The Computational Handoff: A Pre-Rollout Test of Observation-Buffer Reset at Robot Task Boundaries

Yinuo Chen
South China University of Technology

***Abstract*—Full-shift robot evaluation composes many short skills, making runtime state at task boundaries part of the evaluated system. We present a renderer-free case study and reusable conformance contract that asks whether an in-process observation-buffer reset reproduces a fresh OS worker after predecessor history has been exercised. The test matches the recorded successor observation, checkpoint, diffusion seed, hardware, and preprocessing while varying history and worker lifecycle. Across three frozen-before-inspection ordinal RoboFactory task-pair proxies and two checkpoint epochs per successor, injected carry history exceeds a frozen stressor-activation threshold for all 480 held-out action chunks (mean discrepancies 0.1246–0.3178), whereas explicit deque reset matches a fresh process exactly for 480/480 probes and 60/60 independent one-shot workers. On 120 disjoint probes, one predecessor observation activates the stressor throughout, but the H2–H1 contrast is positive for only one of three pairs and confounds history count with frame content. We therefore reject a general monotonic-depth interpretation. This is a pre-rollout action-level gate, not evidence of an observed deployment bug, task success, or eight-hour autonomy. An accompanying artifact supplies the frozen manifests, scripts, and result records needed to recompute the reported checks.**

I. Introduction

A full work shift contains repeated handoffs between skills, controllers, and tasks. Existing long-horizon evaluations show that chained terminal states can be harder than clean skill starts [1], [2]. A separate systems question is whether the policy runtime is computationally fresh when an orchestrator advances to the next task. RoboTwin's deployment interface, for example, exposes a reset hook for recurrent memory or history buffers [3], yet calling a reset API is not itself evidence that the resulting behavior matches a newly started worker.

We define an *observation-buffer reset conformance* relation for a frozen stateful policy. Given an identical recorded successor observation, checkpoint, preprocessing path, hardware, and diffusion seed, an in-process reset is conformant when its proposed action chunk matches that of a fresh OS process within a numerical tolerance frozen before result inspection. We stress the contract by injecting two predecessor-task observations into a three-frame Diffusion Policy [4]. Unlike general metamorphic tests that perturb robot inputs [5], the relation here targets runtime lifecycle scope.

Our contribution is deliberately narrow: (i) a reusable fresh-process-referenced pre-rollout contract, (ii) a worked case study of RoboFactory's public `DPRunner.reset_obs()` path with fail-closed process and byte provenance, (iii) cross-checkpoint replication on three frozen-before-inspection offline RoboFactory [6] task-pair proxies, and (iv) a disjoint-content characterization that exposes the limits of a simple history-depth interpretation. We also measure the lifecycle stages surrounding reset and restart. We make no claim about live task success, safety, or full-shift autonomy.

II. Related Work

Long-horizon manipulation benchmarks increasingly expose failures that short, independently reset episodes conceal. BOSS studies observation-space shift when skills are chained from predecessor terminal states rather than canonical starts [1]; policy-chaining methods improve transition feasibility or jointly refine adjacent skills [7], [8]; and recent work on semantic handoff failures asks whether an agent orchestrator composes VLA skills coherently [2]. These studies concern the physical state, transition distribution, or semantic command presented to a skill. Our test isolates a complementary computational boundary: with the current successor observation held fixed, does an explicitly reset policy runtime behave like a fresh one?

Metamorphic testing avoids requiring a complete behavioral oracle by checking relations among transformed inputs and outputs. DeepXplore cross-references multiple models [9], while DeepCruiser develops coverage criteria for stateful recurrent networks [10]; recent VLA testing applies metamorphic relations to image and instruction transformations [5]. We instead transform worker lifecycle while holding the current observation and model fixed, using a fresh process as the operational reference. This audits deployed wrapper state rather than model disagreement, neuron coverage, or task outcome. The relation is especially relevant to policies that deliberately consume observation history [11] and to Diffusion Policy [4], whose inference wrapper assembles a fixed temporal stack even when the neural network itself is feed-forward. RoboTwin exposes a state-reset hook [3], but a callable hook alone does not establish fresh-start equivalence.

