LEARNING COUNTERFACTUAL PREDICTIVE MEMORY FOR LONG-HORIZON AERIAL EXPLORATION

Anonymous authors
Paper under double-blind review

ABSTRACT

An exploration robot must remember not only where it intended to fly, but which observations would make that intention obsolete. Recent aerial planners exploit geometric passability and cached global tours, yet neither a binary frontier label nor a visit order predicts the decision consequences of a new observation. We propose counterfactual predictive memory (CPM): an action-conditioned representation of how a short sensing maneuver changes newly observed volume, frontier connectivity, and continuation cost. Paired simulator branches share the same observed history and differ in either the unobserved environment or the executed maneuver. A single learned transition law evaluates immediate motion and observation-conditioned continuation; its value discrepancy also determines which historical details can be compressed. We derive a conditional action-regret bound and evaluate the method through aerial comparisons, component ablations, and planar learning controls. In the forest comparisons, CPM reaches 99% coverage in 251.6 s with LiDAR and 507.4 s with limited-FoV depth sensing, the lowest completion times within the respective comparison groups. Component ablations reveal complementary contributions to exploration efficiency, with tradeoffs in prediction error and planning latency.

1 INTRODUCTION

Efficient exploration is a problem of acting before the map is complete. A frontier can be a short-lived occlusion, an entrance to a large region, or a residual patch that requires a final observation. These possibilities can look alike from one viewpoint but require different flight and revisit decisions. Frontier and next-best-view methods establish a practical foundation (Yamauchi, 1997; Bircher et al., 2016). Hierarchical exploration systems improve global coverage through incremental frontier structures, tours, and regional planning (Zhou et al., 2021a; Cao et al., 2021; Tang et al., 2023). Their success motivates a more specific learning question: *which parts of observation history must a robot retain to predict how a sensing action will change its next decision?*

PACE (Gao et al., 2026) makes this question concrete. It estimates frontier passability from cluster geometry, caches a global visit sequence with an anchor, and couples that intent to traverse or link trajectories. It repeatedly updates classifications and uses safe corridors and backup trajectories; it is not an immutable-label or safety-unconstrained baseline. Nevertheless, its memory records a preferred order rather than a distribution of action-conditioned outcomes. A temporarily hidden passage and a genuinely closed region can therefore induce similar current labels while having different future consequences. PACE explicitly discusses dynamic occlusion and computation at high frontier density as limitations. Its reported experiments do not isolate whether a historical observation would change the preferred action after a new measurement.

Replacing the label with a neural classifier does not resolve this gap. Passability is only a proxy: a physically traversable opening may lead to little new volume, while an untraversable recess may be worth observing now to avoid a later return. Likewise, maximizing map entropy reduction can prioritize uncertainty that never affects the route. We instead learn an *action-conditioned predictive memory*: a compact state whose usefulness is measured by its ability to preserve continuation decisions. This follows the general motivation of predictive state representations and value-aware models (Littman et al., 2001; Grimm et al., 2020), specialized to evolving frontier topology and continuous aerial motion.

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

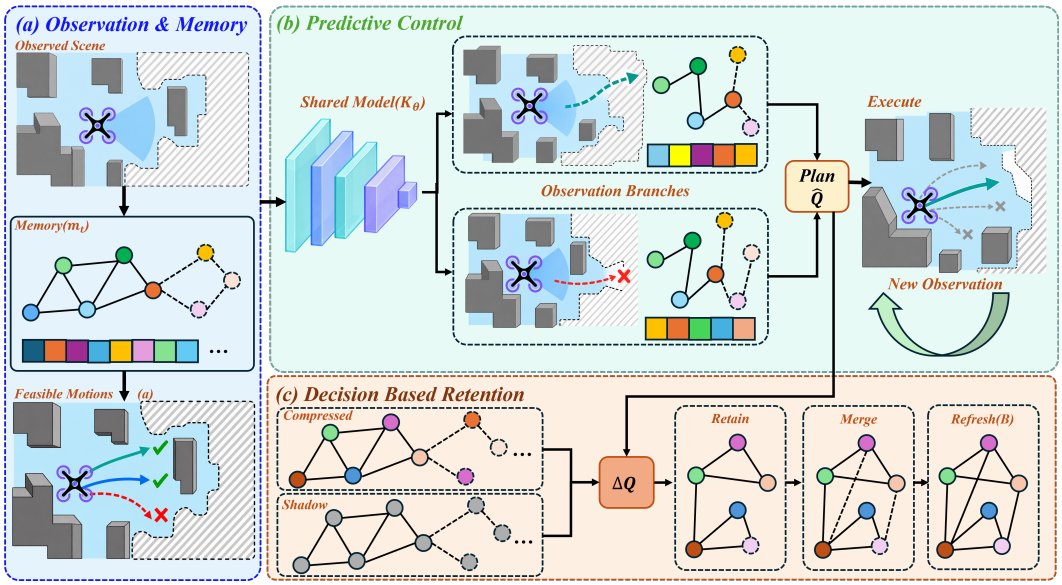


Figure 1: **Counterfactual predictive memory for aerial exploration.** (a) Observations update frontier memory m_t, while an established geometric planner supplies feasible maneuvers a. (b) The shared outcome law K_θ predicts observation-dependent memory transitions; $\widehat{Q}$ evaluates their continuation value to select a feasible maneuver. Execution provides the next observation (blue feedback). (c) Candidate-value discrepancy ΔQ between compressed and shadow memories guides retention, merging, and refresh within token budget B (orange feedback). Both memories use the same K_θ and $\widehat{Q}$. Scene insets and trajectories are schematic.

Our contribution has three connected parts. First, paired counterfactual training teaches one transition law to predict what different sensing maneuvers reveal under observation-equivalent hidden worlds. Second, the same law evaluates observation-contingent continuation, jointly selecting a frontier and its approach motion. Third, memory compression is trained and triggered by changes in predicted action values, with a conditional regret bound that makes the approximation explicit. Figure 1 summarizes how the shared predictive state and objective couple motion selection with memory maintenance. The central falsifiable claim is that this representation preserves decision-relevant history at lower memory cost than geometry reconstruction or a cached visit order. We evaluate this claim using matched exploration comparisons, component ablations, and planar learning controls.

2 Related Work

Exploration and flight. Sampling and regional planners trade global coverage against update cost, including UFOExplorer, FAEL, and HEROS (Duberg & Jensfelt, 2022; Huang et al., 2023; Long et al., 2025). FALCON and FASTEX exploit coverage guidance (Zhang et al., 2025b;a); EPIC and EPICON operate on point-cloud representations (Geng et al., 2025; Yuan et al., 2026); RUSH and SCOPE reduce the cost of spatial reasoning (Bu et al., 2025; Li et al., 2026). FAEP, HIGHSTAR, and kinodynamic exploration explicitly address sensing or motion efficiency (Zhao et al., 2024; Dong et al., 2025; Mendes et al., 2026). They are direct aerial comparisons. GCOPTER, EGO-Planner, and SUPER supply trajectory and safety ideas rather than complete exploration baselines (Wang et al., 2022; Zhou et al., 2021b; Ren et al., 2025). We do not count local navigation alone as an exploration competitor.

