
Measuring Failure Yield and Reflection Cost in a Frozen Robot Policy

Qinzheng Ma¹ Shichen Tang¹ Qingxi Meng¹ Jialin Wu²
¹Rice University ²UC San Diego
qm18@rice.edu

Abstract

We measure failed rollouts and the resource cost of structured reflection for a frozen Action Chunking with Transformers (ACT) policy in MuJoCo bimanual cube transfer. Five primary methods and three post-primary controls each use five search seeds and 64 rollouts per run. In a retrospective analysis, reflection averages 20.4 failures versus 17.6 for otherwise matched language-guided selection: a paired difference of 2.8 ± 1.92 (sample standard deviation), with 25.6% more measured time and 6.85% more tokens. The comparison with inexpensive search is less favorable. Sequential QD averages 18.8 failures; a subsequently frozen nearest-neighbor selector using past failure labels averages 29.0 ± 12.49 , with 166.1 s/run and no model calls, versus 315.8 s/run for reflection. All 140 repeat executions of 70 unique primary archive scenes remain failures. These observations show that an increment over language selection need not justify reflection’s cost against inexpensive alternatives. They do not establish statistical superiority, cross-task generalization, or improved policy robustness.

1 Question and contribution

Failure-search procedures expose unsuccessful robot tests under limited simulation budgets. We ask: *does structured reflection and persistent language memory increase failed rollouts over otherwise matched language selection, and is that increment competitive with inexpensive feedback-based selection?* Each unsuccessful rollout contributes one occurrence to this count, whether or not similar failures have already been observed.

Our contribution is an empirical assessment of adding structured reflection to an existing adaptive robot-test generator. We compare failed tests and measured resource costs against both the same language selector without reflection and inexpensive search alternatives. The failure-count analysis is retrospective; the original descriptor-guided archive and prompts remain unchanged. Thus the question concerns the measured consequences of adding reflection to this particular generator, not the best achievable performance of a system optimized for failure count. Candidate distributions adapt from feedback; policy weights and dynamics remain fixed. We retain all methods and seeds and check persistence of selected failures, without policy training or repair.

Quality-diversity (QD) failure testing is established. MAP-Elites retains representatives in user-defined niches [5]; Mazouni et al. directly search initial conditions for diverse reinforcement-learning policy faults [4]. Contextual failure discovery uses Bayesian active learning and expert feedback [6], while RADIUM studies robot failure prediction and repair through gradient-accelerated sampling [1]. Here simulator reward and geometry provide the outcome oracle, with a frozen policy and no repair stage.

Rainbow Teaming generates diverse adversarial language prompts [7], whereas our model selects from physically validated numerical candidates and does not generate actions or judge success. Reflexion motivates feedback-conditioned text memory without weight updates [8]. Our empirical

distinction is the matched reflection comparison and its cost accounting for a frozen visual ACT policy [9], rather than a new QD algorithm or general language-agent capability claim.

2 Failure search with a frozen policy

Task and validity. We use the public `act_aloha_sim_transfer_cube_human` ACT checkpoint [3] in `gym_aloha/AlohaTransferCube-v0` [2]. Scenarios are cube positions $s = (x, y) \in [0, 0.2) \times [0.4, 0.6)$ meters, the native initialization support; height is 0.05 m and orientation is fixed. Robot initialization, camera, mass, and friction are unchanged. Each proposal receives a real reset and checks for finite state, tabletop containment, penetration no deeper than 0.1 mm, and absence of initial success. The rollout reset is checked against the requested pose. Sampling adaptively within this support changes its distribution, so search failure rates do not estimate native deployment risk.

ACT receives a 480×640 RGB top view and 14-dimensional robot state. Its 14-dimensional commands are absolute arm joint positions and normalized gripper controls. The preserved checkpoint predicts and executes 100-action chunks before replanning. Evaluation uses at most 400 steps at 50 Hz, ending at native success. The 400-step horizon follows checkpoint configuration and differs from the Gym registration’s default. No training or parameter tuning occurs.