III. Conformance Protocol

A. Conditions

For each ordinal source/successor episode pair, we read the last two source observations and the first successor observation. These recordings are an adversarial task-pair proxy, not a physically coherent handoff. We compare:

- **E (injected carry):** a long-lived runner receives two source frames followed by the current successor frame;

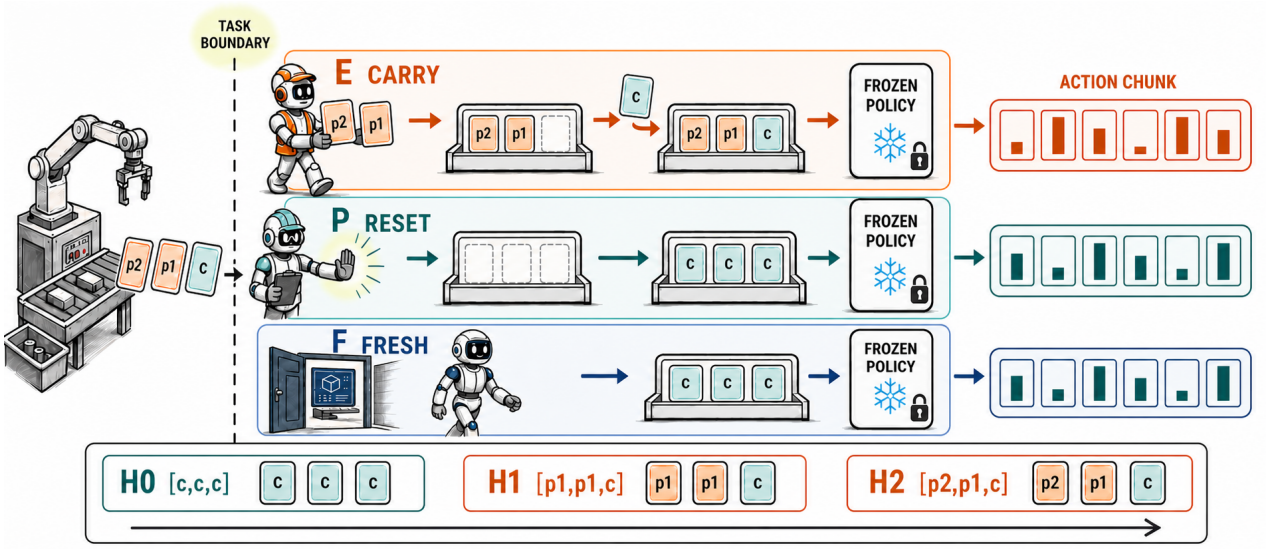


Fig. 1. Pre-rollout computational-handoff contract. E carries two predecessor observations into the successor prediction. P clears only the tested observation deque in the already exercised runner and repopulates it with the matched current observation. F launches a process-isolated worker. The lower strip defines native-padding history-content conditions: H0=$[c, c, c]$, H1=$[p_1, p_1, c]$, and H2=$[p_2, p_1, c]$. AI-generated schematic; scene thumbnails are illustrative, not dataset observations. The test concerns computational freshness, not physical handoff success.

- **P (deque reset):** after E is exercised, the same runner clears its observation deque and receives only the current frame, using the runner's documented padding rule;
- **F (fresh worker):** an independently launched OS process loads the same checkpoint and receives only the current frame;
- **F-repeat:** the fresh worker repeats the prediction after resetting and repopulating its deque, providing a numerical-noise control.

The frozen implementation exposes a three-frame observation deque and produces a 6×8 action chunk. Dimensions 0–6 are Franka joint-position targets; dimension 7 is a gripper command on $[-1, 1]$. We do not generalize the intervention to action queues, temporal ensembles, or hidden model state absent from this stack.

