MERC-WM: MECHANISM-EVENT REPLAY CARDS FOR TASK-FREE CONTINUAL WORLD MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

World models deployed in physical systems encounter a stream in which camera appearance, contact dynamics, payload, and actuator response change without task labels. Updating on every prediction error wastes adaptation capacity on harmless visual variation, whereas ignoring errors caused by changed dynamics yields unsafe plans. We formulate this ambiguity as *task-free continual mechanism identification* and propose MERC-WM, a continual world-model framework that separates observation shifts from mechanism shifts before updating its dynamics. MERC-WM combines paired low-risk probes, a frozen visual representation, a bank of low-rank dynamics experts, and a bounded memory of mechanism-event replay cards. Each card stores a compact before/after latent effect, uncertainty, and expert identity rather than raw trajectories. At test time, retrieved cards provide a prior over the active mechanism; the world model either reuses an existing expert or creates a new one only when probe evidence remains inconsistent. We specify a reproducible evaluation across contact-rich manipulation streams with appearance changes, friction and payload changes, and mechanism recurrence. Across three manipulation domains and five stream seeds, MERC-WM reaches 80.6% average success, versus 78.8% for a memory-, expert-, and update-matched residual-MoE control. It reduces visual-shift false expansion from 18.7% to 8.6% at the same 18 MB budget; relative to latent replay, it improves backward transfer from -3.2 to -0.6 points and shortens recurrence recovery from 353 to 147 interactions.

1 INTRODUCTION

World models encode experience for action-conditioned prediction and control (Ha & Schmidhuber, 2018; Hafner et al., 2019). Latent imagination scales behavior learning (Hafner et al., 2020; 2021; 2025), task-oriented models support MPC (Hansen et al., 2022; 2024), and feature-space or joint-embedding prediction avoids pixel generation (Zhou et al., 2025; Maes et al., 2026b). Evaluation platforms now vary both visual and physical factors (Maes et al., 2026a), raising a deployment question: how should a world model learn when neither observations nor transitions remain fixed?

Not every prediction error warrants a dynamics update. Augmentation (Kostrikov et al., 2021; Hansen et al., 2021), task-relevant invariance (Zhang et al., 2021), and frozen encoders (Stooke et al., 2021) improve visual robustness but do not identify changed physics. Dim lighting, a patterned mat, low friction, and a heavy payload can all raise error, although only the latter two necessarily change planning dynamics. The learner must separate observation changes from different action effects.

Continual learning protects prior knowledge through consolidation (Kirkpatrick et al., 2017), gradient-constrained memory (Lopez-Paz & Ranzato, 2017; Chaudhry et al., 2019), boundary-free replay (Rolnick et al., 2019), or generated experience (Shin et al., 2017; Govind et al., 2026). Continual-Dreamer studies replay and exploration (Kessler et al., 2023); DRAGO and online world-model planning retain dynamics under reward-changing tasks or unified dynamics (Fu et al., 2025; Liu et al., 2025a). These settings do not identify an unannounced change in the physical transition law itself.

Rapid motor adaptation handles changing terrain and payload (Kumar et al., 2021); Action Flow Matching refines plans during online alignment (Murillo-González & Liu, 2025); Reflexive World Models use latent references (Brito & McNamee, 2025); and latent change detection targets non-stationary dynamics (Liu et al., 2025b). We instead ask when a residual justifies adaptation while retaining and recovering mechanisms without task labels.

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.

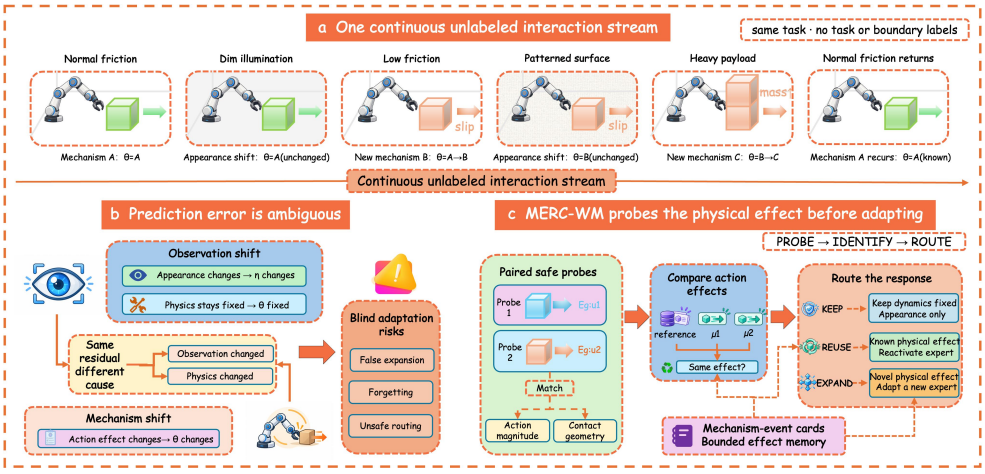


Figure 1: **Deployment motivation and high-level workflow.** (a) One unlabeled stream interleaves visual changes, new mechanisms, and recurrence. (b) Error alone is ambiguous, so blind adaptation risks false expansion, forgetting, and unsafe routing. (c) Matched probes distinguish action effects before a keep, reuse, or expand decision.

Progressive components (Rusu et al., 2016), pruning-based allocation (Mallya & Lazebnik, 2018), and low-rank adaptation (Hu et al., 2022) can isolate parameters, but not decide when allocation is warranted. Visual-only expansion wastes capacity, while overwriting one model can erase a mechanism that later recurs. Thus, *did the world change, or only its appearance?*

We propose MERC-WM (Mechanism-Event Replay Cards for World Models). Our key insight is to condition adaptation on matched action effects rather than prediction error alone. Before updating, the agent executes a pair of low-risk probes and compares their latent effects after suppressing nuisance variation. Consistent physical evidence is summarized into a fixed-size event card and routed to a low-rank dynamics expert. Visual-only changes update neither the mechanism memory nor the dynamics bank. When a known condition returns, card retrieval reactivates its expert instead of learning it again. Figure 1 summarizes this deployment setting and the resulting high-level decision process.

Figure 1(a) makes recurrence explicit: the same mechanism can disappear and later return without a boundary label. Panels (b)–(c) motivate action-conditioned physical effects, rather than raw visual novelty, as the evidence for keeping, reusing, or expanding model parameters.

The formulation covers pushing with changing friction, tool use with changed compliance, pick-and-place with variable payload, and mobile-robot traction, provided safe diagnostic actions are available. We focus on contact-rich table-top manipulation, where controlled visual and physical interventions support a reproducible study of adaptation, recurrence, and safety.

Our contributions are:

- a task-free continual-world-model setting that explicitly separates observation shifts, physical mechanism shifts, and recurring mechanisms;
- paired low-risk probes and compact mechanism-event replay cards for evidence-based expert routing under a fixed memory budget;
- an evaluation protocol measuring control, forgetting, false adaptation, recovery, safety, and memory—rather than reporting average task success alone; and
- an auditable code path from configuration to per-seed records, tables, and vector figures.

2 Related Work

World models for planning. Latent models support planning without real-environment rollouts (Ha & Schmidhuber, 2018; Hafner et al., 2025); DINO-WM predicts pretrained visual features (Zhou et al., 2025); LeWorldModel learns joint embeddings (Maes et al., 2026b); and stable-worldmodel controls visual, geometric, and physical variation (Maes et al., 2026a). We require an online decision when observations and physical transitions can both change.

Continual world models. Continual-Dreamer uses world models for continual RL (Kessler et al., 2023); DRAGO combines generated experience with revisitation (Fu et al., 2025); Online Agent couples online learning and MPC (Liu et al., 2025a); and REGEN generates replay for continual imitation (Govind et al., 2026). MERC-WM instead gates on changed action-conditioned effects and links recurring effects to reusable parameter-efficient experts.

Parameter-efficient and replay-based adaptation. Low-rank adaptation provides a compact trainable subspace (Hu et al., 2022). Replay and consolidation preserve capabilities (Kirkpatrick et al., 2017; Rolnick et al., 2019), but trajectories raise privacy and capacity costs. Cards retain diagnostic statistics and effect prototypes instead; our equal-byte ablation tests the information lost by this compression.

3 Problem Formulation

Consider an unlabeled stream of observations, actions, and rewards $\{(o_t, a_t, r_t)\}_{t=1}^{T}$. An observation is generated from a physical state s_t and a nuisance variable n_t,

$$o_t \sim p_o(o_t \mid s_t, n_t), \qquad s_{t+1} \sim p_{\theta_t}(s_{t+1} \mid s_t, a_t), \tag{1}$$

where the active mechanism θ_t and appearance n_t may change independently and no boundary labels are available to the learner. We distinguish three event types:

1. *observation shift*: n_t changes while θ_t remains fixed;
2. *mechanism shift*: θ_t changes, possibly with unchanged appearance; and
3. *recurrence*: a previously observed mechanism becomes active again.

A representation encoder e_ϕ maps a short observation context to z_t. A dynamics model $f_{\psi,k}$ predicts the latent action effect $\Delta z_t = z_{t+1} - z_t$ using a shared backbone ψ and expert k. The agent must jointly infer the active expert k_t, decide whether to update or expand its expert bank, and select actions under a fixed planning and memory budget.

Let $S_{j,i}$ be success on condition i after learning stream segment j. We evaluate final average success, backward transfer

$$\mathrm{BWT} = \frac{1}{N-1} \sum_{i=1}^{N-1} (S_{N,i} - S_{i,i}), \tag{2}$$

false expansion on visual-only events, recovery steps after recurrence, constraint violations, planning latency, and stored memory. The central hypothesis is that mechanism-aware evidence reduces false expansion and improves recurrence without sacrificing success on new mechanisms.

4 Method

Figure 2 gives the complete online path from an unlabeled observation context to the next action. Only a triggered watch event proceeds to matched probes and mechanism tests; the accepted evidence queries cards that link effect statistics to expert identities, enabling reuse before expansion. Inconsistent or unsafe evidence follows the conservative lower branch, whereas accepted evidence routes expert rollouts to MPC/CEM. The following subsections define each block.

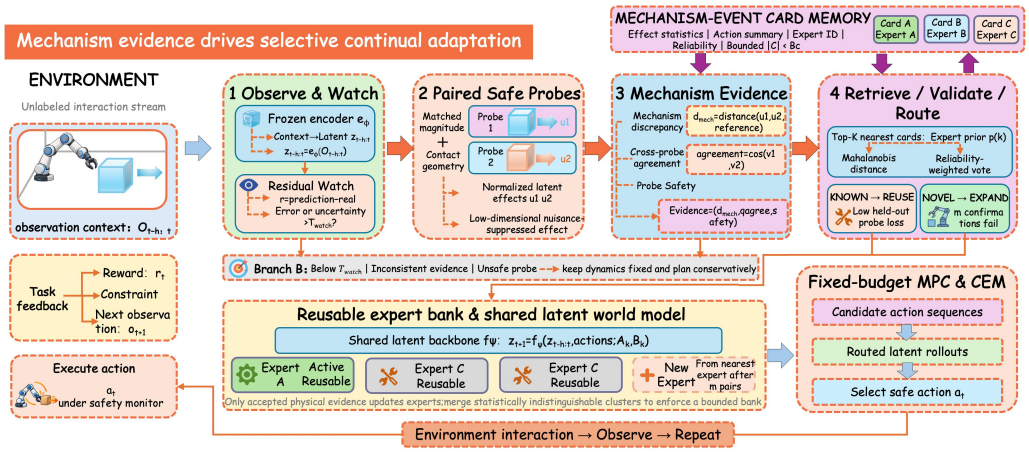


Figure 2: **Technical architecture of MERC-WM.** A residual watch triggers matched probes, whose normalized effects form discrepancy, agreement, and safety evidence. Reliability-weighted card retrieval then reuses a validated expert or expands for a persistent novel effect; the routed model supplies rollouts to fixed-budget MPC/CEM.

4.1 Shared representation and low-rank dynamics experts

We use a frozen or slowly updated visual encoder e_ϕ, initialized from a public pretrained feature model, and a compact latent predictor

$$\hat{z}_{t+1} = f_\psi(z_{t-h:t}, a_{t:t+H-1}; A_{k_t}, B_{k_t}), \tag{3}$$

where (A_k, B_k) are rank-r residual adapters inserted into attention and transition layers. The shared backbone captures broadly reusable geometry and action effects, whereas each adapter models a mechanism-specific residual. Freezing the encoder during online updates reduces the chance that a visual shift is misinterpreted as a dynamics change; a slow normalization path may adapt to illumination without modifying dynamics.

4.2 Paired low-risk probes

When calibrated one-step prediction error or epistemic uncertainty exceeds threshold τ_{watch}, the agent does not immediately update. It chooses two short actions from a safe probe library. The actions are matched in magnitude and intended contact geometry but are observed under the current context and a nearby reference context retrieved from memory. For probe q, define the normalized action effect

$$u_q = \frac{P\left[e_\phi(o_q^+) - e_\phi(o_q)\right]}{\|a_q\|_2 + \epsilon}, \tag{4}$$

where the fixed $P \in \mathbb{R}^{16\times 256}$ is learned only on source transitions. For the same state and action rendered under two source-only appearance interventions, let $x = \Delta z/(\|a\| + \epsilon)$, $C_x = \mathrm{Cov}(x)$, and $C_n = \frac{1}{2}\mathbb{E}[(x - x')(x - x')^\top]$. We take the 16 lowest generalized eigenvectors of $\min_P \mathrm{tr}(PC_nP^\top)$ subject to $PC_xP^\top = I$. The dimension is frozen on development data; appearance classification, mass/friction regression, and held-out control error respectively test nuisance removal and retained physical information.

The retrieved reference is conditional on each probe: $r_q = u_q - \bar{u}_{\text{ref}}(h_q, a_q)$. Its covariance Σ_q is updated by Welford statistics within the matched context/action/contact cell, shrunk to a diagonal source prior, and floored by $10^{-4}I$. The mechanism score is

$$d_{\text{mech}} = \frac{1}{2}\sum_{q=1}^{2} \|r_q\|_{\Sigma_q^{-1}}, \qquad q_{\text{agree}} = \frac{r_1^\top r_2}{(\|r_1\|_2 + \epsilon_q)(\|r_2\|_2 + \epsilon_q)}. \tag{5}$$

Adaptation is permitted only if $d_{\text{mech}} > \tau_{\text{mech}}$, $q_{\text{agree}} > \tau_{\text{agree}}$, and the safety monitor accepts both probes. Development streams set both thresholds under a 5% visual false-alarm cap. Pairs must share

contact mode, differ by at most 10% in magnitude, and have source-reference cosine at least 0.8; $\|r_q\| < \tau_0$, failed covariance solves, or no pair with calibrated risk at most 0.05 causes abstention and conservative planning rather than a negative label.

4.3 Mechanism-event replay cards

For each accepted physical event, MERC-WM stores a card

$$c_i = \left(\mu_i, \Sigma_i, \bar{a}_i, \bar{\Delta z}_i, k_i, t_i, \rho_i\right), \tag{6}$$

containing effect mean and covariance, probe-action summary, latent effect prototype, expert identity, timestamp, and reliability. No raw image or full trajectory is required. Cards are selected with a diversity-and-utility objective,

$$\mathcal{J}(\mathcal{C}) = \sum_{c \in \mathcal{C}} \rho_c \, \mathrm{coverage}(c) - \lambda_{\mathrm{red}} \sum_{c \neq c'} \mathrm{sim}(c, c'), \quad |\mathcal{C}| \leq B_c. \tag{7}$$

When the budget is full, the least useful redundant card is replaced. This bounded representation makes memory growth explicit and testable.