Failure count and search archive. Let $R(\tau) = \max_t r_t$ for native reward $r_t \in \{0, 1, 2, 3, 4\}$. A valid failure has $R < 4$ and severity $q(\tau) = 1 - R(\tau)/4$. The evaluation measure is $F = \sum_{i=1}^{64} \mathbf{1}[\text{valid}_i \wedge R(\tau_i) < 4]$, the number of failed rollouts in a search run. We count from the complete rollout log, including failures that do not replace an archive entry. Each rollout contributes at most one failure, regardless of how many unsuccessful motions it contains. Invalid scenes and system errors are excluded; F does not count distinct failure types. Native success requires designated left-finger contact with the cube and no table contact; it does not establish sustained stable handover. The descriptor combines maximum cube-center height (including reset), with thresholds (0.06, 0.15) m, and final cube-center x , with thresholds $(-0.05, 0.05)$ m. These coarse design choices were fixed before primary search; equality enters the upper bin. Each cell retains the highest-severity failure, keeping earlier observations on ties. This archive is an internal candidate-generation mechanism; archive occupancy and severity sums are not evaluation outcomes in this paper.

Methods and intervention. All methods share eight uniform initial scenarios per seed. Thereafter, *uniform* samples the domain; *greedy* mutates the most severe observed failure; *sequential QD* uniformly chooses an occupied-cell elite to mutate. Mutations have independent coordinate SD 0.025 m, 20% uniform restarts, and clipping to the half-open domain. An empty archive triggers uniform sampling. Identifiers hash positions rounded to nine decimals to reject duplicates. Every completed run uses distinct requested positions at this precision, so repeated execution of one identical failed request cannot inflate F . Nearby requests can still expose the same weakness.

The two language conditions construct a pool of 24 valid QD proposals and select eight with one model call, then execute that frozen batch before constructing the next pool. Both receive the protocol, remaining budget, observed history, archive, and candidate states; neither sees unevaluated outcomes or validation results. Every proposed candidate, including the 16 unselected ones, is marked seen. Seven calls complete each run. The original prompts target discovering unoccupied valid-failure behavior cells; they were not rewritten or optimized for this retrospective failure-count analysis. *LLM selection* emits a short rationale that is logged but not reused. *LLM + reflection* emits hypothesis, evidence, uncertainty, and next-test fields, retaining them for later calls. This is a combined structured-generation and memory intervention, not a memory-only ablation or a multi-agent comparison.

3 Experiment and results

Protocol and costs. The frozen primary design contains five methods, seeds 1201–1205, and 64 rollouts per run: 1,600 completed rollouts and 509,336 control transitions. Within each seed, method order rotates; one GPU workload runs at a time. Setup and warmup are excluded. Wall time includes evaluation, geometry resets, visualization, and provider waits. Sequential methods use 64 proposal prechecks per run, whereas language methods use 176. Thus rollout budgets match, but candidate

Table 1: Failure occurrences and resource use. Mean $\pm$ sample SD over five runs per method; F is the number of failed rollouts out of 64. Tokens/run reports the mean only. Added controls are post-primary experiments.

Method	F	Time (s/run)	Tokens/run
Uniform	11.40 ± 5.18	145.7 ± 9.9	0
Greedy	17.00 ± 6.78	147.7 ± 10.7	0
Sequential QD	18.80 ± 8.14	147.8 ± 10.8	0
LLM selection	17.60 ± 3.71	251.5 ± 5.5	123,032
LLM + reflection	20.40 ± 5.08	315.8 ± 12.2	131,457
Batch random (added)	17.40 ± 6.91	157.6 ± 10.1	0
Geometric (added)	13.00 ± 2.00	150.1 ± 3.1	0
Failure k NN (added)	29.00 ± 12.49	166.1 ± 12.0	0