Learned exploration and predictive control. Large-scale learned exploration, HDPlanner, DARE, and P^2 Explore learn navigation choices or unseen structure (Cao et al., 2024; Liang et al., 2025; Cao et al., 2025; Song et al., 2025). Their planar action settings motivate a separate matched comparison. World-model methods learn action-relevant predictions for control (Chua et al., 2018; Schrittwieser et al., 2020; Hafner et al., 2025; Hansen et al., 2024); Plan2Explore uses predicted

disagreement to drive exploration (Sekar et al., 2020). DeepMDP, bisimulation-based representations, and value equivalence clarify what a latent model should preserve (Gelada et al., 2019; Zhang et al., 2021; Grimm et al., 2020). Our distinction is to define frontier memory by the maneuver-dependent continuation decisions it preserves across observation-matched worlds. This criterion supplies both training targets and the compression objective through the same outcome law. These established principles are not themselves claimed as new. A learned classifier attached to a cached route supplies a direct control for predictive coupling in the ablation study.

3 Problem and Decision-Relevant Memory

Let $h_t = (o_0, a_0, \ldots, o_t)$ denote observation history, x_t the measured vehicle state, and M_t the observed occupancy map. The hidden state includes static geometry and moving occluders. We use the belief-state interpretation of a partially observable decision process (Kaelbling et al., 1998). An action is a short executable maneuver

$$a = (i, \xi, T), \qquad \xi = (p_{\text{end}}, v_{\text{end}}, \psi_{\text{end}}, \text{observation schedule}), \tag{1}$$

where i is a candidate frontier, T its duration, and ξ parameterizes approach and sensing. Different velocities and viewing directions at the same frontier are different actions. This temporal abstraction follows semi-Markov options (Sutton et al., 1999). It avoids assuming that a frontier's utility is independent of how it is approached.

The recurrent encoder produces $m_t = F_\theta(m_{t-1}, a_{t-1}, o_t, x_t, M_t)$, corresponding to the observation-to-memory update in Figure 1(a). Its target is a joint conditional law

$$K_\theta(y, m' \mid m_t, a), \quad y = (\Delta c, \Delta e, d, T, j, x', \omega), \tag{2}$$

where Δc is newly observed volume, Δe denotes frontier birth, merge, closure, or connectivity events, d is traveled distance, j is integrated squared jerk, and ω is an encoded observation. m' is the memory after processing that observation. Predicting these quantities jointly preserves correlations between revealing a passage and incurring a turn or return cost. They are not assumed conditionally independent.

We distinguish an unobserved event from a third physical class called "unknown." For example, the event "a dynamically feasible continuation becomes observed within horizon H" is binary, while uncertainty is a property of its predictive distribution. This distinction prevents epistemic uncertainty from being mislabeled as environmental geometry. Frontier identities use persistent spatial supports with recorded split/merge ancestry; ambiguous identity matches remain multiple hypotheses.

4 Counterfactual Predictive Memory

Figure 1 connects two feedback loops through one outcome model. Execution supplies new observations to update memory, while predicted action-value discrepancies determine which historical details to retain. Geometric feasibility restricts execution independently of learned task value.

4.1 A shared prediction and control objective

For bounded, normalized rewards, define

$$r(y) = w_c \Delta c - w_d d - w_t T - w_j j, \qquad \widehat{Q}(m, a) = \mathbb{E}_{K_\theta}\left[r(y) + \gamma^{T/\Delta t} \widehat{V}(m') \right]. \tag{3}$$

Normalization constants and weights are fixed using validation maps. Costs use consistent physical units before normalization. The continuation model receives updated frontier relations, visited-region summaries, and terminal vehicle state; it does not receive a ground-truth test map. The controller selects $\arg\max_{a \in \mathcal{A}_{\text{safe}}(h_t)} \widehat{Q}(m_t, a)$. It scores frontier identity, observation direction, and terminal velocity together, rather than choosing a destination and subsequently switching a separate trajectory mode.

The safe action set is constructed by an established geometric planner using observed free space, vehicle inflation, dynamics, and a feasible braking backup. Predicted free space cannot independently authorize execution. Dynamic-obstacle extrapolation is a shared experimental component,

with failures reported separately. Thus learning supplies task value, while the paper makes no new collision-free guarantee. We use a finite candidate set with continuous trajectory refinement, avoiding claims of globally optimal continuous planning.

4.2 Learning from observation-matched simulator branches

Training examples are organized by an observed prefix. We sample multiple hidden-world completions consistent with the same rendered prefix, and simulate multiple candidate maneuvers from each completion. Prefix-consistent worlds differ only where the sensor has not supplied evidence. Moving occluders use fixed exogenous random seeds within a paired comparison. This permits supervision of unchosen actions without pretending that a single real-world rollout reveals their outcomes.

Let $\mathcal{D}_{\text{fork}}$ contain (h, a, y, h') tuples and paired returns $\widehat{G}_a, \widehat{G}_b$ computed with a fixed continuation policy and horizon. Write $m = F_\theta(h)$ and $z' = \text{sg}[F_{\bar{\theta}}(h')]$. A return $\widehat{G}_h$ starts at h under the same continuation policy, horizon, and terminal bootstrap. Training minimizes

$$\begin{aligned}
\mathcal{L}_{\text{pred}} &= -\mathbb{E}_{\mathcal{D}_{\text{fork}}} \log K_\theta(y, z' \mid m, a), \\
\mathcal{L}_{\text{pair}} &= \mathbb{E}\ \text{Huber}\Big([\widehat{Q}(m,a) - \widehat{Q}(m,b)] - [\widehat{G}_a - \widehat{G}_b] \Big), \\
\mathcal{L}_{\text{value}} &= \mathbb{E}\big(\widehat{Q}(m,a) - \text{sg}[r + \gamma^{T/\Delta t} \widehat{V}_{\bar{\theta}}(z')]\big)^2 \\
&\quad + \mathbb{E}\big(\widehat{V}(m) - \text{sg}[\widehat{G}_h]\big)^2.
\end{aligned} \tag{4}$$

A stop-gradient target encoder prevents the likelihood target from moving with the predictor within an update; event and measured-cost supervision prevent a constant latent collapse. Continuous targets use likelihoods with bounded variance floors. Paired rollouts and Bellman targets use the same slowly updated continuation policy and terminal bootstrap; paired labels are refreshed when that policy changes. Terminal states have zero continuation value. A joint discrete latent mixture over frontier events and continuous outcome heads implements Equation 2; mixture components share a trunk and remain correlated through the latent variable.

We combine the three losses with the value-preservation loss under a fixed memory budget described below. Training-only privileged maps provide labels, never encoder inputs. Prefixes from the same physical layout remain in one split. Paired labels are finite-horizon estimates under the specified continuation policy, not optimal action values or causal claims about arbitrary unseen environments. An optional proposal policy uses off-policy soft actor-critic training (Haarnoja et al., 2018); the policy is an implementation option, not the claimed contribution.

4.3 Observation-contingent continuation without an entropy bonus

An informative maneuver is useful when it improves a subsequent choice. The branches in Figure 1(b) illustrate how a new view can reveal an opening or a blocked continuation, leading to different subsequent choices. They represent possible observations under K_θ; the controller executes only the selected current maneuver and replans after receiving the actual observation. For a sensing action ending at the same state under all compared observation branches, its estimated value of information is

$$\widehat{\mathcal{I}}(m,a) = \mathbb{E}_\omega \max_b \widehat{Q}(m'_\omega, b) - \max_b \mathbb{E}_\omega \widehat{Q}(m'_\omega, b). \tag{5}$$

The same continuation-action set b is used in both terms. This quantity is nonnegative by convexity of the maximum, and equals zero when every observation gives the same maximizing action. A view that reveals irrelevant texture can reduce entropy without changing Equation 5. The information value is already present in observation-contingent Bellman evaluation; we do not add it again as an independent reward. We use it as a diagnostic of observation-contingent decisions.