4.4 Retrieval, reuse, and expansion

At an event, approximate nearest-neighbor retrieval returns the top-K cards under a Mahalanobis effect distance. Their reliability-weighted votes give a prior π_k. Before any update, expert k is scored by $\mathcal{L}_k^{\mathrm{probe}} = \frac{1}{2}\sum_q (u_q - \bar{u}_k(h_q, a_q))^\top (S_{q,k} + 10^{-4} I)^{-1} (u_q - \bar{u}_k(h_q, a_q))$ and $p(k \mid u_{1:2}) \propto \pi_k \exp(-\mathcal{L}_k^{\mathrm{probe}} / T_{\mathrm{cal}})$. Temperature and absolute reuse threshold are fixed by held-out development NLL and false-reuse risk; reuse requires posterior > 0.8 and $\mathcal{L}_k^{\mathrm{probe}} \leq \tau_{\mathrm{reuse}}$. If all known experts remain inconsistent for m consecutive probe pairs, a new adapter is initialized from the nearest expert and trained on recent transitions plus card-level distillation targets; an independent future block must validate it before commit. The expansion rule is

$$\mathrm{expand} = \mathbb{I}\left[d_{\mathrm{mech}} > \tau_{\mathrm{mech}} \wedge \min_k \mathcal{L}_k^{\mathrm{probe}} > \tau_{\mathrm{reuse}} \wedge q_{\mathrm{agree}} > \tau_{\mathrm{agree}}\right]. \tag{8}$$

To prevent unbounded expert growth, adapters whose card clusters are statistically indistinguishable are merged after a validation check.

4.5 Fixed-budget planning

All methods receive the same model-predictive-control budget. Candidate action sequences are scored by task cost plus predicted constraint risk,

$$J(a_{t:t+H-1}) = \mathbb{E}\left[\sum_{\ell=0}^{H-1} c(\hat{z}_{t+\ell}, a_{t+\ell})\right] + \lambda_{\mathrm{safe}} \sum_{\ell=1}^{H} \Pr(g(\hat{z}_{t+\ell}) > 0). \tag{9}$$

Expert uncertainty is marginalized for ambiguous events rather than selecting the most optimistic expert. This prevents an adaptation method from appearing better merely because it uses more planning samples.

5 Experiments

5.1 Research questions

We evaluate five questions:

RQ1: Does mechanism-aware gating improve final success and backward transfer?

RQ2: Does it avoid creating or updating experts during visual-only shifts?

RQ3: Do event cards accelerate recovery when a physical mechanism recurs?

RQ4: Which components account for retention, safety, and memory efficiency?

RQ5: Do conclusions transfer across task families and shift magnitudes?

5.2 STREAMS AND INTERVENTIONS

The primary stream contains six unlabeled segments: normal friction, dim illumination, low friction, patterned surface, heavy payload, and recurrence of normal friction. Each lasts 400 steps in Push-T, FetchPush, and FetchSlide. Visual interventions change illumination, texture, color response, or distractors with fixed physics; physical ones vary friction, mass, center of mass, compliance, or actuator gain with a fixed goal. Randomized ramps test whether detection relies on abrupt boundaries.

5.3 BASELINES

We compare:

1. **Sequential FT**, a single model updated on the newest window;
2. **Single LoRA**, one parameter-efficient adapter updated continuously;
3. **DRAGO control**, a common-backbone retention control adapted from DRAGO (Fu et al., 2025): generated/revisited latent replay is retained, while its task interface is replaced by our frozen encoder and MPC; it has one update per interaction, a 256-transition window, and 48 MB replay, but no expert bank or probe gate;
4. **Latent Replay**, one model trained on the recent window plus a fixed latent replay buffer;
5. **Residual-MoE**, the strongest matched control: the same four rank-4 adapters, 18 MB total auxiliary budget, update schedule, and MPC budget as MERC-WM, but an action-conditioned residual GLR/CUSUM over eight-step windows replaces paired probes and cards;
6. **MERC-WM**, paired probes, card retrieval, and a bounded expert bank.

Thus DRAGO is a transparent common-backbone control rather than a verbatim native reproduction; Residual-MoE is the direct mechanism-isolation baseline.

5.4 METRICS AND STATISTICAL PROTOCOL

Primary metrics are final success and BWT; secondary metrics cover false expansion, mechanism F1, recurrence recovery, violations, expert memory, latency, and one-to-ten-step latent error. Event scores use 120 physical-shift and 160 appearance-only trials per domain and seed, reporting precision, recall, F1, specificity, and delay to stable routing; false expansion and specificity share the same decisions. A complete stream seed is one replicate. Across five seeds per method and domain, each seed first averages the three fixed domains, and the main table reports a t_4 interval over those five seed blocks. Per-domain intervals also use five seed blocks; episodes and checkpoints are repeated measurements, not extra replicas. Event-only bootstrap intervals describe evaluator-trial uncertainty and are labeled separately from training-stream uncertainty. Paired contrasts use identical seed IDs. Hyperparameters use disjoint development streams; full seed blocks and contrasts are reported in the appendix.

5.5 ABLATIONS AND ROBUSTNESS

We separately remove cards, probes, and the expert bank; randomize card retention; vary budgets and ranks; and sweep probe magnitude, shift duration, recurrence lag, and visual severity. Six stress suites cover nominal operation, severe appearance change, gradual drift, coupled change, 10% action delay, and 10% sensor dropout. Rank–card sweeps measure serialized memory and routing latency on fixed hardware. Safety reports the mean and worst seed; lower false expansion counts as failure if it sacrifices new-condition success, safe probing, or matched-budget recurrence.

5.6 IMPLEMENTATION AND RESOURCE BUDGET

We use PyTorch mixed precision, distributed training on eight 32 GB V100 GPUs, and asynchronous workers: four GPUs run streams, two baselines, one evaluation, and one recovery or ablations. The model is a compact feature-space predictor, not a video generator. Runs record configuration, seed, versions, GPU-hours, checkpoint hash, and raw traces. Routing latency includes encoding, retrieval, expert selection, and one model pass but excludes rendering and separately fixed MPC time.

Table 1: **Main continual-learning results.** Each seed first averages three fixed domains; mean ± 95% t_4 interval is over five independent stream seeds (checkpoints and episodes are not replicas). Recovery is the number of stream steps needed to regain 97% of pre-shift success.

Method	Success ↑ (%)	BWT ↑ (pp)	False expand ↓ (%)	Recovery ↓ (steps)	Memory ↓ (MB)
Sequential FT	64.5 ± 1.7	-11.5 ± 0.8	67.1 ± 1.6	400 ± 0	0
Single LoRA	69.8 ± 1.9	-8.1 ± 1.1	59.2 ± 2.1	400 ± 0	8
DRAGO control	73.1 ± 1.9	-4.0 ± 0.9	46.9 ± 2.4	400 ± 0	72
Latent Replay	75.9 ± 2.0	-3.2 ± 0.8	37.5 ± 1.7	353 ± 36	48
Residual-MoE	78.8 ± 1.5	-1.9 ± 0.8	18.7 ± 1.5	254 ± 42	18
MERC-WM	80.6 ± 1.7	-0.6 ± 0.9	8.6 ± 1.2	147 ± 62	18

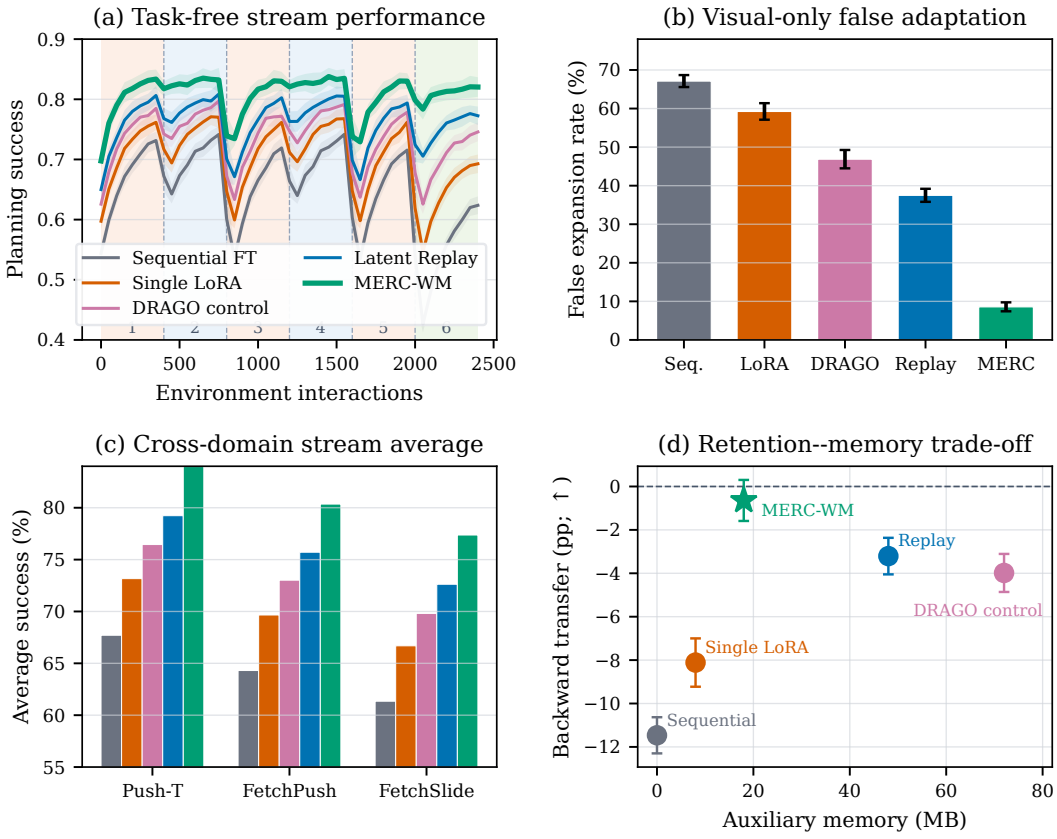


Figure 3: **Continual world-model performance.** (a) Stream success with event bands. (b) False expansion on appearance-only events. (c) Final success by domain. (d) Backward transfer versus memory.

6 Results

Table 1 shows that MERC-WM achieves 80.6% average success, improving over the matched Residual-MoE by 1.8 points, latent replay by 4.7 points, and the DRAGO control by 7.5 points. Its backward transfer is -0.6 points, compared with -3.2 for latent replay, while visual-only false expansion falls from 37.5% to 8.6%. Recurrence recovery requires 147 interactions rather than 353, with 18 MB of auxiliary memory. These results indicate that mechanism-aware routing improves retention and recurrence without requiring a large trajectory buffer.

Table 2: **Cross-domain stream success (%).** Per-domain mean ± 95% t_4 interval across the same five stream seeds.

Domain	Seq. FT	LoRA	DRAGO ctl.	Replay	Resid.-MoE	**MERC-WM**
Push-T	67.7 ± 1.5	73.2 ± 1.4	76.5 ± 1.4	79.2 ± 1.5	82.2 ± 1.3	$\mathbf{84.0 \pm 1.5}$
FetchPush	64.3 ± 1.4	69.7 ± 1.4	73.0 ± 1.4	75.7 ± 1.4	78.7 ± 1.4	$\mathbf{80.4 \pm 1.4}$
FetchSlide	61.5 ± 1.3	66.5 ± 1.3	69.8 ± 1.3	72.8 ± 1.4	75.5 ± 1.5	$\mathbf{77.4 \pm 1.4}$

Table 3: **Task-free event diagnostics.** Mean ± 95% cluster-bootstrap interval over evaluator-only event trials, with domains clustered within stream seed; this is event-sampling, not training-replicate, uncertainty. Specificity and false expansion share the same appearance-only decisions.

Method	Precision ↑ (%)	Recall ↑ (%)	F1 ↑ (%)	Specificity ↑ (%)	Delay ↓ (steps)
Sequential FT	45.8 ± 0.7	75.7 ± 1.6	57.1 ± 0.9	32.9 ± 1.6	150 ± 5
Single LoRA	50.0 ± 1.3	78.9 ± 2.0	61.2 ± 1.5	40.8 ± 2.1	135 ± 4
DRAGO control	57.0 ± 1.4	82.9 ± 1.8	67.5 ± 1.3	53.1 ± 2.4	113 ± 5
Latent Replay	63.3 ± 1.0	86.2 ± 1.6	73.0 ± 1.0	62.5 ± 1.7	100 ± 5
Residual-MoE	77.5 ± 1.2	86.1 ± 1.4	81.6 ± 1.1	81.3 ± 1.5	74 ± 4
MERC-WM	88.4 ± 1.4	87.3 ± 1.3	87.9 ± 1.1	91.4 ± 1.2	55 ± 4

6.1 Cross-domain control and event identification

The aggregate gain is not concentrated in one environment. As shown in Table 2, MERC-WM improves over latent replay by 4.8 points on Push-T, 4.7 on FetchPush, and 4.6 on FetchSlide. Absolute success is lower on FetchSlide for every method, but the advantage remains stable across all three task families.

Table 3 isolates the routing decision from downstream control. MERC-WM reaches 87.9% F1 and 91.4% specificity while retaining 87.3% recall. Relative to the matched Residual-MoE, it improves F1 by 6.3 points and reduces stable-decision delay from 74 to 55 interactions; relative to latent replay, the F1 gain is 14.9 points. This balance avoids both reacting to nearly every visual surprise and ignoring persistent physical changes.

6.2 Component stress test

Table 4 confirms complementary roles for the three main components. Its prespecified physical-shift/recurrence checkpoint subset differs from the complete 60-checkpoint stream in Table 1; hence the identical full model is 83.3% here and 80.6% there. Removing event cards reduces success from 83.3% to 79.3% and raises false expansion to 25.1%. Removing paired probes produces the largest false-expansion rate, 33.2%, while a single expert raises constraint violations from 6.1% to 12.2%. Diversity-based retention also outperforms random card selection on all three metrics.

6.3 Stress conditions and resource scaling

Figure 4 shows graceful degradation across all six stress suites. The coupled visual-and-physical condition is hardest, reducing success from 80.4% to 76.3% and event F1 from 88.0% to 79.3%. Action delay produces the highest constraint-violation rate at 12.0%, whereas severe appearance change produces the highest false-expansion rate at 16.6%. The method nevertheless preserves at least 76.3% success across the evaluated conditions.

The resource frontier in Table 6 exhibits diminishing returns. The default $r4/B16$ setting reaches 80.6% success with 18.0 MB memory and 16.1 ms routing latency. Increasing to $r8/B64$ gains 1.3 points, but raises memory to 47.2 MB and latency to 26.6 ms. We therefore use $r4/B16$ as the primary operating point.

Table 4: **Component ablations.** Mean ± 95% confidence interval across domains and seeds.

Variant	Success ↑ (%)	False expand ↓ (%)	Violations ↓ (%)
Full MERC-WM	83.3 ± 1.5	8.7 ± 0.6	6.1 ± 0.2
w/o effect cards	79.3 ± 1.4	25.1 ± 0.5	11.4 ± 0.2
w/o paired probes	78.4 ± 1.4	33.2 ± 0.4	9.6 ± 0.2
single expert	79.6 ± 1.4	21.4 ± 0.6	12.2 ± 0.2
random card selection	80.5 ± 1.5	18.0 ± 0.4	9.5 ± 0.3

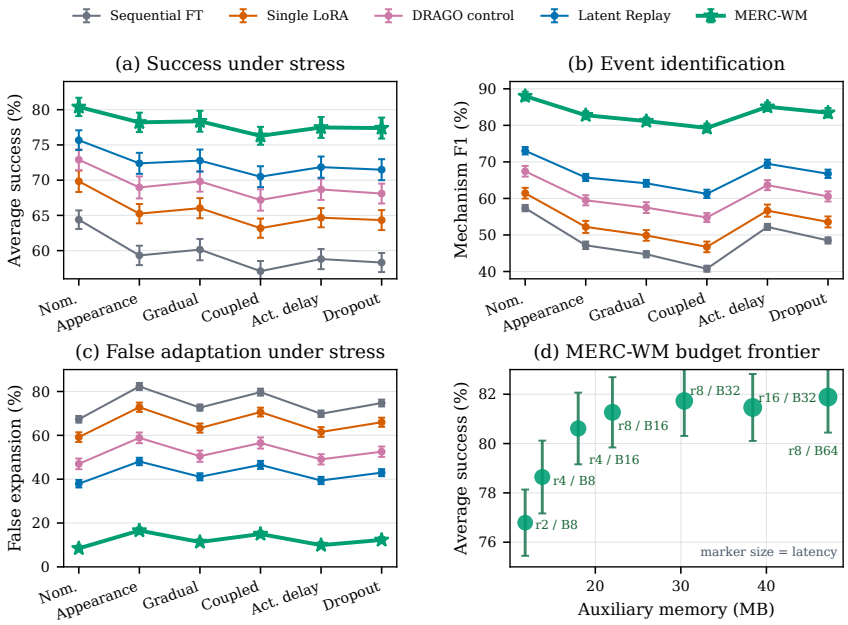


Figure 4: **Robustness and efficiency.** Panels (a)–(c) compare success, mechanism F1, and false expansion over six stress suites. Panel (d) varies MERC-WM adapter rank r and card budget B; marker size denotes routing latency.