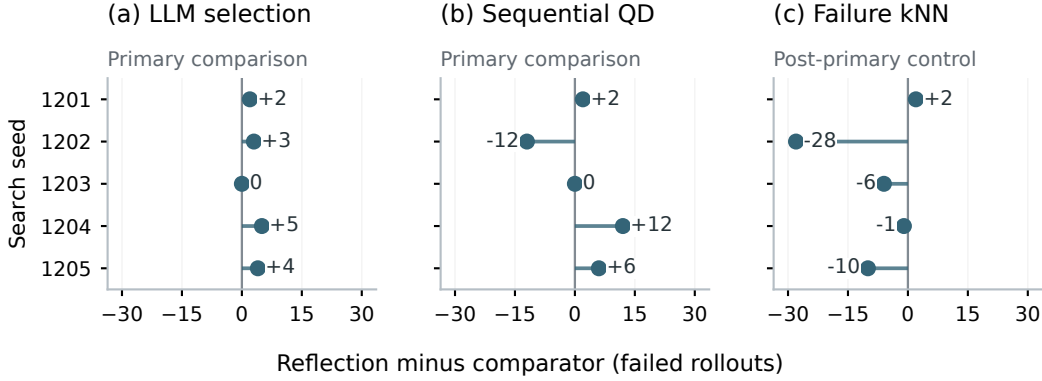


Figure 1: Paired failure-count differences at 64 rollouts. Each point is one search seed; all panels use the same horizontal scale. The k NN comparison uses a separately executed post-primary control. Lines connect observed differences to zero and do not represent uncertainty intervals.

screening, wall time, and tokens do not. Requests specify gpt-6-astra, medium effort; the serving revision was unavailable. All 70 calls completed without tool use, retry, or fallback. Appendix A provides runtime and provenance.

Failure yield and measured cost. The statistical unit is a complete search run; adaptive episodes are not independent replicates. Table 1 reports dispersion, not confidence intervals. Reflection finds 20.4 failures per 64 rollouts versus 17.6 for selection without reflection. Paired differences are $[2, 3, 0, 5, 4]$, averaging 2.8 ± 1.92 failures (Figure 1); four pairs favor reflection and one ties. All 40 initialization inputs and success flags match between these conditions, so the entire count difference arises in rollouts 9–64. This excludes an initial failure-count imbalance, without isolating a pure memory effect.

Sequential QD averages 18.8 failures. Reflection-minus-QD differences are $[2, -12, 0, 12, 6]$, or 1.6 ± 8.88 failures, while reflection takes 2.14 times as much mean recorded run time. The large between-seed variation and small mean count gap do not establish a reliable advantage over this inexpensive comparator. Retrospective prefix counts further show that the reflection-minus-selection mean difference is -0.4 at 16 rollouts, zero at 24 and 32, and 2.8 at 64. All eight specified prefix lengths are retained in Appendix H; they are nested observations, not independent replications or new stopping-rule experiments.

Reflection adds 64.4 s/run of observed end-to-end search time: 25.6% more time and 6.85% more reported tokens, ratios of method means. Recorded model-call intervals account for most of the time difference (Appendix A). Aggregated failures divided by recorded run time give 4.20 failures/minute for selection and 3.88 for reflection: higher count at fixed rollout budget does not imply higher measured time efficiency. These are descriptive ratios, not results from a fixed-wall-time experiment.

Matched batch controls. Sequential baselines differ from language conditions in candidate-pool construction and feedback timing. We added three separately frozen controls after primary outcomes were known, each with five runs of 64 rollouts (960 additional rollouts). All build 24 QD proposals and freeze eight selections per batch. *Batch random* samples eight without replacement; *geometric greedy* maximizes minimum normalized positional distance to evaluated and already selected points. They average 17.4 and 13.0 failures. *Failure kNN* instead ranks candidates by the failure fraction among their five nearest previously executed positions, including successes and failures. Its fixed distance and tie rules use no unobserved labels (Appendix F).

Failure *kNN* averages 29.0 ± 12.49 failures and 166.1 ± 12.0 s/run without language calls. Reflection-minus-*kNN* differences are $[2, -28, -6, -1, -10]$, averaging -8.6 ± 11.78 , while reflection takes 1.90 times as much mean recorded run time. Thus reflection’s increment over language selection does not translate into the highest observed failure count or measured time efficiency. All controls finish without invalid scenes or system errors. They match the pool/batch procedure and precheck counts, but execute in separate phases and can acquire different adaptive pools; their gaps do not isolate language reasoning’s causal effect.

Native performance and selected-scene reproduction. A predeclared validation rule uses 128 native resets independently of search and two repeat policy executions of every unique final primary archive scene, without pruning. Native resets produce 27 failures out of 128 tests (21.1%; 95% Wilson interval 14.9–29.0%). The archive union has 70 unique scenes across 109 memberships. All 140 repeat executions remain failures. These are selected-scene reproducibility counts, not an IID estimate of deployment failure probability. No validation outcomes informed primary search.