When action endpoints differ, Equation 5 is computed within each action, at its own terminal state. Comparing full action values in Equation 3 then accounts for travel and braking costs. Mixture-weighted outcome samples approximate the expectation, with a learned continuation value for each predicted observation. More samples can improve estimation but do not create a real-time guarantee; candidate generation, ray casting, neural inference, and trajectory feasibility all enter the measured latency.

4.4 MEMORY COMPRESSION BY DECISION DISCREPANCY

Memory holds observed-region tokens, persistent frontier-support tokens, and a recurrent summary. A compressor C_B produces a representation with at most B tokens. We train it to preserve the values of the currently feasible candidate maneuvers:

$$\mathcal{L}_{\text{keep}} = \mathbb{E} \max_{a \in \mathcal{A}_{\text{safe}}} \left| \widehat{Q}(m, a) - \widehat{Q}(C_B(m), a) \right|, \quad \mathcal{L} = \mathcal{L}_{\text{pred}} + \lambda_p \mathcal{L}_{\text{pair}} + \lambda_v \mathcal{L}_{\text{value}} + \lambda_k \mathcal{L}_{\text{keep}}. \tag{6}$$

At runtime, a shadow representation formed from retained raw observations is periodically refreshed. In Figure 1(c), ΔQ denotes the maximum absolute candidate-value discrepancy between the shadow and compressed memories, evaluated using the same K_θ and $\widehat{Q}$ over a shared feasible action set. This discrepancy guides retention and merging of memory tokens and triggers expansion or reconstruction within budget B. The shadow uses observed data only. The trigger is an empirical error indicator, not a certified bound on an unknown true value function. We charge shadow construction and retention to compute and memory totals.

Proposition 1 (Conditional one-decision regret)**.** *For a fixed history and shared feasible candidate set, assume* $\sup_a |Q^*(h, a) - \widehat{Q}(m, a)| \le \delta$ *and* $\sup_a |\widehat{Q}(m, a) - \widehat{Q}(C_B(m), a)| \le \epsilon$*. If* $\hat{a}$ *maximizes the compressed value, then* $Q^*(h, a^*) - Q^*(h, \hat{a}) \le 2(\delta + \epsilon)$*. If the best-to-second-best true margin exceeds this bound,* $\hat{a} = a^*$*.*

The proof uses two approximation errors and the maximizing inequality; it appears in Appendix A. The statement motivates value-preserving compression but does not certify learned δ, establish mission-level optimality, or cover actions excluded by candidate generation. Task-oriented state abstraction is established theory (Gelada et al., 2019; Zhang et al., 2021); the contribution is its operationalization for observation-dependent frontier memory.

4.5 TRAINING AND CLOSED-LOOP EXECUTION

Algorithm 1 in Appendix B.1 specifies the complete training and receding-horizon execution procedure. Training collects observed prefixes, samples consistent hidden completions, and evaluates candidate maneuvers under a fixed continuation-policy version. This policy is greedy with respect to the slowly updated value model over the geometric candidate set. Paired returns use the same continuation horizon and terminal bootstrap, so their differences compare actions under a common evaluation rule. Prediction, continuation value, and retention parameters are updated from these groups; target parameters are updated slowly and paired labels are refreshed when the continuation policy changes. Finite training and mission budgets bound the two stages.

Deployment freezes the learned parameters and alternates observation updates, candidate evaluation, and execution. The bounded log L_t contains only measurements and executed actions. Replaying it constructs a shadow memory s_t for periodic compression audits. Shadow and compressed memories are scored on identical feasible candidates with common model-sampling seeds. Refresh selects the lowest-discrepancy representation from a fixed number of budget-feasible retention proposals, including the incumbent. If discrepancy remains above the validation threshold τ, the controller uses the shadow values for that decision and records the exceedance. This fallback is included in memory and latency accounting. The threshold does not certify true-value error.

Before execution, the selected maneuver and its braking backup are revalidated against the latest measured state and map. Empty candidate sets or failed revalidation invoke the retained braking backup. Each executed maneuver, including a backup, enters the next observation update with its realized duration in the semi-Markov target. The operator $C_B(\cdot; \mathcal{A}_t)$ conditions retention on the feasible candidates; geometric mapping and backup validation retain their own state independently. Predicted observations are used only for scoring and never enter the observed occupancy map. Termination uses controlled braking and returns the observation, action, and resource logs.

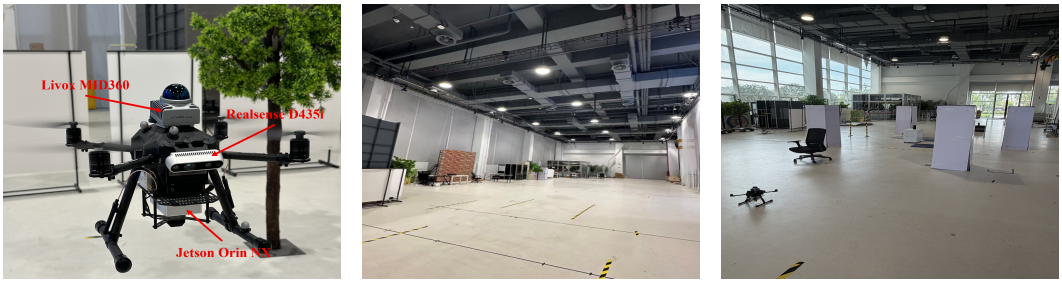
5 EXPERIMENTAL EVALUATION

5.1 FAIR COMPARISONS AND DATA PROVENANCE

Table 1 compares CPM with ten recent aerial exploration methods. LiDAR and limited-FoV tracks use separate sensor configurations and reference rows. Within each track, the comparison fixes sen-

Table 1: Main aerial comparison. Subscripts report signed directions and absolute differences relative to CPM within each sensor track. Header arrows indicate the preferred metric direction; sensor tracks are not pooled.

Method	$T_{99}\downarrow$ (s)	Distance↓ (m)	Plan↓ (ms)	Coverage↑ (%)
LiDAR track				
EPIC (Geng et al., 2025)	$318.4_{(\uparrow 66.8)}$	$987.2_{(\uparrow 113.2)}$	$53.6_{(\downarrow 35.7)}$	$99.87_{(\downarrow 0.04)}$
RUSH (Bu et al., 2025)	$279.3_{(\uparrow 27.7)}$	$890.4_{(\uparrow 16.4)}$	$85.7_{(\downarrow 3.6)}$	$99.90_{(\downarrow 0.01)}$
EPICON (Yuan et al., 2026)	$259.8_{(\uparrow 8.2)}$	$861.5_{(\downarrow 12.5)}$	$75.1_{(\downarrow 14.2)}$	$99.94_{(\uparrow 0.03)}$
PACE (Gao et al., 2026)	$281.2_{(\uparrow 29.6)}$	$902.4_{(\uparrow 28.4)}$	$124.8_{(\uparrow 35.5)}$	$99.88_{(\downarrow 0.03)}$
CPM (ours)	251.6	874.0	89.3	99.91
Limited-FoV depth track				
FAEP (Zhao et al., 2024)	$645.7_{(\uparrow 138.3)}$	$1053.4_{(\uparrow 75.2)}$	$112.5_{(\uparrow 29.8)}$	$99.14_{(\downarrow 0.16)}$
FALCON (Zhang et al., 2025b)	$548.6_{(\uparrow 41.2)}$	$979.1_{(\uparrow 0.9)}$	$79.2_{(\downarrow 3.5)}$	$99.32_{(\uparrow 0.02)}$
SCOPE (Li et al., 2026)	$519.2_{(\uparrow 11.8)}$	$992.3_{(\uparrow 14.1)}$	$24.9_{(\downarrow 57.8)}$	$99.25_{(\downarrow 0.05)}$
HIGHSTAR (Dong et al., 2025)	$521.8_{(\uparrow 14.4)}$	$1001.4_{(\uparrow 23.2)}$	$86.1_{(\uparrow 3.4)}$	$99.24_{(\downarrow 0.06)}$
FASTEX (Zhang et al., 2025a)	$550.3_{(\uparrow 42.9)}$	$955.8_{(\downarrow 22.4)}$	$63.8_{(\downarrow 18.9)}$	$99.21_{(\downarrow 0.09)}$
KAEP (Mendes et al., 2026)	$564.2_{(\uparrow 56.8)}$	$1038.6_{(\uparrow 60.4)}$	$72.4_{(\downarrow 10.3)}$	$99.38_{(\uparrow 0.08)}$
CPM (ours)	507.4	978.2	82.7	99.30



