WHAT SHOULD SURVIVE A FAILED PLAN? COUNTERFACTUAL REUSE WITH WORLD MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

World models can reject a plan before execution, but a failed prediction does not say which earlier decisions the next search should keep. We formulate this problem as counterfactual plan reuse: nominate a single action edit, decide whether to retain the prefix it defines, and search for a completion under a shared prediction budget. ROOTCAUSE scores every candidate edit by its effect on the same diagnosed failure, so a repair can be placed before the first visible symptom, then decides with paired world-model predictions whether the resulting branch deserves the search it would concentrate. Its central quantity is the value of one more check against the candidates that check displaces, so confirmation depth follows the budget instead of being fixed in advance. On four manipulation families with a frozen World Action Planner, ROOTCAUSE raises matched-budget success from 80.7% for the strongest baseline to 84.6%, with the largest gains on long-horizon tasks and a margin over fixed-depth confirmation that grows from 0.3 to 4.6 points as the budget quadruples. Held-out audits of action-response fidelity and calibration transfer locate the regime in which retaining actions pays and where it stops.

1 INTRODUCTION

World models let a robot test a plan in imagination before paying for it in the world (Finn & Levine, 2017; Hansen et al., 2024; Zhou et al., 2025), so a predicted failure lets the planner revise a command before its physical cost is incurred. But rejection alone leaves an important decision unresolved: how much of the rejected plan should survive? Reopening every action throws away the structure a nominal plan already encodes; keeping the wrong prefix removes the very change needed for success.

Consider a robot that predicts a cube will land beside a stack. An off-center grasp and a misplaced release produce the same visible outcome. If the grasp is stable, changing the placement preserves useful acquisition and transport commands; if the grasp is unstable, the repair has to happen before the cube leaves the table. The visible symptom says what went wrong, not which earlier decisions are still worth keeping. The distinction matters because a planner with a fixed prediction budget can afford only so many complete candidates: every action it keeps removes a free variable from the search and concentrates the candidates on what is left, and every action it keeps wrongly removes the solutions.

Existing methods provide the ingredients. Action-conditioned video models predict the consequences of robot commands (Zhang & Du, 2026); temporal backtracking reuses verified trajectory prefixes (Jun et al., 2026); geometric evaluation improves the choice among generated candidates (Zhao et al., 2026). We ask a complementary question: which *action constraint* should the next search inherit? A single replacement at an earlier site defines such a constraint: the actions before it and the edit itself stay fixed, and every later action is free. The object under evaluation is therefore a branch, a family of plans whose quality depends on the actions it preserves and on the continuations it leaves open.

The difficulty is that evaluating this branch consumes the same computation as searching it. A promising edit may be worth using immediately; an ambiguous edit that locks most of a plan may justify more evidence; checking every proposal equally spends search on obvious decisions and can still accept restrictive mistakes. Evidence pays off only when the improved branch choice compensates for the candidates it displaces (Figure 1).

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.

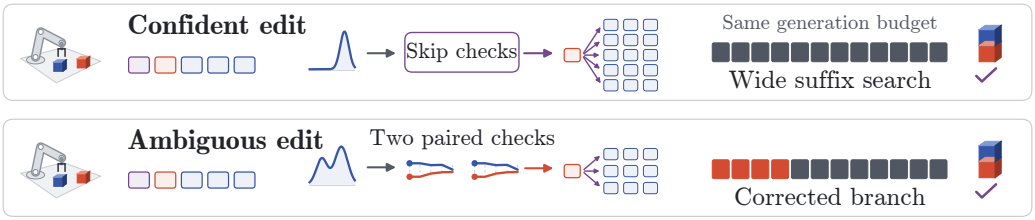

Figure 1: **The same budget can buy confidence or continuations.** A clear repair reserves computation for suffix search; an ambiguous, restrictive prefix can benefit from checking first. The value of the remaining search determines how much confirmation to buy.

We introduce ROOTCAUSE, a test-time planner that decides both where to reopen a rejected plan and whether to commit the resulting prefix. It is named for what it keeps: rather than restarting at the boundary where a failure becomes visible, it replays candidate edits to the diagnosed failure and retains the earliest edit that removes it, even when that edit precedes the first visible symptom. Paired rollouts then provide evidence about the edit, but the evidence is valued by the success of the search it displaces, not by confidence alone (Glasserman & Yao, 1992). Under a shared budget, ROOTCAUSE chooses among immediate reuse, another paired check, and full reopening, which makes prefix commitment an explicit budgeted planning decision (Russell & Wefald, 1991).

Our contributions are threefold:

- **Failure-conditioned site selection.** We formulate a single edit as a retained-prefix constraint and score every candidate edit by its effect on the same diagnosed failure. On early faults this locates the injected action in 72.1% of cases, against 38.5% for scoring each edit at its own endpoint, and it is the largest single source of the planner's gain.
- **Calibrated, budget-aware commitment.** A small belief-state controller values a retained prefix by the calibrated success of the search it leaves and buys another check only when the information is worth the candidates it displaces; its margin over fixed-depth confirmation grows with the budget, from 0.3 to 4.6 points.
- **Matched-cost evidence on four manipulation families.** Under identical prediction budgets and a frozen World Action Planner, ROOTCAUSE reaches 84.6% macro success at $b = 32$ against 80.7% for the strongest baseline, with gains concentrated on long-horizon tasks; audits of action-response fidelity and calibration transfer show where the gains stop.

2 Deciding What to Keep

A failed plan leaves three options: keep an edited prefix, reopen the whole sequence, or buy evidence before deciding. ROOTCAUSE compares them by a single quantity, the probability that the search remaining after the decision returns a successful plan. The three subsections below nominate the edit, value the branch it defines, and decide when one more check is worth its cost (Figure 2).

2.1 Nominate an Edit by Its Effect on the Diagnosed Failure

Planning interface. Given an observed history $\mathcal{H}_0$, a goal, and a nominal plan $u_{1:H}$, each primitive specifies a pose target and a legal gripper schedule over one control interval. The frozen world model predicts one native block per primitive and joins blocks autoregressively, so a complete prediction costs $L = H$ blocks and a prediction through primitive s costs $L_s = s$. The budget B counts blocks, and every planner spends the same B per decision (Appendix B gives the frame layout).

Locate a symptom, without assuming its cause. Two independently seeded nominal predictions identify the earliest phase boundary s with valid geometry and residual $d_s \geq 0.5$ in both predictions. The residual $d_t \in [0, 1]$ combines acquisition centering, object–gripper displacement, clearance, and support error according to the active phase, and the path score $D_s = \max_{t \leq s} d_t$ also penalizes an edit that moves a violation earlier. Masks and object tracks come from ordinary RGB observations; simulator state is reserved for evaluation. If the two predictions disagree on a boundary, the planner reopens the root; either way, diagnosis costs $2L$.

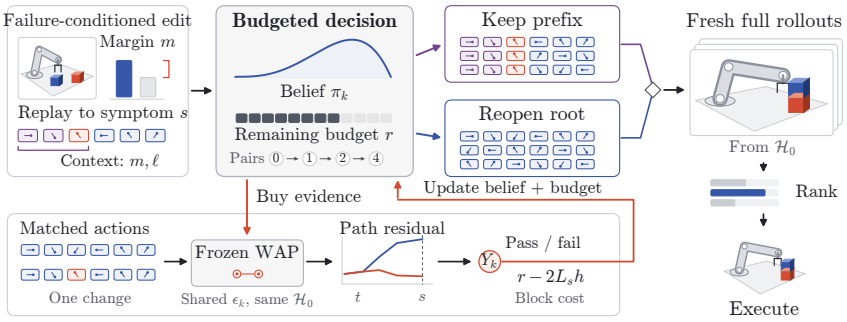

Figure 2: **An edit proposes a branch; evidence decides its use.** Nomination propagates each edit to the diagnosed endpoint; the controller compares prefix reuse, root search, and another paired check, which shares history and noise between the nominal and edited predictions. Purple commands and the red edit stay fixed only in the retained branch; blue commands can change; fresh complete predictions rank the chosen branch.

Score the same endpoint for every edit. An edit at $i \le s$ is useful only if it changes the diagnosed failure, so every proposal is evaluated through s rather than at its own endpoint i. A cheap geometric replay inserts a candidate pose or gripper-timing change, propagates the resulting object–gripper transform through the nominal contact sequence, and recomputes the path residual with all later commands fixed: during attachment the estimated object follows the gripper, and after release it is tested against the nominal support geometry. The replay is a contact-conditioned proposal, not a substitute for predicting contact dynamics, and ambiguous attachments or missing geometry receive no positive score. For each eligible site, a fixed low-discrepancy proposal bank covers bounded continuous translations, rotations, and phase-legal gripper timing under the same workspace and controller limits as the final search. Write $\widetilde{D}_s(i, v)$ for the replayed path residual under replacement v. The score, nomination margin, and retained fraction are

$$a_s(i, v) = D_s^{\text{nom}} - \widetilde{D}_s(i, v), \qquad m = a_{(1)} - \max\{0, a_{(2)}\}, \qquad \ell = i^*/H, \tag{1}$$

where (i^*, v^*) is the best positive-scoring edit and $a_{(2)}$ the runner-up score. An early grasp change can improve the residual at a later transport boundary because it changes the carried object's relative pose, whereas an unrelated local improvement receives no credit unless it changes the path to s; this is what lets nomination reach a site before the symptom. Nonpositive nominations mean root search.

Turn the edit into a search constraint. The retained branch fixes the original actions before the edit and the replacement itself, and frees every subsequent primitive:

$$\mathcal{B}_{i,v} = \{(u_1, \dots, u_{i-1}, v, \widetilde{u}_{i+1:H}) : \widetilde{u}_{i+1:H} \in \mathcal{U}_{i+1:H}\}. \tag{2}$$

Root search may change any action; retaining more actions concentrates the search but removes ways to recover from an incorrect edit, and ℓ measures this restriction, not a failure probability. Nomination proposes a branch without committing to it.

2.2 Translate paired evidence into the value of a search branch

Check consequences beyond the geometric replay. A paired check predicts the nominal and edited sequences through s with the frozen world model. Both start from $\mathcal{H}_0$ and share the per-block random tape, so only the nominated action differs; independent pairs, diagnosis, and final search use separate tapes. Shared randomness reduces the variance of the comparison when paired outcomes covary positively (Glasserman & Yao, 1992; Ng & Jordan, 2000; Ma et al., 2026). Let V_k indicate valid geometry in both members of pair k. A passing comparison reduces the path residual by at least $\delta = 0.25$ and brings the edited path below $\epsilon = 0.25$:

$$\begin{aligned} Y_k &= \mathbf{1}\{V_k = 1,\ D_s^{\text{nom}} - D_s^{\text{edit}} \ge \delta,\ D_s^{\text{edit}} \le \epsilon\}, \\ \pi_k &= \text{Beta}\left(a_0 + \sum_{j=1}^{k} Y_j,\ b_0 + k - \sum_{j=1}^{k} Y_j\right), \end{aligned} \tag{3}$$

Algorithm 1 One budgeted plan-reuse decision

Require: History $\mathcal{H}_0$, nominal plan $u_{1:H}$, goal, block budget B
1: Reserve $4L$; if diagnosis is unaffordable, **return** root search
2: Charge $2L$; diagnose s and nominate (i^*, v^*) by Eq. 1
3: If no valid nomination exists, **return** root search with the remainder
4: Fix (m, ℓ, s); initialize π_0, $k = 0$, and $r = B - 2L$
5: **while** another check maximizes Eq. 6 **do**
6: Generate the next h pairs; update π, $k \leftarrow k + h$, $r \leftarrow r - 2L_s h$
7: **end while**
8: Choose keep or root; rank $\lfloor r/L \rfloor$ fresh full predictions and execute

with prior $(a_0, b_0) = (1, 1)$. The posterior describes the repeatable pass probability θ of this model–verifier comparison; noisy checks can reject useful prefixes and convincing predictions can fail in execution, so neither θ nor a pass count is the planning objective.

