames-to-k are indistinguishable across arms at every k.

Method	Frames to 15 rooms (M)	Frames to 20 rooms (M)	Rooms
Go-Explore	0.6 [0.4, 0.9]	1.6 [1.1, 2.1]	67 [58, 73]
TopoExplore	0.5 [0.5, 0.6]	1.6 [1.3, 2.1]	67 [56, 76]
TopoExplore + ricci	0.6 [0.4, 0.7]	1.3 [0.8, 2.2]	74 [57, 78]
TopoExplore + texture	0.5 [0.4, 0.7]	1.6 [1.3, 2.2]	**78** [73, 83]
TopoExplore + both	0.6 [0.3, 0.7]	1.7 [1.0, 2.9]	72 [55, 78]
TopoExplore + debounce	0.5 [0.3, 1.0]	2.3 [1.7, 3.3]	75 [57, 80]
PPO+RND (100M)	—	—	9 [9, 10]
PPO+ICM (100M)	—	—	7 [7, 7]

Table 16: Pitfall!, the same runs as Table 15: best archived trajectory score (median, 95% bootstrap CI). The PPO baselines keep no archive and so have no archived score.

Method	Score
Go-Explore	6,926 [4,000, 6,956]
TopoExplore	6,646 [3,976, 6,856]
TopoExplore + ricci	6,739 [3,861, 6,910]
TopoExplore + texture	**6,956** [5,426, 7,340]
TopoExplore + both	5,427 [0, 11,970]
TopoExplore + debounce	5,803 [3,855, 7,328]

as coverage fills them, and the remainder absorb bonus draws until retirement; rim draws account for ∼1.6% of selections. This is the intended behaviour when there is nothing to detect, and its cost is the bounded per-class cost and the throughput cost of the pass (Appendix C.5).

Texture finds more rooms but no more reward. As on Montezuma's Revenge, the texture arm is the one that moves: its `underground` slot gives a bonus on tunnel cells, the only route past the walls that partition the surface, and it discovers more rooms than every other arm (78 [73, 83] against 67 [58, 73] for Go-Explore, 16% more), an effect that survived the seed extension. Score does not follow (6,956 against 6,926): Pitfall!'s reward sits in a handful of treasure rooms, so the extra rooms add hazards but no reward. The two-term arm is within Go-Explore's interval and the curvature arm just above it (74 against [58, 73]), so the two terms do not compose here, and the debounce arm is slowest to 20 rooms. Read together with Montezuma's Revenge, the two games suggest the signal's value depends on enclosed structure to detect and on that structure being worth entering, and that the first condition is measurable from the agent's own logs rather than requiring prior knowledge of the environment.

C.3 Negative controls: cleaning the signal

Pitfall!'s artefact classes (Appendix C.2) invite a filter that removes them and leaves the real chambers. Two such filters were designed, calibrated, and rejected. Both are reported here rather than in the main text because neither is part of TopoExplore: they are ablations of S_{loc} that failed.

Spatial magnitude thresholding (rejected before running). The natural first remedy for Pitfall!'s artefact classes is to ignore classes below a magnitude threshold. Calibrating that threshold against a coverage-noise null — classes arising in Go-Explore runs' reconstructed visited sets, plus synthetic Bernoulli-visited lattices matched to each game's room geometry — shows that null classes concentrate at inradius 1 (99th percentile 1 on Pitfall-shaped lattices, 2 on Montezuma-shaped ones), *and so do the real chambers* at this cell resolution. A threshold that removes the artefacts also silences the signal, so the variant was recorded as a negative control rather than run.

Temporal debouncing (run, `debounce`). This variant weights a class in proportion to how improbable its survival is under a coverage-noise null: the arm multiplies a class's mass by the empirical

CDF of null class lifetimes evaluated at the class's current age, so a class earns funding by outliving what noise would predict. The null was measured treatment-blind — from Go-Explore logs and synthetic lattices only, never from TopoExplore runs — which leaves the rule with no threshold or percentile parameter to tune. It fails, and the calibration says why: measured null lifetimes are long (median ≈ 480 topology passes, 90th percentile in the thousands), because a noise class persists until exploration happens to fill its one cell, so withholding funding until a class outlives that distribution also withholds it from genuine chambers. On Montezuma's Revenge the arm completes level 1 in 6 of its 10 seeds, against 8 of 10 for plain TopoExplore on the same seeds; on Pitfall! it is slowest to 20 rooms (Table 15). We report it because a plausible fix that fails under its own calibration is more informative than omitting it.

C.4 Chamber worlds: survival, cost, and grids

The k-sweep suites behind Figure 2 emit the same set of figures for every family; the main text shows only the all-k probability panels. Figure 15 gives the goal-found curves and a chamber grid, Figure 16 gives the survival curves at the high k values (fraction of seeds with at least j chambers at budget end), Figure 17 the interactions to enter all k (successful seeds only) and the per-chamber cost curves, whose bend locates where each method dies, and Figure 18 the remaining chamber grids, including the EpicChase spiral. Every statistic plotted, with its confidence interval and n, is in `figures/topogym/*/suite_stats.csv`; the per-world outcome behind every cell of Table 1 is in `figures/topogym/holdout/holdout_goal_found_by_world.csv`, and Figure 14 shows every unit with its outcome.