(a) Platform (b) Arena (c) Task scene

Figure 2: Representative real-world deployment context: (a) aerial platform and payload, (b) indoor flight arena, and (c) a cluttered task scene.

sor rays, map bounds, vehicle dynamics, completion thresholds, and hardware budgets. Replanning latency includes mapping, candidate generation, neural inference, and feasibility checks, so differences in map interfaces are reflected in the reported cost.

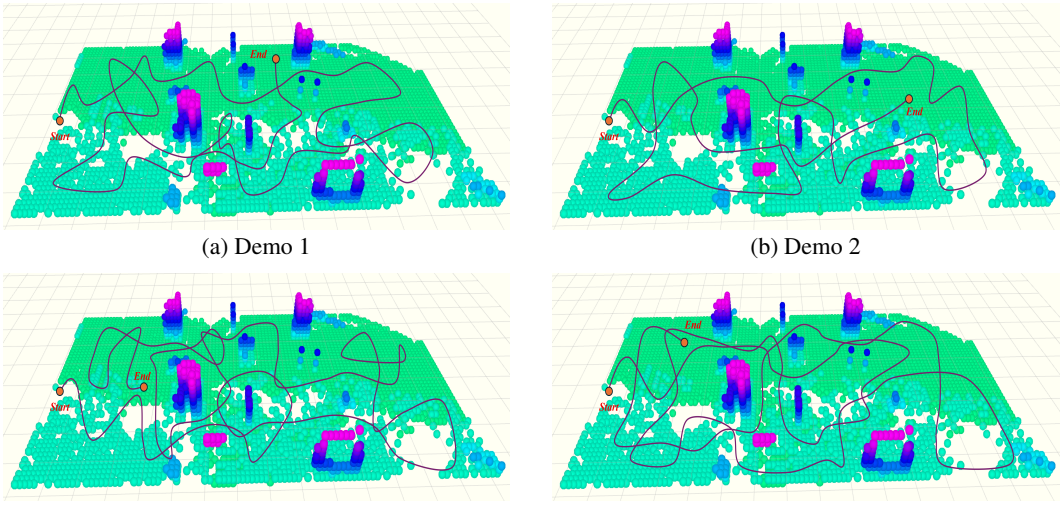
Tables report within-track comparisons. Subscript arrows indicate the sign of the difference between a baseline and the CPM reference in the same panel; the accompanying magnitude is the absolute difference. Metric-header arrows indicate the preferred direction. Sensor tracks are evaluated separately.

The aerial benchmark uses forest, maze, and indoor layouts; Table 1 reports the forest comparison. Map dimensions are $100 \times 100 \times 6\,\mathrm{m}$ for LiDAR and $50 \times 50 \times 4\,\mathrm{m}$ for limited-FoV depth. The corresponding maximum velocities are $5.0/2.5\,\mathrm{m/s}$, acceleration is $2.0\,\mathrm{m/s^2}$, and sensing ranges are $30.0/4.5\,\mathrm{m}$. Sensor ray geometry is fixed within each track. MARSIM provides the point-cloud simulation framework (Kong et al., 2023); FAST-LIO2 supplies a compatible LiDAR-inertial estimation interface for deployment (Xu et al., 2022).

Figure 2 shows the platform, bounded arena, and occluded task scene that motivate frontier-aware deployment; these photographs are separate from the controlled benchmark in Table 1.

Figure 3 shows four recorded task runs in the same indoor deployment setting. The trajectories cover different portions of the cluttered scene and terminate at different locations, illustrating repeated closed-loop execution under the same sensing and planning stack. These demonstrations are qualitative deployment evidence; the controlled benchmark metrics remain those reported in Table 1.

The reported metrics are time to 99% reachable-volume coverage (T_{99}), traveled distance D at that threshold, mean end-to-end replanning latency (Plan), and final coverage C. Reachability and



(a) Demo 1 (b) Demo 2

(c) Demo 3 (d) Demo 4

Figure 3: Four representative real-world demonstrations visualized in RViz. Each panel shows the reconstructed point-cloud scene, the executed exploration trajectory, and the start/end markers for one task run.

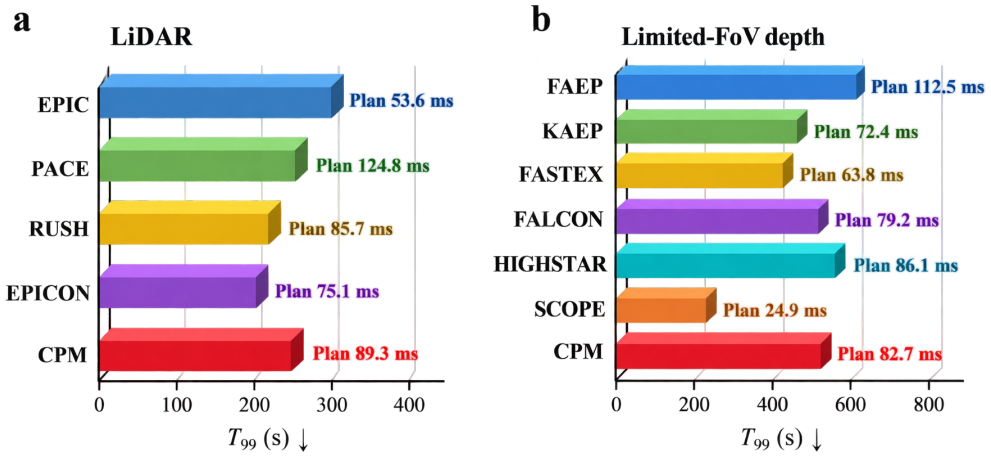


Figure 4: Completion time and replanning latency in the forest comparisons (Table 1). Horizontal bars show reported T_{99} values; the label on each bar gives the corresponding mean planning latency. Sensor tracks are ranked and compared within panels.

coverage denominators are computed by the evaluator from ground truth and withheld from planners. Mission caps are 900 s for LiDAR and 1,500 s for depth. In aggregate evaluation, timeouts and collisions receive the common capped completion time; distance at T_{99} is conditioned on reaching the threshold. Final coverage is measured at the cap or complete coverage. Common start and noise seeds support paired comparisons. The tables present point estimates; their numerical differences do not establish statistical significance (Agarwal et al., 2021).