Separate prefix usefulness from search capacity. The default value model, which we call *Factored*, has two parts. A logistic model $q(\theta, m, \ell)$ estimates whether the nominated prefix admits useful continuations; its development label $G = 1$ requires at least seven successful executions from an independent bank of 32 suffixes. The label is acquired offline, and neither injected fault locations nor reference executions are available to the test-time controller. Let $S_{\text{good}}(n)$ and $S_{\text{bad}}(n)$ estimate selected-plan success after searching n retained-branch candidates, conditional on $G = 1$ and $G = 0$, and let $S_{\text{root}}(n)$ estimate success after root search. The branch values combine these capacity curves with the current evidence:

$$\begin{aligned} v_K(n, \theta, m, \ell) &= q(\theta, m, \ell) S_{\text{good}}(n) + [1 - q(\theta, m, \ell)] S_{\text{bad}}(n), \\ v_R(n, \theta, m, \ell) &= S_{\text{root}}(n). \end{aligned} \tag{4}$$

Evidence changes the estimated usefulness of the retained prefix, the candidate count changes search capacity, root value does not depend on the paired posterior, and all three curves include the errors of the common final ranker. Each leave-one-family-out fold fits q and the capacity curves on development origins from the other three families and freezes them for the held-out family; curve targets are terminal outcomes of search and ranking at each reachable candidate count, from reference, evidence, and search streams independent of one another (Appendix A, which also specifies the seven-feature Direct alternative). For remaining budget r, let $n(r) = \lfloor r/L \rfloor$. The stopping value of branch b is

$$U_b(\pi_k, r) = \mathbb{E}_{\theta \sim \pi_k}[v_b(n(r), \theta, m, \ell)], \qquad b \in \{K, R\}. \tag{5}$$

The posterior average integrates uncertainty in the paired comparison without identifying check-pass probability with task success; calibration is judged on the selected plan, not on accepting an edit.

2.3 Spend on a check only when it improves the remaining decision

After $k \in \{0, 1, 2\}$ completed pairs the next batch has size $h = 1, 1, 2$, and four pairs is the cap. The J passes of a batch follow the Beta–binomial predictive distribution and produce the posterior $\pi_{k+h,J}$. Backward induction compares the two stopping actions with the value of checking:

$$W_k(\pi_k, r) = \max\{U_K(\pi_k, r),\ U_R(\pi_k, r),\ \mathbb{E}_J W_{k+h}(\pi_{k+h,J}, r - 2L_s h)\}. \tag{6}$$

Checking is available only below the cap and while at least four full candidates remain affordable; ties favor stopping, then root search. Zero checks is an ordinary action: a strong nomination need not pay for confirmation, and a weak one can be discarded at once. The controller maximizes calibrated predicted success, and its calibration matters most at small budgets, where one unnecessary pair displaces much of the search.

Why a check must repay the search it displaces. The policy is an instance of value-of-computation reasoning (Russell & Wefald, 1991; Hay et al., 2012), and one identity makes its trade-off explicit (Figure 1). Let $M(\pi, r) = \max_b U_b(\pi, r)$ be the stopping value and $r' = r - 2L_s h$ the budget after the next batch. Adding and subtracting the stopping value at the reduced budget gives

$$\begin{aligned} \mathbb{E}W_{k+h}(\pi', r') - M(\pi, r) = {} & \underbrace{\mathbb{E}M(\pi', r') - M(\pi, r')}_{\text{information gain}} - \underbrace{M(\pi, r) - M(\pi, r')}_{\text{displaced search}} \\ & + \underbrace{\mathbb{E}[W_{k+h}(\pi', r') - M(\pi', r')]}_{\text{future checking option}}. \end{aligned} \tag{7}$$

Posterior averaging makes the first term nonnegative and the option to stop makes the last term nonnegative; the middle term is nonnegative whenever selected-plan value increases with the candidate count (Appendix A.1). At the final batch the option vanishes, and information must repay the search it displaces. The identity separates information from confidence: a batch can sharpen the posterior yet be worthless if the same branch stays preferable across its likely outcomes, whereas near a branch-value tie one more batch can change where the whole remaining search is spent. Edit depth and check cost are decoupled: i^* fixes which actions survive, s the horizon charged per pair.

Search, execution, and accounting. After stopping, the planner samples the chosen branch, includes its anchor, and predicts every complete candidate afresh from $\mathcal{H}_0$; goal satisfaction, maximum phase residual, and terminal geometric error determine the common ranking, and the selected plan is executed once with the baseline controller. With k checks and N final candidates the ledger enforces $2L + 2kL_s + NL \leq B$; nomination reuses diagnosis geometry without a world-model call, and a generation error charges every started block and sends the remainder to root search.

3 Experiments

The experiments ask where the planner's gain comes from, how much of it survives when evidence must be paid for, and whether its value model transfers to an unseen task family; a fidelity audit of the world model's action response frames all three.

3.1 Experimental Setup

Origins, families, and cohorts. An *origin* is one evaluation instance, a physical simulator reset with its observation history, goal, nominal plan, and failure source, shared by every planner and seed; a *block* is one native prediction step, and every planner receives the same budget of $B = bL$ blocks per decision. The four task families are StackCube and PickPlaceCan in robosuite, and LIBERO-Object and LIBERO-Long (Zhu et al., 2020; Liu et al., 2023a); they vary support geometry, transport clearance, object identity, and sequential structure, with horizons $H = L = 6, 8, 12, 18$ blocks. The primary cohort holds 100 single-fault origins per family and three planner seeds; LIBERO origins span ten tasks with ten initializations each. Nominal plans come from waypoint controllers (robosuite) or retargeted demonstrations (LIBERO), and injected errors affect acquisition, transport, or release, before or at the first visible symptom; two further cohorts, untargeted retargeting failures and coupled two-fault errors, appear with their results. Every reported success rate is the outcome of executing the selected plan in the simulator from the origin's reset with the shared baseline controller; imagined outcomes are never counted.

Backbone and planners. The released WAP checkpoints are frozen (Zhang & Du, 2026); ROOTCAUSE adds no backbone training, and its value model is fitted only on development origins disjoint from every evaluation reset. All planners share the same edit bounds, nominal warm start, RGB perception, and final ranker. Full-root sampling, continuous CEM (Rubinstein, 1999; Chua et al., 2018), and MPPI (Williams et al., 2017) optimize complete sequences. WAP-style best-of-N, a TBS-style verified restart (Jun et al., 2026), Geo-gated selection (Zhao et al., 2026), and a Twin-Rollouts-style paired branch search (Ma et al., 2026) are external baselines. ROOTCAUSE-NoChk commits to the nominated branch without evidence; ROOTCAUSE-F1, -F2, and -F4 are fixed-depth controls that keep the branch while pairs pass, reopen at the first failed pair, and stop after at most one, two, or four pairs; ROOTCAUSE-Direct swaps the value model (Appendix A); ROOTCAUSE+Geo adds geometric candidate filtering. Budgets are $b \in \{16, 32, 64\}$. Repair methods pay for diagnosis and both check arms, every complete candidate pays for its full prediction, retained prefix included, and at least four final candidates remain affordable after checking.

Calibration and statistics. The main comparison uses leave-one-family-out (LOFO) calibration: the value model is fitted on development origins from three families and frozen for the fourth. A nested same-family study instead fits on 5 to 80 development origins of the evaluation family with the evaluation resets fixed. Success is averaged over seeds within origin, then equally over origins and families; 2,000 family-stratified origin bootstraps give 95% intervals and paired differences, computed before rounding.

Table 1: **Held-out counterfactual fidelity audit.** Validity is the fraction of attempted predictions with a usable response; each slashed cell gives agreement among eligible cases / eligibility coverage among all attempts (%) for displacement direction, a later failure event, and the ordering of edits. Matched-reset simulator execution supplies the reference response and is never available to the planner.

Edit scale	Validity	Direction / cov.	Delayed failure / cov.	Edit order / cov.
Small (0.25)	94.9	86.9 / 86.7	82.8 / 52.2	90.0 / 91.1
Medium (0.50)	94.0	85.2 / 85.2	81.6 / 50.2	88.1 / 90.2
Large (1.00)	92.0	81.6 / 82.9	77.7 / 49.5	85.8 / 88.5

Table 2: **Complete-planner comparison at $b = 32$.** Terminal success (%) with held-out-family calibration. Methods share origins, seeds, and block budget. Rows prefixed ROOTCAUSE are variants of our planner; all other rows are baselines. Bold numbers are column-wise maxima (ties retained); macro intervals cluster by origin.

Planner	Stack	Can	Object	Long	Macro [95% CI]
Full-root	92.3	68.0	54.7	39.0	63.5 [61.0, 66.1]
Continuous CEM	93.7	77.0	68.3	53.0	73.0 [70.7, 75.5]
MPPI	94.3	77.7	71.0	56.0	74.8 [72.3, 77.1]
WAP-style BoN	95.0	75.7	68.0	51.0	72.4 [69.9, 75.0]
TBS-style adapter	94.7	78.7	71.7	57.0	75.5 [73.2, 77.9]
Paired branch search	95.3	81.3	73.7	65.3	78.9 [76.7, 81.2]
Geo-gated adapter	95.7	83.3	78.7	65.0	80.7 [78.5, 82.8]
ROOTCAUSE-NoChk	**96.3**	74.7	70.3	57.7	74.8 [72.3, 77.0]
ROOTCAUSE-F1	95.7	85.0	74.3	60.0	78.8 [76.7, 80.9]
ROOTCAUSE-F2	96.0	82.7	78.0	63.7	80.1 [77.9, 82.2]
ROOTCAUSE-F4	94.7	80.7	77.0	65.3	79.4 [77.2, 81.6]
ROOTCAUSE	**96.3**	**88.0**	80.0	74.0	84.6 [82.5, 86.6]
ROOTCAUSE+Geo	95.3	87.3	**84.7**	**75.7**	**85.8 [83.8, 87.6]**

World-model counterfactual fidelity. Before comparing planners we test the premise that an edited action changes frozen WAP the way it changes the simulator, by comparing nominal and edited predictions with matched-reset execution under fixed later commands (Table 1; the horizon-by-scale grid is in Appendix B). As the edit scale grows from a quarter to the full legal range, validity falls from 94.9% to 92.0%, direction agreement from 86.9% to 81.6%, and edit-order agreement from 90.0% to 85.8%. Delayed-failure recall, the quantity an early repair relies on, is the weakest at 82.8% to 77.7%, with about half of all attempts eligible. The audit sets the reuse envelope, small edits and moderate horizons, which is where the planner spends its checks.

3.2 Main Results

Retaining structure beats optimizing complete sequences. Table 2 compares thirteen planners at $b = 32$. ROOTCAUSE reaches 84.6% macro success, 3.9 points $[2.0, 5.8]$ above the strongest baseline, Geo-gated selection at 80.7%, 9.8 points above MPPI, and 11.6 above continuous CEM. Continuous optimization improves on unstructured full-root search, but retaining the right actions concentrates the same candidates on fewer free variables. The gain lives where continuations are scarce: ROOTCAUSE and its fixed-depth control differ by 0.3 points on Stack and by 10.3 on Long, and on Long every ROOTCAUSE variant, even the one that never checks, beats every complete-sequence optimizer. On Stack every planner exceeds 92% and the no-check variant ties the full controller at 96.3%, so evidence has nothing to buy; on Can and Object the margin over the strongest baseline is 4.7 and 1.3 points (6.0 with gating), and on Long it reaches 8.7. Simple tasks admit many successful continuations; when a restrictive mistake removes the few that can complete a long sequence, keeping the right actions is what remains.