C.5 Cost against archive size

The gradient baselines spend their whole budget on policy learning; the archival methods use it on discovery and turn what they find into a policy afterwards (Phase 2, Section 3.3, not analysed here). The topological pass costs 16–21% of wall-clock at every iteration on archives of ∼6,000 cells, a ceiling since the cadence is configurable. On Montezuma's Revenge RND and ICM required ∼7.5–8.3 GPU-hours per seed to reach 1–3 rooms; a TopoExplore seed reached 24 rooms in ∼8.0 CPU-hours.

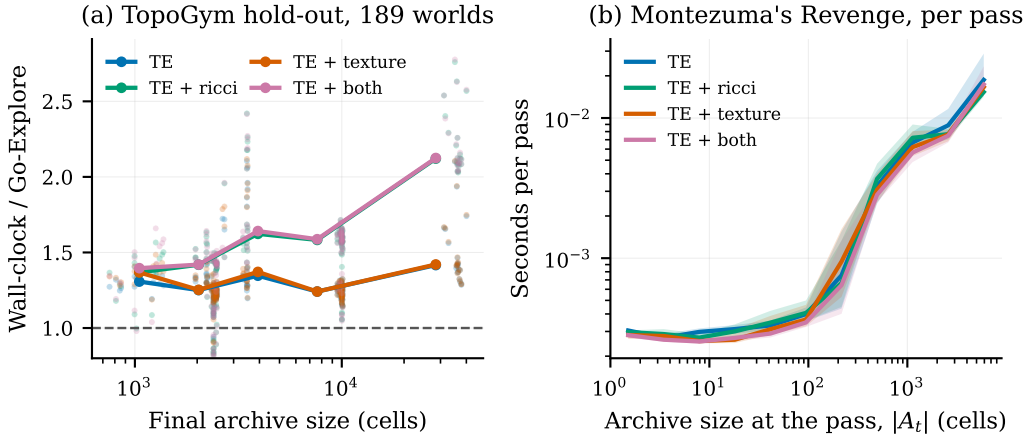


Figure 19: What the topological pass costs and how the cost scales. (a) Hold-out benchmark, 189 worlds: wall-clock of each TopoExplore arm divided by Go-Explore's on the same world (one point per world, medians per archive-size bin), against the archive size the run ended with. The texture arm coincides with the plain term and the both arm with the curvature arm. (b) Montezuma's Revenge, every topology pass of every Phase-1 seed: seconds per pass against the archive size at that pass, medians with interquartile bands per bin, log axes.

Table 17: Overhead of each arm. Left: wall-clock relative to Go-Explore on the 189 hold-out worlds, the median with its interquartile range and the medians over the worlds in the smallest and largest quartile of final archive size (756 to 2,392 and 9,946 to 39,560 cells). Right: the log-log slope b of per-pass latency against archive size on Montezuma's Revenge over the passes with at least 100 cells.

Arm	Wall-clock / Go-Explore, 189 worlds			Pass latency $\propto \lvert A_t \rvert^{\,b}$, $\lvert A_t \rvert \geq 100$
	median [IQR]	smallest quartile	largest quartile	b
TE	1.27 [1.22, 1.32]	1.28	1.27	0.82
TE + ricci	1.49 [1.30, 1.62]	1.28	1.61	0.80
TE + texture	1.27 [1.23, 1.34]	1.29	1.27	0.76
TE + both	1.50 [1.31, 1.63]	1.28	1.62	0.93

Figure 19 and Table 17 answer two questions the compute paragraph of Appendix C.5 leaves open: what the pass adds over Go-Explore when both run the same world, and whether that addition grows with the archive.

Per world, against Go-Explore. On the hold-out benchmark every archive arm ran on every world in the same shard, one after the other, so dividing an arm's wall-clock by Go-Explore's on the same world pairs out the machine's load; the runs shared an 18-core machine with 16 shards, so absolute times are only indicative (Go-Explore took a median of 210 seconds per world, interquartile range 155 to 350) and are shipped in `figures/topogym/holdout/cost_vs_archive_topogym.csv`. The plain term costs a median $1.27\times$ Go-Explore's wall-clock (interquartile range 1.22 to 1.32, ninetieth percentile 1.42, worst world 2.11), and the ratio is flat in archive size: 1.28 on the quartile of smallest worlds and 1.27 on the quartile of largest, from about 800 to about 40,000 cells. The texture arm costs the same as the plain term, since a slot is a per-cell attribute read at scoring time and adds no pass of its own. The curvature arm costs a median $1.49\times$ and its ratio does grow, from 1.28 on the smallest quartile to 1.61 on the largest (worst world 2.76), and the both arm tracks it. In steps per second the plain term runs at 3,510 against Go-Explore's 4,439 (medians over worlds), the curvature arms at about 3,470.