Aerial results. CPM reduces LiDAR completion time from 281.2 s for PACE to 251.6 s, a 10.5% reduction, while lowering traveled distance by 28.4 m. EPICON travels 12.5 m less and attains 0.03 percentage points more coverage than CPM, but takes 8.2 s longer. In the depth track, CPM completes in 507.4 s, 11.8 s faster than SCOPE, the next fastest method. SCOPE has lower replanning latency (24.9 versus 82.7 ms), FASTEX has shorter distance, and KAEP has higher final coverage. These comparisons support a completion-time advantage without implying dominance on every metric.

Table 2: Ablation study in the LiDAR track. P: predictive coupling; F: paired forks; R: decision-based retention; O: observation-contingent continuation. ✓/×: enabled/disabled. Brier measures continuation-event prediction error.

Variant	P	F	R	O	$T_{99}\downarrow$ (s)	Distance↓ (m)	Brier↓
w/o predictive coupling	×	✓	✓	✓	$282.3_{(\uparrow 30.7)}$	$927.1_{(\uparrow 53.1)}$	$0.142_{(\uparrow 0.059)}$
w/o paired forks	✓	×	✓	✓	$273.9_{(\uparrow 22.3)}$	$905.6_{(\uparrow 31.6)}$	$0.127_{(\uparrow 0.044)}$
w/o decision retention	✓	✓	×	✓	$267.5_{(\uparrow 15.9)}$	$920.3_{(\uparrow 46.3)}$	$0.086_{(\uparrow 0.003)}$
w/o contingent continuation	✓	✓	✓	×	$263.2_{(\uparrow 11.6)}$	$896.8_{(\uparrow 22.8)}$	$0.081_{(\downarrow 0.002)}$
CPM (ours)	✓	✓	✓	✓	251.6	874.0	0.083

Table 3: Matched planar learning comparison with common 2-D actions and observations. Subscript differences are relative to CPM in this setting.

Method	Distance↓ (m)	Plan↓ (ms)	Coverage↑ (%)
ARiADNE (large-scale) (Cao et al., 2024)	$1058.0_{(\uparrow 72.0)}$	$69.4_{(\downarrow 19.1)}$	$99.20_{(\downarrow 0.20)}$
HDPlanner (Liang et al., 2025)	$994.0_{(\uparrow 8.0)}$	$81.2_{(\downarrow 7.3)}$	$99.30_{(\downarrow 0.10)}$
DARE (Cao et al., 2025)	$978.0_{(\downarrow 8.0)}$	$107.6_{(\uparrow 19.1)}$	$99.50_{(\uparrow 0.10)}$
P^2 Explore (Song et al., 2025)	$1012.0_{(\uparrow 26.0)}$	$76.3_{(\downarrow 12.2)}$	$99.10_{(\downarrow 0.30)}$
CPM (planar)	986.0	88.5	99.40

Figure 4 makes the latency–completion tradeoff explicit: CPM has the shortest reported completion time in both panels, while several alternatives have shorter planning-latency labels. Mean computation time alone therefore does not explain mission completion time.

5.2 Mechanism tests, learning controls, and ablations

Observation aliasing is examined through geometrically similar entrances: an occluded passage, a closed recess, and a branch whose opening does not alter the preferred route. These cases distinguish useful uncertainty resolution from generic entropy reduction; Appendix C specifies the diagnostic constructions. The ablation table reports the Brier score for the event that a dynamically feasible continuation becomes observed within the prediction horizon. Lower scores indicate more accurate probabilistic predictions. Calibration methods and ensembles motivate this diagnostic but do not provide execution-safety guarantees (Guo et al., 2017; Lakshminarayanan et al., 2017).

The ablations in Table 2 alter supervision or coupling while retaining candidate generation, execution constraints, training budget, and as much parameter capacity as possible. "w/o predictive coupling" retains a separately trained passability head and a cached route, giving the direct module-addition control. "w/o paired forks" uses an equal number of ordinary rollouts. "w/o decision retention" uses uniform-age retention with the same token budget. "w/o contingent continuation" commits to a continuation before receiving the hypothetical observation. This last variant tests the mechanism in Equation 5, not merely the presence of a neural network.

Ablation results. All four ablations increase completion time relative to full CPM. Removing predictive coupling produces the largest increase (30.7 s), followed by removing paired forks (22.3 s), decision-based retention (15.9 s), and contingent continuation (11.6 s). The last variant slightly improves Brier score (0.081 versus 0.083) while increasing time and distance. Thus aggregate prediction accuracy alone does not determine exploration efficiency.

Figure 5(a) summarizes the cost of removing each component. In panel (b), the $-O$ bar is slightly shorter than the CPM bar, indicating a lower Brier score, while panel (a) shows its larger completion-time penalty. The distinction supports evaluating decision consequences alongside prediction error.

Table 3 compares learned exploration methods within a common planar action space, robot graph, and occupancy observation interface. CPM is restricted to those actions and receives no extra 3-D cues. Its 986.0 m path is shorter than those of ARiADNE, HDPlanner, and P^2 Explore, while DARE achieves 978.0 m and 0.10 percentage points higher coverage. CPM reduces latency relative

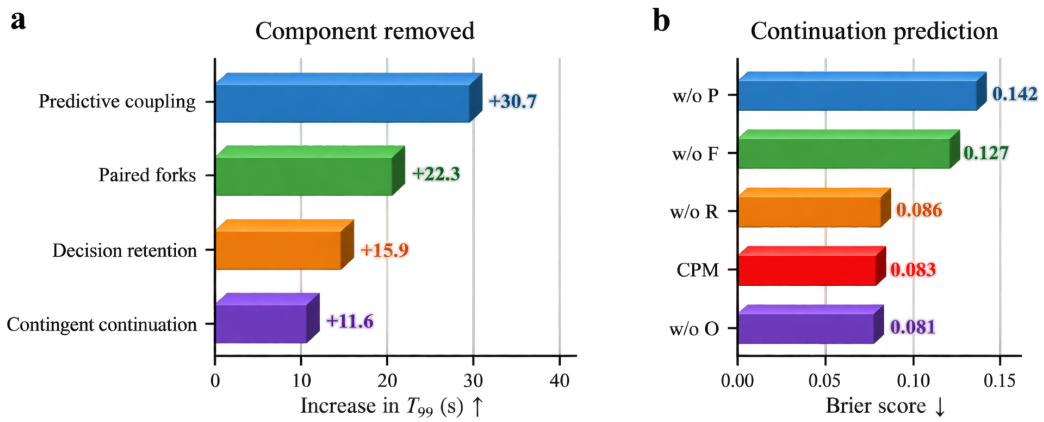


Figure 5: Component contributions from Table 2. (a) Horizontal bars show completion-time increases relative to full CPM. (b) Horizontal bars show continuation-event Brier scores for the four ablations and full CPM; $-P$, $-F$, $-R$, and $-O$ remove predictive coupling, paired forks, decision retention, and contingent continuation, respectively. Bars are aggregate point estimates without inferred uncertainty intervals.

to DARE by 19.1 ms, but has higher latency than the other three methods. The planar comparison therefore characterizes an efficiency tradeoff rather than a universal advantage over learned policies.

5.3 Computational scope

The reported Plan metric measures mean end-to-end replanning latency. Sparse frontier attention limits one component of computation, but ray casting, candidate construction, and the periodic shadow audit also contribute. Task-memory budget B excludes the occupancy map, safety state, and bounded observation log; total-system memory includes all of them. The comparisons in the tables do not establish tail-latency bounds, total-memory savings, or hardware-transfer performance. These distinctions delimit the empirical claims and the conditional analysis.