The gain comes from choosing the site, not from ranking edits within it. For early faults, failure-conditioned replay returns the injected action in 72.1% of cases against 38.5% for scoring each edit at its own endpoint (Figure 4); the symptom boundary cannot reach an earlier site by construction and finds a useful branch in 49.2% of these cases against 68.6%. When cause and

Table 3: **Budget and calibration-data requirements.** Macro terminal success (%). LOFO rows exclude the evaluation family from development; same-family rows give the development origins per family, with evaluation resets unchanged. The shaded column is our controller; bold marks the best method in each row. Intervals and paired contrasts are in Appendix B.

Calibration	b	ROOTCAUSE-NoChk	ROOTCAUSE-F2	Geo-gated	ROOTCAUSE
LOFO	16	72.8	73.8	71.6	**74.1**
LOFO	32	74.8	80.1	80.7	**84.6**
LOFO	64	79.7	84.2	86.5	**88.8**
Same family: 5	32	74.8	80.1	**80.7**	76.5
Same family: 10	32	74.8	80.1	**80.7**	78.1
Same family: 20	32	74.8	80.1	**80.7**	80.2
Same family: 40	32	74.8	80.1	80.7	**82.5**
Same family: 80	32	74.8	80.1	80.7	**84.9**

symptom coincide the symptom boundary is the sharper locator, 96.1% against 80.3%, so searching earlier trades some local precision for the ability to recover delayed causes. Within a fixed site, however, the ranking rule hardly matters: failure-conditioned, endpoint-local, and random within-site ranking reach 69.8%, 71.2%, and 70.6% success at a matched depth (Appendix B). Yet replacing failure-conditioned nomination by endpoint-local scoring in the full planner costs 6.2 points $[4.2, 8.3]$, the largest effect in any ablation. What the diagnosed-endpoint score buys is the site: an edit is credited only when it changes the path to the symptom, which lets the planner restart earlier.

Commit on calibrated value, not on pass counts. Committing to the nominated branch without evidence yields 74.8%, no better than MPPI: a good site is not yet a good branch. One, two, or four pairs under a pass-count rule raise this to 78.8%, 80.1%, and 79.4%. ROOTCAUSE, with the same pairs, reaches 84.6%, 4.5 points $[2.7, 6.4]$ above ROOTCAUSE-F2 and 5.2 above ROOTCAUSE-F4. The difference is the decision rule: the fixed-depth controls keep while checks pass and reopen when one fails, whereas ROOTCAUSE weighs the posterior against the calibrated success of both branches at the candidate count that will actually remain. Capping ROOTCAUSE at two pairs changes success by only -1.2 points $[-2.9, 0.4]$ (Appendix B), so at $b = 32$ most of the benefit comes from how the evidence is used rather than from how much is collected.

Evidence must earn back its search cost. The budget sweep in Table 3 and Figure 3 shows when collecting more evidence does pay. The advantage over ROOTCAUSE-NoChk grows from 1.2 points $[-1.0, 3.4]$ at $b = 16$ to 9.8 at $b = 32$ and 9.1 at $b = 64$; the advantage over ROOTCAUSE-F2 grows from 0.3 points $[-1.9, 2.6]$ to 4.5 and 4.6 $[2.6, 6.6]$. The controller's stopping depth tracks the same quantity: one pair is the most common depth at $b = 16$ (59.8% of decisions), the three depths split 18.2%, 36.6%, and 45.2% at $b = 32$, and four pairs dominate at $b = 64$ (65.4%), as the opportunity cost in Eq. 7 predicts. The smallest budget also shows the controller's limit. Evidence barely pays there, yet the controller chooses zero checks in only 2.0% of decisions, and the gated no-check variant, ROOTCAUSE-NoChk+Geo, is the strongest configuration at $b = 16$, 2.3 points $[0.2, 4.5]$ above ROOTCAUSE+Geo (Appendix B): the regime Section 2.3 identifies as the most sensitive to calibration error, where one pair removes roughly one candidate in ten from a small search.

Calibration transfers across families; what it needs is origins. LOFO calibration, with no development data from the evaluation family, reaches 84.6%; same-family calibration climbs from 76.5% with five development origins to 84.9% with eighty, passing the fixed-depth control at twenty origins and the strongest baseline at forty (Table 3). The value model is therefore not family-specific, three hundred out-of-family origins being worth about eighty in-family ones, but it needs a few dozen origins before it beats a calibration-free rule, the price of deciding by calibrated value. That price is paid offline: fitting on eighty origins per family costs about 54,000 full-rollout equivalents, sixteen times the cost of fitting on five (Appendix B).

3.3 Analysis

Candidate filtering and evidence overlap. Geometric gating raises ROOTCAUSE-NoChk from 74.8% to 78.8% but ROOTCAUSE only from 84.6% to 85.8%, a gain of 1.2 points $[-0.7, 2.9]$ with

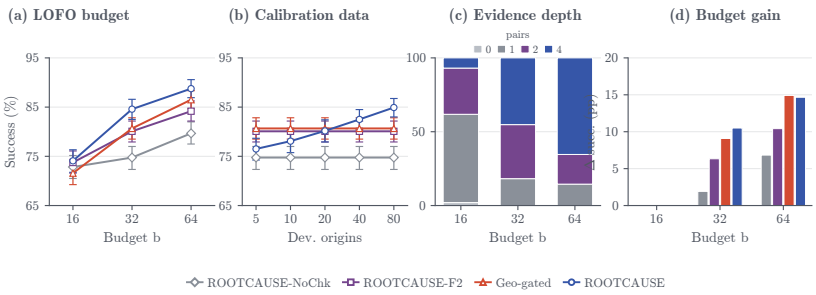

Figure 3: **Budget transfer and calibration have distinct signatures.** (a) LOFO success as the prediction budget grows; (b) nested same-family calibration with fixed test origins; (c) the ROOTCAUSE stopping depth selected at each budget; (d) the success gain relative to $b = 16$. Intervals cluster the held-out origins; panels (a,b) use a 65–95% axis.

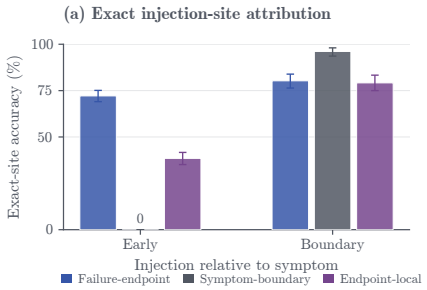

Figure 4: **Localizing a repair.** Exact injection-site accuracy under three nomination rules.

Table 4: **Held-out-family calibration.** Out-of-fold predictions cover 2,400 branch records from 400 held-out origins; scores pool keep and root outcomes.

Model	Brier ↓	Skill ↑	AUC ↑	ECE ↓
Train prevalence	0.222	0.000	0.517	0.031
Factored	0.193	0.131	0.652	0.026
Direct	0.192	0.134	0.687	0.039

a negative interaction of -2.8 points $[-5.7, -0.1]$ on matched origins and budget (Appendix B): both mechanisms remove continuations with poor geometry, so the second finds fewer to remove. The combination still adds 1.9 points $[0.3, 3.6]$ at $b = 64$.

Natural and coupled failures. Untargeted retargeting produces 30, 44, 49, and 62 nominal failures per 100 attempts across the four families; ROOTCAUSE recovers 77.8%, 71.2%, 63.3%, and 60.8% of them, against 55–69% for ROOTCAUSE-F2 and Geo-gated selection, for final success of 93.3%, 87.3%, 82.0%, and 75.7%. Coupled two-fault errors mark the boundary of the single-edit assumption: on that cohort ROOTCAUSE reaches 75.0%, still the best of the six planners tested but well below its 84.6% on single faults, and a third of its failures there are coordinated pre-symptom edits that one nomination cannot express, because suffix search cannot undo a committed prefix.

Robustness. Across thresholds, priors, caps, and batch schedules, only a stricter diagnosis threshold moves success by more than a point (-1.8 $[-3.2, -0.7]$ at 0.7); removing the margin, retained fraction, or budget dependence from the value model, or decoupling the noise of the two check arms (0.7 points $[-0.7, 2.0]$), changes no outcome beyond noise; raising the cap to eight pairs adds 0.6 points $[-0.2, 1.6]$ for 3.60 pairs per decision (Appendix B).

Value-model quality and practical cost. The Factored value model is modestly discriminative but well calibrated (Table 4): Brier 0.193 against 0.222 for prevalence, AUC 0.652, ECE 0.026, and its Brier score beats prevalence in every held-out family (0.178–0.215 against 0.201–0.242). The Direct alternative has higher AUC (0.687) but worse keep-branch ECE (0.070 against 0.030) and no higher decision utility, 75.6% against 76.2% in a one-shot branch choice (-0.58 points $[-1.92, 0.67]$; Appendix B.5), so Factored remains the default; modest discrimination suffices because the controller needs only the ordering of two branch values near a tie. Evidence cannot fix everything: extra pairs reduce sampling noise but not a systematic action-response mismatch, and Figure 5 sketches a false acceptance and a useful rejection. At $b = 32$, ROOTCAUSE uses 348.2 blocks and 26.66 final candidates per decision, and a decision takes 2,417.9 s of wall-clock time, 2,383.5 s of it in

world-model inference and 33.7 s in SAM 2, so the planner runs offline or asynchronously until generation gets cheaper.

4 Related Work

World models for control. Dyna, PILCO, PETS, PlaNet and the Dreamer family, IRIS, TD-MPC, MuZero, and MPPI establish imagined search and sequence optimization with learned or sampled dynamics (Sutton, 1991; Deisenroth & Rasmussen, 2011; Chua et al., 2018; Hafner et al., 2019; 2020; 2021; 2025; Micheli et al., 2023; Hansen et al., 2022; 2024; Ha & Schmidhuber, 2018; Schrittwieser et al., 2020; Williams et al., 2017). Video world models, from visual foresight to UniSim, Genie, Cosmos, V-JEPA 2-AC, DINO-WM, and the World Action Planner, turn the model into a frozen, general-purpose asset that ROOTCAUSE takes as given (Finn & Levine, 2017; Yang et al., 2024; Bruce et al., 2024; NVIDIA, 2025; Assran et al., 2025; Zhou et al., 2025; Zhang & Du, 2026; Nair et al., 2023; Xiao et al., 2022; Oquab et al., 2023; Assran et al., 2023; Fu et al., 2026; Yin et al., 2026).

Reusing, verifying, and repairing imagined trajectories. Temporal backtracking search returns to the last verified prefix boundary and regenerates from there (Jun et al., 2026); geometric test-time scaling ranks complete candidates by zero-shot geometric evaluation (Zhao et al., 2026); CheckVLA, Embodied Tree of Thoughts, and Video Language Planning verify, deliberate over, or compose action-conditioned rollouts (Liu et al., 2026; Xu et al., 2025; Du et al., 2024). These methods decide at the boundary where a failure becomes visible or among finished candidates. On the repair side, Inner Monologue and REFLECT correct failures with language feedback (Huang et al., 2023; Liu et al., 2023b), counterfactual failure synthesis and Grasp-Then-Plan learn or attribute recovery (Li et al., 2026; Xu et al., 2026), FACT trains world-action models on failure-aware causal data (Peng et al., 2026), and CausalFlow and Causal Agent Replay attribute LLM-agent failures by counterfactual replay (Bonagiri et al., 2026; Shah, 2026). ROOTCAUSE evaluates a different object, a single edit placed before the symptom and scored by its effect on the diagnosed failure, at test time on a frozen model, and treats that edit as a hypothesis to be priced rather than a conclusion.