Implementation boundary. We test RoboFactory commit `5868242` and class `dp_runner.DPRunner`. The persistent field is `obs`, a Python `deque`; `reset_obs()` calls `obs.clear()`, `update_obs()` appends the current observation, and `stack_last_n_obs()` left-pads a short stack with its earliest frame. P directly calls this public runner method after E while retaining the same policy and runner objects; it is not an observed orchestrator integration. The harness binds the imported runner file by SHA-256. P/F equality is a known-good positive control for this implementation, not a claim that clearing a deque is surprising: the contribution is the lifecycle-scope oracle and fail-closed pre-rollout gate that can expose output-visible state omitted by a reset implementation. Mutations to other wrapper state, caches, queues, or random-number state could cause failure after E. Fixed seeds isolate lifecycle residue, and the one-process-per-probe stage guards against contamination of the fresh baseline.

B. Contract semantics

Let $A_\theta(o, h; \xi)$ denote the complete proposed action chunk from frozen checkpoint θ, current observation o, explicit history-stack construction h, and diffusion seed ξ. This notation includes preprocessing and the deployed runner, not merely the neural-network forward call. For a probe i, the three outputs are

$$a_i^E = A_\theta(o_i, h_i^{\text{carry}}; \xi_i), \tag{1}$$
$$a_i^P = A_\theta(o_i, h_i^{\text{reset}}; \xi_i), \tag{2}$$
$$a_i^F = A_\theta(o_i, h_i^{\text{fresh}}; \xi_i). \tag{3}$$

Reset conformance asks whether $\|a_i^P - a_i^F\|_\infty \le \epsilon$; stressor adequacy separately asks whether $\text{MAE}(a_i^E, a_i^F) \ge 10^{-3}$. The latter is only a frozen stressor-activation threshold; it has no calibrated physical-severity meaning. Keeping these predicates separate is important. E/F equality can mean that the chosen predecessor content is weak, but cannot turn a P/F mismatch into a reset pass. Conversely, E/F divergence is expected from a temporal policy and is not evidence of a defect unless the deployment contract actually declares a task boundary.

The matched-current-observation design removes physical-state mismatch from the relation. This makes the oracle unusually precise but deliberately changes the question: it tests whether lifecycle handling preserves fresh-start computation, not whether the successor can recover from the predecessor's terminal world state. The latter requires synchronized simulator or hardware snapshots and task-level outcomes. Our protocol is therefore best placed before closed-loop endurance evaluation, where it can reject an ambiguous policy-runtime boundary without consuming rollout budget.

C. Split, tolerance, and gates

Each task pair contains 100 probes selected without replacement by a frozen seed. Probe IDs 0–19 calibrate the numerical tolerance; IDs 20–99 are held out. Let $d_{\max}$ be maximum absolute chunk error. We set

$$\epsilon = \min\left(10^{-5}, \max\left(10^{-7}, 10Q_{.99}(d_{\max}^{F,F'})\right)\right), \qquad (4)$$

and fail if the unclipped candidate exceeds 10^{-5}. The reset batch gate requires both F-repeat/F and P/F to be within ϵ for 80/80 held-out probes. Independently, the stressor gate requires E/F chunk MAE of at least 10^{-3} for at least 64/80 probes. IDs 20–29 are then rerun with one fresh OS process per probe; all 10 must pass repeat, batch-F, and P comparisons.

The harness rejects provenance mismatches in checkpoint bytes, Zarr metadata, selected episodes, seeds, observation-stack hashes, action shapes/hashes, deterministic settings, GPU identity, runner identity, and process identity. Full chunks are stored in append-only JSONL before an atomic final artifact.

D. Cross-checkpoint and history-content extension

The six-page extension adds two frozen-before-execution tests without changing the primary result. M1 evaluates the other available successor checkpoint for each task pair while preserving the primary episode IDs, selection seed, and diffusion seeds. Its reset gate requires held-out F-repeat/F and P/F conformance for 80/80 probes plus 10/10 same-GPU one-shot workers at all three alternate checkpoints. Its independent stressor gate requires threshold-exceeding E/F discrepancy on at least 64/80 probes for at least two checkpoints; a weak E stressor therefore cannot invalidate an otherwise conformant reset.