Per pass, against the archive. The flat ratio of the plain term is not because the pass is cheap in the limit; it is because Go-Explore's own iteration is already linear in the archive. Each iteration draws a cell from the whole archive, so Go-Explore pays $O(|A_t|)$ per iteration, and panel (b) shows the pass paying about the same order. Over the 5.1 million passes of the 16 plain-term seeds on Montezuma's Revenge, latency sits on a floor of about 0.3 ms while the archive is below 100 cells (the fixed cost of building the complex), then grows as $|A_t|^{\,b}$ with $b = 0.82$ (0.76 to 0.93 across arms): 0.75 ms at about 200 cells, 3.3 ms at 500, 6.7 ms at 1,100, and 19 ms at about 6,000, the largest archives a Phase-1 seed reaches. At the pass cadence this is the 14 to 21% share of wall-clock of Appendix C.5. Pitfall! gives the same curve (0.25 ms floor, 20 ms at about 4,000 cells; `cost_vs_archive_pass_pitfall.csv`). A pass whose latency grew faster than the archive would show as a rising ratio in panel (a); the plain term's does not, and the curvature arm's, whose Ollivier–Ricci computation touches every edge of the archive graph, does. Both figures are produced by `analysis/iclr/cost_vs_archive.py` from the hold-out result files and the epoch logs.

D Reproducibility

The method is stated in full in Algorithm 1, with every hyperparameter, backend and class-identity rule in Appendix B; the theory's model, assumptions and complete proofs are in Appendix A. The hold-out benchmark is a public, frozen split whose tuning and test worlds are disjoint (Section 4); the Montezuma test's hypothesis, seeds, statistic and decision rule are stated in Appendix C.1; the embodied pilot's per-seed evaluation rows, trajectory pool recipe and training configuration are in Appendix B.2.

Every figure and table in this paper is generated from the logs written during training or from the per-run summaries derived from them; no number is transcribed by hand. The paper-level script `papers/iclr/reproduce.sh` has tiered targets:

- `setup` — virtual environment and dependencies, including the TopoGym release the paper used;
- `tables` — every table from the committed per-run summaries, with no training;
- `figures` and `topogym-figs` — every figure from the released logs, with no training;
- `check` — verifies every number the prose quotes against the generated tables and summaries;
- the campaigns themselves, per domain: `runs`, `baselines` and `phase2` in `experiments/{montezuma,pitfall}/reproduce.sh`, resumable, so that any (environment, method, seed, budget) combination already complete is skipped and an interrupted run costs only the jobs in flight;
- `experiments/topogym/topogym_benchmarks.sh` — the TopoGym studies behind Table 1 and Figures 2 and 15: the 189-world hold-out benchmark, the three chamber-count k-sweeps, the chamber grids, their figures, and the port into the paper tree. Every shard's exact argv and the TopoGym commit it ran at are shipped as run manifests, and the script replays those argv verbatim; Table 1 is recomputed from the episode logs by `analysis/topogym/holdout_table.py` (verified identical to the shipped CSVs);
- `figures` also runs the Montezuma diagnostics over the same episode rows and archives: `analysis/montezuma_door_draws.py` (Table 10), `analysis/montezuma_regime.py` (Figure 8) and `analysis/montezuma_room_heatmap.py` (Figure 10; re-renders rooms in the ALE, on the room layout `analysis/montezuma_room_layout.py` learns from the step logs), with `analysis/montezuma_where_topo_helps.py` printing the paired frames-to-k tests quoted in Section 5.2, `analysis/montezuma_rooms_wave.py` the room-scale wave's tables (Tables 11 to 14), and `analysis/montezuma_room12.py` Figure 3 from the first-visit events of every run that logged them;
- the embodied pilot (Table 3) lives in a separate repository whose recipes R14–R18 name the trajectory pool, the arm lists, the training recipe, and the evaluation quotas; `analysis/embodied/pilot_table.py` recomputes the table from the per-episode evaluation rows, the composite and training-example figures come from that repository's `make_pilot_figures.py` and `make_n1_paper_assets.py`, and the per-seed summaries, source commit, and handoff notes ship under `figures/embodied/`.

Seeds, budgets, and hyperparameters are pinned in code. Archival runs are deterministic given a seed (deterministic emulator, seeded selection), and the determinism survives a change of platform: a 200,000-step run of the curvature arm executed on x86-64 Linux reproduces the arm64 macOS run of the same seed exactly, agreeing on the action taken and the resulting cell at every one of the 200,000 steps. Seeds produced on either platform are therefore poolable, which is what lets the seed extension run on cloud CPU. The learned baselines are not bit-deterministic on GPU, so every training episode logs its environment reset seed and action sequence, making any logged episode exactly replayable. Runs persist their archives, so trajectories, demonstrations, and per-room first-entry frames are recoverable after the fact rather than only during the run.