Counterfactual comparison and compute allocation. Common random numbers reduce the variance of paired comparisons (Glasserman & Yao, 1992; Ng & Jordan, 2000; Yadav et al., 2026), and Twin Rollouts couple the noise of counterfactual branches in interactive video models (Ma et al., 2026); where Twin Rollouts couple branches, ROOTCAUSE prices them. Recent audits ask whether robotic world models follow their actions at all (Chen et al., 2026), which our fidelity audit asks of the edits ROOTCAUSE relies on. Metareasoning and value-of-computation (Russell & Wefald, 1991; Hay et al., 2012; Callaway et al., 2018), ranking and selection (Chick & Inoue, 2001), anytime-valid sequential tests (Howard et al., 2021; Ramdas et al., 2023; Waudby-Smith & Ramdas, 2024; Snyder et al., 2025; Sadhuka et al., 2025; Ota et al., 2026), and budget-aware verification and sparse imagination (Montgomery et al., 2025; Chun et al., 2026) all decide how much computation to spend before committing. ROOTCAUSE brings this question to action-prefix reuse, where a check both informs the decision and shrinks the search the decision is about.

5 Conclusion and Limitations

ROOTCAUSE treats a rejected plan as a choice of search spaces. A failure-conditioned edit proposes which actions should survive, calibrated paired evidence decides whether they do, and the budget decides how much of that evidence is worth buying, because every check is paid for in candidates the final search will not see. On four manipulation families this recovers more of a failed plan than restarting at the symptom, optimizing the whole sequence, or confirming to a fixed depth, with a margin that grows with the budget and the horizon. The approach has a definite scope: it nominates one edit, so coupled early errors fall outside it and success on the two-fault cohort drops to 75.0%; its replay and residuals are defined for phase-structured manipulation with observable geometry; its evidence is only as good as the world model's action response, which the fidelity audit bounds at 78–83% recall for delayed effects; and a decision in simulation currently takes about forty minutes. Multi-edit nomination, learned residuals, and cheaper generation all plug into the same budgeted choice of what to keep.

AI USE STATEMENT

We used AI-based tools to assist with language editing, LaTeX typesetting, and figure scripting. All research ideas, methods, experiments, analyses, and conclusions are the authors' own, and every AI-assisted passage was checked by the authors.

REFERENCES

Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15619–15629, 2023.

Mahmoud Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Mojtaba Komeili, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, Sergio Arnaud, Abha Gejji, Ada Martin, Francois Robert Hogan, Daniel Dugas, Piotr Bojanowski, Vasil Khalidov, Patrick Labatut, Francisco Massa, Marc Szafraniec, Kapil Krishnakumar, Yong Li, Xiaodong Ma, Sarath Chandar, Franziska Meier, Yann LeCun, Michael Rabbat, and Nicolas Ballas. V-JEPA 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025.

Akash Bonagiri, Devang Borkar, Gerard Janno Anderias, Setareh Rafatirad, and Houman Homayoun. CausalFlow: Causal attribution and counterfactual repair for LLM agent failures. *arXiv preprint arXiv:2605.25338*, 2026.

Jake Bruce, Michael Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, Yusuf Aytar, Sarah Bechtle, Feryal Behbahani, Stephanie Chan, Nicolas Heess, Lucy Gonzalez, Simon Osindero, Sherjil Ozair, Scott Reed, Jingwei Zhang, Konrad Zolna, Jeff Clune, Nando de Freitas, Satinder Singh, and Tim Rocktäschel. Genie: Generative interactive environments. In *International Conference on Machine Learning*, 2024.

Frederick Callaway, Sayan Gul, Paul M. Krueger, Thomas L. Griffiths, and Falk Lieder. Learning to select computations. In *Conference on Uncertainty in Artificial Intelligence*, pp. 776–785. AUAI Press, 2018.

Sixiang Chen, Jiaming Liu, Jixian Wu, Yichen Guo, Tinghao Wang, Siyuan Qian, Hao Chen, Jiajun Cao, Jian Tang, and Shanghang Zhang. Do robotic world models really follow actions? diagnosing and aligning action-conditioned generation for policy learning. *arXiv preprint arXiv:2608.24885*, 2026.

Stephen E. Chick and Koichiro Inoue. New two-stage and sequential procedures for selecting the best simulated system. *Operations Research*, 49(5):732–743, 2001.

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

Junha Chun, Youngjoon Jeong, and Taesup Kim. Sparse imagination for efficient visual world model planning. In *International Conference on Learning Representations*, 2026.

Marc Peter Deisenroth and Carl Edward Rasmussen. PILCO: A model-based and data-efficient approach to policy search. In *International Conference on Machine Learning*, pp. 465–472, 2011.

Yilun Du, Mengjiao Yang, Pete Florence, Fei Xia, Ayzaan Wahid, Brian Ichter, Pierre Sermanet, Tianhe Yu, Pieter Abbeel, Joshua B. Tenenbaum, Leslie Kaelbling, Andy Zeng, and Jonathan Tompson. Video language planning. In *International Conference on Learning Representations*, 2024.

Chelsea Finn and Sergey Levine. Deep visual foresight for planning robot motion. In *IEEE International Conference on Robotics and Automation*, 2017.

Minghao Fu, Fan Feng, Nicklas Hansen, and Biwei Huang. Back to parsimonious latents: Learning task-centric world models from visual foundations. *arXiv preprint arXiv:2605.25620*, 2026.

Paul Glasserman and David D. Yao. Some guidelines and guarantees for common random numbers. *Management Science*, 38(6):884–908, 1992.

David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2555–2565. PMLR, 2019.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. In *International Conference on Learning Representations*, 2020.

Danijar Hafner, Timothy Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering Atari with discrete world models. In *International Conference on Learning Representations*, 2021.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, 640:647–653, 2025.

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

Nicklas A Hansen, Hao Su, and Xiaolong Wang. Temporal difference learning for model predictive control. In *International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 8387–8406. PMLR, 2022.

Nicholas Hay, Stuart Russell, David Tolpin, and Solomon Eyal Shimony. Selecting computations: Theory and applications. In *Conference on Uncertainty in Artificial Intelligence*, pp. 346–355. AUAI Press, 2012.

Steven R. Howard, Aaditya Ramdas, Jon McAuliffe, and Jasjeet Sekhon. Time-uniform, nonparametric, nonasymptotic confidence sequences. *The Annals of Statistics*, 49(2):1055–1080, 2021.

Wenlong Huang, Fei Xia, Ted Xiao, Harris Chan, Jacky Liang, Pete Florence, Andy Zeng, Jonathan Tompson, Igor Mordatch, Yevgen Chebotar, Pierre Sermanet, Tomas Jackson, Noah Brown, Linda Luu, Sergey Levine, Karol Hausman, and Brian Ichter. Inner monologue: Embodied reasoning through planning with language models. In *Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, pp. 1769–1782. PMLR, 2023.

Sejoon Jun, Zheng Ding, Huangyuan Su, Weirui Ye, and Yilun Du. Temporal backtracking search for test-time generative video reasoning. *arXiv preprint arXiv:2606.13861*, 2026.

Dayou Li, Jiuzhou Lei, Hao Wang, Lulin Liu, Yunhao Yang, Zihan Wang, Bangya Liu, Minghui Zheng, and Zhiwen Fan. Learning actionable manipulation recovery via counterfactual failure synthesis. *arXiv preprint arXiv:2603.13528*, 2026.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. LIBERO: Benchmarking knowledge transfer for lifelong robot learning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 44776–44791. Curran Associates, Inc., 2023a.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, and Lei Zhang. Grounding DINO: Marrying DINO with grounded pre-training for open-set object detection. In *European Conference on Computer Vision*, 2024.

Yushan Liu, Peibo Sun, Xintao Chao, Zhenyang Yang, Yifan Xie, Lingfeng Zhang, Shoujie Li, Chenyu Tang, Fang Chen, Xiao-Ping Zhang, and Wenbo Ding. CheckVLA: Execution-time verification with action-conditioned world model for long-horizon mobile manipulation. *arXiv preprint arXiv:2607.26789*, 2026.

Zeyi Liu, Arpit Bahety, and Shuran Song. REFLECT: Summarizing robot experiences for failure explanation and correction. In *Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 3468–3484. PMLR, 2023b.

Yu Ma, Hongli Shi, and Xinran Xu. Twin Rollouts: Noise-coupled counterfactual branching in interactive video world models. *arXiv preprint arXiv:2608.08982*, 2026.

Vincent Micheli, Eloi Alonso, and François Fleuret. Transformers are sample-efficient world models. In *International Conference on Learning Representations*, 2023.

Kyle Montgomery, Sijun Tan, Yuqi Chen, Siyuan Zhuang, Tianjun Zhang, Raluca Ada Popa, and Chenguang Wang. Budget-aware test-time scaling via discriminative verification. *arXiv preprint arXiv:2510.14913*, 2025.

Suraj Nair, Aravind Rajeswaran, Vikash Kumar, Chelsea Finn, and Abhinav Gupta. R3M: A universal visual representation for robot manipulation. In *Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, pp. 892–909. PMLR, 2023.

Andrew Y. Ng and Michael I. Jordan. PEGASUS: A policy search method for large MDPs and POMDPs. In *Conference on Uncertainty in Artificial Intelligence*, pp. 406–415, 2000.

NVIDIA. Cosmos world foundation model platform for physical AI. *arXiv preprint arXiv:2501.03575*, 2025.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Hirofumi Ota, Naoto Iwase, Yuki Ichihara, Junpei Komiyama, and Masaaki Imaizumi. CITE: Anytime-valid statistical inference in LLM self-consistency. *arXiv preprint arXiv:2605.05873*, 2026.

Quanquan Peng, Yutong Liang, Rui Yan, Nicklas Hansen, and Xiaolong Wang. FACT: Failure-aware causal training for world-action models. *arXiv preprint arXiv:2608.10232*, 2026.

Aaditya Ramdas, Peter Grünwald, Vladimir Vovk, and Glenn Shafer. Game-theoretic statistics and safe anytime-valid inference. *Statistical Science*, 38(4):576–601, 2023.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. SAM 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024.

Reuven Rubinstein. The cross-entropy method for combinatorial and continuous optimization. *Methodology and Computing in Applied Probability*, 1(2):127–190, 1999.

Stuart Russell and Eric Wefald. Principles of metareasoning. *Artificial Intelligence*, 49(1–3):361–395, 1991.

Shuvom Sadhuka, Drew Prinster, Clara Fannjiang, Gabriele Scalia, Bonnie Berger, Aviv Regev, and Hanchen Wang. E-valuator: Reliable agent verifiers with sequential hypothesis testing. *arXiv preprint arXiv:2512.03109*, 2025.

Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, Timothy Lillicrap, and David Silver. Mastering Atari, Go, chess and shogi by planning with a learned model. *Nature*, 588:604–609, 2020.

Jaineet Shah. Causal Agent Replay: Counterfactual attribution for LLM-agent failures. *arXiv preprint arXiv:2606.08275*, 2026.

David Snyder, Asher James Hancock, Apurva Badithela, Emma Dixon, Patrick Miller, Rares Andrei Ambrus, Anirudha Majumdar, Masha Itkina, and Haruki Nishimura. Is your imitation learning policy better than mine? policy comparison with near-optimal stopping. In *Robotics: Science and Systems*, 2025.

Richard S. Sutton. Dyna, an integrated architecture for learning, planning and reacting. *SIGART Bulletin*, 2:160–163, 1991.

Ian Waudby-Smith and Aaditya Ramdas. Estimating means of bounded random variables by betting. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 86(1):1–27, 2024.

Grady Williams, Nolan Wagener, Brian Goldfain, Paul Drews, James M. Rehg, Byron Boots, and Evangelos A. Theodorou. Information theoretic MPC for model-based reinforcement learning. In *IEEE International Conference on Robotics and Automation*, pp. 1714–1721. IEEE, 2017.