4 Limitations and conclusion

The evidence concerns one checkpoint, task, two-dimensional position domain, and requested provider configuration. It does not measure dynamics shifts, cross-policy generalization, or real hardware. The native reward criterion does not prove a sustained stable handover. Textual hypotheses are not verified causes. ACT uses deterministic evaluation settings, but tiny rendered-image differences can occur at identical physical states; their downstream influence is not fully isolated. Policy seeds are provenance, not independent stochastic-policy draws. Unknown serving revision and external-service latency limit exact replication of language behavior and timing.

Structured reflection found more failed rollouts than language selection in the primary runs, at higher time and token cost. However, a cheap failure-label selector achieved a higher mean count with less measured time in the added phase. This retrospective case study therefore does not support a practical advantage for reflection over the tested inexpensive alternatives. Five runs per method, separate execution phases, and one task limit that conclusion. A prospective failure-count objective for both language conditions, broader tasks, and policy adaptation on discovered failures remain future work.

References

- [1] Charles Dawson, Anjali Parashar, and Chuchu Fan. RADIUM: Predicting and repairing end-to-end robot failures using gradient-accelerated sampling, 2024. URL <https://arxiv.org/abs/2404.03412v1>. Preprint version 1; a later journal publication exists.
- [2] Hugging Face. gym-aloha: A gym environment for ALOHA. <https://github.com/huggingface/gym-aloha>, n.d. Version 0.1.4 used; accessed 2026-09-13.
- [3] Hugging Face LeRobot. ACT/ALOHA transfer cube checkpoint: act_aloha_sim_transfer_cube_human. https://huggingface.co/lerobot/act_aloha_sim_transfer_cube_human, n.d. Revision ba73b2766f1371cdc133ca4efb97eb090d744625; accessed 2026-09-13.
- [4] Quentin Mazouni, Helge Spieker, Arnaud Gotlieb, and Mathieu Acher. Testing for fault diversity in reinforcement learning. In *Proceedings of the 5th ACM/IEEE International Conference on Automation of Software Test*, 2024. doi: 10.1145/3644032.3644458. URL <https://arxiv.org/abs/2403.15065>.
- [5] Jean-Baptiste Mouret and Jeff Clune. Illuminating search spaces by mapping elites, 2015. URL <https://arxiv.org/abs/1504.04909>.
- [6] Anjali Parashar, Joseph Zhang, Yingke Li, and Chuchu Fan. Cost-aware discovery of contextual failures using bayesian active learning. In Joseph Lim, Shuran Song, and Hae-Won Park, editors, *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pages 2239–2267. PMLR, 2025. URL <https://proceedings.mlr.press/v305/parashar25a.html>.
- [7] Mikayel Samvelyan, Sharath Chandra Raparthy, Andrei Lupu, Eric Hambro, Aram H. Markosyan, Manish Bhatt, Yuning Mao, Minqi Jiang, Jack Parker-Holder, Jakob Foerster, Tim Rocktäschel, and Roberta Raileanu. Rainbow teaming: Open-ended generation of diverse adversarial prompts. In *Advances in Neural Information Processing Systems*, volume 37, pages 69747–69786. Curran Associates, Inc., 2024. doi: 10.52202/079017-2229. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/8147a43d030b43a01020774ae1d3e3bb-Paper-Conference.pdf.
- [8] Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, pages 8634–8652. Curran Associates, Inc., 2023. doi: 10.52202/075280-0377. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/1b44b878bb782e6954cd888628510e90-Paper-Conference.pdf.
- [9] Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware, 2023. URL <https://arxiv.org/abs/2304.13705>.

A Execution and reproducibility

Fixed policy and environment. Checkpoint migration preserves all 234 policy state tensors, including parameters and buffers, and the external normalization statistics; loading into LeRobot is strict. The model has 51,613,582 parameters. Redundant ImageNet backbone initialization is skipped before the full checkpoint is loaded; no trained weights are omitted. A zero latent and disabled dropout are used at evaluation, without temporal ensembling or action clipping that would corrupt absolute joint positions. Exact immutable identities are:

Hugging Face revision	ba73b2766f1371cdc133ca4efb97eb090d744625
LeRobot commit	675c07664279659525b1d1e52b6e33538c957ee3

Execution uses Windows, Python 3.12.14, PyTorch 2.10.0+cu128, gym-aloha 0.1.4, MuJoCo 3.8.1, dm-control 1.0.41, and an RTX 5090 Laptop GPU. Computation is float32, TF32 is disabled, and one CPU thread is used. The monitor reuses native observer images at up to 5 Hz without modifying policy inputs or introducing real-time pacing. Waiting for the language provider is explicitly shown as a paused phase. Post-completion display retention is excluded from run timing.