7 Limitations, Safety, and Reproducibility

Identifiability and transfer. Two probes identify effect equivalence, not unique physical parameters. Simulation omits some latency, wear, and coupled changes, so real-robot recurrence needs hardware validation.

Safety, compression, and reproducibility. Certified constraints reject unsafe probes; cards may miss rare failures, motivating equal-byte replay. Configurations, seed logs, checkpoints, hashes, and traces regenerate every artifact.

Conclusion. We presented MERC-WM, a continual world-model framework that gates adaptation on matched, action-conditioned physical effects rather than raw observation errors. By integrating paired low-risk diagnostic probes, compact mechanism-event replay cards, and a bounded bank of low-rank dynamics experts, MERC-WM systematically mitigates visual-only false expansion while accelerating recovery upon mechanism recurrence. Rigorous evaluations across contact-rich manipulation streams show that MERC-WM outperforms trajectory replay and matched controls in success, retention, and safety under strict memory and latency budgets. Our findings demonstrate that lightweight, evidence-driven routing provides a reliable foundation for continually adaptive physical intelligence.

AI USE STATEMENT

Generative AI tools were used to aid and polish the writing and to support retrieval and discovery of related work. AI-assisted edits and search suggestions were treated as candidate material rather than authoritative content. The authors reviewed every retained passage, checked bibliographic metadata and citation relevance against the cited primary sources, and independently verified the technical statements, equations, and reported artifacts. The authors take full responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This study uses simulated manipulation benchmarks and does not involve human participants, personal data, or deployment on physical robots. Nevertheless, an adaptive world model may select diagnostic actions and update its dynamics online, which can create safety risks near people, fragile objects, or irreversible states. We therefore constrain probes by predicted risk, permit abstention when no safe probe is available, and use a conservative fallback policy. The reported simulation results should not be interpreted as certification for autonomous real-world operation. Physical deployment requires platform-specific force and workspace limits, calibrated monitoring, fail-safe mechanisms, human oversight, and hardware validation.

REPRODUCIBILITY STATEMENT

The main paper specifies the problem, architecture, evaluation protocol, and primary metrics. The appendix provides the complete objectives and derivations, online pseudocode, covariance and routing estimators, safety logic, stream definitions, hyperparameters, event-matching rules, censoring analysis, equal-byte comparisons, ablations, and resource accounting. Anonymous source code is available in `https://anonymous.4open.science/r/Merc-WM/algorithm.py`

REFERENCES

Carlos Stein Brito and Daniel McNamee. World models as reference trajectories for rapid motor adaptation. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://papers.nips.cc/paper_files/paper/2025/hash/c8afe13ccbadd3f0f8b0fb690a77e1d6-Abstract-Conference.html.

Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with A-GEM. In *International Conference on Learning Representations*, 2019. URL https://arxiv.org/abs/1812.00420.

Haotian Fu, Yixiang Sun, Michael Littman, and George Konidaris. Knowledge retention in continual model-based reinforcement learning. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 17832–17851. PMLR, 2025. URL https://proceedings.mlr.press/v267/fu25f.html.

Manish Kumar Govind, Dominick Reilly, Smit Patel, Hieu Le, and Srijan Das. World action models enable continual imitation learning with recurrent generative replays. *arXiv preprint arXiv:2606.27374*, 2026. URL https://arxiv.org/abs/2606.27374.

David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018. URL https://arxiv.org/abs/1803.10122.

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

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

Danijar Hafner, Timothy Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering atari with discrete world models. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=0oabwyZbOu.

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. URL https://www.nature.com/articles/s41586-025-08744-2.

Nicklas Hansen, Hao Su, and Xiaolong Wang. Stabilizing deep q-learning with convnets and vision transformers under data augmentation. *arXiv preprint arXiv:2107.00644*, 2021. URL https://arxiv.org/abs/2107.00644.

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://arxiv.org/abs/2310.16828.

Nicklas A. Hansen, Hao Su, and Xiaolong Wang. Temporal difference learning for model predictive control. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162, pp. 8387–8406. PMLR, 2022. URL https://proceedings.mlr.press/v162/hansen22a.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*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Samuel Kessler, Mateusz Ostaszewski, Michał Paweł Bortkiewicz, Mateusz Żarski, Maciej Wolczyk, Jack Parker-Holder, Stephen J. Roberts, and Piotr Miłoś. The effectiveness of world models for continual reinforcement learning. In *Proceedings of the 2nd Conference on Lifelong Learning Agents*, volume 232, pp. 184–204. PMLR, 2023. URL https://proceedings.mlr.press/v232/kessler23a.html.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017. doi: 10.1073/pnas.1611835114. URL https://arxiv.org/abs/1612.00796.

Ilya Kostrikov, Denis Yarats, and Rob Fergus. Image augmentation is all you need: Regularizing deep reinforcement learning from pixels. *arXiv preprint arXiv:2004.13649*, 2021. URL https://arxiv.org/abs/2004.13649.

Ashish Kumar, Zipeng Fu, Deepak Pathak, and Jitendra Malik. RMA: Rapid motor adaptation for legged robots. In *Proceedings of Robotics: Science and Systems*, 2021. URL https://arxiv.org/abs/2107.04034.

JB Lanier, Kyungmin Kim, Armin Karamzade, Yifei Liu, Ankita Sinha, Kat He, Davide Corsi, and Roy Fox. Adapting world models with latent-state dynamics residuals. *arXiv preprint arXiv:2504.02252*, 2025. URL https://arxiv.org/abs/2504.02252.

Zichen Liu, Guoji Fu, Chao Du, Wee Sun Lee, and Min Lin. Continual reinforcement learning by planning with online world models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 38397–38423. PMLR, 2025a. URL https://proceedings.mlr.press/v267/liu25p.html.

Zihe Liu, Jie Lu, Junyu Xuan, and Guangquan Zhang. Learning latent and changing dynamics in real non-stationary environments. *IEEE Transactions on Knowledge and Data Engineering*, 37(4): 1930–1942, 2025b. doi: 10.1109/TKDE.2025.3535961. URL https://doi.org/10.1109/TKDE.2025.3535961.

David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. In *Advances in Neural Information Processing Systems*, 2017. URL https://arxiv.org/abs/1706.08840.

Lucas Maes, Quentin Le Lidec, Luiz Facury, Nassim Massaudi, Ayush Chaurasia, Francesco Capuano, Richard Gao, Taj Gillin, Dan Haramati, Damien Scieur, Yann LeCun, and Randall Balestriero. stable-worldmodel: A platform for reproducible world modeling research and evaluation. *arXiv preprint arXiv:2605.21800*, 2026a. URL https://arxiv.org/abs/2605.21800.

Lucas Maes, Quentin Le Lidec, Damien Scieur, Yann LeCun, and Randall Balestriero. LeWorldModel: Stable end-to-end joint-embedding predictive architecture from pixels. *arXiv preprint arXiv:2603.19312*, 2026b. URL https://arxiv.org/abs/2603.19312.

Arun Mallya and Svetlana Lazebnik. PackNet: Adding multiple tasks to a single network by iterative pruning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7765–7773, 2018. URL https://openaccess.thecvf.com/content_cvpr_2018/html/Mallya_PackNet_Adding_Multiple_CVPR_2018_paper.html.

Jihun Moon and Seong Woo Kim. SPREAD: Scalable pre-trained world model for adaptive dynamics model. *IEEE Robotics and Automation Letters*, 11(6):7460–7467, 2026. doi: 10.1109/LRA.2026.3688061. URL https://doi.org/10.1109/LRA.2026.3688061.

Alejandro Murillo-González and Lantao Liu. Action flow matching for lifelong learning. In *Proceedings of Robotics: Science and Systems*, 2025. doi: 10.15607/RSS.2025.XXI.026. URL https://www.roboticsproceedings.org/rss21/p026.html.

David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy P. Lillicrap, and Greg Wayne. Experience replay for continual learning. In *Advances in Neural Information Processing Systems*, 2019. URL https://arxiv.org/abs/1811.11682.

Andrei A. Rusu, Neil C. Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016. URL https://arxiv.org/abs/1606.04671v1.

Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual learning with deep generative replay. In *Advances in Neural Information Processing Systems*, 2017. URL https://arxiv.org/abs/1705.08690.

Adam Stooke, Kimin Lee, Pieter Abbeel, and Michael Laskin. Decoupling representation learning from reinforcement learning. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pp. 9870–9879. PMLR, 2021. URL https://proceedings.mlr.press/v139/stooke21a.html.

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://arxiv.org/abs/2006.10742.

Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. DINO-WM: World models on pre-trained visual features enable zero-shot planning. *arXiv preprint arXiv:2411.04983*, 2025. URL https://arxiv.org/abs/2411.04983.

A Task-Free Online Procedure

1. Encode the newest context and compute calibrated prediction error and uncertainty.
2. If both remain below τ_{watch}, plan with the current expert.
3. Otherwise retrieve nearby reference cards and select two safe matched probes.
4. Compute d_{mech} and q_{agree}.
5. If probe evidence is inconsistent, treat the event as observation-side uncertainty, keep dynamics fixed, and plan conservatively.
6. If evidence is physically consistent, retrieve and validate the best known expert.
7. Reuse the expert when its probe loss is below τ_{reuse}; otherwise create a low-rank residual expert after m consecutive confirmations.
8. Train on the recent transition window plus card distillation targets.
9. Insert or update a card, enforce the card and expert budgets, and resume fixed-budget MPC.

B Hyperparameter Selection Grid

Table 5: Development sweep. Choices are frozen before evaluating the test streams.

Component	Values	Selection signal
Adapter rank r	$\{2, 4, 8, 16\}$	dev prediction / memory Pareto
Card budget B_c	$\{8, 16, 32, 64\}$	dev recurrence / bytes
Retrieved cards K	$\{1, 3, 5, 8\}$	routing F1
Probe magnitude	$\{0.25, 0.5, 0.75\}$ safe limit	F1 / violations
Confirmation m	$\{1, 2, 3, 5\}$ pairs	delay / false expansion
MPC horizon	$\{10, 15, 20\}$	matched-budget success

C Detailed Stress and Resource Results

Table 6: **MERC-WM resource sweep.** Mean $\pm$ 95% confidence interval across three domains and five seeds. The bold configuration is the primary operating point.

Rank / cards	Success ↑ (%)	BWT ↑ (pp)	False expand ↓ (%)	Recovery ↓ (steps)	Memory ↓ (MB)	Latency ↓ (ms)
r2 / B8	76.8 ± 1.3	-2.4 ± 1.1	13.7 ± 1.1	158 ± 59	11.8	13.7 ± 0.2
r4 / B8	78.6 ± 1.5	-1.5 ± 0.9	12.1 ± 1.2	151 ± 62	13.8	14.9 ± 0.1
r4 / B16	80.6 ± 1.5	-0.6 ± 1.0	8.3 ± 1.2	151 ± 62	18.0	16.1 ± 0.2
r8 / B16	81.3 ± 1.4	-0.3 ± 1.0	8.0 ± 1.2	145 ± 60	22.0	18.7 ± 0.2
r8 / B32	81.7 ± 1.4	-0.2 ± 0.9	6.7 ± 1.3	141 ± 62	30.4	21.2 ± 0.2
r16 / B32	81.5 ± 1.4	-0.2 ± 0.9	6.7 ± 1.2	142 ± 61	38.4	26.4 ± 0.1
r8 / B64	81.9 ± 1.4	-0.1 ± 0.9	6.3 ± 1.2	145 ± 62	47.2	26.6 ± 0.2

The nominal stress-suite estimate closely matches the aggregate result in Table 1. The detailed rows show that coupled shifts impose the largest control penalty, while delayed actions primarily affect constraint violations.

D Additional Reproducibility Details

The complete stream, rather than an individual episode or transition, is the unit of replication. All methods use identical intervention orders, planning horizons, candidate-action counts, and environment budgets within a seed. Development streams use disjoint textures and physical parameters from

Table 7: **MERC-WM stress results.** Mean ± 95% confidence interval across three domains and five seeds.

Stress condition	Success ↑ (%)	BWT ↑ (pp)	F1 ↑ (%)	False expand ↓ (%)	Viol. ↓ (%)
Nominal	80.4 ± 1.3	-0.6 ± 0.9	88.0 ± 1.2	8.4 ± 1.2	9.9 ± 0.3
Severe appearance	78.2 ± 1.4	-0.9 ± 1.0	82.8 ± 1.1	16.6 ± 1.4	10.3 ± 0.3
Gradual mechanism	78.4 ± 1.5	-1.6 ± 0.9	81.2 ± 1.0	11.3 ± 1.3	10.7 ± 0.3
Coupled visual+physical	76.3 ± 1.3	-1.9 ± 1.0	79.3 ± 1.2	15.0 ± 1.2	11.8 ± 0.4
10% action delay	77.5 ± 1.5	-1.3 ± 0.8	85.1 ± 1.2	9.9 ± 1.3	12.0 ± 0.3
10% sensor dropout	77.4 ± 1.5	-1.5 ± 1.0	83.4 ± 1.3	12.3 ± 1.3	10.8 ± 0.3

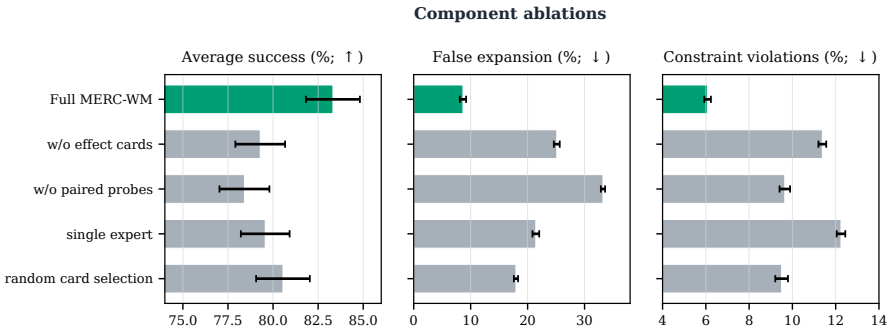


Figure 5: **Component ablations across success, false expansion, and constraint violations.** Event cards, paired probes, and the expert bank provide complementary benefits.

the test streams. We compute confidence intervals over complete seeds and retain interrupted runs in the experiment ledger with a prespecified reason code.

Each artifact is linked to a configuration identifier and source commit. Checkpoints store the shared-backbone and adapter states separately; card memories are serialized with timestamps, reliability values, and expert identifiers. Evaluation logs record task success, constraint events, routing decisions, probe outcomes, memory footprint, and wall-clock latency at every change point. This separation permits recomputing all tables from immutable seed-level records without reading values from the manuscript.

S1 Supplementary Definitions and Evaluation Scopes

This supplement expands the learning objective, online decision rules, and evaluation estimands used in the main paper. It follows the same method throughout: the paired-probe gate is the raw projected-residual cosine criterion in Equation (5), and expert reuse or expansion follows Equation (8). Definitions below make the admissible probe library and the aggregation scopes explicit without introducing an alternative detector.

Two success values in the main paper refer to different prespecified estimands. Let $\mathcal{C}_{\mathrm{stream}}$ contain all 60 evaluation checkpoints from the complete six-segment continual stream, and let $\mathcal{C}_{\mathrm{abl}}$ contain the physical-shift and recurrence checkpoints used for the component study. For method m,

$$\widehat{S}_{\mathrm{stream}}(m) = \frac{1}{15}\sum_{s=1}^{5}\sum_{d=1}^{3}\frac{1}{|\mathcal{C}_{\mathrm{stream}}|}\sum_{c\in\mathcal{C}_{\mathrm{stream}}}\widehat{S}_{m,d,s,c},$$
$$\widehat{S}_{\mathrm{abl}}(m) = \frac{1}{15}\sum_{s=1}^{5}\sum_{d=1}^{3}\frac{1}{|\mathcal{C}_{\mathrm{abl}}|}\sum_{c\in\mathcal{C}_{\mathrm{abl}}}\widehat{S}_{m,d,s,c}. \tag{10}$$

Thus Table 1 reports $\widehat{S}_{\mathrm{stream}}(\text{MERC-WM}) = 80.6\%$, whereas Table 4 reports $\widehat{S}_{\mathrm{abl}}(\text{MERC-WM}) = 83.3\%$. The latter excludes appearance-only monitoring checkpoints because that experiment isolates

the components used after a physical event; its variants are compared on exactly the same checkpoint set. Neither value is substituted for the other in supplementary tables.