M2 uses exactly the 50 episode IDs omitted by each primary 100-probe selection and new diffusion seeds. After resetting before every condition, it compares H0=$[c, c, c]$, H1=$[p_1, p_1, c]$, and H2=$[p_2, p_1, c]$ against a new process-isolated F. Here H1 contains one distinct predecessor repeated by native earliest-frame padding, whereas H2 replaces that repeated slot with an earlier, different predecessor. The H2–H1 contrast therefore changes both distinct-history count and frame content; it is a native-padding history-content test, not an isolated causal depth intervention. Ten probes calibrate the repeat-derived tolerance and 40 are held out. We report paired-probe 10,000-resample percentile intervals for $d(H2, F) - d(H1, F)$ and retain the originally frozen monotonicity gate only as a deliberately failed diagnostic.

Both extensions bind accepted primary-artifact bytes, checkpoint bytes, full dataset manifests, raw JSON and append-only JSONL, and analysis hashes. The aggregate re-derives every Boolean decision from numeric evidence and requires all six configurations.

E. Why the fresh process is the oracle

The relation is intentionally operational rather than architectural. We do not attempt to enumerate every mutable tensor or Python object reachable from the policy wrapper. Instead, F reconstructs the worker through the deployed loading path, providing the behavior expected after maximal computational reinitialization. P is acceptable only if it reproduces F under matched successor bytes and pseudorandom seed. F-repeat separates lifecycle error from numerical repeatability, while the one-shot stage prevents a single long-lived F process from silently becoming a second stateful baseline. This design tests a reset implementation without assuming that its source-level field list is complete.

IV. Experiments

We use the public RoboFactory Franka manipulation benchmark and its Diffusion Policy implementation. The primary task-pair proxies were fixed before inspection of v2 results: StrikeCube→PickMeat with PickMeat epoch 300, PickMeat→StackCube with StackCube epoch 300, and StackCube→StrikeCube with StrikeCube epoch 150. Selection/diffusion seeds are respectively $(260808, 520000)$, $(260809, 620000)$, and $(260810, 720000)$. M1 pairs these with PickMeat epoch 150, StackCube epoch 150, and StrikeCube epoch 300. M2 returns to the primary checkpoints on the exact unused episode complements.

All comparisons within a pair run sequentially on the same physical NVIDIA A40. GPU keepalive load is present throughout and may inflate absolute latency, so the timing result is node-specific rather than an unloaded hardware baseline. The container can run CUDA inference but cannot initialize SAPIEN RGB rendering because its NVIDIA Vulkan userspace and host driver are incompatible; consequently, live rollouts are outside the present evidence.

Using the PickMeat epoch-300 checkpoint, we separately benchmark 1,000 deque reset-plus-current-observation operations after 100 warmups and 10 fresh worker launches. The latter report spawn-to-observation-ready, spawn-to-first-valid-action, and spawn-to-exit, together with import, checkpoint-hash, policy-load, and first-action phases. The sequential benchmark is warm-cache-prone: each worker hashes the 1.5-GB checkpoint before `torch.load`.

V. Results

A. Reset conformance replicates across checkpoints

Table I reports the frozen primary checkpoints and the M1 alternate epochs. Injected carry history activates the stressor for all 480 held-out checkpoint-conditioned probes. The primary E/F means are 0.3135, 0.1262, and 0.1797; their paired alternate-epoch values are 0.3178, 0.1246, and 0.1790, respectively. The close within-pair values are a replication observation, not a claim that checkpoint epoch is irrelevant more generally.