Native reset and valid proposals. The environment performs its native reset before only the named cube free joint is edited and physics is forwarded. Robot joints, velocities, and control targets are preserved. The actual seven-dimensional pose is compared with the request during both precheck and rollout reset. Checks include full cube-footprint containment in the tabletop mesh, finite physical state, native-domain bounds, initial contact penetration tolerance, and no reward-4 start. Invalid proposals and infrastructure failures cannot enter the physical-failure archive. No invalid scene or system error was recorded in the completed primary and 268-rollout validation phases.

Language and provenance. Candidate identifiers and uniqueness are validated locally against the supplied pool. A malformed response or provider failure stops the run with its costs retained, without retries or fallback choices. Prompts, schemas, receipts, responses, and runtime manifests are preserved locally. All 70 prompt hashes match after normalizing Windows CRLF to the LF text sent to the provider. Reasoning-output tokens are a subset of output tokens and are not added twice. Appendix E gives schema and memory examples.

Table 2: Actual aggregate language usage across 35 calls per condition. Both conditions use seven calls per search run.

Condition	Input	Output	Total
LLM selection	604,496	10,662	615,158
LLM + reflection	636,898	20,386	657,284

Timing decomposition. Per-run mean sums of model-call elapsed time are 100.316 s for selection and 162.874 s for reflection; mean sums of episode elapsed time are 148.856 and 150.626 s respectively. Total means are 251.486 and 315.838 s, with the small remainder accounting for other search work. The mean numbers of control transitions are 20,612.0 and 20,794.4. Model-call intervals include local bridge startup, network/service waiting and execution, and response validation; they do not isolate provider compute. The 64.352 s paired mean total difference decomposes into 62.558 s for calls, 1.770 s for episodes, and 0.024 s for other work. This does not establish a hardware-independent or service-independent latency penalty for reflection.

Validation definition. The native-reset seed range is 920000–920127, with fixed policy seed 940000. Final primary elites are deduplicated by exact candidate identity, sorted by identifier, and each receives episode seeds $930000 + 2i + r$ for scene index i and repetition $r \in \{0, 1\}$. All 70 scenes are retained, even when multiple methods discovered them. The 140 executions reproduce the saved exact starting scenarios; they are not neighborhood perturbations or independent stochastic policies. Both executions fail for all 70 scenes. Membership-expanded counts would duplicate shared scenes and are not used as independent observations.

B Statistical unit and interpretation

Five complete search runs per condition provide descriptive paired observations. A shared seed aligns initialization, not later adaptive trajectories. Repeated positions across methods do not become independent scenarios by being reevaluated. We make no significance, superiority, equivalence, or power claim from these five observations. The Wilson interval for native failure frequency is the only binomial interval reported: it applies to native initial-state draws, not adaptively selected failures or archive memberships. Every run’s failure count and costs are listed in Table 4.

C Search procedure

Initialize empty history, archive, and proposed-identifier set. For the first eight trials, sample a valid native-domain position and evaluate it. Thereafter, a sequential method generates and evaluates one candidate per step. A batch method generates 24 unique valid QD proposals, marks all proposed identifiers seen, and chooses eight distinct identifiers without seeing their outcomes. Execute the eight selected scenarios in order, record all transitions and resource counters, and update the archive after each observed failure. Construct the next candidate pool only after the entire selected batch finishes. Stop at exactly 64 counted rollouts or record and stop on an infrastructure error. Successes and invalid outcomes never become failure elites.

D AI use and scope of responsibility

Language models are used as an experimental selection mechanism under the recorded configurations above. AI assistance was also used for implementation, analysis scripts, literature checking, manuscript drafting, and editing. The manuscript’s numerical statements are tied to executed logs and recomputed summaries; model-generated explanations are not treated as experimental evidence.