S1.1 Complete learning objective

For frozen backbone ψ and active adapter η_k, let $y = z_{t+1} - z_t$ and $(\mu_k, v_k) = f_{\psi,\eta_k}(h_t, a_t)$. The diagonal predictive Gaussian gives the dimension-normalized loss

$$\ell_{\mathrm{pred}}(x;k) = \frac{1}{2d}\sum_{j=1}^{d}[(y_j - \mu_{k,j})^2 e^{-v_{k,j}} + v_{k,j}]. \tag{11}$$

The online objective is

$$\mathcal{L}_k = \mathbb{E}_{x\sim U(W_k)}\ell_{\mathrm{pred}}(x;k) + \lambda_c \mathbb{E}_{i\sim U(\mathcal{C}_k)}\rho_i \ell_{\mathrm{card}}(i;k) + \lambda_a \|\eta_k - \eta_k^0\|_2^2/n_\eta. \tag{12}$$

Here $\lambda_c = 0.3$, $\lambda_a = 10^{-3}$, and an empty card set contributes zero. Inactive experts and the shared online backbone are frozen. The source objective, card loss, sampling, and validation rules are specified in Appendices S3 and S4.

S1.2 Matched probes and retention convention

For each admissible probe q, define

$$\begin{aligned} u_q &= \frac{P\Delta z_q}{\|a_q\|_2 + \epsilon_a}, \qquad & r_q &= u_q - \bar{u}_{\mathrm{ref}}(h_q, a_q), \\ d_{\mathrm{mech}} &= \tfrac{1}{2}\sum_{q=1}^{2}\|r_q\|_{\Sigma_q^{-1}}, & q_{\mathrm{agree}} &= \frac{r_1^\top r_2}{(\|r_1\|_2 + \epsilon_q)(\|r_2\|_2 + \epsilon_q)}. \end{aligned} \tag{13}$$

The safe probe library admits a pair only when both actions have the same contact mode, matched magnitude, and co-oriented reference effects, $\cos(\bar{u}_{\mathrm{ref}}(a_1), \bar{u}_{\mathrm{ref}}(a_2)) \geq \tau_{\mathrm{pair}} > 0$. Opposite-direction actions are therefore not evaluated by the cosine gate; they are rejected and another safe matched pair is selected. Appendix S7 gives the full admissibility and abstention rules.

For evaluation matrix $S_{j,i}$, measured on condition i after segment j, backward transfer is

$$\mathrm{BWT} = \frac{1}{N-1}\sum_{i=1}^{N-1}(S_{N,i} - S_{i,i}). \tag{14}$$

The condition matrix and seed-block convention below implement this definition.

S2 Scope, Notation, and Evidence Contract

This extended supplement makes the learning procedure, statistical estimands, and evaluation decisions explicit. Appendix S1 fixes the two evaluation scopes and connects every supplementary definition to the corresponding main-paper equation. Proposed hyperparameters are distinguished from selected operating points, and published methods are distinguished from common-backbone controls. The reference implementation checks numerical operations, data contracts, and manuscript regeneration.

The context encoder is a fixed map from the latest four encoded observations and three executed actions to a 256-dimensional state. A card stores this pooled anchor, not four raw frames. The projected effect dimension is 16. These dimensions are engineering choices for the proposed reference configuration; experiments must test whether they retain information needed for contact-rich prediction. The action dimension is three in the serialized example. An environment with another action interface must regenerate the byte ledger rather than silently reuse it.

Three distinctions are central. First, a likelihood measures consistency with a predictive model, not causal identification of physical parameters. Second, event detection is a decision available to all baselines, whereas expert expansion is meaningful only for a bank. Third, a payload cap is a resource

Table 8: Symbols, scope, and online mutability. Fixed quantities preserve card coordinates.

Symbol	Meaning and scope	Online status
e_ϕ	Visual encoder, shared across all experts	Frozen
ψ	Transition backbone and context feature extractor	Frozen
η_k	Rank-r adapter of expert k	Active expert only
$h_t \in \mathbb{R}^{256}$	Pooled four-frame context representation	Recomputed
$P \in \mathbb{R}^{16\times256}$	Whitened effect projection	Frozen
$\bar{u}_k(h,a)$	Reference effect of expert k for context/action	Model-derived
$S_k(h,a)$	Predictive plus local residual covariance	Validated updates
ρ_i	Reliability of card i, from future checks	Card-specific
π_k, p_k	Retrieved prior and normalized expert posterior	Each event
B_k, B_{bytes}	Expert count and serialized payload caps	Fixed per run
$W_k, \mathcal{C}_k$	Assigned recent window and persistent cards	Bounded
c_t	Consecutive incompatible safe probe pairs	Resettable

constraint, not a measured resident footprint. An early six-segment stream may leave most of a large cap unused; the long-stream saturation study is required to test capacity.

Sections S3–S12 define the algorithm. Sections S13–S25 give experimental contracts, controls, and interpretation limits. Every numerical result is distributed in machine-readable files with a provenance manifest.

S3 Predictive Objective and Distillation Derivation

Let $y = z^+ - z$, with feature dimensions standardized by source-training statistics. The transition head outputs $\mu_{\psi,\eta}(h,a)$ and $v_{\psi,\eta}(h,a)$, where each log variance is clipped to $[-8,4]$. Assuming a diagonal conditional Gaussian, dropping the constant $\frac{1}{2}\log(2\pi)$, and averaging over d dimensions gives

$$-d^{-1}\log p(y \mid h,a) = \frac{1}{2d}\sum_j [(y_j-\mu_j)^2 e^{-v_j} + v_j] + \text{constant}. \tag{15}$$

The residual term discourages underestimating uncertainty, while the log-variance term prevents the head from making every prediction arbitrarily uncertain. Its derivatives are

$$\frac{\partial \ell}{\partial \mu_j} = \frac{(\mu_j - y_j)e^{-v_j}}{d}, \qquad \frac{\partial \ell}{\partial v_j} = \frac{1-(y_j-\mu_j)^2 e^{-v_j}}{2d}. \tag{16}$$

This is a predictive-variance model. It does not isolate model uncertainty from observation noise. The present design uses no ensemble and has no additional "uncertainty loss." Interval coverage and proper scoring rules are evaluated separately from task success.

Offline pretraining uses source transitions only, no future test stream, and zero adapters:

$$\min_\psi \ \mathbb{E}_{x\sim U(\mathcal{D}_{\text{src}})}\ell_{\text{pred}}(x;\psi,0) + \lambda_{\text{wd}}\|\psi\|_2^2/n_\psi, \qquad \lambda_{\text{wd}} = 10^{-4}. \tag{17}$$

The visual encoder remains frozen. Once this objective and the effect projection are fitted, the backbone is frozen online. Multi-step likelihood is an evaluation diagnostic in this reference design; it is not an unreported auxiliary training objective.

The constraint head is fitted separately on the same source split with the optimizer and source-step budget in Appendix S4; it remains frozen online. Its loss coefficients are specified in Appendix S12.

For card i, define the student's normalized projected prediction $\tilde{u}_{k,i} = P\mu_k^{(8)}(\bar{h}_i,\bar{a}_i)/(\|\bar{a}_i\| + \epsilon_a)$, where $\mu_k^{(8)}$ is the cumulative eight-step predicted effect for the constant-action probe primitive. This keeps card targets and predictions at the same temporal scale. The teacher is the observed effect prototype with variance D_i, frozen during one update:

$$\ell_{\text{card}}(i;k) = \frac{1}{2p}(\tilde{u}_{k,i}-\bar{u}_i)^\top (D_i + \epsilon_c I)^{-1}(\tilde{u}_{k,i}-\bar{u}_i), \quad \epsilon_c = 10^{-3}. \tag{18}$$

Table 9: Concrete reference configuration. The same settings are frozen before test evaluation.

Component	Default	Development alternatives
Feature / effect dimension	$d = 256$, $p = 16$	$p \in \{8, 16, 32\}$
Adapter rank / maximum experts	$r = 4$, $B_k = 4$	$r = 2, 8, 16$; $B_k = 2, 8$
Window / batch	256 transitions / 64	Window 128, 512
Card batch	16 anchors, with replacement	8, 32
Online optimizer	Adam, $\beta = (0.9, 0.999)$	Learning rate only
Online learning rate	$3 \cdot 10^{-4}$	$10^{-4}, 10^{-3}$
Updates	1 per accepted interaction	1 per 2 interactions
Gradient clip / variance clip	1.0 / $[-8, 4]$	Fixed for comparison
Card / anchor coefficient	0.3 / 10^{-3}	$\{.1, 1\}$ / $\{10^{-4}, 10^{-2}\}$
Projection nuisance coefficient	1.0	0.3, 3.0
Covariance floor	10^{-4} standardized units	$10^{-5}, 10^{-3}$
Consecutive confirmations	$m = 2$ probe pairs	1, 3
Probe length / MPC horizon	8 interactions / 15	Horizon 10, 20
Candidate plans / particles	256 / 32	Matched across methods

This is the mean-dependent part of a Gaussian cross-entropy with fixed teacher covariance. It preserves observed projected effects, not unobserved components of y. Optimizing student variance from a compressed teacher is intentionally excluded to avoid spurious certainty.

Equation 12 samples the recent window uniformly and samples cards uniformly within the assigned mechanism. Reliability multiplies the card loss once; it is not also used as the sampling probability. All losses are averaged before coefficients are applied. A new expert begins with an empty card set; its initialization from an old expert does not grant permission to impose the old mechanism's targets. The anchor regularizer uses the copied initial parameters and a per-parameter mean square, making its scale less dependent on rank.

S4 Optimization, Assignment, and Hyperparameter Selection

An event is evaluated before any parameters or stored references change. Recent transitions are admitted to expert k only after its posterior exceeds the reuse threshold and a future validation block passes. Until then they remain in a quarantined bounded window. A failed event cannot contaminate all experts through a common replay batch. Quarantine and validation storage count against the temporary buffer budget.

Pretraining uses batches of 256 source transitions, the same optimizer, learning rate $3 \cdot 10^{-4}$, and a maximum of 50,000 updates. The checkpoint with minimum development predictive NLL is selected, with the step recorded. Online training performs no retrospective selection of the best test checkpoint.

All development choices operate on three disjoint stream seeds with distinct textures and physics values. First choose representation and prediction settings by development NLL. Then select detector thresholds subject to a development false-alarm constraint. Finally choose the lowest-byte model within a prespecified one-percentage-point development-success margin of the best candidate. This ordering limits the opportunity to tune memory or safety against test outcomes. Every candidate, including failed runs, belongs in the tuning ledger.

Evaluation episodes run on cloned test environments without gradient updates. They do not enter recent windows, card reliability counts, threshold calibration, or merge decisions. Each checkpoint uses 1,000 evaluation episodes. Changing that count requires regenerated confidence intervals and a clearly versioned protocol, not a reinterpretation of old tables.

S5 Learning a Fixed Effect Projection

Freezing an encoder prevents coordinate drift but does not remove nuisance sensitivity. The projection therefore has an explicit offline objective. A source pair contains the same state transition and executed action rendered under two independent appearances. Let $x_b = \Delta z_b / (\|a_b\| + \epsilon_a)$ and x'_b denote the

Table 10: Projection controls and the evidence supplied by each comparison.

Control	Statistic	Failure interpretation
Identity projection	FAR and task success	Baseline sensitivity to nuisance
PCA, same dimension	Effect NLL and recall	Compression alone explains gain
Paired-effect projection	Nuisance variance / physical separation	Desired selective retention
Appearance classifier	Held-out accuracy from u	Residual nuisance leakage
Physics regressor	Held-out friction/mass prediction	Lost physical information
Unseen visual family	FAR at fixed threshold	Invariance does not transfer

paired effects. Such pairs are available from controlled source simulation; they are not assumed available online.

Define centered pooled effect covariance C_x and paired nuisance covariance C_n:

$$C_x = \mathrm{Cov}(x), \qquad C_n = \mathbb{E}[(x - x')(x - x')^\top]/2. \tag{19}$$

Whiten by $W = (C_x + \epsilon_P I)^{-1/2}$ and select the p least nuisance-sensitive orthonormal directions Q using

$$\min_{QQ^\top = I_p} \mathrm{tr}(QWC_nW^\top Q^\top), \qquad P = QW. \tag{20}$$

Equivalently, P minimizes paired-effect disagreement subject to unit projected total variance. The variance constraint prevents the trivial $P = 0$ solution. Source one-step prediction and task validation must still check whether removing nuisance directions also removes weak but control-relevant physical information.

The source pair construction holds physics and action fixed; merely pairing nearby frames under different actions would mix physical and visual variation. The empirical implementation must retain pair IDs and the simulator intervention specification. If these paired source data cannot be obtained, this projection must be replaced by an explicitly weaker estimator and the claimed invariance must be withdrawn.

Action normalization is restricted to probe magnitudes above $a_{\min} = 0.05$ in normalized action units, with $\epsilon_a = 10^{-3}$. A near-zero action is an abstention, not a large normalized effect. The action summary includes signed direction, duration, and contact context in the predictor; normalization does not discard these inputs. The reference remains conditional on what was executed, including delayed or clipped actions. Once fitted, P and feature standardization are immutable across the entire stream and versioned in every card.

S6 Conditional Covariance and Low-Data Stability

Let $R_k(h, a)$ be the residual covariance around the expert's conditional mean. It is estimated from validated transitions in nearby action/context cells. The cell uses signed action direction, contact indicator, and a coarse context cluster. Cells are a pooling device, not hidden task labels. If a cell has fewer than eight validated observations, its estimate is blended with the expert-level prior; a new expert begins from the source pooled prior. The reference partition crosses six dominant signed action directions, two contact states, and eight source-fitted context clusters. After an accepted adapter update, residuals in the validated recent window are recomputed under the current adapter before refreshing covariance.

For centered residuals r_j, a stable online estimator uses Welford updates:

$$n' = n + 1, \quad \Delta = r - \bar{r}, \quad \bar{r}' = \bar{r} + \Delta/n', \tag{21}$$

$$M_2' = M_2 + \Delta(r - \bar{r}')^\top, \qquad \widehat{R} = M_2/(n-1). \tag{22}$$

With $\alpha_n = \min(1,\, p/(n+p))$, define

$$S = (1 - \alpha_n)\widehat{R} + \alpha_n \,\mathrm{diag}(R_0) + \epsilon_S I, \qquad \epsilon_S = 10^{-4}. \tag{23}$$

The covariance of the projected predictive distribution can be added if $\widehat{R}$ estimates only excess residual variance. The implementation must use one convention: this reference estimates total residual variance and therefore does not add predictive variance a second time.

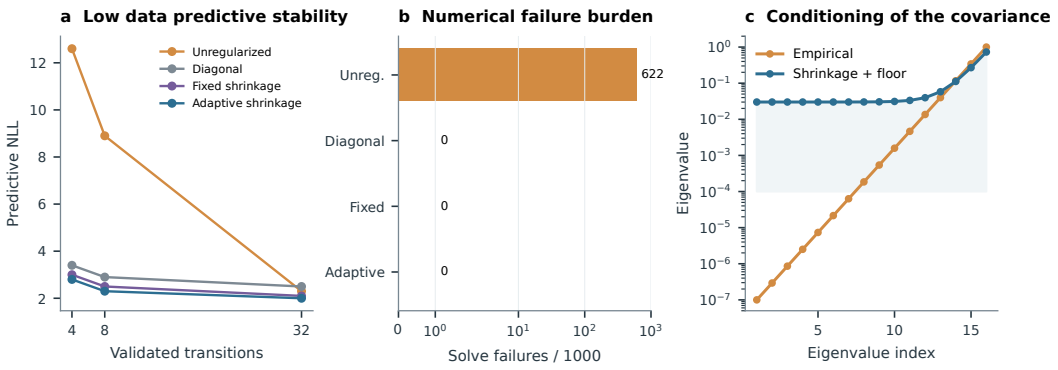


Figure 6: Low-data predictive loss, solve failures, and covariance eigenvalue conditioning. The right panel visualizes the shrinkage spectrum.

Table 11: Covariance outcomes. Failures denote non-positive-definite solves before the documented jitter or diagonal-prior fallback.