Metrics and hardware. The evidence throughout is training-time exploration: frames to the k-th distinct room, right-censored at the budget and always reported with the fraction of seeds that reached k; completion; mechanism logs (live and pruned classes, the realised share δ_t, attempt counters); and cost. PPO+RND and PPO+ICM use dual value heads, non-episodic normalised intrinsic returns, observation whitening and sticky actions, as in their reference implementations. The Atari and embodied campaigns ran on one 18-core Apple-silicon workstation, with archival methods CPU-bound and PPO and Phase 2 on its GPU; the seed extensions and the TopoGym waves ran on cloud CPU (`cloud/README.md`).

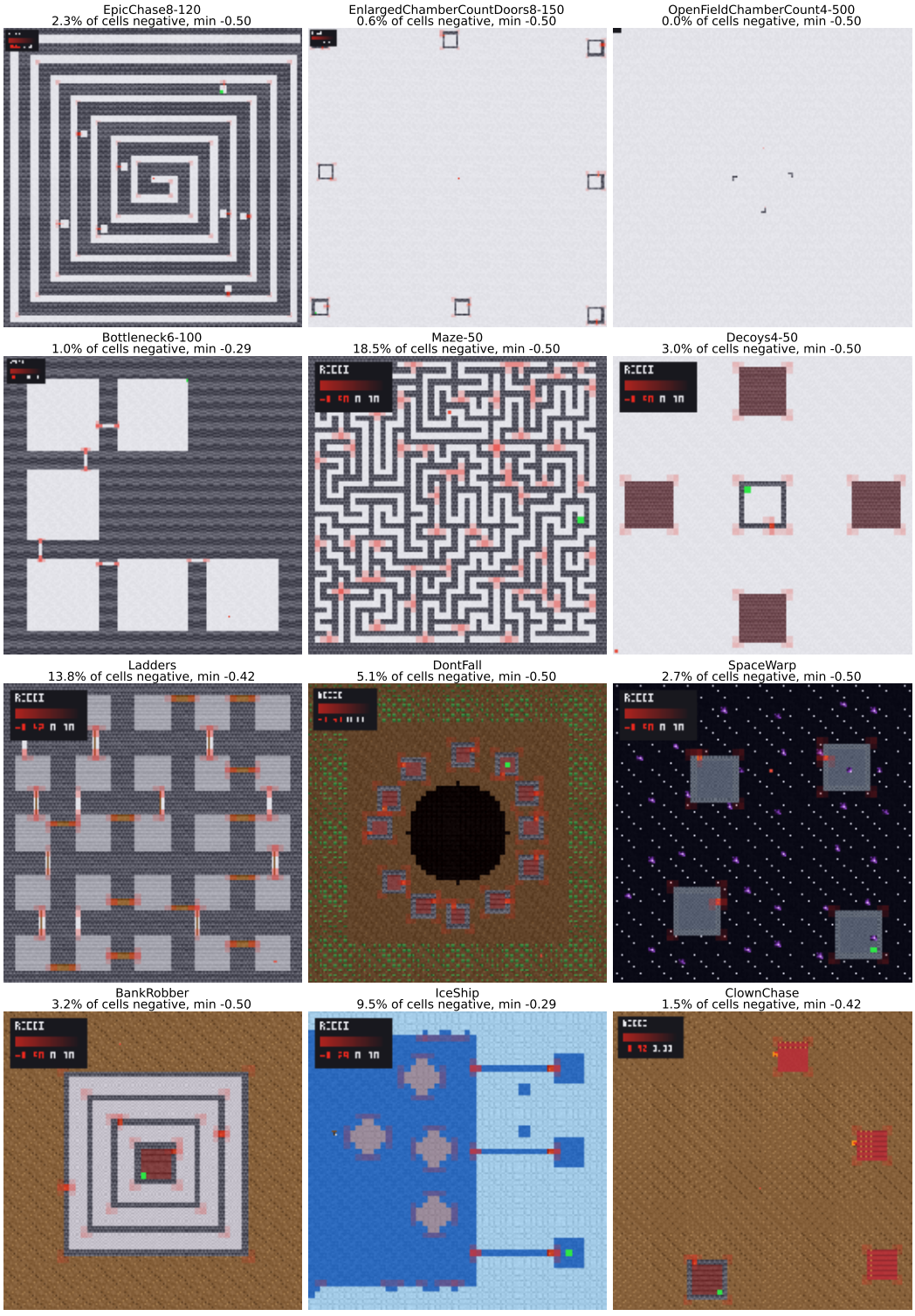


Figure 11: Ollivier–Ricci curvature on TopoGym: twelve worlds rendered by TopoGym with its curvature heat map, every free cell tinted red in proportion to how negative its curvature is over the world's free-cell graph (`analysis/topogym_ricci_panels.py`); the negative cells are doorways, corners and junctions.

