ORBITPO: LEARNING INTERFACE EQUIVARIANCE FOR TOOL-USING LANGUAGE AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Tool-using language agents act through software interfaces that are routinely refactored—tools renamed, fields reordered, entity handles relabeled—while the underlying tasks remain unchanged. Because such exact interface changes preserve task semantics yet still break capable agents, robustness to them is a prerequisite for integrating agents into evolving software. First, existing consistency objectives compare candidate-normalized action distributions, which conceal cross-interface shifts in unconditional action probability and yield no signal when only one candidate is present. Second, view-local reward normalization and prefix matching discard cross-interface supervision precisely when equivalent rollouts fail or diverge, a loss that grows as episodes lengthen. To address the first challenge, we propose Orbit-Equivariant Policy Optimization (ORBITPO), whose mass-preserving semantic alignment matches inverse-mapped unconditional candidate probabilities together with an explicit residual mass, supported by an exact information decomposition and non-asymptotic bounds that link retained-mass dispersion to alignment. To address the second, ORBITPO introduces orbit-relative credit and counterfactual complete-history replay, which pool rewards across equivalent interfaces and align every visited history without requiring matched prefixes or successful continuations, with a coverage-dependent bound from replay loss to cross-interface return. Experiments on six core benchmarks, four frozen-transfer suites, nine adapted backbones, and controlled OrbitLab programs show that ORBITPO improves worst-view aggregate over exposure-matched augmentation on every backbone, by 6.7–9.9 points while keeping one policy call per decision.

1 INTRODUCTION

Tool-using language agents act through software interfaces that are routinely refactored while the tasks they serve remain unchanged. Renaming `lookup_order(order_id)` to `fetch_purchase(ref)`, reordering tools or JSON fields, and consistently relabeling entity handles are exact coordinate changes: they preserve task semantics, dynamics, and rewards, yet they still break capable agents (Ye et al., 2024; Huang et al., 2024; Rabinovich & Anaby-Tavor, 2025; Zheng et al., 2026). Robustness to such changes is therefore a prerequisite for integrating agents into evolving software. Exposure to transformed interfaces does not provide this robustness: in our six-benchmark comparison on Qwen3-8B, exposure-matched augmentation (Aug-GRPO) reaches 63.1 on clean interfaces but only 47.5 worst-view aggregate (WVA).

Two challenges keep augmentation from converting exposure into robustness. **Challenge 1 (what to align):** existing consistency objectives compare action distributions renormalized over a selected candidate set. Renormalization discards the probability outside the candidates, so two views can agree on their renormalized candidates while assigning the same semantic action very different unconditional probability; with a single candidate, the discrepancy is identically zero. **Challenge 2 (where to supervise):** existing credit-assignment and matching schemes lose cross-view supervision exactly when equivalent rollouts fail or diverge. View-local reward normalization cannot contrast a failed view with a successful equivalent, and prefix matching discards every history after the first divergence; natural cross-view history matches fall to 12.4% at calls 17–32.

Exact interface changes, however, arrive with a resource that generic augmentation ignores: the compiler that renders a view also records its invertible map g. We therefore require policy equivariance,

$$\pi_\theta(ga \mid gh, g\mathcal{T}) = \pi_\theta(a \mid h, \mathcal{T}), \tag{1}$$

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.