Tete Xiao, Ilija Radosavovic, Trevor Darrell, and Jitendra Malik. Masked visual pre-training for motor control. *arXiv preprint arXiv:2203.06173*, 2022.

Jiahao Xu, Peiyuan Wang, Hanzhuo Zhang, Zihao Yu, Tianyu Fu, Hao Chen, Xuanhao Xiang, Jianbo Yu, Chenchen Fu, and Wanyuan Wang. Grasp-Then-Plan with failure attribution: A closed two-stage framework for precise and generalizable robotic manipulation. *arXiv preprint arXiv:2606.03385*, 2026.

Wenjiang Xu, Cindy Wang, Rui Fang, Mingkang Zhang, Lusong Li, Jing Xu, Jiayuan Gu, Zecui Zeng, and Rui Chen. Embodied tree of thoughts: Deliberate manipulation planning with embodied world model. *arXiv preprint arXiv:2512.08188*, 2025.

Sandarbh Yadav, Frederic J Maliakkal, Harshad Khadilkar, and Shivaram Kalyanakrishnan. Using common random numbers for simulation-based planning with rollouts. *arXiv preprint arXiv:2605.04732*, 2026.

Sherry Yang, Yilun Du, Seyed Kamyar Seyed Ghasemipour, Jonathan Tompson, Leslie Kaelbling, Dale Schuurmans, and Pieter Abbeel. Learning interactive real-world simulators. In *International Conference on Learning Representations*, 2024.

Shicheng Yin, Kaixuan Yin, Weixing Chen, Yang Liu, Guanbin Li, and Liang Lin. DDP-WM: Disentangled dynamics prediction for efficient world models. In *International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, 2026.

Xiangcheng Zhang and Yilun Du. World Action Planner: Generalizable decision-making with action-conditioned world models. *arXiv preprint arXiv:2607.27599*, 2026.

Zesen Zhao, Minkyoung Cho, Hui Shen, Boyuan Zheng, Kunxiao Gao, Yulong Cao, and Z. Morley Mao. Test-time scaling for world action models via zero-shot geometric evaluation. *arXiv preprint arXiv:2607.17454*, 2026.

Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. DINO-WM: World models on pre-trained visual features enable zero-shot planning. In *International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 79115–79135. PMLR, 2025.

Yuke Zhu, Josiah Wong, Ajay Mandlekar, Roberto Martín-Martín, Abhishek Joshi, Kevin Lin, Abhiram Maddukuri, Soroush Nasiriany, and Yifeng Zhu. robosuite: A modular simulation framework and benchmark for robot learning. *arXiv preprint arXiv:2009.12293*, 2020.

A Implementation and Decision Analysis

Failure-conditioned replay. The task program assigns acquisition, transport, and release phases to the nominal sequence. For an acquisition edit, replay recomputes the relative object–gripper transform at capture. Later gripper poses carry this new transform until the nominal release event. A transport edit modifies the interpolated gripper path over its fixed interval and the transition into the next interval. Release changes the candidate support location. All other commands, object identities, and phase labels are held fixed. The replay recomputes the maximum residual through the diagnosed boundary s; the endpoint-local control instead recomputes only d_i. Thus the two controls can rank different edits at exactly the same site.

Each site receives 24 scrambled Sobol proposals in normalized translation, rotation, and phase-timing coordinates. The nominal command is added as a zero-change reference. Proposals violating workspace, inverse-kinematics, or phase-order limits are discarded. Scores break ties by smaller edit norm, then later edit index, then proposal index. Missing object tracks or an ambiguous attachment invalidate the proxy rather than assuming successful contact. The replay uses no world-model call; all candidate video predictions are charged in the ledger. It approximates transport geometry, but does not predict friction, compliance, or recovery after a novel contact.

Residuals and model validity. Centering measures the distance between the planned capture point and the interior of the observed object mask; slip measures the change in the object–gripper offset after acquisition. Placement measures distance outside the eroded target support region. Clearance measures intrusion into the projected obstacle margin. Distances are normalized by object width for capture and support, and by the configured safety margin for clearance, then clipped to $[0, 1]$. The active-phase residual is their maximum. Invalid masks, inconsistent multi-view geometry, and lost tracks produce an invalid comparison, counted as $Y = 0$.

The default Factored value model receives no additional fidelity feature. Track validity affects its evidence through Y, and ambiguous geometry can prevent nomination altogether. The Direct comparison additionally uses $f = v \exp(-e)$, where v is the valid endpoint fraction and e is the median center disagreement between the two nominal predictions, normalized by object width. This observable agreement score is distinct from simulator ground truth and does not change the default verifier.

Development labels and the Factored value model. For each development origin and nominated edit, we draw 32 reference suffixes from the retained branch, predict each resulting complete plan with the world model, then execute it from the same reset state, and label the prefix $G = 1$ if at least seven succeed. The bank is independent of both the paired evidence and the candidate banks used to fit search capacity. A separate evidence stream supplies eight paired checks per development origin; its posterior mean $(1 + \text{passes})/10$ is the fitting input for repeatable check probability. The four-feature logistic model uses clipped log-odds:

$$\begin{aligned} t &= \operatorname{logit}(\operatorname{clip}(\theta, 10^{-4}, 1 - 10^{-4})), \\ q(\theta, m, \ell) &= \operatorname{sigmoid}(w_q^\top [1, t, m, \ell]). \end{aligned} \tag{8}$$

Fitting minimizes summed binary logistic loss against G, with ℓ_2 penalty 0.25 and intercept penalty 10^{-6}. Newton updates stop after 60 iterations or coefficient change below 10^{-8}. Features retain their specified scales; task identity, injected fault sites, and simulator state are excluded. At test time, 32-node quadrature integrates q over the current Beta posterior.

Capacity-dependent search targets. For every reachable positive candidate allowance n, a development search bank supplies the terminal outcome of the plan returned by the common search-and-rank procedure. Root and retained banks each contain 64 candidates and use independent random streams. Within a branch, the same ordered bank serves every reachable allowance: rank its first n candidates and select their best-scoring plan. Reset-based execution of each distinct candidate supplies its terminal label. Across origins, these banks capture proposal variability without using prediction scores as success targets.

Let $y_{K,j,n}$ and $y_{R,j,n}$ denote these binary selected-plan outcomes, and let $\mathcal{D}_g$ contain development records with reference label g. Beta$(1, 1)$ smoothing gives

$$\widehat{S}_g(n) = \frac{1 + \sum_{j \in \mathcal{D}_g} y_{K,j,n}}{2 + |\mathcal{D}_g|}, \quad g \in \{0, 1\}, \qquad \widehat{S}_{\text{root}}(n) = \frac{1 + \sum_{j \in \mathcal{D}} y_{R,j,n}}{2 + |\mathcal{D}|}. \tag{9}$$

Here $S_{\text{good}} = S_1$, $S_{\text{bad}} = S_0$, and all three values are zero at $n = 0$. Both label groups require observations at each fitted allowance. Every candidate count reachable under the budgets, diagnostic depths, and check schedules has its own fitted value; a missing count disables that configuration rather than invoking an extrapolated value. Raw fitted curves need not be monotone because an additional candidate can mislead the ranker. The controller uses these unaltered curves in its stopping decisions.

Offline generation is normalized to a complete prediction of L native blocks. One development origin costs $32 + 128 + 16L_s/L$ full-rollout equivalents: 32 reference plans, 64 candidates for each branch, and eight paired checks. The learning-curve cost sums this allowance over the nested development sets of all four families. Candidate prefixes are shared across capacities; each allowance does not require another bank. Simulator control steps and CPU parameter-fitting time are separate costs, excluded from this generation column and from the online budget B. Held-out usefulness-audit banks of 16, 32, and 64 suffixes are independent of training and never update the fitted decision model.

Splits and the Direct comparison. Development and evaluation origins have disjoint physical reset identifiers, including aliases and all associated random seeds. Each of four leave-one-family-out fits uses development origins from the other three families for both q and the capacity curves. The held-out family supplies only evaluation outcomes. Same-family learning curves instead use nested development subsets with an unchanged, separate evaluation cohort. Neither reference labels nor injected-site annotations enter a test-time decision.

Direct replaces the factorization with two seven-feature logistic models. Writing $\nu_n = \log(1 + n)/\log 65$, its branch responses are

$$v_b^{\text{Direct}}(n, \theta, m, \ell, f) = \text{sigmoid}(w_b^\top [1, t, m, \ell, f, \nu_n, \theta f]), \qquad b \in \{K, R\}. \tag{10}$$

Both responses are zero at $n = 0$. They fit terminal search-and-rank outcomes, not usefulness labels, using the same origin splits, candidate banks, and regularization as the default model. The comparison reports Brier score, Brier skill relative to training-fold branch prevalence, AUC, ten-bin ECE, and selected-branch utility. Undefined AUC for a single-class set and Brier skill with a zero reference Brier are reported as missing values.

A calibration audit at one recorded candidate allowance evaluates only probability quality and the resulting choice between its two branch targets. It supplies neither a capacity curve nor an online adaptive-planner result. The full decision comparison requires the evidence loop and final search to run again with the fitted value model at every visited allowance.

Finite-state computation. After k pairs with z passes, the posterior is $\text{Beta}(a_0 + z, b_0 + k - z)$. For a batch of size h,

$$\Pr(J = j \mid a, b, h) = \binom{h}{j} \frac{\text{B}(a + j, b + h - j)}{\text{B}(a, b)}, \qquad j = 0, \ldots, h. \tag{11}$$

Backward induction enumerates the posterior states at depths $0, 1, 2, 4$. The two-pair batch sums all three predictive outcomes; it is not evaluated by squaring a posterior-mean pass probability. The context, nomination, and diagnostic endpoint are locked throughout the evidence loop. Memoization uses the count of passes, completed pairs, and remaining native blocks. The four-candidate reserve is checked before diagnosis and before every batch, and unaffordable evidence actions are removed from the maximization.

Checks within a batch are charged separately but the adaptive policy makes its next decision only after the complete batch. A generation exception charges started blocks, cancels the unfinished batch, and invokes root search using the actual remaining allowance. Fixed-depth controls may stop after a failed pair; their realized costs, rather than nominal caps, enter the ledger. Diagnosis, checks, and final search never reuse stochastic tapes across stages.

A.1 Why information must repay displaced search

Equation 7 is an accounting identity for the calibrated decision model: adding and subtracting $\mathbb{E}M(\pi', r')$ and $M(\pi, r')$ from $\mathbb{E}W_{k+h}(\pi', r') - M(\pi, r)$ separates the information gained at the reduced budget, the capacity lost by paying for a check, and the option value of subsequent checks. The signs of the three terms are the only claims made here; they follow from the structure of the value model rather than from any property of the environment.

For each branch, posterior averaging gives $\mathbb{E}_{\pi'} U_b(\pi', r') = U_b(\pi, r')$ by the tower property. Since M is the maximum of two functions linear in the posterior, $\mathbb{E}M(\pi', r') \geq M(\pi, r')$. Also, $W_{k+h}(\pi', r') \geq M(\pi', r')$ because stopping remains available. These facts establish nonnegative information and future-option terms. If both fitted branch values are nondecreasing in candidate count, paying for evidence cannot improve the stopping value, making the capacity term nonnegative as well. Without monotonicity, the identity still holds with a signed capacity term. At the final batch the option term is zero, so checking is preferred exactly when its information gain exceeds its capacity cost. These statements explain the Bellman policy within its calibrated model; calibration error determines how well that policy transfers to physical outcomes.

B Evaluation Details

This appendix documents the world-model interface, the evaluation cohorts, the comparison rules, and the statistical definitions behind every reported number. All experiments share physical origin identifiers, proposal streams, and native-block budgets. Separate development and evaluation cohorts distinguish the cost of fitting a decision model from its performance on held-out tasks.