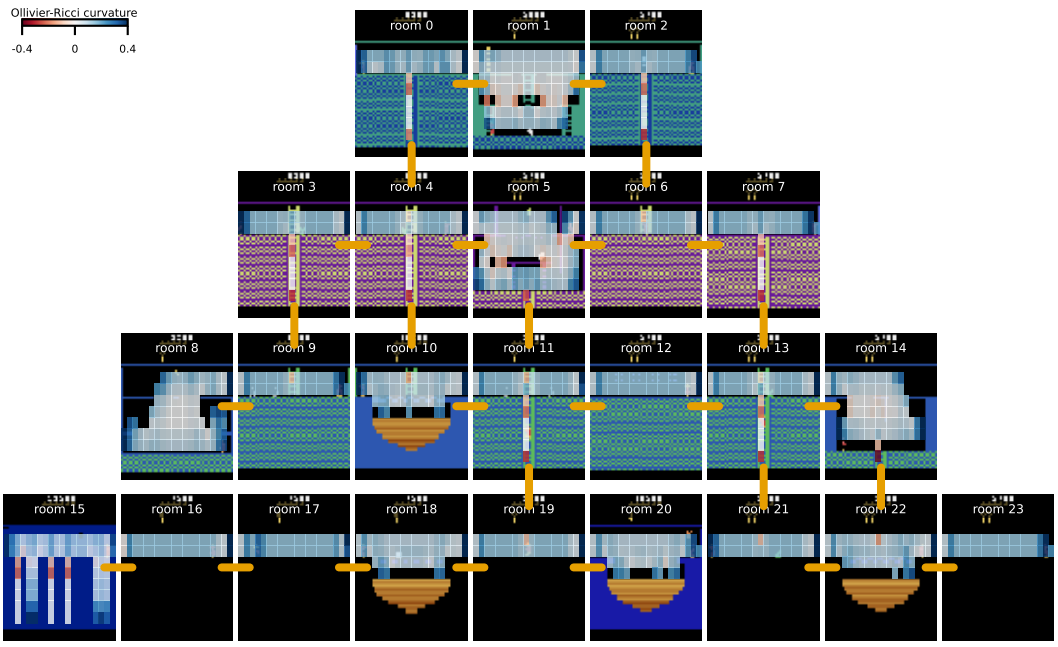


Figure 12: Ollivier–Ricci curvature on level 1 of Montezuma's Revenge: the curvature of the archive graph of one plain TopoExplore seed, lattice neighbours inside a room plus the crossed passages, on the learned room layout (red negative, blue positive; `analysis/montezuma_ricci_map.py`).

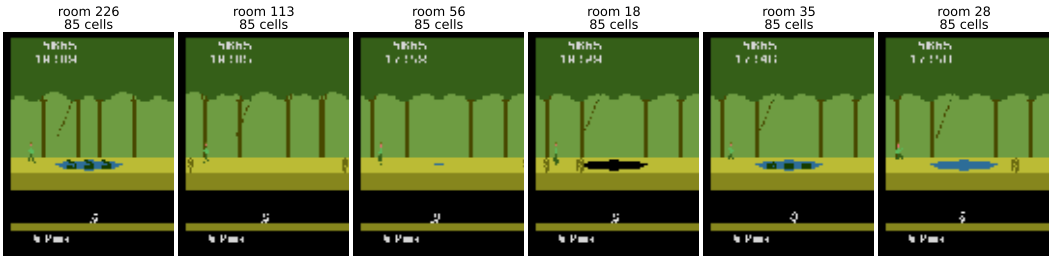


Figure 13: Archived states on Pitfall!, replayed from a TopoExplore run: six of the 52 rooms it reached, each a thin strip with one hazard (crocodiles, rolling logs, a tar pit, quicksand) and a tunnel below. Every room saturates at the same $\sim$85 lattice cells, so the archive has almost no per-room shape for the homology term to read.