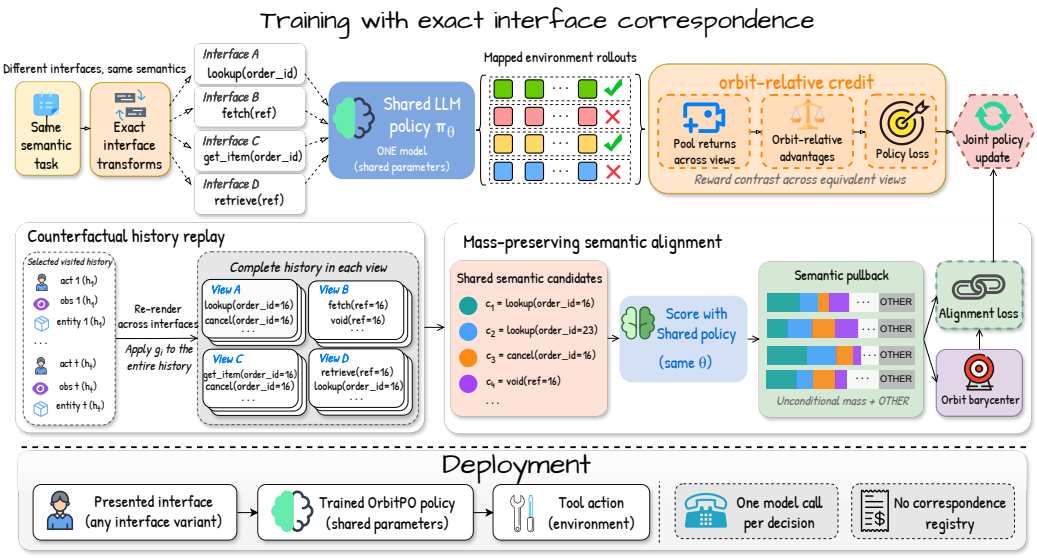


Figure 1: Overview of ORBITPO. Exact interfaces are coordinate charts of one semantic task. Training pools returns across equivalent views and replays every visited complete history in all views (Contribution 2), and aligns inverse-mapped candidate probabilities plus the OTHER residual mass to the orbit barycenter (Contribution 1). Deployment uses one policy call in the presented interface.

for every exact interface map g, history h, interface $\mathcal{T}$, and structured action a; Theorem 1 shows that this constraint is compatible with optimal control. The recorded map motivates one remedy per challenge. **Motivation 1 (for Challenge 1):** pulling every view back through g^{-1} places unconditional candidate probabilities and the residual mass in one shared semantic space, so alignment can compare them directly instead of renormalizing the residual away. **Motivation 2 (for Challenge 2):** because admitted maps preserve return and can re-render any visited history, all views of a task form one credit group and remain comparable at every visited history.

We introduce ORBITPO, Orbit-Equivariant Policy Optimization (Figure 1), with three contributions. **(1) Mass-preserving semantic alignment** (Challenge 1, Motivation 1). Unlike candidate-normalized consistency, it aligns inverse-mapped unconditional candidate probabilities together with an explicit OTHER residual; an exact information decomposition shows that it retains the mass term that conditional alignment deletes, with non-asymptotic bounds and an objective-to-return certificate (Sections 3 and 4). **(2) Orbit-relative credit and complete-history replay** (Challenge 2, Motivation 2). Unlike per-prompt normalization, prefix matching, or successful-behavior transfer, they normalize rewards across the whole orbit and align re-rendered visited histories without a within-view success, a matched prefix, or a successful continuation, with a coverage-dependent bound from replay loss to cross-interface return. **(3) Extensive experiments** span six core benchmarks with 17 exact views per task, four frozen-transfer suites, nine adapted backbones, controlled OrbitLab programs that separate horizon from consequential interface-decision density, and an API-migration audit (Sections 5–6).

In summary, ORBITPO improves WVA over matched augmentation on all nine backbones by 6.7–9.9 points with one call per decision, and at equal episodes on Qwen3-8B it exceeds LocalCanon-GRPO and FLOP-matched Success-Xfer-MF by 1.3 and 3.1 points (LocalCanon leads by 0.8 at equal FLOPs). Retained mass adds 2.8 WVA points over conditional alignment and 4.8 in the high-dispersion stratum (Contribution 1), and the replay advantage grows from essentially zero on unused-tool changes to 4.8 points when transformed decisions become consequential (Contribution 2). Paraphrases, permission mutations, and runtime faults remain tied, delimiting the gains to exact coordinate changes.

2 RELATED WORK

Tool learning. Toolformer, ToolLLM, API-Bank, ToolACE, and APIGen develop training and data recipes for function calling (Schick et al., 2023; Qin et al., 2023; Li et al., 2023; Liu et al., 2024a;b), and verifiable-reward methods optimize tool choice and multi-turn interaction (Qian et al., 2025;

Zhong et al., 2026). These methods mainly improve *which* action is chosen within one fixed interface. ORBITPO is complementary: it targets whether that choice survives an exact change of the interface.

Interface robustness. RoTBench and follow-up studies expose function-calling brittleness (Ye et al., 2024; Huang et al., 2024; Rabinovich & Anaby-Tavor, 2025), ToolRobustBench and AgentChaos diagnose stage-wise perturbations and runtime faults (Zheng et al., 2026; Tan et al., 2026), and long-horizon suites add state, dual control, retrieval, and cross-application workflows (Trivedi et al., 2024; Yao et al., 2024; Lu et al., 2024; Berkeley Function Calling Leaderboard Team, 2026; Barres et al., 2025; Shi et al., 2026; Li et al., 2025). This line mainly *diagnoses* sensitivity or trains on perturbed inputs (RoTTuning), and description rewriting and programmatic calls instead *change the deployed interface* (Guo et al., 2026; Patel et al., 2026). ORBITPO instead uses the recorded correspondence as a training constraint (Equation 1) and leaves the deployed interface unchanged.