E Artifact status and language-call details

Artifact availability. The appendices specify the search procedures, all run-level failure counts and costs, and fixed-prefix counts. A separate anonymized evidence package containing outcomes, protocols, sanitized calls, and offline analysis code is prepared, but no reviewer-accessible URL is available at this writing. Private provider code and credentials are excluded; the package alone cannot regenerate language-model responses. The serving revision remains unknown.

Selection contract. Both conditions receive 24 validated candidates, the current archive, compact observed history, and remaining budget. Their JSON schemas require `candidate_ids` (exactly eight identifiers from the supplied enumeration) and `evidence_sha256` (the input digest). Additional output keys are forbidden; identifier uniqueness and digest equality are checked locally. Selection additionally requires a 1–500-character `rationale`. Reflection instead requires a nested `reflection` object containing exactly four 1–1,200-character strings: `hypothesis`, `evidence`, `uncertainty`, and `next_test`. These are output fields, not verified causal annotations. The shared prompt includes: “Infer no unobserved candidate outcomes.” Tools and changes to the protocol are prohibited.

Observed memory transmission. Table 3 describes the actual first two calls for search seed 1201. The suffix denotes completed rollouts before selection; equal row counts do not imply identical histories. In reflection, the second input’s sole memory object exactly equals the first response’s four-field reflection. In selection, the `memories` field is absent in both inputs, and the first `rationale` is absent from the second input.

Table 3: Actual input structure for four recorded calls; no additional experiment.

Condition / call suffix	History rows	Pool size	Prior prose memory
Selection / 008	8	24	Field absent
Selection / 016	16	24	Field absent
Reflection / 008	8	24	Empty list
Reflection / 016	16	24	One four-field object

Verbatim model-text excerpts. The following short excerpts are from reflection call 008; the complete object, including unquoted sentences, was passed to call 016:

Hypothesis: “Sampling near observed success/failure transitions and extending initial-position coverage may reveal failures in unoccupied behavioral cells.”

Evidence: “Eight observed trials produced four successes in behavioral cell 2,1 and four failures occupying only 0,2 and 2,2.”

Uncertainty: “No selected candidate has an observed outcome.”

Next test: “Evaluate these eight distinct scenarios with the fixed protocol.”

Selection call 008 instead stated, within its nonpersistent `rationale`: “spatial diversity does not establish behavioral diversity, and selected outcomes remain unknown.” These excerpts illustrate the implemented information flow, not validated explanations or a reasoning-quality score. The intervention jointly changes structured generation and persistent memory; it does not isolate memory’s causal contribution.

F Post-primary controls and matching limits

All three additional controls were designed after primary results were known. Each design and its implementation hashes were frozen before its own execution. They use the original evaluator, checkpoint, proposal generator, failure descriptor, severity, and archive update without modifying the primary implementation or results. Five search seeds (1201–1205), eight initial uniform scenarios, and 64 rollouts per run are retained. Each run builds seven pools of 24 proposals and selects eight per pool, with 176 geometric prechecks and all 24 candidates marked seen. The full batch is selected before any of its outcomes are observed; the next pool uses the resulting archive. No language calls or automatic retries are used.

Random batch selection. The proposal generator retains `default_rng(search_seed)`. Uniform selection without replacement uses an independent generator initialized by `SeedSequence([search_seed, 3028025345])`, so selection draws never consume the proposal stream. The ordered selection, entire pool, and pre-selection history/archive digests are written before executing the batch.

Geometric batch selection. Normalize position as $u(s) = (x/0.2, (y - 0.4)/0.2)$. Let H contain *all* already evaluated requested positions, including successes and failures; initially the shared eight trials supply H . For remaining pool P and current batch selection B , select

$$s^* = \arg \max_{s \in P} \min_{r \in H \cup B} \|u(s) - u(r)\|_2^2,$$

add s^* to B , remove it from P , and repeat until eight are chosen. Exact computed distance ties use ascending candidate identifier. There is no random or artificial starting anchor. The selector reads positions and identifiers without reward or behavior, although the upstream QD proposal generator uses the failure archive. Thus only selection conditional on the pool is outcome-blind.