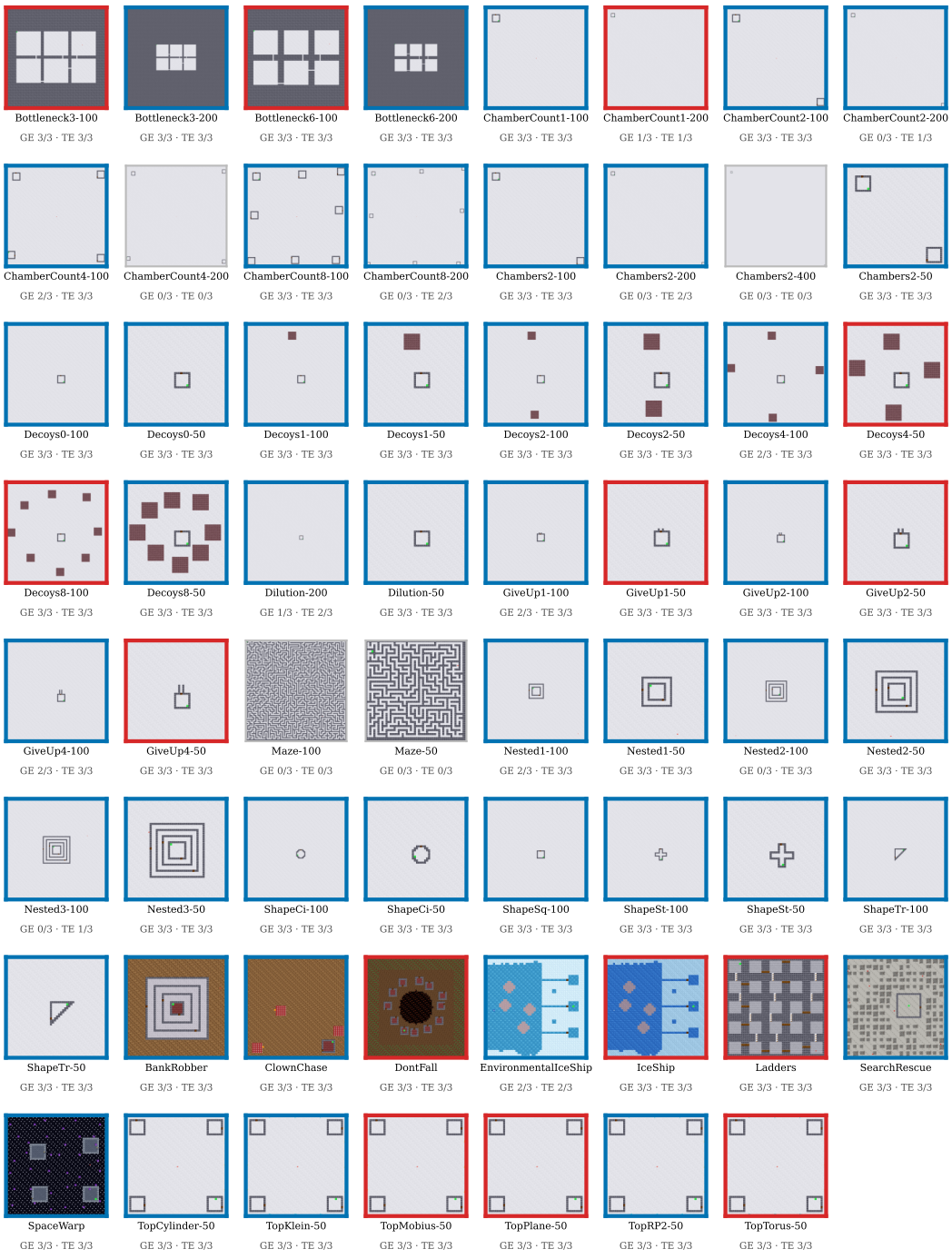


Figure 14: The 63 units of the TopoGym hold-out benchmark (Table 1), one of each unit's three layout seeds, drawn whole at their true extent and ordered by slice (GridWorld2D, Texture, non-trivial topologies) and family. Under each: the number of its three seeds in which Go-Explore and plain TopoExplore found the goal. Frame: who found it first, i.e. the method that solved more seeds or, on a tie, the lower median steps-to-goal over the seeds both solved; blue TopoExplore (45 units), red Go-Explore (14), grey neither solved any seed (4).