B.1 World-model and action interfaces

The world-model interface uses the released task-domain WAP variants in Table 5 (Zhang & Du, 2026). Their upstream fine-tuning is part of the release; ROOTCAUSE adds no backbone training. The model repository is `XiangchengZhang/world-action-planner` on Hugging Face, with checkpoints at `world_models/<name>/checkpoints/latest.ckpt` and accompanying `.hydra` configurations. All planners use identical checkpoints and inference settings.

Table 5: **World-model interface and primary cohorts.** Each family contains 100 evaluation origins. Native-block horizons define the online allowance.

Task group	Checkpoint name	Primary origins	$H = L$
StackCube	`robosuite_ft`	100	6
PickPlaceCan	`robosuite_ft`	100	8
LIBERO-Object	`libero_object_ft`	100	12
LIBERO-Long	`libero_90_base`	100	18

Native prediction. Four 224×224 RGB views form a 448×448 grid containing agent, overhead, wrist, and side views. The released configuration selects history indices $0, 3, \ldots, 60$ from 61 frames at 20 Hz and predicts 20 future frames at 20 Hz per block, so the 21-frame history is approximately 7 FPS and a prediction block spans one second. The server uses 20 denoising steps, overriding the checkpoint configuration's 40. The backbone and VAE are Wan2.1-T2V-1.3B. Autoregression joins native blocks; predicting through primitive s costs $L_s = s$ blocks, not one arbitrarily long model call.

Action rendering and randomness. Each primitive maps to 20 operational-space controller actions with translation, rotation, and gripper coordinates. The official client advances an object-free robot model and renders future robot poses into the four camera views. The server consumes historical RGB, historical pose images, and future pose images; it does not directly consume primitive labels or action vectors. The paired-check adapter exposes a per-block random tape; the released WebSocket interface does not accept an explicit seed or noise tensor. The two arms of a check use the same initial noise and scheduler randomness. Different pairs, diagnosis, and final search use independent streams. Each complete candidate begins at the observed history, never at an imagined prefix.

B.2 Cohorts, calibration, and perception

Origin manifest. An origin specifies the task, physical reset, observation history, goal, nominal action sequence, controller, and failure source. Aliases of one physical reset are grouped before any split. The primary cohort contains 100 single-injected-fault origins per family. Injection logs record the altered command independently of the planner and are inaccessible during planning. Faults at the symptom boundary are included alongside earlier faults. Nominal successes, failed diagnosis, and abstentions remain in the relevant denominator; origins cannot be filtered using a method's later performance.

Stack and Can nominal plans use fixed acquisition, transport, and placement waypoints. LIBERO plans extract end-effector targets and gripper transitions from held-out demonstrations, retarget them to the new reset, and resample to H intervals. Phase labels derive from assigned subgoals and gripper events, not terminal outcomes. Natural failures use a separate, untargeted retargeting noise schedule, with 100 origins per family and three seeds per origin. Every retargeting attempt is logged before selecting the nominal-failure subset. A disjoint two-fault cohort contains 80 origins (20 per family) and three seeds, yielding 240 runs per method; it is excluded from single-site attribution.

Development and transfer. The development pool contains 100 independent origins per family. Each primary LOFO fold fits on the other three families and executes on the held-out family's evaluation origins. All seeds and branches of an origin remain in the same split. Hyperparameters are fixed in development, using inner origin-grouped validation when selection is needed; outer-fold success does not choose between Factored and Direct. Same-family learning curves use content-stable nested subsets of 5, 10, 20, 40, and 80 development origins per family. Larger sets contain smaller sets; evaluation origins, proposal streams, and online budgets do not change. Calibration must cover every reachable candidate capacity, rather than one middle-budget allowance.

The default Factored model uses a fixed 32-suffix development bank to label branch usefulness, independent paired evidence to fit usefulness probability, and separate candidate-bank outcomes to fit search capacity. Each branch's 64-candidate bank supplies nested search capacities. The resulting generation allowance is $32+128+16L_s/L$ full-rollout equivalents per development origin, including eight paired checks. Simulator execution and parameter-fitting time are tracked separately from this allowance. *Test audit* banks of 16, 32, or 64 suffixes evaluate returned branches without changing decisions. Larger banks append independent suffixes rather than forcing label agreement.

Ordinary RGB geometry. Grounding DINO initializes text-conditioned boxes, and SAM 2 propagates masks independently through each branch and view (Liu et al., 2024; Ravi et al., 2024). The estimator uses camera calibration, robot proprioception, and calibrated workspace planes. Wrist extrinsics update through forward kinematics. Approximate multi-view object centers must agree with mask projections; mask centroids are not exact point correspondences. Eroded support regions account for object extent. Missing tracks, implausible area changes, and large reprojection errors invalidate geometry. Simulator state supplies terminal predicates and offline references.

All planners use translation bounds of 20 mm, rotation bounds of 10°, and gripper-event shifts of at most three control steps within the legal phase. Every primitive still occupies 20 steps. Nomination evaluates 24 deterministic low-discrepancy edits per eligible site, plus a zero-change reference. Workspace, inverse-kinematics, and phase constraints are shared. Equal scores break ties by smaller edit norm, later site, then proposal index.

B.3 Comparisons, generation cost, and decision traces

Complete-sequence and related planners. Full-root samples complete plans. CEM and MPPI use four generations, populations $4, 8, 16$ at $b = 16, 32, 64$, and the same total complete-evaluation allowance. CEM keeps $1, 2, 4$ elites, uses smoothing 0.25, and floors standard deviation at 10% of its initial value. MPPI uses temperature 1.0. Both return the best evaluated sample; an unevaluated proposal mean is not a free candidate. WAP-style BoN applies primitive-conditioned proposals and complete ranking. The TBS-style adapter restarts at a process-verified semantic boundary and can use an earlier gripper-opening boundary. Geo-gated selection adds action–future consistency filtering and cross-view geometric reranking. Their common-interface status is retained in result labels.

The Twin-Rollouts-style paired branch search compares two nominated anchors with two shared-noise rounds through s, costing $4L_s$ blocks, then searches the selected branch; it adopts the noise coupling of Twin Rollouts within our interface rather than reproducing the original planner. ROOTCAUSE-NoChk commits immediately to the nominated branch. The fixed-depth controls ROOTCAUSE-F1, -F2, and -F4 keep the branch while pairs pass and reopen after their first failed pair; their one-, two-, or four-pair limit is a maximum subject to the search reserve. ROOTCAUSE uses the Bellman decision after each completed batch. ROOTCAUSE+Geo applies the gate to the chosen branch's candidate bank and charges its full computation.

Generation ledger. The allowance is $B = bL$. Diagnosis costs $2L$, one complete check pair costs $2L_s$, and a final candidate costs L, including any retained prefix. For the confirmation family,

Table 6: **Full LOFO budget sweep.** Macro success (%) and origin-cluster intervals use the same held-out cohort at every budget.

Planner	$b = 16$ [95% CI]	$b = 32$ [95% CI]	$b = 64$ [95% CI]
Full-root	54.0 [51.4, 56.6]	63.5 [61.0, 66.1]	75.3 [72.9, 77.7]
Continuous CEM	63.4 [60.9, 66.0]	73.0 [70.7, 75.5]	82.0 [79.8, 84.2]
MPPI	67.8 [65.4, 70.3]	74.8 [72.3, 77.1]	83.6 [81.5, 85.5]
WAP-style BoN	63.9 [61.4, 66.5]	72.4 [69.9, 75.0]	81.2 [79.0, 83.3]
TBS-style adapter	67.8 [65.3, 70.2]	75.5 [73.2, 77.9]	82.6 [80.4, 84.8]
Paired branch search	69.9 [67.4, 72.4]	78.9 [76.7, 81.2]	84.8 [82.8, 86.8]
Geo-gated adapter	71.6 [69.3, 74.1]	80.7 [78.5, 82.8]	86.5 [84.7, 88.3]
ROOTCAUSE-NoChk	72.8 [70.5, 75.2]	74.8 [72.3, 77.0]	79.7 [77.5, 82.0]
ROOTCAUSE-F1	73.0 [70.5, 75.5]	78.8 [76.7, 80.9]	84.3 [82.3, 86.3]
ROOTCAUSE-F2	73.8 [71.6, 76.1]	80.1 [77.9, 82.2]	84.2 [82.2, 86.3]
ROOTCAUSE-F4	72.0 [69.4, 74.6]	79.4 [77.2, 81.6]	85.8 [83.8, 87.8]
ROOTCAUSE	74.1 [71.8, 76.3]	84.6 [82.5, 86.6]	88.8 [86.9, 90.6]
ROOTCAUSE+Geo	**75.1 [72.8, 77.4]**	**85.8 [83.8, 87.6]**	**90.7 [88.9, 92.3]**

Table 7: **Prespecified paired contrasts.** Percentage-point differences with 95% intervals, computed on paired origins before rounding, so a contrast can differ by 0.1 from the difference of two rounded means. Comparators are fixed before testing; the final row uses the matched gated ROOTCAUSE-NoChk cell.

Contrast (percentage points)	$b = 16$	$b = 32$	$b = 64$
ROOTCAUSE − Geo-gated adapter	2.5 [0.2, 4.8]	3.9 [2.0, 5.8]	2.2 [0.4, 4.2]
ROOTCAUSE − ROOTCAUSE-F2	0.3 [-1.9, 2.6]	4.5 [2.7, 6.4]	4.6 [2.6, 6.6]
ROOTCAUSE − ROOTCAUSE-NoChk	1.2 [-1.0, 3.4]	9.8 [7.7, 11.9]	9.1 [7.0, 11.0]
ROOTCAUSE+Geo − ROOTCAUSE	1.0 [-1.0, 3.0]	1.2 [-0.7, 2.9]	1.9 [0.3, 3.6]
ROOTCAUSE+Geo − ROOTCAUSE-NoChk+Geo	-2.3 [-4.5, -0.2]	7.0 [5.1, 8.8]	7.5 [5.7, 9.4]

$2L + 2kL_s + NL \leq B$. Root-only optimizers pay no unused diagnosis cost. At least $4L$ remains reserved before another check. Every started block is charged even if inference fails; an unstarted reservation is released. Logs distinguish diagnosis, checking, search, failed generation, and unused blocks. A partially completed batch is recorded as such, not rounded to a planned depth.

Table 8: **Generation ledger at $b = 32$.** Per-decision means follow the native-block accounting. Task-specific allowances apply before aggregation; failed generation remains charged.

Planner	Pairs	Final candidates	Used blocks	Failed blocks	Unused
Full-root	0.00	31.99	351.89	0.01	0.11
Continuous CEM	0.00	31.99	351.92	0.01	0.08
MPPI	0.00	31.99	351.88	0.01	0.12
WAP-style BoN	0.00	31.98	351.82	0.02	0.18
TBS-style adapter	0.00	29.98	351.85	0.01	0.15
Paired branch search	2.00	27.33	347.75	0.01	4.25
Geo-gated adapter	0.00	29.99	351.88	0.01	0.12
ROOTCAUSE-NoChk	0.00	29.98	351.83	0.02	0.17
ROOTCAUSE-F1	1.00	28.47	347.65	0.01	4.35
ROOTCAUSE-F2	1.62	27.76	347.78	0.01	4.22
ROOTCAUSE-F4	2.51	26.77	347.96	0.01	4.04
ROOTCAUSE	2.72	26.66	348.17	0.01	3.83
ROOTCAUSE+Geo	2.72	26.66	348.22	0.02	3.78

Table 9: **ROOTCAUSE check allocation.** Fractions include all decisions, zero-check fallback, and incomplete batches. The stopping depths and consumed blocks are summarized separately.