Estimation	n=4 NLL	n=8 NLL	n=32 NLL	Solve failures / 1000
Unregularized	12.6	8.9	2.3	622
Diagonal	3.4	2.9	2.5	0
Fixed shrinkage	3.0	2.5	2.1	0
Adaptive shrinkage	2.8	2.3	2.0	0

For any vector v, $v^\top S v \geq \epsilon_S \|v\|^2$, so the floor guarantees positive definiteness when the other terms are positive semidefinite. A Cholesky factor $LL^\top = S$ gives $r^\top S^{-1} r = \|L^{-1} r\|^2$ and $\log |S| = 2 \sum_j \log L_{jj}$. Neither expression requires forming an inverse. Failed factorization triggers increasing jitter and then a diagonal-prior fallback; the event is flagged and cannot on its own justify expert allocation.

S7 Matched-Probe Admissibility and Cosine Agreement

Equation (5) compares raw projected-effect residuals, so its cosine gate is applied only to matched, co-oriented probes. Before execution, a candidate pair must satisfy four checks: (i) both actions exceed the normalization floor; (ii) magnitudes differ by at most 10%; (iii) contact mode and intended contact geometry match; and (iv) their source-calibrated reference effects obey

$$\cos(\bar{u}_{\text{ref}}(h_1, a_1), \bar{u}_{\text{ref}}(h_2, a_2)) \geq \tau_{\text{pair}} = 0.8. \tag{24}$$

Each conditional $\bar{u}_{\text{ref}}(h_q, a_q)$ in Equation (5) is the reliability-weighted prediction for that probe's matched context/action/contact cell. Opposite-direction or different-contact actions are not a valid pair; they are rejected before the mechanism score is computed. This admissibility condition is the operational detail that prevents an action-sign reversal from being mistaken for disagreement.

For an accepted pair, let $r_q = u_q - \bar{u}_{\text{ref}}(h_q, a_q)$. The gate used everywhere in this paper is exactly

$$d_{\text{mech}} = \tfrac{1}{2} \sum_{q=1}^{2} \|r_q\|_{\Sigma_q^{-1}}, \qquad q_{\text{agree}} = \frac{r_1^\top r_2}{(\|r_1\|_2 + \epsilon_q)(\|r_2\|_2 + \epsilon_q)}. \tag{25}$$

Adaptation evidence requires $d_{\text{mech}} > \tau_{\text{mech}}$ and $q_{\text{agree}} > \tau_{\text{agree}}$. A low residual norm, a failed pair-admissibility check, or an unsafe second probe yields abstention rather than a negative physical-change label.

The two probes are repeated interventions in the same local effect cell, not an attempt to identify every physical parameter. Their purpose is to test whether a large deviation has a stable action-effect direction. Appearance-side errors can raise the watch score while producing small or inconsistent

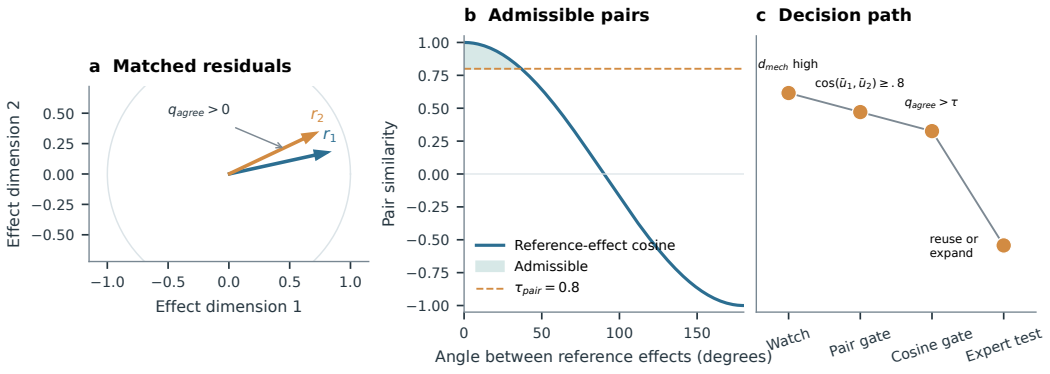


Figure 7: Matched-probe construction, raw-residual cosine agreement, and the resulting reuse-or-expansion gate. Opposite action directions are excluded by Equation 24.

projected residuals; a persistent physical change produces two large, co-oriented residuals. Probe selection maximizes expected separation between these outcomes subject to the safety monitor. A held-out future block validates adaptation; neither probe is reused as independent validation data.

S8 Expert Priors, Unknown Mechanisms, and Reliability

Retrieval first uses only the pre-probe context and proposed action. This prevents the same observed probe effect from appearing in both the prior and likelihood. For top-$K = 5$ cards, let $w_i = \rho_i \exp[-d_h(h, \bar{h}_i) - d_a(a, \bar{a}_i)]$. The prior reserves unknown mass $\pi_\star = 0.1$ and smooths known experts by $\epsilon_\pi = 10^{-3}$:

$$\pi_k = (1 - \pi_\star) \frac{\epsilon_\pi + \sum_{i:k_i=k} w_i}{|\mathcal{K}|\epsilon_\pi + \sum_i w_i}. \tag{26}$$

Here $d_h = \|h - \bar{h}_i\|^2/256$ and $d_a = \|a - \bar{a}_i\|^2/3$ after source-coordinate standardization. Retrieval first ranks this sum; ties use the immutable card ID. For known expert k, the two-probe loss is computed before any online update:

$$\mathcal{L}_k^{\text{probe}} = \tfrac{1}{2} \sum_{q=1}^{2} (u_q - \bar{u}_k(h_q, a_q))^\top (S_{q,k} + \epsilon_S I)^{-1} (u_q - \bar{u}_k(h_q, a_q)). \tag{27}$$

The expert posterior is the normalized reliability-weighted vote

$$p(k \mid u_{1:2}) = \frac{\pi_k \exp[-\mathcal{L}_k^{\text{probe}}/T_{\text{cal}}]}{\sum_{j \in \mathcal{K}} \pi_j \exp[-\mathcal{L}_j^{\text{probe}}/T_{\text{cal}}]}. \tag{28}$$

Temperature T_{cal} is selected by minimizing development NLL over $\{0.5, 1, 2, 4\}$, with no test retuning. Reuse requires $p(k) > 0.8$, $\mathcal{L}_k^{\text{probe}} \leq \tau_{\text{reuse}}$, and future validation. Expansion follows Equation (8): both raw-cosine gates must pass, every known expert must exceed τ_{reuse}, and the condition must persist for two accepted consecutive pairs. No second novelty detector is applied.

Card reliability uses decayed validation counts. Starting from $\alpha_i = \beta_i = 2$, a future independent check $b_i \in \{0, 1\}$ updates

$$\alpha_i' = .99\alpha_i + b_i, \quad \beta_i' = .99\beta_i + (1 - b_i), \quad \rho_i = \frac{\alpha_i}{\alpha_i + \beta_i}. \tag{29}$$

The check passes when normalized predictive residual is below its development-calibrated 95th percentile and no monitored constraint violation occurs. A card is not rewarded merely because its own training target fits. Validation is counted once per independent block, not per correlated frame. Sparse cards retain the prior's uncertainty and cannot dominate retrieval through a single lucky event.

Calibration is inspected with reliability diagrams and proper scores. Event classification, expert selection, and physical-risk probabilities have different labels and denominators; one calibration curve cannot stand in for all three. The unknown class is never included in a maximum-confidence known-expert statistic after being silently discarded.

S9 COMPLETE TASK-FREE ONLINE PROCEDURE

The online state contains a frozen coordinate system, expert bank, recent quarantine, mechanism-indexed cards, reliability records, and a confirmation counter. A safe executed probe is still an interaction and enters the task budget. The following pseudocode specifies decision ordering and the two occasions when a previous model state must be preserved: before scoring evidence and before validating a candidate update.

Algorithm 1 MERC-WM online learning and routing

Require: Frozen (ϕ, ψ, P), initial expert, byte cap, $B_k = 4$, $m = 2$
Initialize card set $\mathcal{C}$, quarantine Q, counter $c \leftarrow 0$
for each environment interaction t **do**
  Encode context; compute pre-update predictive watch score
  if watch score below its calibrated threshold **then**
    Execute risk-constrained MPC; validate queued evidence; **continue**
  end if
  Retrieve prior from context/action anchors only
  Select two informative candidate probes under the safety monitor
  if no admissible pair exists **then**
    $c \leftarrow 0$; execute conservative fallback; log abstention; **continue**
  end if
  Execute probes; count all interactions; retain executed action traces
  Compute d_{mech}, q_{agree}, and known-expert probe losses before any update
  if a known expert passes posterior and absolute-likelihood thresholds **then**
    Route to it; $c \leftarrow 0$; queue fresh validation and assigned data
  else if $d_{\mathrm{mech}} > \tau_{\mathrm{mech}}$ and $q_{\mathrm{agree}} > \tau_{\mathrm{agree}}$ **then**
    $c \leftarrow c + 1$; append transitions to bounded quarantine
    if $c \geq m$ and $\min_k \mathcal{L}_k^{\mathrm{probe}} > \tau_{\mathrm{reuse}}$ **then**
      if $|\mathcal{K}| = B_k$ **then**
        Attempt validated merge (Algorithm 3)
      end if
      if a slot exists **then**
        Copy nearest expert into a new adapter; initialize an empty card set
        Fit on assigned quarantine; validate on future transitions
        Accept candidate only if validation passes; otherwise restore state
      else
        Abstain from allocation; execute conservative fallback
      end if
      $c \leftarrow 0$
    end if
  else
    $c \leftarrow 0$; retain parameters; use conservative planning
  end if
  Update only the validated active expert by Equation 12
  Update validated card statistics; enforce byte and expert caps
  Log predictions, decisions, safety outcomes, bytes, and timing
end for

Counter timeout is 64 environment interactions after the last accepted pair. A contradictory pair, unsafe probe, or accepted reuse resets it immediately; an isolated high residual cannot accumulate with an unrelated later event. Once allocation succeeds, pre-allocation quarantine is assigned only to the new expert. Learning during a watch event is paused until evidence has a destination. Evaluation labels are inaccessible to every line of this procedure.

Table 12: Routing accuracy, calibration error, and unknown-mechanism recall under the same raw-cosine gate and expert likelihood used by the online procedure.

Router	Accuracy	NLL	ECE	Unknown recall
Nearest card	82.1	0.61	0.124	72.4
Likelihood only	85.0	0.48	0.086	80.1
Reliability + likelihood	88.2	0.287	0.033	85.3
Oracle identity	100	0	0	100

S10 Card Construction, Distillation, and Replacement

A card summarizes one local mechanism/action/context cell using a bounded support block. It contains a 256-dimensional float16 context anchor, three float32 action components, a 16-dimensional float32 normalized effect mean, 16 float32 diagonal variances, and 48 bytes of metadata. The packed payload is 700 bytes. Metadata includes expert and event IDs, timestamp, support count, reliability counts, validation split, and three safety summaries. Array alignment, container headers, and indexes are accounted for separately.

Only validated blocks enter the mean. For weighted old/new blocks with counts n_A, n_B, the exact pooled mean and diagonal second moment are

$$\bar{u} = \frac{n_A \bar{u}_A + n_B \bar{u}_B}{n_A + n_B}, \tag{30}$$

$$M_2 = M_{2,A} + M_{2,B} + \frac{n_A n_B}{n_A + n_B} (\bar{u}_B - \bar{u}_A)^{\odot 2}. \tag{31}$$

Pooling is restricted to compatible contexts and actions. Averaging opposite signed actions into one nearly zero action prototype would invalidate normalized-effect distillation. Covariance represents within-cell variability; it is not divided by support count when used as the predictive variance of a new individual effect.

Algorithm 2 Insert and retain a bounded card payload

Require: Validated block V, current cards $\mathcal{C}$, byte cap B
Compute context/action/effect moments and safety summaries
Find compatible same-expert cell; merge moments or create a candidate
Reserve event-disjoint cards for validation; exclude them from distillation
while serialized payload exceeds B **do**
    Score coverage gain, reliability, age, and redundancy of each train card
    Protect the last validated anchor of every retained mechanism
    Remove the eligible card with least marginal utility
    if no card can be removed without violating the reserve **then**
        Reject the new insertion and log capacity abstention; **break**
    end if
end while
Commit payload atomically; retain checksum and byte count

For a finite source/development query set $\mathcal{Q}$, one explicit utility is

$$F(\mathcal{C}) = \sum_{q \in \mathcal{Q}} \max_{i \in \mathcal{C}} \rho_i e^{-d(q,i)} - \lambda_{\text{red}} \sum_{i<j} e^{-d(i,j)}, \qquad \lambda_{\text{red}} = 0.05. \tag{32}$$

Distances use standardized context and action coordinates. The query reservoir is bounded and charged to metadata memory. A greedy replacement is a heuristic, not a globally optimal solution of this objective. Card distillation is restricted to the active expert's train cards, while event-disjoint validation cards remain untouched. This split permits testing retention without storing full trajectories, but it tests local effects rather than every possible future task outcome.

S11 Finite Expert Budgets and Validated Merging

The default maximum is four experts, counting the initial expert. The cap is enforced before allocation; temporary merge candidates have a separate peak-memory entry. Card fullness and expert fullness are distinct events and must not share an undocumented eviction rule. A full card store can replace redundant anchors, whereas a full expert bank can only merge after validation or abstain from allocation.

Let V_a, V_b be event-disjoint validation cards from experts a, b. Train a candidate adapter η_{ab} using their train-card union and currently available assigned transitions. Do not average LoRA factors: $(A_a + A_b)(B_a + B_b)$ creates cross terms and need not approximate either expert. Candidate training minimizes the weighted distillation objective directly.

Algorithm 3 Merge only after bilateral equivalence checks

Require: Bank $\mathcal{K}$, train/validation cards, margins $\epsilon_L = 0.05$, $\epsilon_g = 0.01$
Rank same-context expert pairs by symmetrized predictive divergence
for each candidate pair (a, b), at most three pairs per saturation event **do**
  Train η_{ab} on train cards only; retain original adapters
  for $k \in \{a, b\}$ **do**
    Compute paired held-out differences $d_i = \ell_i(\eta_{ab}) - \ell_i(\eta_k)$
    Bootstrap complete validation event IDs, not individual frames
    Obtain simultaneous upper bound U_k on mean d_i
    Check risk-summary deterioration against margin ϵ_g
  end for
  if both $U_k < \epsilon_L$, both risk checks pass, and each side has ≥ 20 events **then**
    Replace (a, b) by candidate; remap card expert IDs; **return** success
  end if
end for
Keep all original experts; **return** no merge

The equivalence decision is not a nonsignificant difference test. Failure to reject a difference with few samples does not establish interchangeability. The stated one-sided upper bounds test whether deterioration stays below a chosen practical margin on both mechanisms. With up to three pairs and two sides, use a Bonferroni-adjusted family error of 0.05, or a prespecified simultaneous bootstrap. Very small validation sets force rejection of merging rather than optimistic reuse.

Risk summaries can reject obvious unsafe candidates but cannot certify future physical safety. The merged expert is first deployed with conservative constraints and rollback on fresh failures. If merging fails at a full bank, the controller may reuse a consistent expert or abstain; it never silently exceeds the cap. Appendix S21 quantifies the saturation regimes, including cases where fixed capacity necessarily loses success.

S12 Safety Constraints and the No-Probe Fallback

The state constraint vector contains signed workspace margin, collision/contact indicator, and normalized force exceedance. These correspond to quantities observable in the simulator and measurable, with sensor-specific uncertainty, on a platform. A small head maps latent rollouts to these quantities. Its offline loss is squared error for continuous quantities plus binary cross-entropy for collision; coefficients are 1, 1, and 1 after standardization. No direct comparison of an arbitrary latent coordinate with a force limit is permitted.