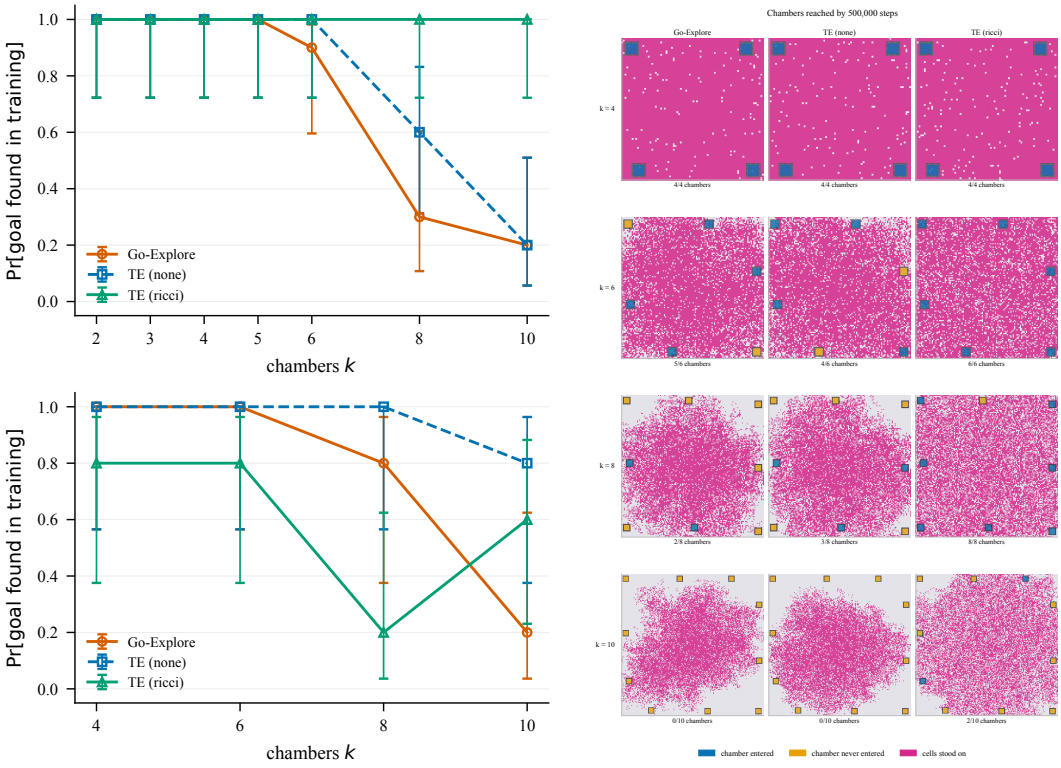


Figure 15: Left, top and bottom: probability of finding the goal during training against k on EnlargedChamberCount and OpenFieldChamberCount (Wilson 95% CIs). Right: which rooms each method had opened at half budget on EnlargedChamberCount (layout seed 0 at each k, 500k of 1M steps): blue chambers were entered, amber never were, magenta marks cells stood on. At $k = 8$ Go-Explore and plain TopoExplore have opened 2 and 3 of 8 chambers and their visited sets are still a disc around the start; the curvature arm has opened all 8 and reached the walls. One row is one world and illustrates the statistic; the claim lives in Figure 2.

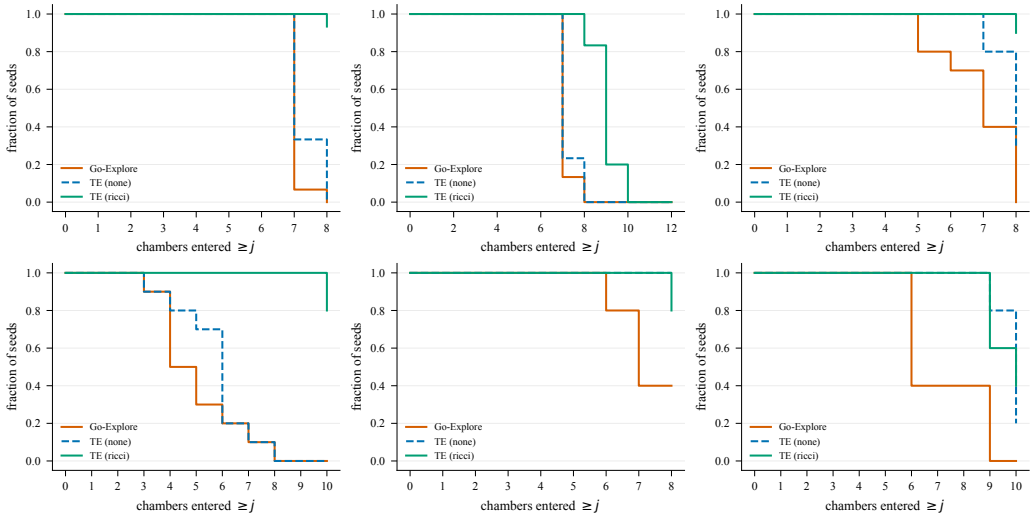


Figure 16: Survival curves at the high k values: fraction of seeds with at least j chambers entered at budget end. Top: EpicChase $k = 8$ and 12 ($n = 30$), EnlargedChamberCount $k = 8$ ($n = 10$). Bottom: EnlargedChamberCount $k = 10$, OpenFieldChamberCount $k = 8$ and 10 ($n = 5$, 2M steps).

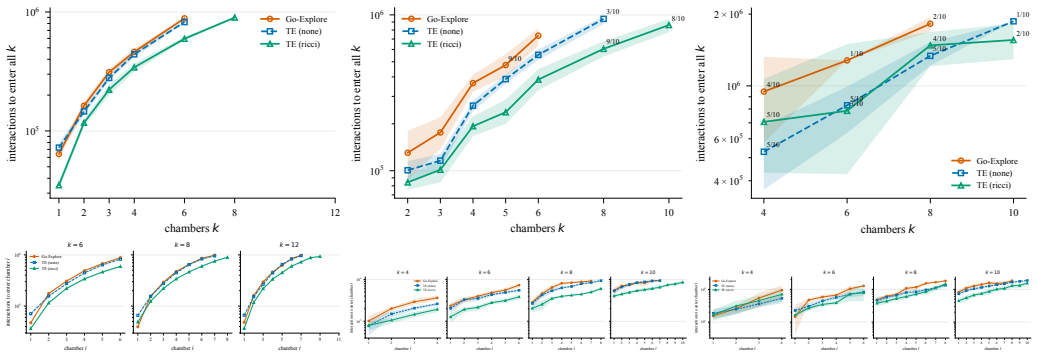


Figure 17: Cost figures for EpicChase, EnlargedChamberCount, and OpenFieldChamberCount (left to right). Top: interactions to enter all k chambers against k, successful seeds only, annotated with success counts. Bottom: interactions to enter the i-th chamber against i, one panel per k; where a method's curve bends upward is where it stops.