Budget	$k = 0$	$k = 1$	$k = 2$	$k = 4$	Incomplete batch
16	2.0	59.8	31.3	6.9	0.0
32	0.0	18.2	36.6	45.2	0.0
64	0.0	14.5	20.1	65.4	0.0

Log integrity. Every decision in the reported runs was logged as a trajectory of diagnosis, nomination, paired residuals, validity flags, posterior updates, Bellman choices, and final search, and all reported statistics are computed from these logs. Before aggregation we verified that adjacent logged states obey the controller's accounting: each posterior includes exactly the completed pairs, the remaining budget subtracts their cost, the selected branch matches the stopping action, and candidate counts agree with the remaining budget (Table 10). These checks certify the bookkeeping of the executed runs, not their task performance.

Table 10: **Log integrity of the reported runs.** Each rule is checked on every logged record of the primary sweep; all reported numbers derive from these logs.

Check	Consistency rule	Records	Violations
Posterior update	$a = 1 + p,\ b = 1 + k - p$	11054	0
Stopping decision	Maximum Bellman value	11054	0
Budget feasibility	$C_{\text{spent}} + C_{\text{left}} = B$	46800	0
Candidate accounting	$C_{\text{search}} = NL + C_{\text{fail}}$	46800	0
Outcome pairing	Common origin and seed keys	46800	0

Setting-specific decision trajectories. Each setting in Table 11 has its own posterior updates, stopping decision, and final search. Diagnosis-threshold changes also repeat symptom selection and nomination. Comparison thresholds are applied to the paired path residuals before updating the posterior. Matched random tapes pair settings without changing their marginal distributions. The final candidate bank follows the newly chosen branch; outcomes and costs are aggregated over that setting's complete decision trajectory. At the default thresholds the required improvement δ is implied by the diagnosis threshold and the terminal tolerance ($D_s^{\text{nom}} \geq 0.5$ and $D_s^{\text{edit}} \leq 0.25$ give an improvement of at least 0.25), which is why its rows in Table 11 leave the decisions unchanged.

Table 11: **Controller sensitivity at $b = 32$.** Success, pair counts, and decision changes are compared with the default under matched origins and random tapes.

Changed setting	Value	Success	Δ to default	Pairs	Changed decisions
Diagnosis threshold	0.3	84.4	-0.2 [-1.8, 1.4]	2.88	9.2
	0.5	84.6	0.0 [0.0, 0.0]	2.72	0.0
	0.7	82.8	-1.8 [-3.2, -0.7]	2.61	6.1
Required improvement	0.15	84.6	0.0 [0.0, 0.0]	2.72	0.0
	0.25	84.6	0.0 [0.0, 0.0]	2.72	0.0
	0.35	84.6	0.0 [0.0, 0.0]	2.72	0.1
Terminal tolerance	0.15	83.9	-0.7 [-2.0, 0.7]	2.32	49.2
	0.25	84.6	0.0 [0.0, 0.0]	2.72	0.0
	0.35	84.6	0.0 [-0.5, 0.5]	2.78	6.1
Beta prior	$(0.5, 0.5)$	84.2	-0.4 [-0.8, -0.0]	2.72	8.8
	$(1.0, 1.0)$	84.6	0.0 [0.0, 0.0]	2.72	0.0
	$(2.0, 2.0)$	84.3	-0.3 [-1.1, 0.4]	2.39	18.5
Maximum pairs	2	83.3	-1.2 [-2.9, 0.4]	1.51	65.8
	4	84.6	0.0 [0.0, 0.0]	2.72	0.0
	8	85.2	0.6 [-0.2, 1.6]	3.60	25.2
Batch schedule	1/1/2	84.6	0.0 [0.0, 0.0]	2.72	0.0
	Sequential	84.6	0.0 [-0.9, 1.0]	2.60	36.8

B.4 Attribution, natural failures, and interacting mechanisms

Site identification and branch quality. For single injected faults, exact-hit accuracy is the number of correctly returned injection sites divided by all eligible origins, including no-site decisions. Site-return coverage is the fraction returning any valid site. Conditional exact-hit among returned sites is supplementary and never replaces unconditional accuracy. Early and symptom-coincident faults remain separate strata. The injection annotation is read only by the evaluator. The symptom-boundary control returns the diagnosed boundary; it is distinct from a process-verified temporal restart.

At a matched site, failure-endpoint, endpoint-local, and random edit ranking use exactly the same legal proposals. The first score propagates an edit to the diagnosed symptom; the second uses its own

Table 12: **Single-fault localization.** Exact-hit, coverage, and useful branch rates (%) use distinct denominators. Abstentions count as misses. The symptom-boundary rule returns the diagnosed boundary itself, so its exact-hit rate on early faults is zero by construction.

Rule	Fault stratum	Exact hit	Coverage	Useful branch
Failure-endpoint	Early	**72.1**	**96.3**	**68.6**
Symptom boundary	Early	0.0	95.4	49.2
Endpoint-local	Early	38.5	95.4	56.0
Failure-endpoint	At symptom	80.3	96.7	**65.8**
Symptom boundary	At symptom	**96.1**	96.1	50.8
Endpoint-local	At symptom	79.2	**96.9**	56.4

local residual. A test audit bank labels a branch useful when at least 20% of its suffix executions succeed (seven of 32 in the default bank). Useful retention conditions on all useful branches; bad commitment conditions on all unhelpful branches. Zero denominators are reported as NA, not zero. These labels measure repair opportunities and are not equivalent to identifying the injected command.

Table 13: **Edit ranking at a matched depth.** The site, proposal bank, suffix allowance, ranker, and reset are shared; only within-site edit ranking changes. Entries are rates (%).

Within-site ranking	Useful branch	Retained useful	Bad commit	Success
Failure-endpoint	**69.8**	87.3	**23.7**	69.8
Endpoint-local	60.4	86.7	26.8	**71.2**
Random edit	44.4	**89.4**	24.2	70.6

Table 14: **Recovery from untargeted retargeting failures.** Each cell gives recovery / final success (%). Recovery conditions on nominal failure; final success uses all retargeting attempts. Injected-site accuracy is inapplicable to this cohort.

Family	All attempts	Nominal failures	ROOTCAUSE-F2	Geo-gated	ROOTCAUSE
StackCube	100	30	64.4 / 89.3	68.9 / 90.7	**77.8 / 93.3**
PickPlaceCan	100	44	61.4 / 83.0	62.1 / 83.3	**71.2 / 87.3**
LIBERO-Object	100	49	57.1 / 79.0	59.9 / 80.3	**63.3 / 82.0**
LIBERO-Long	100	62	55.4 / 72.3	55.4 / 72.3	**60.8 / 75.7**

The four-cell interaction. Let S_{ag} denote success with ROOTCAUSE evidence $a \in \{0, 1\}$ and geometric gating $g \in \{0, 1\}$. The no-evidence cells reuse the same nominated branch; all four cells share proposals and final ranking. We report $S_{11} - S_{10}$, $S_{11} - S_{01}$, and $(S_{11} - S_{10}) - (S_{01} - S_{00})$ under one paired bootstrap draw. Candidate generation, perception, and rejected proposals remain charged. The full-root Geo-gated baseline belongs to the primary comparison, not this four-cell interaction. Table 15 gives the four cells with their pair counts and block consumption.

Table 15: **ROOTCAUSE evidence $\times$ geometric candidate gating.** Four matched cells at $b = 32$. Rejected candidates retain their generation cost. Shaded rows use ROOTCAUSE's evidence controller; bold success is the best cell. The paired interaction is $\Delta_{AG} = (S_{11} - S_{10}) - (S_{01} - S_{00})$.

Evidence	Candidate gate	Success [95% CI]	Pairs	Blocks
No evidence	Off	74.8 [72.3, 77.0]	0.00	351.8
No evidence	On	78.8 [76.7, 80.9]	0.00	351.9
ROOTCAUSE	Off	84.6 [82.5, 86.6]	2.72	348.2
ROOTCAUSE	On	**85.8 [83.8, 87.6]**	2.72	348.2
Interaction	Δ_{AG}	-2.8 [-5.7, -0.1]	–	–

B.5 Calibration and branch-value comparison

The branch-value study cross-fits a separate 400-origin cohort with three seeds and two branch targets per origin, yielding 2,400 out-of-fold records. Its origins and evidence streams are distinct

Table 16: **Same-family development learning curve at $b = 32$.** Each row refits on a nested development set and evaluates unchanged test origins. Fit cost sums full-rollout equivalents across all four families; CPU fitting time and simulator control steps are accounted separately.

Development origins	ROOTCAUSE	ROOTCAUSE-F2	ROOTCAUSE-NoChk	Geo-gated	Fit cost
5	76.5	80.1	74.8	**80.7**	3,369 eq.
10	78.1	80.1	74.8	**80.7**	6,746 eq.
20	80.2	80.1	74.8	**80.7**	13,498 eq.
40	**82.5**	80.1	74.8	80.7	27,053 eq.
80	**84.9**	80.1	74.8	80.7	54,104 eq.

from the primary planning cohort. Each LOFO fit uses 300 origins from three families and predicts the remaining family's 100 origins. All seeds stay together; no held-out label enters its own fit. ROOTCAUSE uses Factored by default; ROOTCAUSE-Direct is the matched value-model ablation. The two estimators share splits, candidate allowances, and terminal targets. This study isolates probability estimation and the choice between two branches at a given candidate allowance. The allocation analysis follows budget-aware verification, confidence sequences, and simulation ranking-and-selection (Montgomery et al., 2025; Howard et al., 2021; Ramdas et al., 2023; Waudby-Smith & Ramdas, 2024; Snyder et al., 2025; Chick & Inoue, 2001; Russell & Wefald, 1991; Hay et al., 2012; Callaway et al., 2018).

Factored fits usefulness from $[1, \operatorname{logit}\theta, m, \ell]$, then uses smoothed conditional keep means and a root mean. Direct fits each branch's terminal label from these four features plus fidelity, $\log(1+n)/\log 65$, and the evidence–fidelity interaction. Both use regularization 0.25. The θ estimate follows eight independent checks, and this calibration comparison evaluates candidate capacities from 22 to 30. The full planner uses capacity-dependent values throughout the online evidence loop; the single-decision comparison isolates value estimation from the choice of how many checks to purchase.

Reference predictions use unsmoothed *training-fold* prevalence separately for each branch. Brier skill is $1 - \mathrm{BS}/\mathrm{BS}_{\mathrm{ref}}$ on the same held-out outcomes. AUC counts score ties as one half and is NA for single-class samples. Ten fixed equal-width probability bins define ECE. Pooled reference AUC need not be 0.5 because prevalence differs across folds and branches; within each fold and branch, the reference is constant. The 2,000 paired origin-bootstrap samples retain all seeds, branches, and model predictions together. Intervals condition on fitted models and exclude refitting uncertainty.

Table 17: **Calibration by held-out family.** Scores combine keep and root records; intervals resample complete origin clusters while retaining all branch predictions.

Held-out family	Model	Brier [95% CI]	Skill	AUC	ECE
StackCube	Train prevalence	0.216 [0.204, 0.228]	0.000	0.486	0.077
	Factored	0.186 [0.172, 0.199]	0.140	0.667	0.018
	Direct	0.186 [0.170, 0.202]	0.137	0.656	0.041
PickPlaceCan	Train prevalence	0.201 [0.190, 0.213]	0.000	0.569	0.078
	Factored	0.178 [0.164, 0.193]	0.113	0.636	0.061
	Direct	0.178 [0.162, 0.195]	0.114	0.666	0.076
LIBERO-Object	Train prevalence	0.228 [0.214, 0.242]	0.000	0.567	0.038
	Factored	0.192 [0.175, 0.209]	0.157	0.688	0.019
	Direct	0.193 [0.177, 0.209]	0.151	0.701	0.055
LIBERO-Long	Train prevalence	0.242 [0.228, 0.256]	0.000	0.582	0.087
	Factored	0.215 [0.196, 0.235]	0.112	0.660	0.053
	Direct	0.210 [0.196, 0.225]	0.132	0.686	0.044

ROOTCAUSE-Direct improves ranking but does not improve branch-selection utility uniformly. Its keep-branch ECE is higher, while root-branch Brier improves. These differences separate probability quality, ranking, and planning value. Factored remains the ROOTCAUSE default; ROOTCAUSE-Direct measures the effect of richer joint conditioning within otherwise matched comparisons.