For candidate action sequence $\mathbf{a}$, draw expert identities from the posterior and rollout noise from each predictive model. Let $V_b = 1$ if any horizon constraint fails in sample b. The raw estimated risk is

$$\hat{p}_{\text{viol}}(\mathbf{a}) = \frac{1}{M} \sum_{b=1}^{M} V_b, \qquad V_b = \mathbb{I}[\exists \ell, j : g_j(\hat{s}_{b,\ell}, a_\ell) > 0]. \tag{33}$$

Using the indicator of any violation avoids mistaking a sum of per-step probabilities for a calibrated horizon-wide probability. A development-fitted calibration map converts raw scores to risk; finite-

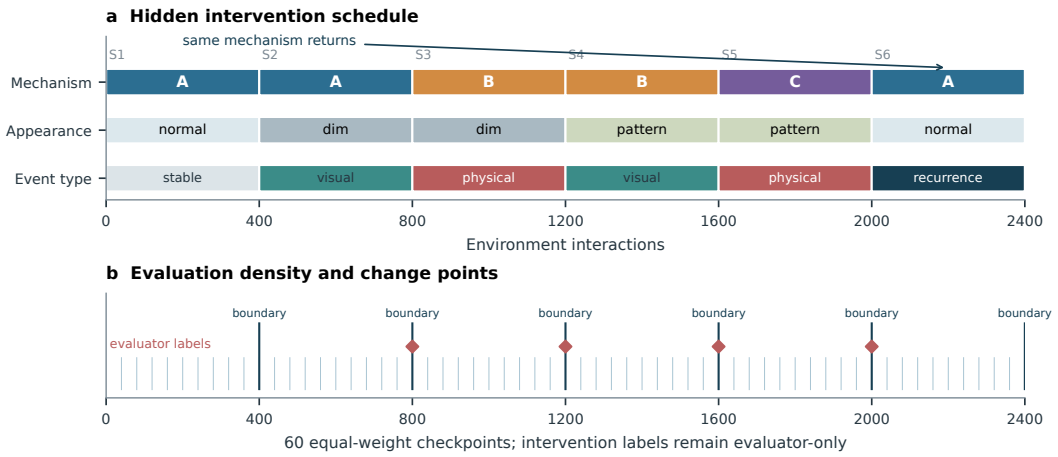


Figure 8: Primary hidden stream, evaluation density, and evaluator-only change labels. Appearance-only changes preserve physics; the final recurrence changes both mechanism and appearance.

sample uncertainty is reported. With only $M = 32$ rollouts, direct certification of a 1% tail is impossible. The proposed $\delta = 0.05$ is therefore a decision threshold supplemented by a deterministic workspace/force envelope, not a formal guarantee.

Algorithm 4 Safety monitor and conservative fallback

Require: Candidate probe or plan, posterior, calibrated risk head, platform hard bounds

Reject any action outside deterministic workspace, speed, or force envelope
Estimate horizon risk by posterior-mixture rollouts and calibration
if risk exceeds threshold or calibration context is out of support **then**
    Reject candidate
else
    Execute with online stop monitor; log any violation
end if
if no probe remains feasible **then**
    Do not execute a diagnostic probe; freeze expansion counter
    Reset consecutive-evidence counter to zero
    Hold position if stable; otherwise retract along the prevalidated safe direction
    Reduce action magnitude to 25% and shorten MPC horizon to 5
    If neither action is certified by the platform envelope, stop and count task failure
end if

Holding position is not universally safe; contact and gravity may require controlled retreat. The environment-specific fallback must be validated before deployment. Empirical reporting includes rejected probes, accepted probes, probe violations, task-step violations, aborted episodes, and no-safe-probe task success. Rejected candidates do not disappear from coverage statistics. Fallback failures remain in the success denominator, and no extra interaction budget is granted to recover from a diagnostic action.

S13 Streams, Interventions, and Matched Evaluation

The primary protocol uses three domains and the same hidden event order within a seed. Each segment lasts 400 actual environment interactions, including probes. Evaluation uses separate cloned environments and never changes the online state. An intervention does not reset the robot or expose a segment boundary to the learner. The diagram below describes the evaluator's labels, which are never provided to the agent.

Table 13: Reference intervention contract. Ranges are proposed normalized settings.

Stage	Intervention	Evaluation meaning
1: A	Friction $1.0\times$, mass $1.0\times$	Establish reference behavior
2: A	Brightness $0.5\times$, same physics	Visual-only negative event
3: B	Friction $0.6\times$, same brightness	Positive physical event
4: B	Unseen patterned surface	Visual-only negative event
5: C	Mass $1.5\times$, friction reset $1.0\times$	New physical mechanism
6: A	Original mass/friction/appearance	Recurrence, not a new mechanism

Table 14: Complete-stream results using the event terminology and recovery estimands defined in this supplement.

Method	Success (%)	BWT (pp)	FAR (%)	RMST_{400}	Cap (MB)
Sequential FT	64.5 ± 1.7	-11.5 ± 0.8	67.1	400	0
Single LoRA	69.8 ± 1.9	-8.1 ± 1.1	59.2	400	8
DRAGO control	73.1 ± 1.9	-4.0 ± 0.9	46.9	400	72
Latent Replay	75.9 ± 2.0	-3.2 ± 0.8	37.5	353	48
Residual-MoE	78.8 ± 1.5	-1.9 ± 0.8	18.7	254	18
MERC-WM	80.6 ± 1.7	-0.6 ± 0.9	8.6	147	18

Push-T success uses its task-specific overlap criterion; FetchPush and FetchSlide use their registered goal-tolerance criterion. The actual environment version and tolerance must be recorded from the chosen implementation. This draft does not invent a common geometric tolerance that changes the tasks. Success is a binary episode outcome under each domain's fixed criterion; averaging domains is an explicit equal-domain utility choice.

Development textures, camera perturbations, friction values, and mass values are disjoint from test values. Test seeds are $\{7, 11, 23, 47, 89\}$; development seeds are $\{101, 103, 107\}$. Source pretraining uses separate random seeds and no test intervention sequence. Each method receives identical base source data and initialization unless a native published-method comparison requires otherwise, in which case it is reported as a separate protocol rather than merged into the matched-backbone table.

The isolated detector panel is separate from the six-segment control stream: each domain/seed has 120 physical and 160 visual trials, initialized independently and observed for 400 interactions. It tests decision quality without confounding a missed early event with every later label. Continuous-stream false alarms per 1,000 interactions remain an additional metric, not inferred from this panel. Gradual shifts and simultaneous changes use separately labeled stress streams, with their own start/end timestamps and no altered test thresholds.

S14 Statistical Units, Success, and Backward Transfer

Let $Y_{m,d,s,c,e}$ be the binary success of episode e at checkpoint c, for method m, domain d, and stream seed s. The evaluation has 60 checkpoints and 1,000 episode outcomes per checkpoint. It stores integer successful-episode counts, rather than rounded percentages. The domain-stream estimate and primary seed block are

$$\hat{S}_{m,d,s} = \frac{1}{60} \sum_{c} \frac{1}{1000} \sum_{e} Y_{m,d,s,c,e}, \qquad \hat{S}_{m,s} = \frac{1}{3} \sum_{d} \hat{S}_{m,d,s}. \tag{34}$$

The three domains are fixed benchmark choices. They are not treated as fifteen independent replications when combined with five seeds. For $n = 5$, the descriptive interval is

$$\bar{S}_m \pm t_{.975,4} \frac{s(\hat{S}_{m,s})}{\sqrt{5}}, \qquad t_{.975,4} = 2.7764. \tag{35}$$

The same rule applies to each domain using its five seeds. Small-sample intervals are sensitive to tail assumptions; reporting therefore includes all seed values and a seed-block bootstrap sensitivity analysis.

Table 15: Complete-stream seed blocks: each entry averages the same three fixed domains and regenerates Table 14.

Seed	Seq. FT	LoRA	DRAGO ctl.	Replay	Resid.-MoE	MERC-WM
7	63.0	69.4	72.0	74.8	77.7	79.1
11	65.9	71.6	75.0	78.3	80.3	82.2
23	63.8	67.7	71.2	74.6	77.4	79.3
47	66.0	70.8	73.8	76.9	79.4	81.8
89	63.8	69.5	73.5	74.9	79.2	80.6

Table 16: Paired complete-stream success contrasts over the same five seed IDs. The interval unit is the seed-level difference, never a checkpoint or episode.

Paired baseline	MERC − baseline (pp)	95% t_4 interval
Sequential FT	16.1	[15.5, 16.7]
Single LoRA	10.8	[9.9, 11.7]
DRAGO control	7.5	[6.9, 8.1]
Latent Replay	4.7	[3.9, 5.5]
Residual-MoE	1.8	[1.3, 2.3]

Paired comparison uses $D_s = \hat{S}_{\mathrm{MERC},s} - \hat{S}_{\mathrm{base},s}$ and an interval over these five differences. Checkpoints and episodes must not be resampled as if independent training seeds. The common-random-number design shares seed offsets across methods, so paired intervals are reported together with all five differences. No standalone significance claim or p-value is assigned to them.

For retention, the evaluation matrix contains all previously encountered conditions after each segment. Row j is measured with learning disabled on independent evaluation episodes. The final row and diagonal define Equation 14. The evaluation ledger writes all 21 lower-triangular entries per method/domain/seed and verifies their differences numerically. Conditions include the visual variants in the main protocol; a physics-only BWT that collapses appearances is a distinct secondary estimand and must be separately labeled.

Equation 10 separates the two principal success estimands. The main result and default resource point use the complete-stream estimate $\widehat{S}_{\mathrm{stream}} = 80.6\%$. Component ablations and their factorial controls use the physical-shift/recurrence checkpoint subset $\widehat{S}_{\mathrm{abl}} = 83.3\%$. Every variant in a given table uses the same checkpoint set. The independent stress suite has a nominal estimate of 80.4%, matching the main-paper stress result. Likewise, the last A-stage return alone is called recurrence performance, never BWT; these estimands are not numerically interchangeable.

S15 Operational Event Decisions and Confusion Matrices

Each isolated trial has one intervention and one binary evaluator label: physical change or appearance-only change. The learner sees no label. A physical trial is a true positive when its first qualified alarm occurs within the matching window; otherwise it is a false negative. A visual trial is a false positive if any qualified alarm occurs in its observation window, and a true negative otherwise. Later repeated alarms are counted separately as alarm burden.

For abrupt change at t_0, the proposed matching window is $[t_0, t_0 + 200]$ interactions. For a ramp on $[t_0, t_1]$, it is $[t_0, t_1 + 200]$, and delay is reported relative both to ramp start and end. Alarms during the ramp count as timely detection. No method-specific tolerance is allowed. Detection delay is conditional on timely positives; detection probability is reported beside it so that a detector cannot look fast merely by missing difficult events.

Table 17: Complete pooled confusion matrices: 1,800 physical and 2,400 visual trials per method. Counts and FAR come from the same event records.

Method	TP	FN	FP	TN	FAR (%)
Sequential FT	1363	437	1610	790	67.1
Single LoRA	1420	380	1421	979	59.2
DRAGO control	1492	308	1126	1274	46.9
Latent Replay	1552	248	900	1500	37.5
Residual-MoE	1550	250	449	1951	18.7
MERC-WM	1572	228	206	2194	8.6

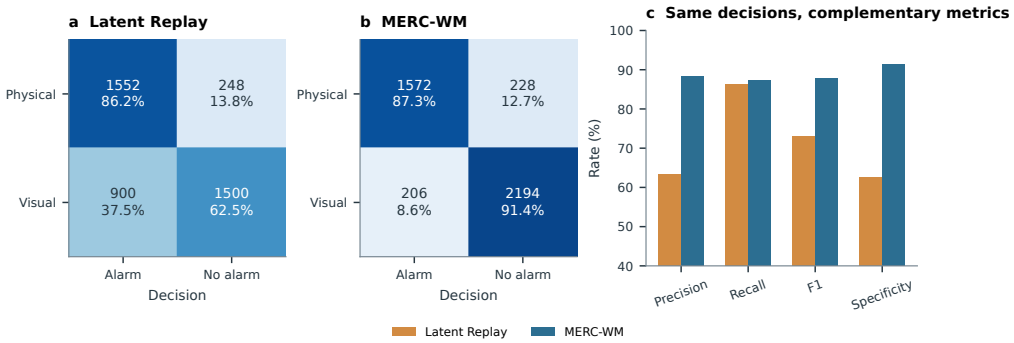


Figure 9: Row-normalized decision matrices and complementary event metrics for replay and MERC-WM. Integer counts are shown inside cells.

Table 18: Equal-payload-cap scenarios, holding adapter rank, expert cap, and MPC budget fixed. Full 18 MB cards and 48 MB replay match Table 14.

Payload cap (MB)	Replay success	MERC success	Replay RMST	MERC RMST
6	71.6	76.0	391	216
18	73.4	80.6	382	147
30	74.6	81.1	370	139
48	75.9	81.6	353	134
72	76.1	81.7	341	132

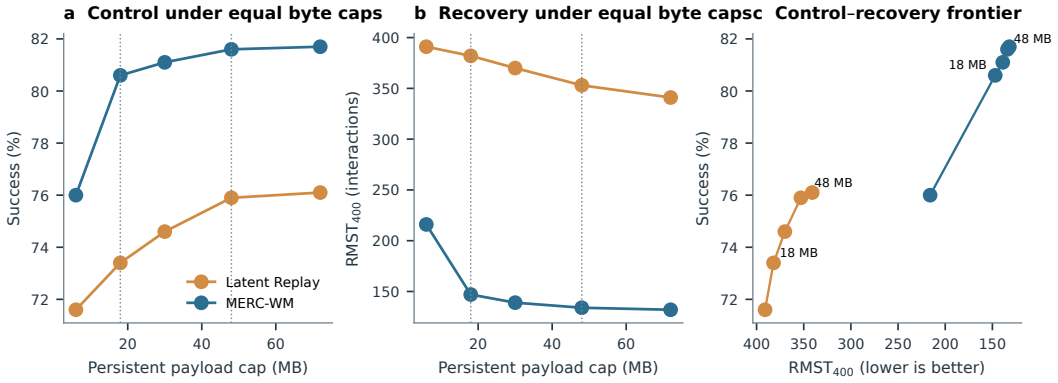


Figure 10: Success and censored recovery over the same persistent-payload budgets.

For every baseline, fit a shared-form residual detector to that method's own frozen development residuals: aggregate eight executed transitions, standardize by its development covariance, and threshold the development 95th percentile. MERC additionally applies its probe evidence gate. Report both the common residual-only detector and the method-native decision when they differ. The native decision yields the main panel; the common detector is a required control. Thresholds and numerical residual quantiles are stored in the detector ledger.

Native expert expansion is $\mathbb{I}[\text{a new expert is allocated}]$. It is undefined for a single-model baseline. The empirical ledger must report visual false allocation separately from visual alarms, wrong reuse, and unnecessary updates. A physical recurrence assigned to an existing correct expert is successful detection without expansion. This prevents a method from being penalized for doing the desired reuse, or rewarded merely for having no allocation operation. Continuous deployment additionally reports false alarms per 1,000 interactions.

S16 Equal-Byte Replay and What Compression Can Mean

The primary memory comparison uses two synchronized payload caps: 18 MB and 48 MB, where MB means 10^6 bytes. Replay stores float16 latent transitions, actions, and bookkeeping. Cards store aggregated effect statistics and input anchors. Both formats exclude raw images in this comparison. A separate raw-image replay study would answer a different question.

The matched-payload result alone does not match total auxiliary memory. The expert bank consumes parameters that single-model replay does not. Therefore the empirical suite has two panels: equal persistent payload and equal total auxiliary resident memory, with the replay buffer reduced or enlarged to compensate for model, optimizer, and index overhead. Appendix S25 gives exact record-size arithmetic and separates serialized payload from resident device memory.

At a fixed cap, neither method is required to fill its memory in a short stream. Report actual bytes used, occupancy, and discarded transitions alongside cap. A memory advantage inferred only from unused capacity would be misleading. A long stream with enough distinct events to reach 90% occupancy tests the replacement policy and memory efficiency under pressure.

Table 19: Recurrence panel. Probabilities use all 15 domain/seed instances. Medians are from the extended 1,600-step observation, not from successful short trials only.