P/F and F-repeat/F maximum errors are exactly zero over all 480 full 6×8 chunks. Sixty distinct one-probe OS processes—10 for every pair/epoch configuration—also have zero maximum error for repeat, batch-F, and P comparisons. The M1 reset gate frozen before execution therefore passes at 3/3 alternate checkpoints; its independent E/F stressor gate also passes at 3/3, exceeding the required 2/3. Across the primary configurations, E/F sensitivity appears in all seven

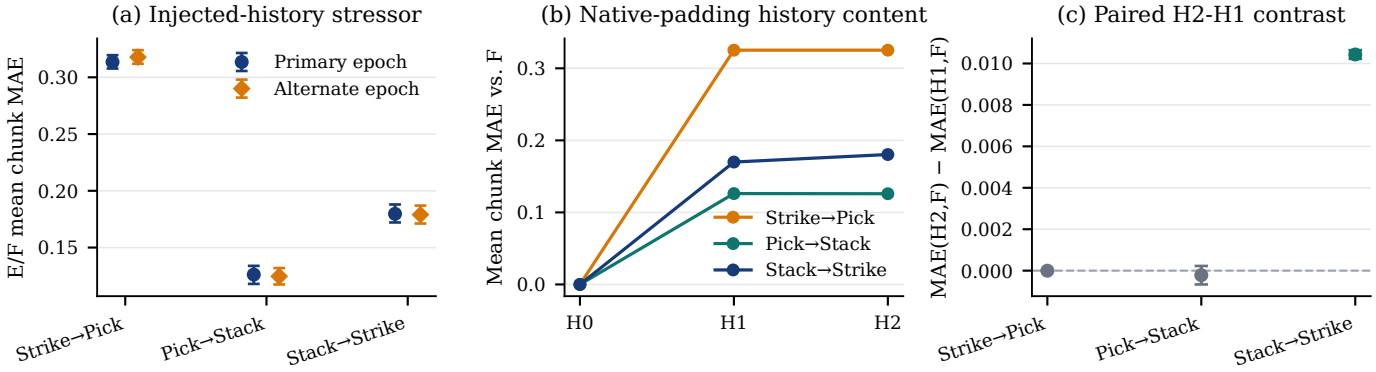


Fig. 2. Cross-checkpoint replication and native-padding history-content characterization. (a) E/F exceeds the stressor threshold at both successor-checkpoint epochs; error bars are probe-bootstrap 95% intervals. P/F is exactly zero and omitted to preserve resolution. (b) On disjoint episodes, H1 and H2 both exceed the threshold relative to fresh/native-padded H0. (c) Paired 95% intervals for the H2–H1 contrast are positive only for Stack→Strike. This contrast changes both history count and content; all values are action-space discrepancies, not physical error or severity.

TABLE I

CROSS-CHECKPOINT HELD-OUT ACTION CONFORMANCE. "E/F ACTIVE" COUNTS CHUNK MAE $\geq 10^{-3}$ UNDER THE FROZEN STRESSOR THRESHOLD; "P/F CONFORM." COUNTS MAXIMUM CHUNK ERROR WITHIN THE CALIBRATED $\epsilon = 10^{-7}$. CONFIDENCE INTERVALS ARE PROBE-BOOTSTRAP 95% INTERVALS.

Task pair	Succ. epoch	E/F mean MAE	95% CI	E/F active	P/F conform.	One-shot
Strike→Pick	300	0.3135	[0.3077, 0.3195]	80/80	80/80	10/10
Strike→Pick	150	0.3178	[0.3119, 0.3239]	80/80	80/80	10/10
Pick→Stack	300	0.1262	[0.1180, 0.1339]	80/80	80/80	10/10
Pick→Stack	150	0.1246	[0.1175, 0.1319]	80/80	80/80	10/10
Stack→Strike	150	0.1797	[0.1721, 0.1879]	80/80	80/80	10/10
Stack→Strike	300	0.1790	[0.1712, 0.1870]	80/80	80/80	10/10

arm dimensions (pairwise dimension-averaged errors 0.0577–0.1360) but is largest in the gripper dimension (0.5884–1.5557 on $[-1, 1]$). These values demonstrate action sensitivity, not physical severity.

B. History presence is stable; content contrast varies

M2 uses the exact 50-episode complement excluded by each primary selection: 10 calibration and 40 held-out probes per pair. H0 matches F exactly for all 120 held-out chunks, verifying the native-padding control on disjoint data. H1 is stressor-active for 120/120 probes; H2 is likewise active for 120/120. Mean H1/H2 discrepancies are $0.325192/0.325190$ for Strike→Pick, $0.126137/0.125918$ for Pick→Stack, and $0.169904/0.180338$ for Stack→Strike.