B.6 Ablations, action response, and coupled errors

Feature-removal variants are refit using the same development folds. The budget-blind comparison fixes the candidate-capacity input at the development reference allowance but still enforces the real

Table 18: **Calibration by branch.** Separate keep and root scores reveal differences hidden by pooling. Outer-fold metrics do not select between the two value models.

Branch	Model	Brier [95% CI]	Skill	AUC	ECE
Keep	Train prevalence	0.239 [0.232, 0.246]	0.000	0.412	0.062
	Factored	0.181 [0.169, 0.193]	0.242	0.727	0.030
	Direct	0.184 [0.172, 0.195]	0.232	0.758	0.070
Root	Train prevalence	0.205 [0.194, 0.215]	0.000	0.452	0.045
	Factored	0.205 [0.194, 0.215]	0.000	0.452	0.045
	Direct	0.201 [0.190, 0.211]	0.020	0.573	0.017

Table 19: **Single-decision branch-choice comparison.** The larger predicted keep/root value is selected after eight checks, with ties broken toward root. Both branch targets are available for paired utility comparisons at the same candidate allowance.

Value scoring	One-shot utility (%) [95% CI]
Train prevalence	71.58 [69.25, 74.00]
Factored	76.17 [73.83, 78.58]
Direct	75.58 [73.42, 77.92]
Direct − Factored (points)	-0.58 [-1.92, 0.67]

Table 20: **ROOTCAUSE value-model calibration.** Paired Direct-minus-Factored contrasts. Positive Brier or ECE differences favor Factored; positive skill or AUC differences favor Direct. Intervals condition on the LOFO fits.

Direct − Factored	Estimate	95% paired interval
AUC	0.0349	[0.0133, 0.0564]
Brier	-0.0008	[-0.0037, 0.0022]
Brier skill	0.0034	[-0.0102, 0.0167]
ECE	0.0131	[-0.0087, 0.0310]

Table 21: **ROOTCAUSE ablations at $b = 32$.** Differences are default minus each variant on matched origins. Direct replaces the value estimator; there is no fidelity-feature removal because Factored has no such input.

Configuration	Success	Δ [95% CI]	Pairs	Candidates
Factored (default)	84.6	0.0 [0.0, 0.0]	2.72	26.66
Without margin	84.6	0.0 [-1.0, 0.9]	2.77	26.63
Without retained fraction	84.1	0.5 [-0.1, 1.2]	2.75	26.63
Without budget dependence	84.2	0.4 [-1.2, 1.8]	3.54	25.64
Endpoint-local nomination	78.3	6.2 [4.2, 8.3]	2.77	26.63
Independent pair noise	83.9	0.7 [-0.7, 2.0]	2.74	26.64
Direct value model	84.2	0.4 [-1.6, 2.2]	2.20	27.12

online ledger. The endpoint-local ablation changes nomination while retaining the same diagnostic endpoint and charge for subsequent checks. The independent-noise comparison changes coupling between check arms while preserving each arm's marginal noise distribution. Paired success differences, check counts, and candidate counts distinguish comparison noise from planning value.

Table 22: **Test audit-bank sensitivity.** Independent suffixes assess returned branches without entering the decision rule. Bank size changes the usefulness assessment, not the chosen actions or their cost.

Test audit suffixes	Useful fraction	Agreement to 32	Outcome change	Cost change
16	65.4	93.9	0.0	0.0
32	69.0	100.0	0.0	0.0
64	68.7	98.2	0.0	0.0

Action-response premise. Matched-reset simulator execution supplies the reference response to an edit, with subsequent commands held fixed. The audit crosses edit scales of 0.25, 0.50, and 1.00 of the legal range with horizons $H/3$, $2H/3$, and H (rounded to native boundaries). Validity uses all attempted predictions. Direction agreement requires valid geometry and nonzero reference displacement; delayed-failure recall requires a reference delayed failure; edit-order agreement requires an available reference ordering. We report agreement and eligibility coverage over all attempted predictions. Invalid predictions count as failed checks and remain in planning success and cost.

Table 23: **Action response across horizon and edit amplitude.** Conditional agreement and coverage (cov.) accompany total validity. Scores compare predicted videos with matched executions.

Horizon	Edit scale	Valid	Direction / cov.	Delay / cov.	Ordering / cov.
$H/3$	0.25	**96.8**	89.8 / 88.8	86.1 / 55.3	**92.7 / 92.5**
$H/3$	0.50	95.1	**90.7 / 86.3**	**86.3 / 51.7**	91.3 / 91.6
$H/3$	1.00	93.6	88.1 / 84.5	82.4 / 48.7	88.5 / 90.2
$2H/3$	0.25	94.8	88.5 / 87.3	84.4 / 50.3	89.1 / 91.7
$2H/3$	0.50	93.8	85.0 / 84.2	82.1 / 52.2	88.5 / 90.3
$2H/3$	1.00	92.1	81.2 / 83.4	77.8 / 49.7	86.2 / 88.5
H	0.25	93.2	82.4 / 84.0	77.9 / 51.1	88.2 / 89.1
H	0.50	93.2	80.0 / 85.2	76.3 / 46.7	84.6 / 88.8
H	1.00	90.4	75.5 / 80.9	73.0 / 50.1	82.6 / 86.8

Table 24: **Coupled-error cohort.** Set hit measures whether a returned site belongs to the injected-site set, not whether every error is identified. Success, coverage, and reviewed joint-error cases are reported.

Planner	Success	Set hit	Coverage	Joint-edit failures
MPPI	65.0	–	–	28
Geo-gated adapter	68.8	–	–	30
ROOTCAUSE-F2	67.5	58.7	92.9	21
ROOTCAUSE-F4	65.4	63.3	91.7	26
ROOTCAUSE	**75.0**	**66.7**	**94.6**	20
ROOTCAUSE+Geo	73.8	60.0	92.5	**18**

Qualitative comparisons. Figure 5 sketches the four outcomes a paired comparison can have. The early and local repairs reach the same target through different retained constraints. The other two separate a persuasive prediction from a useful intervention: an edit can be accepted yet miss the target, or be rejected despite a successful reference path. These distinctions motivate calibrating the final branch decision instead of interpreting every passing comparison as evidence of successful execution.

B.7 Full-pipeline runtime and statistical reporting

Timing boundary. Warm decision latency starts when RGB history and the goal are ready and ends when the selected action plan is returned. Loading, compilation, and cold-start latency are excluded from this warm-decision boundary. Timed spans include world-model inference, Grounding DINO, SAM 2 initialization and prompts, fully consumed video propagation, geometry, nomination,

Table 25: **ROOTCAUSE failures on the coupled-error cohort.** Each failed origin–seed run was categorized by inspecting its logged decision trace together with its execution outcome. Categories distinguish attribution, prediction, ranking, and interaction errors; fractions condition on failed runs, and unresolved cases remain in the denominator.

Failure category	Count	Fraction (%)
Coupled pre-symptom edits	20	33.3
World-model mismatch	11	18.3
Wrong nomination	12	20.0
Final ranking error	10	16.7
Search-budget exhaustion	4	6.7
Unresolved	3	5.0

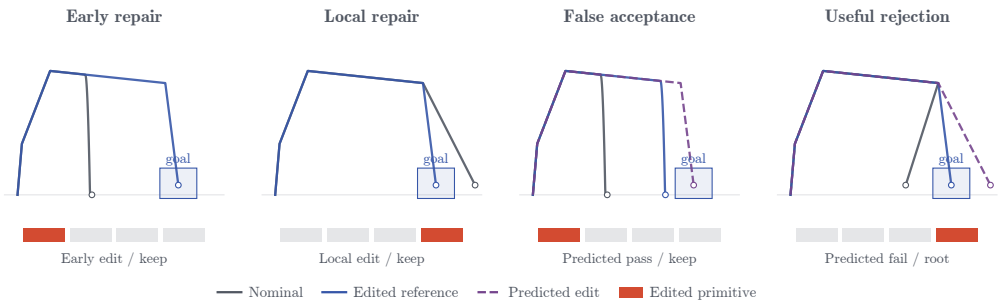

Figure 5: **Schematic: the same comparison can err in either direction.** Gray paths stand for the nominal plan, blue solid paths for the edited plan's reference execution, and purple dashed paths for the edited plan's prediction; shaded boxes mark target regions, and the red block on each four-marker timeline locates the intervention. An accepted prediction may miss in the reference, while a rejected prediction may conceal a useful edit.

posterior integration, Bellman decisions, candidate gating, and ranking. Each generated branch has an independent SAM 2 tracking state. Failed calls, rejected candidates, both check arms, and all views remain inside the boundary. Robot execution time is separate.

Table 26: **Complete-decision latency at $b = 32$.** The timing boundary covers perception, generation, checking, and final ranking. Comparisons share precision, batching, and warm-up settings.

Planner	Mean (s)	p50 (s)	p90 (s)	Blocks	SAM 2 (s)
Full-root	2443.9	2168.1	4086.6	351.9	34.0
Continuous CEM	2444.0	2168.1	4089.7	351.9	34.0
MPPI	2443.8	2168.1	4090.7	351.9	34.0
WAP-style BoN	2443.4	2162.9	4090.7	351.8	34.0
TBS-style adapter	2443.6	2168.1	4090.7	351.8	34.0
Paired branch search	2415.0	2130.0	4033.1	347.8	33.6
Geo-gated adapter	2445.3	2169.7	4091.2	351.9	34.0
ROOTCAUSE-NoChk	2443.4	2162.9	4089.6	351.8	34.0
ROOTCAUSE-F1	2414.4	2128.5	4031.6	347.7	33.6
ROOTCAUSE-F2	2415.3	2126.8	4032.8	347.8	33.6
ROOTCAUSE-F4	2416.5	2140.5	4037.3	348.0	33.6
ROOTCAUSE	2417.9	2133.0	4033.5	348.2	33.6
ROOTCAUSE+Geo	2419.7	2134.2	4034.9	348.2	33.7

Table 27: **ROOTCAUSE latency breakdown.** Exclusive component time separates video generation, visual tracking, geometric scoring, and the decision rule without double-counting nested spans.

Timed component	Mean exclusive (s)	Coverage
World-model inference / RPC	2383.48	100.0
Grounding DINO	0.24	100.0
SAM 2: initialize + prompt	4.91	100.0
SAM 2: consume propagation	28.74	100.0
Geometry + nomination	0.31	100.0
Posterior + Bellman	0.02	100.0
Candidate gate + final ranking	0.21	100.0
Other / unassigned	0.04	100.0

Decision spans give mean, median, and 90th-percentile latency; component percentiles are not additive. GPU work is synchronized at measurement boundaries or timed with device events. Overlapping spans are reconciled before attributing component costs. Offline costs cover reference-bank executions, paired evidence, candidate banks, and model fitting.

Aggregation and missingness. Seeds are averaged within origin, origins within family, and families equally. Paired bootstrap draws resample origins within families, retaining every method, seed, and factorial cell. Conditional rates recompute both counts in each replicate; failed executions remain in the applicable denominator. Undefined rates are NA. Primary comparisons are prespecified; secondary intervals are descriptive and uncorrected for post-selection. Family scores accompany macro averages to expose gains concentrated in a single task group.