The implementation uses a 128-dimensional recurrent state and an eight-component outcome mixture. Deployment evaluates at most 32 candidate maneuvers per decision, using a learned observation-contingent continuation value. Explicit depth-two search is an optional extension and is disabled in this configuration. The reported behavior therefore depends on both the candidate budget and the learned continuation model. Increasing the number of candidates changes the computational workload and the set over which values are compared; it does not change the stated memory-budget definition. Appendix B gives the representation and compression details.

6 Limitations and Conclusion

Counterfactual training requires a simulator distribution that covers plausible hidden completions. A learned memory cannot recover information absent from the observed prefix, and correlated prediction errors can make observation branching overconfident. Finite candidates restrict optimality, frontier split/merge errors can corrupt history, and finite memory can discard a distant branch needed later. These mechanisms limit robustness outside the evaluated conditions. Deployment remains dependent on sensing, localization, and a conservative execution filter.

Counterfactual predictive memory links frontier history, approach motion, and observation-contingent continuation through a shared action-conditioned outcome law. The forest comparisons show lower completion times within both sensor tracks, while component ablations connect the full method to improved exploration efficiency. The conditional regret result explains how value-preserving compression can retain a preferred action under uniform approximation assumptions. The latency, distance, and coverage comparisons also show that faster completion can involve higher replanning latency, longer flight paths, or lower final coverage.

AI USE STATEMENT

AI tools were used only for language polishing, grammar checking, wording refinement, and assistance with understanding code and troubleshooting simple issues. The authors take full responsibility for the final manuscript, including its scientific claims and results.

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 *Advances in Neural Information Processing Systems*, volume 34, pp. 29304–29320, 2021.

Andreas Bircher, Mina Kamel, Kostas Alexis, Helen Oleynikova, and Roland Siegwart. Receding horizon "next-best-view" planner for 3D exploration. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1462–1468, 2016. doi: 10.1109/ICRA.2016.7487281.

Ningbo Bu, Gen Xu, Hao Zheng, Xuehang Wei, Wenshi Chen, Li Lv, Xiaolu Zhang, Jiangjian Xiao, and Zhiqiang Li. RUSH: Rapid UAV spatial hierarchical exploration via regional viewpoint generation for large-scale environments. *IEEE Robotics and Automation Letters*, 10(10):10698–10705, 2025. doi: 10.1109/LRA.2025.3604699.

Chao Cao, Hongbiao Zhu, Howie Choset, and Ji Zhang. TARE: A hierarchical framework for efficiently exploring complex 3D environments. In *Robotics: Science and Systems*, 2021. doi: 10.15607/RSS.2021.XVII.018.

Yuhong Cao, Rui Zhao, Yizhuo Wang, Bairan Xiang, and Guillaume Sartoretti. Deep reinforcement learning-based large-scale robot exploration. *IEEE Robotics and Automation Letters*, 9(5):4631–4638, 2024. doi: 10.1109/LRA.2024.3379804.

Yuhong Cao, Jeric Lew, Jingsong Liang, Jin Cheng, and Guillaume Sartoretti. DARE: Diffusion policy for autonomous robot exploration. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 11987–11993, 2025. doi: 10.1109/ICRA55743.2025.11128196.

Kurtland Chua, Roberto Calandra, Rowan McAllister, and Sergey Levine. Deep reinforcement learning in a handful of trials using probabilistic dynamics models. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Qianli Dong, Xuebo Zhang, Shiyong Zhang, Ziyu Wang, Zhe Ma, Tianyi Li, and Haobo Xi. HIGH-STAR: High-speed and efficient online autonomous UAV exploration. *IEEE Transactions on Automation Science and Engineering*, 22:20770–20787, 2025. doi: 10.1109/TASE.2025.3605375.

Daniel Duberg and Patric Jensfelt. UFOExplorer: Fast and scalable sampling-based exploration with a graph-based planning structure. *IEEE Robotics and Automation Letters*, 7(2):2487–2494, 2022. doi: 10.1109/LRA.2022.3142923.

Yuxiang Gao, Zhuoxuan Wang, Xianlu Tao, Jinle Xu, Shuguo Pan, and Wang Gao. PACE: Passability and memory integrated exploration for UAVs in large-scale and cluttered environments. *IEEE Robotics and Automation Letters*, 11(9):10282–10289, 2026. doi: 10.1109/LRA.2026.3711825.

Carles Gelada, Saurabh Kumar, Jacob Buckman, Ofir Nachum, and Marc G. Bellemare. DeepMDP: Learning continuous latent space models for representation learning. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pp. 2170–2179. PMLR, 2019.

Shuang Geng, Zelin Ning, Fu Zhang, and Boyu Zhou. EPIC: A lightweight LiDAR-based AAV exploration framework for large-scale scenarios. *IEEE Robotics and Automation Letters*, 10(5): 5090–5097, 2025. doi: 10.1109/LRA.2025.3555878.

Christopher Grimm, André Barreto, Satinder P. Singh, and David Silver. The value equivalence principle for model-based reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 33, pp. 5541–5552, 2020.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70, pp. 1321–1330. PMLR, 2017.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pp. 1861–1870. PMLR, 2018.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, 640(8059):647–653, 2025. doi: 10.1038/s41586-025-08744-2.

Nicklas Hansen, Hao Su, and Xiaolong Wang. TD-MPC2: Scalable, robust world models for continuous control. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=Oxh5CstDJU.

Junlong Huang, Boyu Zhou, Zhengping Fan, Yilin Zhu, Yingrui Jie, Longwei Li, and Hui Cheng. FAEL: Fast autonomous exploration for large-scale environments with a mobile robot. *IEEE Robotics and Automation Letters*, 8(3):1667–1674, 2023. doi: 10.1109/LRA.2023.3236573.

Leslie Pack Kaelbling, Michael L. Littman, and Anthony R. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 101(1–2):99–134, 1998. doi: 10.1016/S0004-3702(98)00023-X.

Fanze Kong, Xiyuan Liu, Benxu Tang, Jiarong Lin, Yunfan Ren, Yixi Cai, Fangcheng Zhu, Nan Chen, and Fu Zhang. MARSIM: A light-weight point-realistic simulator for LiDAR-based UAVs. *IEEE Robotics and Automation Letters*, 8(5):2954–2961, 2023. doi: 10.1109/LRA.2023.3264163.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Kai Li, Shengtao Zheng, Lingkun Xiu, Yuze Sheng, Xiao-Ping Zhang, Dongyue Huang, and Xinlei Chen. SCOPE: Skeleton graph-based computation-efficient framework for autonomous UAV exploration. *IEEE Robotics and Automation Letters*, 11(4):5270–5277, 2026. doi: 10.1109/LRA.2026.3671559.

Jingsong Liang, Yuhong Cao, Yixiao Ma, Hanqi Zhao, and Guillaume Sartoretti. HDPlanner: Advancing autonomous deployments in unknown environments through hierarchical decision networks. *IEEE Robotics and Automation Letters*, 10(1):256–263, 2025. doi: 10.1109/LRA.2024.3506281.

Michael L. Littman, Richard S. Sutton, and Satinder Singh. Predictive representations of state. In *Advances in Neural Information Processing Systems*, volume 14. MIT Press, 2001.

Shijun Long, Ying Li, Chenming Wu, Bin Xu, and Wei Fan. HEROS: Hierarchical exploration with online subregion updating for 3D environment coverage. In *IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pp. 1–8, 2025. doi: 10.1109/AIM64088.2025.11175892.