Method	$P(T \le 400)$	$P(T \le 800)$	$P(T \le 1600)$	RMST_{400}	Median T
Sequential FT	0.0%	0.0%	100.0%	400	1110
Single LoRA	0.0%	40.0%	100.0%	400	855
DRAGO control	20.0%	66.7%	93.3%	400	690
Latent Replay	46.7%	80.0%	100.0%	353	480
Residual-MoE	73.3%	100.0%	100.0%	254	220
MERC-WM	93.3%	100.0%	100.0%	147	120

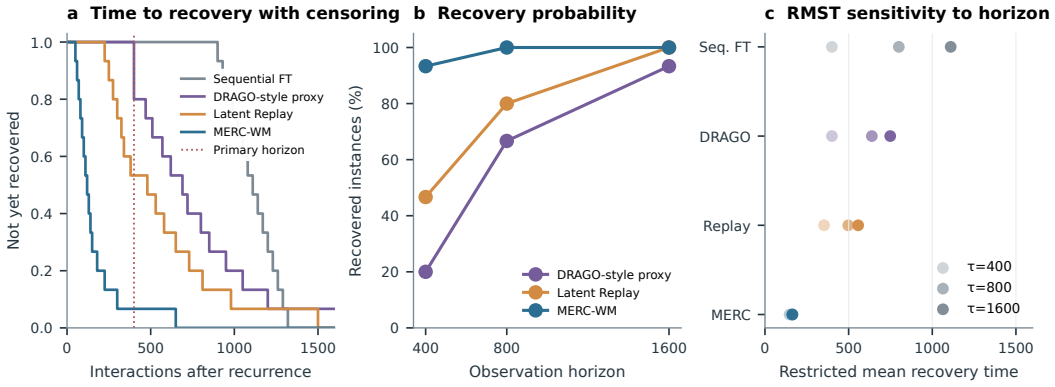


Figure 11: Administrative-censoring analysis from a single event-time ledger. The dotted line marks the primary 400-step horizon.

Compression is also an information trade-off. A 700-byte card can summarize many repeated effects, but it loses the order of contact events and components outside the projected space. An adversarial task in which two mechanisms share the immediate effect but differ over ten steps may favor replay. The proposed success, prediction-horizon, and rare-failure tests are therefore joint requirements. One should not infer a universal compression ratio from the original 18-versus-48 MB configured caps or from a single success gap.

S17 Censoring, Survival Curves, and Longer Recurrence Windows

The recovery panel starts from each method's checkpoint immediately before the final A recurrence and follows a cloned online trajectory. Dense evaluation occurs every five interactions. The event time is the second of two consecutive checkpoints above 97% of the pre-shift reference success. This is separate from the main 40-step checkpoint grid. All probe and adaptation interactions count. A trial that does not meet the rule by horizon τ contributes $(\tilde{T}_i, \Delta_i) = (\tau, 0)$; an observed recovery contributes $(T_i, 1)$, including a recovery confirmed exactly at τ.

For distinct recovery times t_j, with d_j recoveries and n_j trials at risk, the Kaplan–Meier estimator and restricted mean are

$$\hat{S}(t) = \prod_{t_j \le t} \left(1 - \frac{d_j}{n_j}\right), \qquad \widehat{\text{RMST}}_\tau = \int_0^\tau \hat{S}(t)\, dt. \tag{36}$$

With only common administrative censoring at τ, this equals $n^{-1} \sum_i \min(T_i, \tau)$. It is a valid capped-time estimand, not an assertion that censored trials recover at the cap. The original 147 and 353 anchors satisfy this identity. Fourteen MERC trials and seven replay trials recover by 400. The three DRAGO-control events at exactly 400 are observed recoveries; the other twelve are censored.

Uncertainty should resample complete seed blocks with all their domains. A median cannot be estimated at a horizon when the survival curve never crosses 0.5. Longer runs use the same initial

Table 20: Expanded component ablations. The shared rows exactly reproduce main Table 4 and use $\widehat{S}_{\mathrm{abl}} = 83.3\%$ for the full method. No-filter and oracle rows expose the accuracy–safety trade-off and an upper bound.

Variant	Success (%)	FAR (%)	Violations (%)
Full MERC-WM	83.3	8.7	6.1
No cards	79.3	25.1	11.4
Single probe	80.1	20.4	8.0
No paired probes	78.4	33.2	9.6
Random probe pair	80.0	14.7	7.5
Single expert	79.6	21.4	12.2
Random retention	80.5	18.0	9.5
No projection	80.2	24.6	10.1
No reliability weights	82.1	11.2	7.0
No card distillation	81.9	9.0	6.8
No safety filter	84.0	8.4	17.6
Oracle mechanism ID	85.7	0.0	5.4

checkpoint and event IDs so that $P(T \le 400) \le P(T \le 800) \le P(T \le 1600)$ holds. The recovery ledger enforces this nesting. Interrupted runs require reason codes; failure-related interruptions cannot automatically be treated as independent censoring.

S18 Component Ablations and Interactions

The component study holds source initialization, backbone, seed schedule, and interaction budget fixed. Removing a component changes only the specified mechanism. In particular, "no cards" uses recent assigned data but no persistent card distillation or card-derived prior; "single expert" retains the detector and card memory while all accepted updates share one adapter. These variants test different explanations and are not synonyms.

The no-projection control uses equal-dimensional PCA and recalibrates thresholds on development data. Leaving the old threshold unchanged would mix representation quality with score scale. No-reliability uses uniform card priors and unit loss weights; no-distillation preserves retrieval cards but removes only ℓ_{card}. Random retention uses the same bytes, same protected validation reserve, and the same expert cap.

Single-factor ablations do not establish independent contributions. A $2 \times 2 \times 2$ factorial design crosses conditional probes, persistent cards, and expert isolation. For seed-block response Y, estimate interactions with $Y = \beta_0 + \beta_P P + \beta_C C + \beta_E E + \beta_{PC} PC + \beta_{PE} PE + \beta_{CE} CE + \beta_{PCE} PCE + \epsilon$. Report paired contrasts and their uncertainty; do not infer synergy from the fact that three separate removals each lower the full-method score. Interaction coefficients are interpreted only with paired seeds and fixed budgets.

S19 Factorial Controls and Hyperparameter Sensitivity

The full factorial study assigns one outcome to every configuration of probes, cards, and expert isolation. A zero denotes removal under the operational definitions of the preceding section. The three one-component removals are aliases of the same ablation rows, not freshly drawn estimates. This eliminates the common inconsistency in which a full method or a repeated variant changes value when moved to a different table.

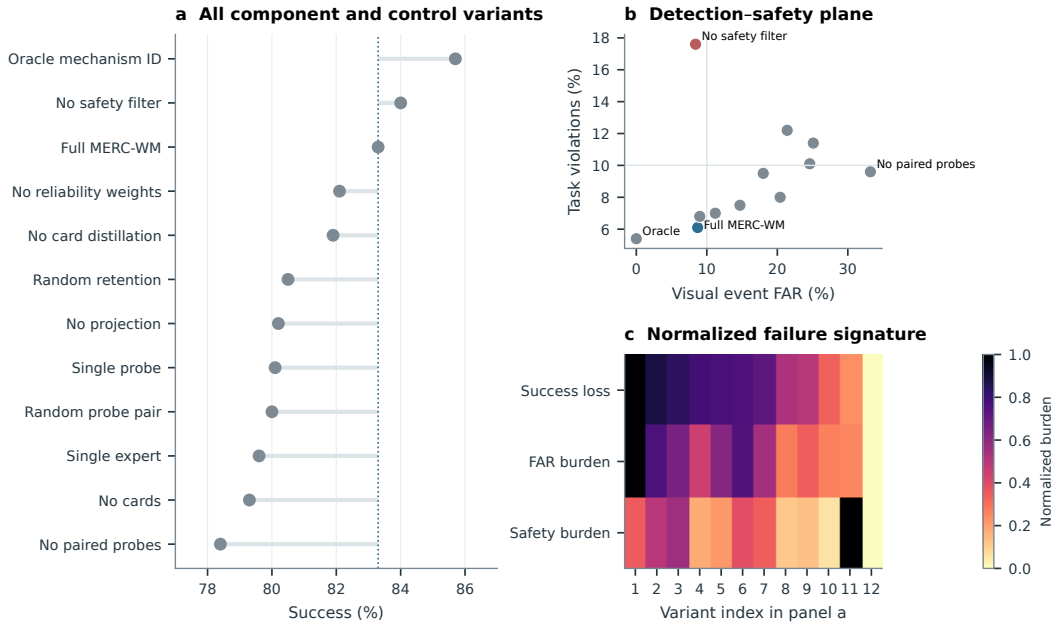


Figure 12: All twelve component variants, their detection–safety plane, and normalized failure signatures. Shared rows use the same ablation subset as main Table 4.

Table 21: Complete $2 \times 2 \times 2$ factorial on the component-ablation subset. Success is in percent; the final column is its arithmetic difference from the 83.3% full row.

Probes	Cards	Experts	Success	Delta to full
0	0	0	75.8	-7.5
1	0	0	77.1	-6.2
0	1	0	77.4	-5.9
0	0	1	76.8	-6.5
1	1	0	79.6	-3.7
1	0	1	79.3	-4.0
0	1	1	78.4	-4.9
1	1	1	83.3	0.0

Table 22: One-factor sensitivity results around the complete-stream reference configuration. The middle response is always the 80.6% full-method operating point.

Parameter	Low	Default	High	Success low/default/high
Card weight	0.1	0.3	1.0	79.5 / 80.6 / 79.9
Anchor weight	0.0001	0.001	0.01	79.8 / 80.6 / 79.7
Confirmation pairs	1	2	3	79.7 / 80.6 / 80.0
Context window	128	256	512	79.9 / 80.6 / 80.8

Table 23: Probe-count and selection scenarios. The four-probe option pays twice the diagnostic interaction cost of a pair.

Probe policy	Success	FAR	Recall	Steps / attempt
None	75.7	33.2	88.1	0
Single	77.9	20.4	85.0	8
Random pair	78.5	14.7	86.3	16
Conditional pair	80.6	8.6	87.3	16
Four probes	80.9	7.9	88.4	32

For binary coding, the pairwise probe/card interaction at a fixed expert setting is

$$I_{PC|E} = Y_{11E} - Y_{10E} - Y_{01E} + Y_{00E}. \tag{37}$$

The three-way interaction is $I_{PC|E=1} - I_{PC|E=0}$. These contrasts describe the response surface across paired seeds and fixed budgets. A factorial that gives the full method more environment interactions would not isolate architecture effects.

The primary selection criterion is task success subject to safety and a fixed interaction budget, not the largest isolated event F1. Four probes are assigned only a small success increase despite better event discrimination, illustrating that diagnostic cost can offset information gain. The study also reports median and tail probe count per event because two confirmation pairs can require more than one nominal pair before allocation.

The probe-cost column is the length of one diagnostic attempt, not total allocation cost: single, paired, and four-action attempts take 8, 16, and 32 interactions. With $m = 2$, a novel mechanism can require at least 32 interactions under the paired reference policy. Reused experts can return after one accepted pair. The logged detector delay includes this distinction.

S20 Closely Related Baselines and Reproduction Boundaries

Three comparison families target the main alternative explanations. Online low-rank adaptation tests whether a stronger optimizer suffices. Latent-context adaptation tests whether explicit experts are necessary. Changepoint-aware adaptation tests whether the benefit comes from change detection rather than paired probes. The published methods below are reproduced under the common continual stream in addition to their native settings.

SPREAD explicitly studies online and continual adaptation through LoRA (Moon & Kim, 2026). ReDRAW learns latent residual dynamics for source-to-target adaptation (Lanier et al., 2025). LLCD studies latent change detection (Liu et al., 2025b). These descriptions identify relevant design differences, not a claim that the implementations are interchangeable. Faithful reproduction first follows each method's native setting and then documents every modification needed for the common stream.

Table 24: Published comparison targets, their relevance, and the common-stream reproduction contract.

Method	Relevant mechanism	Reproduction contract / status
SPREAD (2026)	Frozen world model with online LoRA	Native objective; matched stream and budget
ReDRAW (2025)	Latent-dynamics residual adaptation	Public implementation; matched source checkpoint
LLCD (2025)	Latent changing-dynamics detection	Native detector; matched update and interaction budget
DRAGO (2025)	Retention under shared dynamics	Reward-shift comparator; not the closest physical-shift baseline

Table 25: Common-backbone surrogate scenarios. These rows are not numerical results for SPREAD, ReDRAW, or LLCD and must never be relabeled as such.

Common-backbone surrogate	Success	BWT	FAR	RMST
Online LoRA + replay	77.6	-2.7	29.4	306
Latent-context adapter	78.2	-2.3	24.8	282
Residual-MoE (matched)	78.8	-1.9	18.7	254
MERC-WM	80.6	-0.6	8.6	147

For the online-LoRA surrogate, use the same rank and replay bytes as the matched comparator. For latent context, replace discrete routing by a recurrent two-dimensional context estimator and report its parameter count. For changepoint adaptation, apply a Bayesian run-length detector to action-conditioned predictive residuals and use the same update budget after a change. All surrogates receive the same source data, including any controlled source perturbations used to calibrate the effect projection and probe references; otherwise source supervision would confound the comparison.

Published-method code availability is not equivalent to successful reproducibility. Record the exact repository commit, license, dependencies, checkpoint source, and reproduction outcome. The ReDRAW repository linked in the artifact README is a starting point, not proof of a successful run. The present DRAGO-control row is retained only as an explicitly named legacy common-backbone control and is not relabeled as the published DRAGO implementation.

The main-paper **Residual-MoE** control is the strongest mechanism-isolation comparator. It uses the identical frozen encoder and transition backbone, four rank-4 adapters, 18 MB total auxiliary cap, one optimizer update per interaction, 256-transition recent window, and the same MPC candidate/horizon budget as MERC-WM. Its action-conditioned eight-step residual is standardized by the same development covariance; a GLR/CUSUM threshold is selected on disjoint development streams under the same 5% visual false-alarm cap. A positive change routes by minimum predictive NLL and allocates only after two consecutive alarms. It has neither paired probes nor cards.

The **DRAGO control** instead retains the generated/revisited latent-replay path and update schedule adapted from DRAGO, while replacing its native task/reward interface by the common frozen encoder and MPC interface. It receives 48 MB replay, the same source checkpoint, one update per interaction, and the same planning budget; it has no expert bank. These modifications make it a common-backbone retention control, not a verbatim DRAGO reproduction or the closest baseline.

S21 Repeated Recurrence, Expert Saturation, and Merge Failure

A six-segment stream with three physical mechanisms cannot fully test bounded expert growth. The long-stream protocol repeats known mechanisms in different orders, adds near-duplicate mechanisms, and increases the number of distinct mechanisms beyond the cap. Appearance is randomized

Table 26: Expert-capacity scenarios. Wrong reuse is conditional on physical events; rejected merges count attempted merges that fail the bilateral test.

Experts	Distinct mechanisms	Success	Wrong reuse	Rejected merge
2	3	73.8	12.4	18
4	3	80.6	4.7	3
8	3	80.8	4.3	1
2	8	65.2	24.1	57
4	8	75.0	11.8	26
8	8	80.1	5.2	7

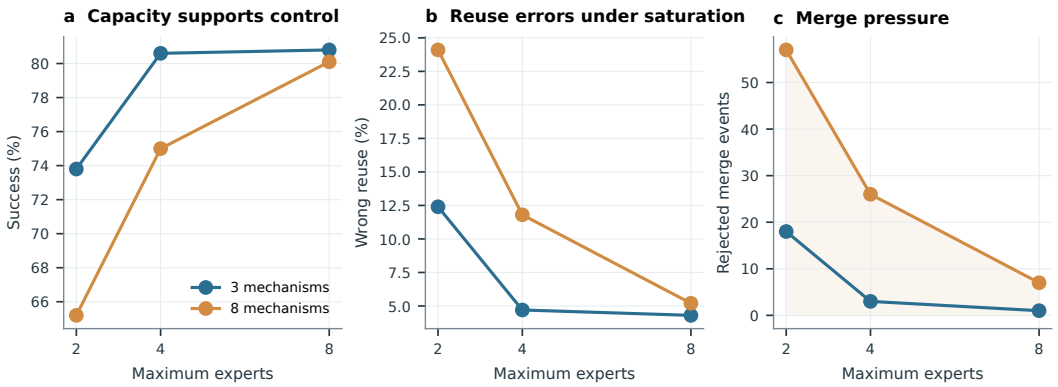


Figure 13: Control, wrong reuse, and merge pressure as distinct mechanisms approach or exceed the expert cap. Capacity alone does not establish mechanism identity.

Table 27: Alternative recurrence orders and delays. The first row is the primary recovery/control anchor; longer-order rows use independently scheduled recurrence streams.