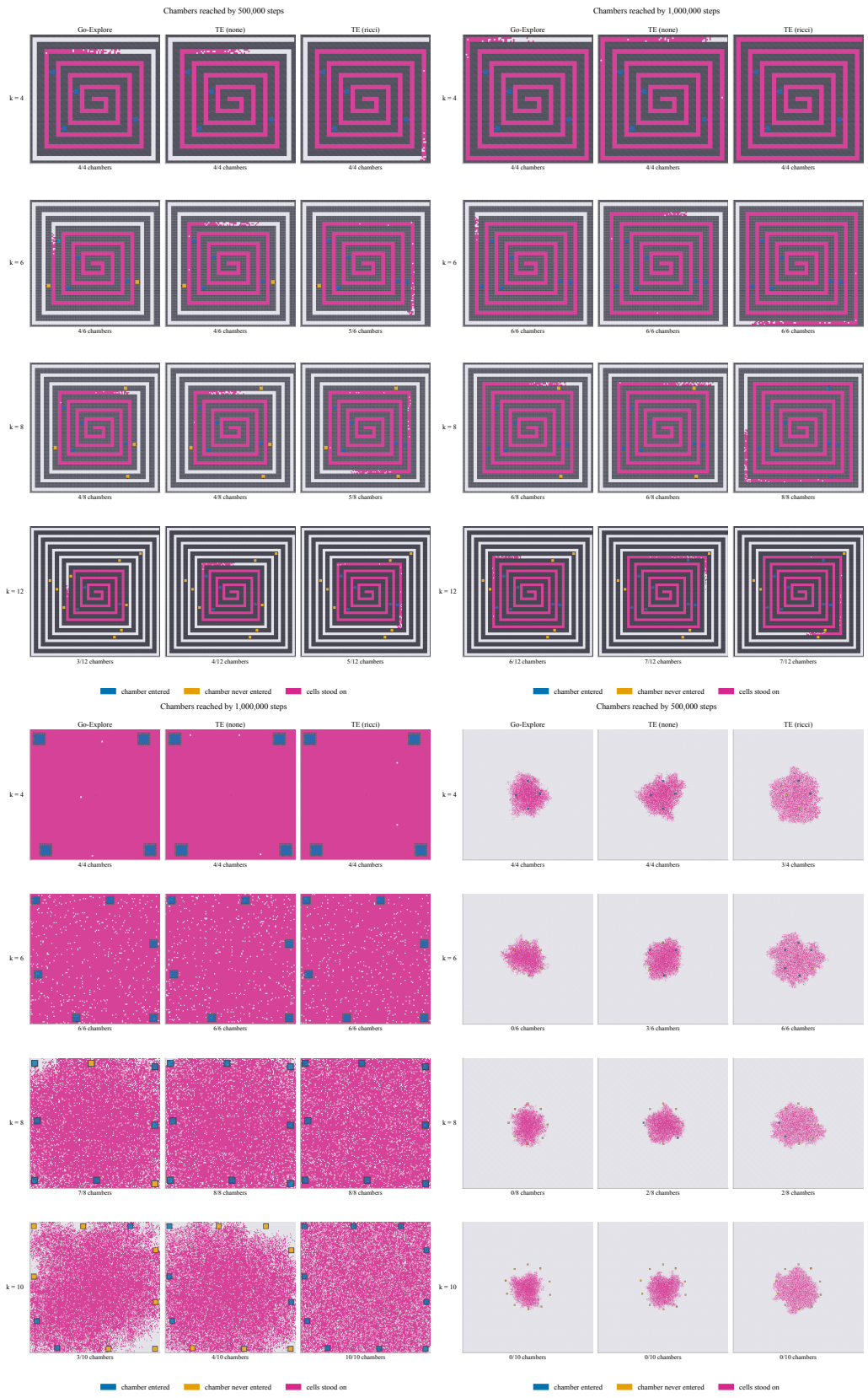


Figure 18: Chamber grids. Top: the EpicChase spiral at 500k and 1M steps (seed 0; the spiral is layout-seed invariant, so one panel per k is the family). Bottom left: EnlargedChamberCount at the full 1M budget. Bottom right: OpenFieldChamberCount at 500k steps. Blue chambers were entered, amber never were; magenta marks cells stood on.

E AI USE STATEMENT

We used Claude (Anthropic), operated by the authors through an agentic coding interface, for tasks with required disclosure: to help develop the theoretical model of Appendix A by turning the claims the authors intended to prove into concrete candidate statements (the proofs themselves were written by the authors), to give feedback on experimental design, and to implement the method and the analysis code. We did not use it to generate synthetic data, to prove mathematical claims, or to interpret results; translation, data cleaning and reformatting, and qualitative or thematic analysis are not applicable to this work. Additionally, for tasks with recommended disclosure, it drafted and edited text, proposed the paper's structure, created and modified figures, suggested experimental parameters, and identified and formatted references. We have reviewed all AI-assisted work: the authors checked every proof, reviewed and tested all code, verified references against the cited venues, and recomputed every reported number from the run logs. We take responsibility for the final content of this work, including text, claims and artifacts produced with the aid of generative AI.