João Félix Mendes, Meysam Basiri, and Rodrigo Ventura. Kinodynamic trajectory planning for efficient UAV exploration and reconstruction of unknown environments. *IEEE Robotics and Automation Letters*, 11(2):1530–1537, 2026. doi: 10.1109/LRA.2025.3641147.

Yunfan Ren, Fangcheng Zhu, Guozheng Lu, Yixi Cai, Longji Yin, Fanze Kong, Jiarong Lin, Nan Chen, and Fu Zhang. Safety-assured high-speed navigation for MAVs. *Science Robotics*, 10(98): eado6187, 2025. doi: 10.1126/scirobotics.ado6187.

Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, Timothy Lillicrap, and David Silver. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588 (7839):604–609, 2020. doi: 10.1038/s41586-020-03051-4.

Ramanan Sekar, Oleh Rybkin, Kostas Daniilidis, Pieter Abbeel, Danijar Hafner, and Deepak Pathak. Planning to explore via self-supervised world models. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pp. 8583–8592. PMLR, 2020.

Kun Song, Gaoming Chen, Masayoshi Tomizuka, Wei Zhan, Zhenhua Xiong, and Mingyu Ding. P^2 Explore: Efficient exploration in unknown cluttered environment with floor plan prediction. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 13090–13096, 2025. doi: 10.1109/IROS60139.2025.11247205.

Richard S. Sutton, Doina Precup, and Satinder Singh. Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence*, 112(1–2):181–211, 1999. doi: 10.1016/S0004-3702(99)00052-1.

Benxu Tang, Yunfan Ren, Fangcheng Zhu, Rui He, Siqi Liang, Fanze Kong, and Fu Zhang. Bubble explorer: Fast UAV exploration in large-scale and cluttered 3D-environments using occlusion-free spheres. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1118–1125, 2023. doi: 10.1109/IROS55552.2023.10342348.

Zhepei Wang, Xin Zhou, Chao Xu, and Fei Gao. Geometrically constrained trajectory optimization for multicopters. *IEEE Transactions on Robotics*, 38(5):3259–3278, 2022. doi: 10.1109/TRO.2022.3160022.

Wei Xu, Yixi Cai, Dongjiao He, Jiarong Lin, and Fu Zhang. FAST-LIO2: Fast direct LiDAR-inertial odometry. *IEEE Transactions on Robotics*, 38(4):2053–2073, 2022. doi: 10.1109/TRO.2022.3141876.

Brian Yamauchi. A frontier-based approach for autonomous exploration. In *IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA)*, pp. 146–151, 1997. doi: 10.1109/CIRA.1997.613851.

Zikang Yuan, Yuan Ren, Yixue Wang, Leyi Zhao, Shangzhe Sun, Chi Chen, Lijun Zhu, Xin Yang, and Kwang-Ting Cheng. Aerial exploration on point cloud maps via coverage path guidance. *IEEE Robotics and Automation Letters*, 11(10):11299–11306, 2026. doi: 10.1109/LRA.2026.3723740.

Amy Zhang, Rowan McAllister, Roberto Calandra, Yarin Gal, and Sergey Levine. Learning invariant representations for reinforcement learning without reconstruction. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=-2FCwDKRREu.

Xiaoxun Zhang, Peiming Duan, Lanxiang Zheng, Junlong Huang, and Hui Cheng. FASTEX: Fast UAV exploration in large-scale environments using dynamically expanding grids and coverage paths. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3541–3548, 2025a. doi: 10.1109/IROS60139.2025.11246816.

Yichen Zhang, Xinyi Chen, Chen Feng, Boyu Zhou, and Shaojie Shen. FALCON: Fast autonomous aerial exploration using coverage path guidance. *IEEE Transactions on Robotics*, 41:1365–1385, 2025b. doi: 10.1109/TRO.2024.3522148.

Yinghao Zhao, Li Yan, Hong Xie, Jicheng Dai, and Pengcheng Wei. Autonomous exploration method for fast unknown environment mapping by using UAV equipped with limited FOV sensor. *IEEE Transactions on Industrial Electronics*, 71(5):4933–4943, 2024. doi: 10.1109/TIE.2023.3285921.

Boyu Zhou, Yichen Zhang, Xinyi Chen, and Shaojie Shen. FUEL: Fast UAV exploration using incremental frontier structure and hierarchical planning. *IEEE Robotics and Automation Letters*, 6(2):779–786, 2021a. doi: 10.1109/LRA.2021.3051563.

Xin Zhou, Zhepei Wang, Hongkai Ye, Chao Xu, and Fei Gao. EGO-Planner: An ESDF-free gradient-based local planner for quadrotors. *IEEE Robotics and Automation Letters*, 6(2):478–485, 2021b. doi: 10.1109/LRA.2020.3047728.

A Conditional Regret Analysis

Let $\widetilde{Q}(a) = \widehat{Q}(C_B(m), a)$ and $e = \delta + \epsilon$. By the assumptions in Proposition 1, $|Q^*(h,a) - \widetilde{Q}(a)| \leq e$ for every common feasible candidate. Hence

$$\begin{aligned} Q^*(h,a^*) - Q^*(h,\hat{a}) &\leq Q^*(h,a^*) - \widetilde{Q}(a^*) + \widetilde{Q}(\hat{a}) - Q^*(h,\hat{a}) \\ &\leq 2e, \end{aligned} \tag{7}$$

where the first inequality uses $\widetilde{Q}(\hat{a}) \geq \widetilde{Q}(a^*)$. A true best-action gap strictly above $2e$ precludes any other maximizer. This is a pointwise guarantee conditional on uniform approximation, not an empirical certificate.

For completeness, consider a discounted fixed-duration abstraction with bounded rewards $|r| \leq R_{\max}$, discount $\gamma < 1$, reward error at most ϵ_r, and transition total-variation error at most ϵ_p. Suppose the true and approximate models share a state/action abstraction, and $\widehat{Q}$ differs from the approximate model's optimal action value by at most ϵ_{solve}. With $\|V\|_\infty \leq R_{\max}/(1-\gamma)$ in both models and the convention $\mathrm{TV}(P,Q) = \frac{1}{2}\|P - Q\|_1$, Bellman perturbation and the triangle inequality give

$$\delta \leq \frac{\epsilon_r}{1-\gamma} + \frac{2\gamma R_{\max}\epsilon_p}{(1-\gamma)^2} + \epsilon_{\text{solve}}. \tag{8}$$

The factor two is required for signed bounded values. The one-step expectation error is bounded by $2\|V\|_\infty\epsilon_p$; Bellman contraction accumulates it by $1/(1-\gamma)$. The term ϵ_{solve} is zero only for exact planning in the approximate model; fitting returns under a fixed continuation policy does not establish this condition. The semi-Markov case additionally requires duration error control and a discount uniformly bounded below one for every option. We do not claim these uniform errors are known for the learned system. Candidate approximation, changed feasibility sets, and localization error lie outside this statement.

B Concrete Implementation Specification

Representation. The model uses 128-dimensional recurrent memory, masked frontier-support attention, and an eight-component outcome mixture. Attention is restricted to spatially adjacent supports. Candidate action tokens include relative position, terminal velocity, yaw, duration, current clearance, and planned observation times. Absolute world coordinates are excluded from learned features; relative distances are normalized by sensor range while vehicle size and dynamics remain explicit inputs. Support ancestry records splits and merges. Ambiguous associations retain multiple hypotheses, keeping geometrically adjacent but topologically disconnected frontiers separate.