Paired H2−H1 means (95% 10,000-resample intervals) are -2.86×10^{-6} $[-5.64\times10^{-6}, -0.06\times10^{-6}]$ for Strike→Pick, -2.19×10^{-4} $[-6.66\times10^{-4}, 2.26\times10^{-4}]$ for Pick→Stack, and 1.043×10^{-2} $[1.023\times10^{-2}, 1.064\times10^{-2}]$ for Stack→Strike. Only the last satisfies the positive, interval-excludes-zero pair gate. The frozen aggregate requires two of three, so it rejects a general monotonic-depth claim. Because H2 also changes predecessor content, these data support only a stable presence-of-history stressor and a pair-dependent H2–H1 contrast; content-controlled stacks are needed to estimate a causal depth effect.

TABLE II

NON-SYMMETRIC LIFECYCLE STAGE TIMINGS UNDER THE STATED WARM-CACHE-PRONE PROTOCOL; NO END-TO-END SPEEDUP RATIO IS IMPLIED.

Operation	Median	95th percentile
Deque operation overhead only	0.569 μs	0.623 μs
Process init. to observation-ready	57.68 s	64.21 s
Process init. plus first action	63.46 s	67.67 s

VI. OPERATIONAL INTERPRETATION

The protocol separates three questions that are often conflated at a task boundary. First, *sensitivity*: can predecessor observations alter the successor action proposal? This is answered by E/F and is expected for a temporal policy; it is not by itself a software defect. Second, *reset conformance*: does the declared reset reproduce fresh-worker behavior? This is answered by P/F and is the main test contract. Third, *deployment utility*: does conformant reset improve success, recovery, or throughput? The present offline study does not answer this third question.

For integration testing, we recommend that an orchestrator declare each stateful runtime field, its scope (step, episode, task, or shift), its reset trigger, and its fresh-start padding rule. A boundary test should then record the current observation bytes, random seed, checkpoint digest, action digest, process

TABLE III
FAIL-CLOSED INTERPRETATION OF THE THREE COMPARISONS. F′ DENOTES THE REPEAT PREDICTION IN THE FRESH WORKER.

Observed relation	Permitted interpretation
F′ $\not\approx$F	Numerical oracle invalid; stop before judging reset.
F′ $\approx$F, P$\not\approx$F	Reset is nonconformant for this probe; inspect lifecycle state.
P$\approx$F, E$\approx$F	Reset conforms, but predecessor injection is an uninformative stressor.
P$\approx$F, E$\not\approx$F	Informative conformance pass: reset removes an exercised influence.

identity, and reset event. Failure of P/F points to incomplete or non-deterministic reinitialization; failure of F-repeat invalidates the numerical oracle; absence of E/F sensitivity means the tested predecessor history is not a useful stressor. These outcomes remain informative even when no renderer or closed-loop simulator is available.

This contract suggests a two-tier evaluation architecture. A cheap, renderer-free tier runs on every policy/checkpoint integration and verifies repeatability, stressor adequacy, and reset conformance. Only artifacts that pass this tier proceed to a renderer or robot tier that evaluates reachable handoffs, success, recovery time, and accumulated failures. The tiers answer different questions and should retain separate pass/fail states. In particular, offline conformance must not be promoted into a task-success result, while a task-success aggregate should not obscure a lifecycle mismatch that happened to be recovered later.

At runtime, the same distinction supports actionable telemetry. A task-boundary event can log the prior and next skill identifiers, declared state scopes, reset calls, padding mode, checkpoint digest, and first post-boundary action digest. Such a log does not require storing raw camera frames indefinitely; a privacy-conscious deployment can retain cryptographic hashes and selectively capture bytes only after a failed canary comparison. Periodic process-isolated canaries can then test whether a long-lived worker still matches its reference without imposing a restart at every handoff. The fresh process remains a diagnostic oracle rather than the default recovery mechanism.