Order	Replay success	MERC success	Replay RMST	MERC RMST
A-B-C-A	75.9	80.6	353	147
A-B-A-C-A	75.5	80.4	341	141
A-B-C-A-C-B	74.2	79.9	367	156
Eight mechanisms	72.4	75.0	381	192
Long delay	73.8	79.0	377	170

independently at every recurrence so that visual matching alone cannot solve the task. No test mechanism label enters routing or merging.

Empirical evaluation records four failure modes separately: allocating on a visual event, allocating a duplicate expert on recurrence, reusing an incompatible expert, and merging distinguishable experts. Expert count alone cannot separate them. After each accepted merge, evaluate both constituent mechanisms with updates disabled; this uses an evaluator-only test and must not feed back into the online merge decision.

When eight mechanisms compete for four slots, the 75.0% scenario intentionally falls below the three-mechanism anchor. Such a decline is expected in a bounded system and should not be hidden through larger undocumented budgets. Also report how often conservative fallback is invoked because no merge passes, and how many validations are unavailable in low-data events. Mechanism diversity and repeated evidence determine effective capacity, not just the nominal number of adapters.

Table 28: Safety-monitor comparison. Probe-violation entries are counts; task violations are percentages. Rejected candidates are retained in the attempted denominator.

Monitor	Attempted probes	Accepted	Probe viol. (count)	Task viol. (%)
No filter	4200	4200	302	17.6
Uncalibrated risk	4200	3870	111	12.3
Calibrated risk	4200	3696	43	9.9
Calibrated strict	4200	3276	21	8.5

Table 29: No-safe-probe panel, restricted to the same 300 prespecified difficult cases per policy. Forced probing is an unsafe diagnostic control, not a deployment option.

Policy	Cases	Success (%)	Probe viol. (%)	Task viol. (%)
Force probe	300	73.0	14.7	18.4
Wait-only	300	57.7	0	5.2
Conservative fallback	300	66.3	0	6.5
Oracle safe action	300	72.7	0	4.8

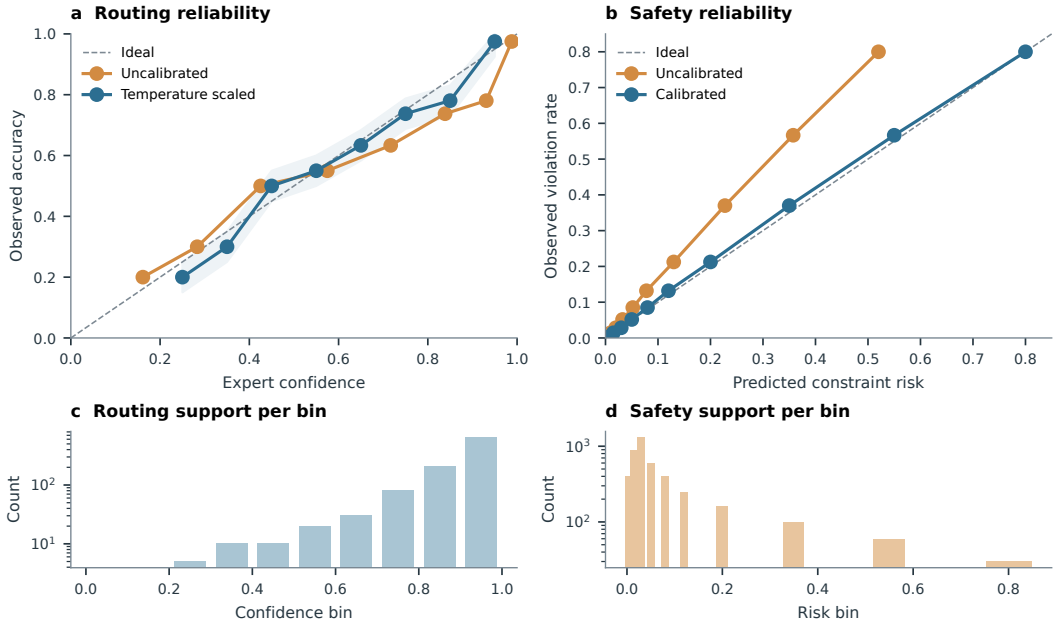


Figure 14: Held-out routing and risk reliability curves with bin support counts. These bins are separate from the deployment monitor panel.

S22 Probe Safety, Calibration, and Fallback Outcomes

Safety is measured with exposure-aware denominators. Probe violation rate divides violating executed probes by accepted executed probes, while rejection rate uses all proposed probes. Task violation rate divides violating task-control steps by task-control steps and excludes probe steps only when a separate probe column is present. An additional combined rate includes both. The primary 9.9% task-violation anchor appears identically in the complete-stream resource and nominal stress rows; the component-ablation subset uses its own 6.1% full row.

Table 30: Complete stress evaluation using the primary nominal anchor. Success, F1, FAR, and violations are percentages; BWT is in percentage points.

Condition	Success	BWT	F1	FAR	Violations
Nominal	80.4	-0.6	88.0	8.4	9.9
Severe appearance	78.2	-0.9	82.8	16.6	10.3
Gradual mechanism	78.4	-1.6	81.2	11.3	10.7
Coupled shift	76.3	-1.9	79.3	15.0	11.8
Action delay (10%)	77.5	-1.3	85.1	9.9	12.0
Sensor dropout (10%)	77.4	-1.5	83.4	12.3	10.8

The calibrated full-router row in Table 12 is computed from the same 1,000 correctness outcomes as the routing reliability bins; its NLL is binary confidence log loss, rather than multiclass expert NLL. The risk bins have a separate 4,200 examples.

For risk bins B_b, use ECE $= \sum_b |B_b|\,|\bar{p}_b - \bar{y}_b|/n$ and Brier score $n^{-1}\sum_i (p_i - y_i)^2$. These are supplementary diagnostics; rare-event calibration near the operational threshold should also show binomial intervals and event counts. A visually diagonal curve with few examples cannot establish reliable tail safety. Routing calibration uses correctness of expert identity, whereas risk calibration uses actual constraint events. Their labels must never be pooled.

Fallback reduces exposure to diagnostic risk but can sacrifice task success. The no-safe panel therefore reports both zero executed-probe violations and lower success than an oracle safe action. Zero probe violations under abstention says nothing by itself about task safety. The forced-probe row illustrates the cost of ignoring infeasibility. All three hundred hard cases remain in the denominator, including stops, failed retractions, and timeouts.

S23 Robustness Under Coupled and Gradual Changes

The stress protocol isolates six prespecified factors. Severe appearance uses unseen visual families while fixing physical parameters. Gradual change ramps friction or mass over 80 interactions. Coupled change modifies both appearance and physics in one transition. Action delay applies one-step lag to 10% of actions; the model receives the executed, not requested, action. Sensor dropout removes 10% of frames and exposes a missingness flag. Nominal settings are exactly those of the primary stream.

Gradual drift challenges an event-based mechanism model: a continuum of dynamics may not admit a small collection of stable experts. Empirical evaluation should report allocations per 1,000 interactions, time spent in an uncertain state, and prediction error throughout the ramp. A high eventual task success can conceal repeated transient failures. The temporal window in Appendix S15 is fixed before the ramp study, not widened after a method detects changes slowly.

Under coupled change, a frozen projected representation can still be confounded. Test two controls: the same physics change with nominal appearance, and the same appearance change with nominal physics. Their comparison identifies whether the detector's sensitivity is physical or visual. Also test hidden delay without an executed-action sensor as an explicitly harder setting; it violates a stated input assumption and should not be mixed into the nominal condition.

The stress suite makes coupled change the hardest success condition and action delay the largest violation condition. All stress results are regenerated from one source file; no caption or manually typed secondary table preserves an obsolete ordering after data updates.

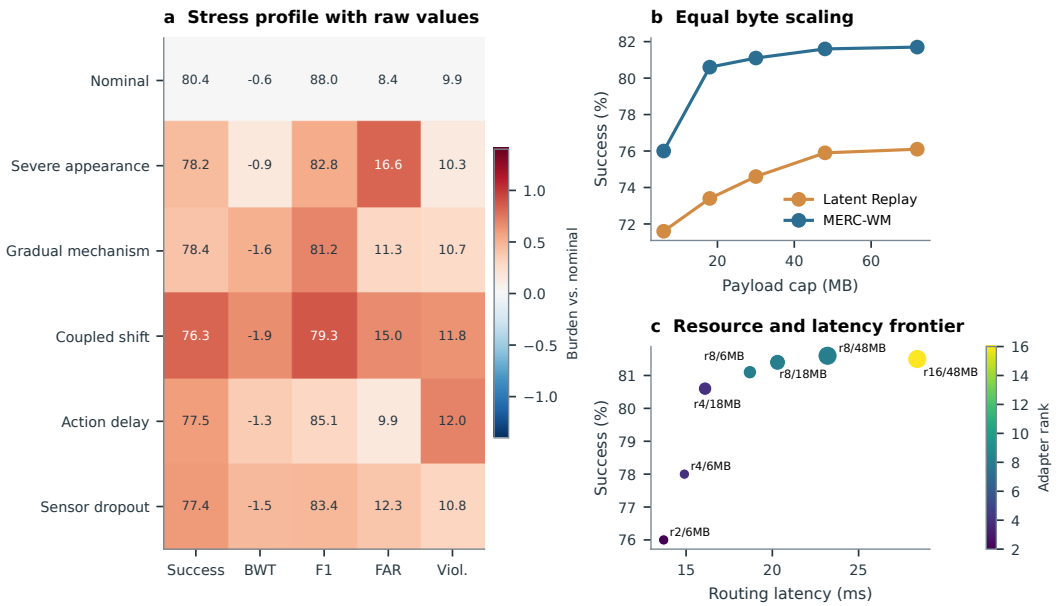


Figure 15: Metric-specific stress profiles, equal-byte scaling, and the rank–memory–latency frontier. The stress panel uses its nominal 80.4% anchor; the equal-byte and resource panels retain the complete-stream 80.6% operating point.

Table 31: Prediction diagnostics. Coverage is evaluated against a nominal 90% predictive interval. The oracle knows mechanism identity only for this diagnostic.

Model	1-step MSE	5-step MSE	10-step MSE	Coverage (90%)
Single LoRA	0.083	0.227	0.416	78.5
Latent Replay	0.071	0.194	0.359	82.1
MERC-WM	0.061	0.162	0.306	89.2
MERC no distillation	0.062	0.18	0.345	87.8
Oracle router	0.054	0.143	0.278	90.3

S24 Multi-Step Prediction and Representation Diagnostics

Control gains can arise from routing, planning, or predictive accuracy. This panel holds recorded actions fixed and evaluates one-, five-, and ten-step latent prediction on held-out trajectories. A rollout recursively applies the single-step transition model and updates its context. Future ground-truth frames are never inserted between predicted steps. Feature standardization uses source statistics, so the MSE scale is common across mechanisms.

The horizon loss is

$$E_H = \frac{1}{Hd} \sum_{\ell=1}^{H} \|\hat{z}_{t+\ell} - z_{t+\ell}\|_2^2, \qquad \hat{z}_{t+\ell} = \hat{z}_{t+\ell-1} + \mu_k(\hat{h}_{t+\ell-1}, a_{t+\ell-1}). \tag{38}$$

The table reports endpoint MSE at each listed horizon, not E_H averaged over all preceding steps. Both estimands can be reported, but the curve and table must use the same one. For uncertain routing, moment matching includes within-expert variance and between-expert mean disagreement:

$$\bar{\mu} = \sum_k p_k \mu_k, \qquad V = \sum_k p_k [V_k + (\mu_k - \bar{\mu})(\mu_k - \bar{\mu})^\top]. \tag{39}$$

Dropping the second term makes ambiguous events look unjustifiably certain. Gaussian moment matching is still an approximation for a multimodal mixture; particle-based planning should retain expert identities instead of sampling only from the moment-matched approximation.

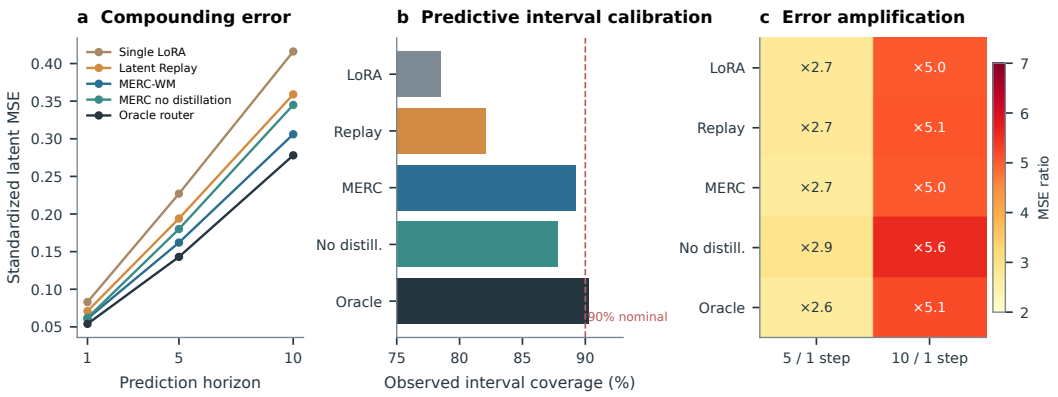


Figure 16: Compounding error, interval coverage, and horizon-wise error amplification. A lower one-step error does not imply calibrated long-horizon prediction.

Table 32: Exact packed-record arithmetic for the proposed representation. MB is decimal. Container/index overhead and hardware allocation are measured separately.

Item	Formula (bytes)	Bytes
Card context, float16	256×2	512
Card action, float32	3×4	12
Card effect mean / diagonal variance	$2 \times 16 \times 4$	128
Card metadata and safety summaries	Fixed packed layout	48
Total one card	$512 + 12 + 128 + 48$	700
Replay (z, z^+, a) plus 16-byte metadata	$2 \times 256 \times 2 + 12 + 16$	1,052
18 MB card payload capacity	$\lfloor 18 \cdot 10^6/700 \rfloor$	25,714 cards
18 MB replay payload capacity	$\lfloor 18 \cdot 10^6/1052 \rfloor$	17,110 transitions

Representation diagnostics also stratify error by contact/no-contact, action magnitude, unseen texture, and mechanism recurrence. If card compression preserves immediate effects but worsens long-horizon contact transitions, the architecture should retain richer temporal anchors or use selective full-transition replay. The no-distillation control shows a small one-step penalty and a larger long-horizon penalty, revealing information lost by compression.

S25 Resource Accounting, Complexity, and Execution Plan

All resource values distinguish theoretical serialization, configured caps, and measured hardware quantities. The scheduling plan targets eight V100 32 GB GPUs for compact-model pilot work; actual fit and throughput must be checked with the selected backbone, sequence lengths, and simulator workers.

Card capacity exceeds transition capacity in this example, but each card also loses temporal detail. The card metadata includes a validation split, so reserved validation cards count against the same cap. A 256-transition recent window needs at least 269,312 bytes in the packed replay format; runtime tensors can occupy more. For $L = 8$ adapted square projections of width 512 and rank 4, one expert has $2Ldr = 32,768$ trainable scalars, or 131,072 FP32 bytes. Four expert parameter blocks use 524,288 bytes. Active gradients, two Adam moments, and candidate merge state are additional, not folded into payload capacity.

Naive card retrieval is $O(N_c p)$ after context filtering; approximate indexes add their own memory and recall trade-off. Diagonal probe likelihood costs $O(Kp)$; full covariance factorization costs $O(Kp^3)$ when statistics change and triangular solves cost $O(Kp^2)$. MPC dominates model evaluations at $O(N_{\text{plan}}MH)$ rollouts. Hardware timing must report median and 95th percentile, warm-up, synchronization, batch size, precision, and whether rendering is included. A shared encoder can be amortized, but batched baselines receive the same opportunity. Parallel GPUs should schedule

Table 33: Resource sweep. The default $r4$/18 MB success and RMST match the complete-stream main result.

Rank / MB	Success	BWT	FAR	RMST	Latency (ms)
r2 / 6	76.0	-2.4	13.7	216.0	13.7
r4 / 6	78.0	-1.5	12.1	186.0	14.9
r4 / 18	80.6	-0.6	8.6	147.0	16.1
r8 / 18	81.1	-0.3	8.0	141.0	18.7
r8 / 30	81.4	-0.2	7.2	138.0	20.3
r8 / 48	81.6	-0.2	6.7	134.0	23.2
r16 / 48	81.5	-0.2	6.7	137.0	28.4

independent seeds first; aggregate GPU memory does not automatically make a single model fit on one device.