Data and optimization. Training, validation, and test partitions are separated by physical layout; all hidden completions and action forks of a layout remain in the same partition. At selected prefixes, eight maneuvers are simulated under four hidden completions. Completions must reproduce every prefix ray within the specified sensor tolerance. Common random numbers reduce the variance of action differences. Optimization uses Adam with learning rate 3×10^{-4} and five independent seeds; loss weights are selected on validation families. No optimal-label claim is made for finite-horizon continuation rollouts.

Planning. The deployment configuration bounds the action set at 32 candidates and evaluates an eight-component outcome mixture with a learned observation-contingent continuation value. Explicit depth-two search is an optional extension, not enabled in the deployment configuration. Each action is a short executable segment followed by an observation update. Cached quantities are invalidated when their supporting map regions or vehicle-state dependencies change. If feasibility filtering leaves no candidate, the controller executes the retained braking backup and regenerates candidates after the next observation. A learned prediction cannot overrule failed collision checks. Candidate counts and branch depth are experimental variables, not hidden budget adjustments for difficult scenes.

Compression. The retention operator accepts a fixed token budget B independently of the occupancy-map resolution. A candidate-conditioned token scorer selects the top B tokens; a continuous gate or straight-through estimator supplies training gradients, while validation evaluates the hard selection. The shared value-discrepancy loss trains the scorer without a separate relevance label. The execution filter retains raw observed geometry independently of learned task memory; total-system memory also includes this geometric state. Shadow-memory construction uses a bounded log of observed supports and is charged to runtime. The value-disagreement threshold is selected on held-out validation histories. A bounded refresh search includes the incumbent representation and therefore does not increase discrepancy against the same shadow scores. Persistent exceedances use the shadow scores for that decision, as specified in Algorithm 1; this empirical rule does not certify the assumptions of Proposition 1.

B.1 Training and execution algorithm

Algorithm 1 Counterfactual predictive memory: training and execution

Require: Training layouts; token budget B; audit period $P \geq 1$; threshold τ; observation-log capacity $L_{\max}$; finite training and mission budgets

Initialize parameters θ, target parameters $\bar{\theta} \leftarrow \theta$, and fork buffer $\mathcal{D}$
for each training round within the training budget **do**
  Collect observed prefixes using the current geometrically filtered policy
  for each selected prefix h **do**
    Construct $\mathcal{A}_{\mathrm{safe}}(h)$ from observed geometry; skip h if the set is empty
    Sample hidden completions reproducing the observed prefix
    Simulate each $a \in \mathcal{A}_{\mathrm{safe}}(h)$ in each completion
    Evaluate the frozen continuation policy from h to obtain $\widehat{G}_h$
    Append (h, a, y, h'), $\widehat{G}_h$, and paired returns under that policy to $\mathcal{D}$
  end for
  Sample fork groups from $\mathcal{D}$; encode histories and stop-gradient target memories
  Update prediction, value, and retention parameters using Equations 4 and 6
  Fit $\widehat{V}$ to continuation returns with the same horizon and terminal bootstrap
  Slowly update $\bar{\theta}$; refresh paired labels when the continuation policy changes
end for
Freeze θ; initialize m_{-1}, empty L_{-1}, initial map, and a feasible braking backup
for $t = 0, 1, \ldots$ until a mission stopping signal or the deadline **do**
  Acquire o_t, x_t; update M_t and append observations/actions to bounded log L_t
  $u_t \leftarrow F_\theta(m_{t-1}, a_{t-1}, o_t, x_t, M_t)$ ▷ null action at $t = 0$
  $\mathcal{A}_t \leftarrow \mathcal{A}_{\mathrm{safe}}(h_t)$
  if $\mathcal{A}_t = \varnothing$ **then**
    $m_t \leftarrow u_t$ with its most recent B tokens and recurrent summary
    Execute the retained braking backup; record it as a_t; continue to the next observation
  end if
  $m_t \leftarrow C_B(u_t; \mathcal{A}_t)$
  $q_t(a) \leftarrow \widehat{Q}(m_t, a)$ for all $a \in \mathcal{A}_t$
  if $t \bmod P = 0$ **then**
    $s_t \leftarrow \mathrm{Replay}_\theta(L_t)$; $q_t^s(a) \leftarrow \widehat{Q}(s_t, a)$ for $a \in \mathcal{A}_t$
    $\Delta_t \leftarrow \max_{a \in \mathcal{A}_t} |q_t^s(a) - q_t(a)|$
    if $\Delta_t > \tau$ **then**
      Refresh m_t from s_t within B tokens; recompute q_t and Δ_t
      if $\Delta_t > \tau$ **then**
        $q_t \leftarrow q_t^s$; record the threshold exceedance
      end if
    end if
  end if
  $a_t \leftarrow \arg\max_{a \in \mathcal{A}_t} q_t(a)$ using a fixed tie-breaking rule
  Revalidate a_t and its braking backup against the latest measured state and map
  If valid, execute a_t and retain its validated braking backup
  Otherwise, execute the retained braking backup and record it as a_t
end for
Terminate with controlled braking; return the observation, action, and resource logs

C Counterexamples and Negative Controls

Passability versus exploration value. Construct two entrances with the same visible endpoint separation. One opens into a short pocket and the other into a long connected corridor. Then construct a small untraversable recess whose unseen surface requires a detour if deferred. These cases test whether a geometric width proxy is sufficient for exploration value. They do not demonstrate that PACE must fail on every such case, since it updates labels and considers travel cost.

A viewpoint-distance diagnostic. PACE's Equation (13) squares an unnormalized soft minimum of sampled viewpoint distances. If all K distances equal d, this expression becomes $(d - \alpha \log K)^2$. Its minimum occurs at $d = \alpha \log K$, not $d = 0$. This algebraic example identifies a possible proxy bias and dependence on sampling count. Whether it changes flight quality depends on the other objective terms and must be tested. Correcting that scalar proxy alone would be an implementation improvement, not the learning contribution proposed here.

Information that cannot change a decision. Add a visually ambiguous region whose possible completions all preserve the same best continuation action. The model should not pay a large detour cost solely to reduce uncertainty there. Then change one completion to open a valuable alternate route while keeping current local geometry fixed. The expected value of observation should rise. This paired construction distinguishes task-relevant information from an entropy bonus.

Against modular additions. A modular control attaches to PACE (i) a supervised passability predictor and (ii) the same predictor plus a recurrent memory, trained with the same fork data. The comparison matches task-memory capacity and the candidate executor wherever possible. An additional diagnostic control can use a TD-MPC2-inspired latent value model with the same safe option set; this is a task-specific adaptation, distinct from the published TD-MPC2 benchmark results. Failure to improve over these controls weakens the claim that the counterfactual predictive representation itself is useful.

D Metric Definitions and Table Conventions

Reachable volume is the evaluation-only set of free voxels reachable by the inflated vehicle under benchmark connectivity rules. Coverage counts observed voxels in that set, with identical treatment of occlusions and maximum-range rays. Dynamic objects are excluded from the static coverage denominator and used for collision and interruption metrics. Mean mission speed is total traveled distance divided by total elapsed time; moving-only speed is a distinct metric and is not interchangeable with it.

For a baseline value x and the full-method value x_{ours} in the same panel, the subscript is $(\uparrow |x - x_{\mathrm{ours}}|)$ when $x > x_{\mathrm{ours}}$, $(\downarrow |x - x_{\mathrm{ours}}|)$ when $x < x_{\mathrm{ours}}$, and $(= 0)$ for equality. This convention denotes arithmetic direction independently of metric desirability. Gray shading identifies the reference row. Comparisons across sensor tracks use their own reference values.