VII. DISCUSSION AND LIMITATIONS

The six checkpoint-conditioned configurations support a worked conformance-test statement: for these offline proxies, explicit deque reset restores fresh-process action equivalence under matched inputs and randomness, while an injected-history stress condition does not. Replication across checkpoint epochs reduces the chance that this conclusion is an idiosyncrasy of one saved model. The disjoint M2 probes provide a second separation: history presence remains consequential, but H2–H1 varies by pair and does not isolate depth from content. A reset contract can therefore be stable even when the measured stress response lacks a simple monotonic model.

The timing decomposition attributes a median 47.92 s to policy loading, 5.22 s to hashing the 1.5-GB checkpoint, 5.20 s to dependency import, and 4.80 s to first-action inference. Absolute values include contention from mandatory keepalive load and should not be read as an unloaded A40 benchmark.

The result does not show that RoboFactory or a deployed orchestrator actually leaves this deque uncleared, and ordinal pairing does not represent a feasible physical transition. The evidence covers one policy family and only its three-frame observation deque; action queues, hidden model state, and other wrappers are untested. A reset should only be applied at a known boundary because resetting useful within-task history could itself be harmful. Live paired snapshots, a stateless control policy, closed-loop success, short shifts, real hardware, and safety outcomes remain future validation. The practical checklist is therefore modest: document stateful runtime fields, clear them at declared boundaries, repopulate buffers according to the fresh-start rule, and verify the first proposed chunk against a process-isolated oracle.

A. Threats to validity

The principal construct threat is that action discrepancy is not task-level harm. Diffusion Policy intentionally consumes temporal context, so E/F sensitivity is expected and may be beneficial within a task. Our intervention becomes a useful counterexample only at an externally declared boundary where predecessor observations are outside the successor contract. Even there, a changed proposal need not reduce success, and the especially large gripper-coordinate discrepancy should not be converted into workspace distance or severity. Closed-loop paired snapshots are required to connect conformance to utility.

The main internal threat is an incomplete lifecycle intervention. E is evaluated before P in the same long-lived runner; an E prediction could in principle mutate state outside the observation deque. Exact P/F equality after E across 480 checkpoint-conditioned held-out probes argues that no such residual affects the tested output under fixed seeds, but it cannot certify unobserved fields or other policy wrappers. Conversely, exact equality is stack- and hardware-specific rather than a universal bitwise guarantee. The repeat-derived tolerance and the 10^{-5} fail-closed cap are intended to make a future nondeterministic backend reject the test instead of silently widening equivalence.

External validity is limited by ordinal cross-dataset pairing, three manipulation tasks, one Diffusion Policy implementation, and checkpoints trained in the same benchmark family. Two epochs per successor improve checkpoint coverage but are correlated training snapshots, not six independent policy families. Predecessor and successor frames need not form a reachable physical transition. This deliberately strengthens stress coverage but weakens deployment realism. Testing synchronized live boundary snapshots, additional temporal architectures, and a stateless negative control would distinguish a reusable lifecycle contract from an implementation-specific observation-deque test.

B. Implications for full-shift evaluation

A full-shift benchmark can treat runtime state as a scoped resource. Each field should declare whether it survives a step, episode, task, or shift; the orchestrator should emit the boundary event that changes the scope. Conformance then becomes a pre-rollout gate: repeatability must hold, a chosen stressor must exercise the field, and the inexpensive reset path must match maximal reinitialization. This gate does not replace endurance metrics, but prevents an ambiguous runtime lifecycle from contaminating subsequent measurements of recovery, jobs per hour, or survival.

The latency result also motivates separating correctness from availability. A fresh process is a strong testing oracle yet took roughly one minute to reload and propose an action on the occupied node. It is therefore unsuitable as evidence that restarts are operationally cheap. The microsecond measurement covers only deque manipulation and is not end-to-end comparable with restart-to-action; production systems must additionally account for post-reset inference, communication, synchronization, and hardware contention.