Failure-probability batch selection. This control uses the same normalized position $u(s)$. Let $N_5(s; H)$ be the five nearest previously executed valid trials in this run’s history, including both successes and failures. It ranks each candidate by

$$\hat{p}_F(s) = \frac{1}{5} \sum_{j \in N_5(s; H)} \mathbf{1}[R(\tau_j) < 4].$$

The shared eight initial trials provide sufficient history. Neighbors are ordered by squared Euclidean distance and then ascending candidate identifier; candidates are ordered by descending score and then ascending identifier. The full eight-candidate batch is fixed using the same observed history, without adding imagined labels for its selections. The fixed $k = 5$, normalization, and tie rules were chosen before this control ran, without tuning on its results. These scores are ranking heuristics, not calibrated probability estimates. The selector uses only past binary outcomes and positions; the upstream QD proposal generator remains unchanged.

Initialization and phase matching. All 40 initial requested/actual states and episode/policy seeds per control exactly match primary inputs. Rendered observations do not always match. Relative to primary uniform search, initial image hashes differ in 11/40 random-control and 10/40 geometric-control trials. Initial outcome signatures (status, success, maximum reward, cell) differ from primary LLM selection in two and one trials respectively, and from reflection in one and zero trials. The first pool matches both language conditions in four of five seeds for batch random and all five for geometric selection. At random-control seed 1205, trial eight changes from the agents’ cell (2, 1) to (1, 1) with the same maximum reward of 2, altering the initial archive. All observations are retained without favorable reruns or seed removal.

For failure k NN, initial image hashes differ from primary uniform search in 10/40 trials and from reflection in 16/40. Initial outcome signatures differ from LLM selection in one trial and from reflection in zero trials; all five first pools match both language conditions. Every control retains the same initial failure counts [4, 1, 0, 0, 5]. These checks establish initial input and failure-count alignment, not identical images or later trajectories.

Later pools can diverge because selectors choose different positions and acquire different archives. The procedures, rollout budgets, and precheck counts match; this does not establish identical downstream pools, observations, tokens, or wall time. Controls execute sequentially after the primary

study, rather than being randomized within its timing schedule. These limits prevent interpreting the observed gaps as perfectly isolated causal effects of language reasoning.

Completed results and integrity. Each control completes all 320 rollouts with no invalid scene or infrastructure error. Batch random finds 87 physical failures; geometric selection finds 65; failure k NN finds 145. Independent CPU checks reconstruct all 105 batch selections, including each geometric distance calculation and each k NN neighbor/score ordering, verify all 960 numerical traces, and recompute failure counts, rewards, and the final archives used by the search mechanism. Primary and predecessor files retain their recorded hashes. The original 140 re-executions validate only the 70-scene *primary* archive union; added-control elites have not received that repeat-execution validation. Table 4 gives every search run, distinguishing primary and added-control phases.

G Complete numerical run results

Table 4 retains every completed search run. Each uses 64 rollouts. Added controls were designed after primary outcomes were known and executed in separate phases; E3 was designed after E1 and E2. Wall times use the end-to-end scope in Appendix A. Values are rounded only for display.

Table 4: All 40 search runs. F : valid native-task failure occurrences among 64 rollouts; tokens: reported input plus output.

Method	Seed	F	Time (s)	Tokens
<i>Primary methods</i>				
Uniform	1201	15	157.08	0
	1202	13	143.90	0
	1203	7	141.47	0
	1204	5	132.28	0
	1205	17	153.53	0
Greedy	1201	18	160.30	0
	1202	24	151.58	0
	1203	14	144.93	0
	1204	7	131.20	0
	1205	22	150.43	0
Sequential QD	1201	19	160.41	0
	1202	31	157.33	0
	1203	13	141.10	0
	1204	10	135.02	0
	1205	21	145.11	0
LLM selection	1201	19	243.99	124,447
	1202	16	254.21	123,916
	1203	13	253.19	121,806
	1204	17	257.98	122,733
	1205	23	248.05	122,256
LLM + reflection	1201	21	324.25	133,354
	1202	19	325.24	133,011
	1203	13	295.86	129,113
	1204	22	312.80	129,872
	1205	27	321.04	131,934
<i>Post-primary controls</i>				
Batch random (E1)	1201	16	161.41	0
	1202	26	172.71	0
	1203	7	145.92	0
	1204	18	153.86	0
	1205	20	153.99	0
Geometric (E2)	1201	15	152.34	0
	1202	14	153.06	0
	1203	10	151.00	0
	1204	14	148.61	0
	1205	12	145.39	0
Failure kNN (E3)	1201	19	162.80	0
	1202	47	186.01	0
	1203	19	158.81	0
	1204	23	155.57	0
	1205	37	167.16	0