C. Reproducibility boundary

All configurations frozen before result inspection are reported without post-hoc selection. The primary and M1 stages comprise 480 checkpoint-conditioned held-out evaluations over 240 unique episode proxies plus 60 independent one-shot processes. M2 contributes 120 new held-out proxies from exact selection complements. Every stage stores calibration/test IDs, dataset episode indices, checkpoint and Zarr digests, observation-stack and action hashes, diffusion seeds, process IDs, GPU UUID, and full 6×8 chunks. Raw records are appended before atomic summaries. The aggregate re-derives its gates and fails on absent, duplicate, or relabeled configurations.

The accompanying supplementary artifact contains the exact probe and analysis scripts, frozen experiment/deployment manifests, aggregate and raw result records, and a SHA-256 inventory. It permits regeneration of the numerical decisions and paper tables from the included records. Checkpoints and RoboFactory datasets are omitted because of size, so rerunning inference additionally requires the upstream assets identified by their recorded digests. This makes the submitted evidence checkable without implying public archival permanence.

VIII. Conclusion

Task orchestration needs a computational handoff contract in addition to a physical postcondition. Fresh-worker-referenced action conformance provides a low-cost, fail-closed pre-rollout test for one observation-buffer contract. Across three offline ordinal task-pair proxies, two checkpoints per successor, and disjoint history-content probes, explicit deque reset is exactly fresh-equivalent under fixed inputs and randomness, while history presence consistently activates the stressor. The content-confounded H2–H1 contrast cautions against a universal monotonic-depth story. Whether the verified contract improves closed-loop or shift-level outcomes remains an open deployment question.

References

[1] Y. Yang, L. Zhao, M. Ding, G. Bertasius, and D. Szafir, "BOSS: Benchmark for observation space shift in long-horizon task," *arXiv preprint arXiv:2502.15679*, 2025.

[2] K. Rui, Y. Zuo, J. Wang, H. Jia, J. Ma, W. Zhou, and M. Li, "Diagnosing semantic handoff failures in agent-orchestrated vision-language-action skill composition," *arXiv preprint arXiv:2607.06256*, 2026.

[3] RoboTwin Team, "RoboTwin 2.0: Deploy your policy," Online documentation, 2025, accessed 2026-08-08. [Online]. Available: https://robotwin-platform.github.io/doc/usage/deploy-your-policy.html

[4] C. Chi, S. Feng, Y. Du, Z. Xu, E. Cousineau, B. C. M. Burchfiel, and S. Song, "Diffusion policy: Visuomotor policy learning via action diffusion," in *Robotics: Science and Systems*, 2023.

[5] P. Valle, S. Segura, S. Ali, and A. Arrieta, "Metamorphic testing of vision-language action-enabled robots," *arXiv preprint arXiv:2602.22579*, 2026.

[6] Y. Qin, L. Kang, X. Song, Z. Yin, X. Liu, X. Liu, R. Zhang, and L. Bai, "RoboFactory: Exploring embodied agent collaboration with compositional constraints," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

[7] Y. Chen, C. Wang, L. Fei-Fei, and C. K. Liu, "Sequential dexterity: Chaining dexterous policies for long-horizon manipulation," in *Proceedings of the 7th Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, vol. 229, 2023, pp. 3809–3829.

[8] Z. Chen, Z. Ji, J. Huo, and Y. Gao, "SCaR: Refining skill chaining for long-horizon robotic manipulation via dual regularization," in *Advances in Neural Information Processing Systems*, 2024.

[9] K. Pei, Y. Cao, J. Yang, and S. Jana, "DeepXplore: Automated whitebox testing of deep learning systems," in *Proceedings of the 26th Symposium on Operating Systems Principles*, 2017, pp. 1–18.

[10] X. Du, X. Xie, Y. Li, L. Ma, J. Zhao, and Y. Liu, "DeepCruiser: Automated guided testing for stateful deep learning systems," *arXiv preprint arXiv:1812.05339*, 2018.

[11] S. Haldar, Z. Peng, and L. Pinto, "BAKU: An efficient transformer for multi-task policy learning," in *Advances in Neural Information Processing Systems*, vol. 37, 2024.