H Fixed-budget failure-count diagnostic

This diagnostic was specified after the original seven-method results were known. All five search seeds and the fixed prefixes 8, 16, 24, 32, 40, 48, 56, and 64 are retained; no favorable budget is selected. The first eight failure counts are $[4, 1, 0, 0, 5]$ for both language conditions, with all 40 initial requested/actual states, candidate identifiers, episode/policy seeds, and success flags matching. Their paired difference over trials 9–64 is $[2, 3, 0, 5, 4]$, exactly the full-budget difference. This rules out an additive initial failure-count imbalance, but does not isolate the causal effect of memory or reflection.

Prefixes are nested views of the same five complete runs, not independent samples. Every prefix truncates a run planned for 64 rollouts; the language conditions received the remaining budget of that 64-rollout run. These are not fresh runs configured for the smaller prefix budgets. Entries are cumulative physical failure occurrences, mean $\pm$ sample SD. Only full-run measured wall time is available for the cost comparison; no prefix time or equal-time result is inferred.

The failure-probability control was designed after these seven-method outcomes; its identical analysis rule was fixed before its execution. It is reported as a separate post-primary control.

Table 5: All fixed-prefix failure counts for the five primary methods. Mean $\pm$ sample SD across five search seeds.

Prefix (rollouts)	Uniform	Greedy	Sequential QD	LLM selection	LLM + reflection
8	2.00 $\pm$ 2.35	2.00 $\pm$ 2.35	2.00 $\pm$ 2.35	2.00 $\pm$ 2.35	2.00 $\pm$ 2.35
16	2.80 $\pm$ 3.11	3.20 $\pm$ 3.11	3.20 $\pm$ 3.11	4.00 $\pm$ 3.16	3.60 $\pm$ 3.36
24	4.20 $\pm$ 3.42	5.60 $\pm$ 4.93	6.40 $\pm$ 4.62	5.40 $\pm$ 4.34	5.40 $\pm$ 4.28
32	5.20 $\pm$ 4.02	7.80 $\pm$ 5.26	8.00 $\pm$ 4.90	7.40 $\pm$ 4.93	7.40 $\pm$ 3.78
40	7.60 $\pm$ 4.77	10.20 $\pm$ 5.07	11.40 $\pm$ 6.69	10.00 $\pm$ 4.90	11.00 $\pm$ 4.64
48	9.00 $\pm$ 5.34	12.40 $\pm$ 5.55	13.20 $\pm$ 7.05	12.40 $\pm$ 5.13	14.60 $\pm$ 4.16
56	9.60 $\pm$ 5.32	14.20 $\pm$ 6.22	15.20 $\pm$ 6.69	15.20 $\pm$ 4.66	17.40 $\pm$ 4.04
64	11.40 $\pm$ 5.18	17.00 $\pm$ 6.78	18.80 $\pm$ 8.14	17.60 $\pm$ 3.71	20.40 $\pm$ 5.08

Table 6: All fixed-prefix failure counts for the separately executed post-primary controls. Mean $\pm$ sample SD across five search seeds.

Prefix (rollouts)	Batch random	Geometric	Failure k NN
8	2.00 $\pm$ 2.35	2.00 $\pm$ 2.35	2.00 $\pm$ 2.35
16	3.80 $\pm$ 3.49	3.20 $\pm$ 2.68	4.20 $\pm$ 3.56
24	5.40 $\pm$ 4.62	4.60 $\pm$ 2.30	7.20 $\pm$ 4.87
32	7.80 $\pm$ 5.02	6.00 $\pm$ 2.24	11.60 $\pm$ 6.19
40	10.00 $\pm$ 5.79	8.20 $\pm$ 2.49	15.60 $\pm$ 8.11
48	12.20 $\pm$ 6.06	10.40 $\pm$ 2.51	20.20 $\pm$ 10.21
56	14.80 $\pm$ 5.81	11.80 $\pm$ 2.59	24.40 $\pm$ 11.39
64	17.40 $\pm$ 6.91	13.00 $\pm$ 2.00	29.00 $\pm$ 12.49