Invariance and consistency. Equivariant architectures and MDP-homomorphic RL hard-code symmetry into the model (Cohen & Welling, 2016; van der Pol et al., 2020); IRM, REx, and GroupDRO align or reweight risks across environments (Arjovsky et al., 2019; Krueger et al., 2020; Sagawa et al., 2019); and R-Drop aligns two dropout passes of one input (Liang et al., 2021). These methods mainly compare predictions or risks in a shared output space, whereas interface maps transform both the input and the structured output, so predictions must first be pulled back through the map before they can be compared.

Positioning of ORBITPO. ORBITPO differs from this literature in *what* it compares—unconditional pullback probabilities plus residual mass, rather than candidate-normalized or same-input predictions (Challenge 1)—and in *where* it compares—re-rendered complete visited histories, rather than matched prefixes or successful anchors (Challenge 2). Our baselines draw robustness from shared inputs (Aug-GRPO), reweighted view risk (GroupDRO), imitated successful actions (Success-Xfer), visible-schema canonicalization (LocalCanon), four-call voting (Orbit-TTA), or the deployment registry (FullCanon). Only ORBITPO aligns unconditional mass at complete visited histories with neither a registry nor extra calls (Appendix Table 16).

3 PROBLEM AND METHOD

An episode has history $h_t = (o_1, a_1, \ldots, o_t)$, typed interface $\mathcal{T}$, structured action $a_t \in \mathcal{A}(\mathcal{T})$, and bounded return $R(\tau)$. A canonical grammar decoder maps every completed structured action to one token path and adds a length-cap failure atom. A compiler morphism $g : \mathcal{T} \to \mathcal{T}'$ rewrites schemas, prompt-visible references, calls, observations, and handles through a bidirectional symbol table. Admitted maps are invertible and commute with the dynamics and the reward:

$$gP(o_{t+1} \mid h_t, a_t, \mathcal{T}) = P(go_{t+1} \mid gh_t, ga_t, g\mathcal{T}), \qquad R(g\tau) = R(\tau). \tag{2}$$

We use tool and field permutations, tool and argument alpha-renaming, entity handle renaming, and their compositions; Appendix B specifies their executable contracts and audits. ORBITPO uses Equation 2 twice: return invariance makes rewards comparable across views, and dynamics commutation renders any visited history in every view.

Orbit-relative credit (Contribution 2). For each semantic task, we sample m exact views and K trajectories per view. Aug-GRPO normalizes rewards within each rendered prompt, so a view whose K rollouts all fail receives no contrast; ORBITPO instead normalizes across the whole semantic orbit:

$$A^{\mathrm{orb}}_{i,k} = \frac{R_{i,k} - \mu_x}{\sigma_x + \epsilon}, \quad \mu_x = \frac{1}{mK}\sum_{i,k} R_{i,k}, \quad \sigma_x^2 = \frac{1}{mK}\sum_{i,k}(R_{i,k} - \mu_x)^2. \tag{3}$$

The clipped PPO term uses the rollout policy's masked token likelihood ratio $r_{i,k,t}$:

$$\mathcal{L}_{\mathrm{orbRL}} = -\mathbb{E}_{i,k,t}\left[\min\left(r_{i,k,t}A^{\mathrm{orb}}_{i,k}, \mathrm{clip}(r_{i,k,t}, 1-\epsilon_c, 1+\epsilon_c)A^{\mathrm{orb}}_{i,k}\right)\right]. \tag{4}$$

Because Equation 2 makes returns comparable across views, a failed view is contrasted with its successful equivalents; unlike per-prompt normalization, the contrast survives when every rollout of one view fails, and unlike successful-behavior transfer, it copies no successful trajectory into that view.

Mass-preserving complete-history alignment (Contributions 1 and 2). At up to eight complete histories visited by any rollout, we re-render the same semantic history in all sampled charts. A candidate set C_x contains the parsed actions observed at that exact history and a bounded prior-action reservoir; gold actions are never inserted. For each view, teacher-forced canonical scoring defines the pulled-back law

$$\widetilde{q}_{\theta,i}(c) = p_\theta^\tau(g_i c \mid g_i x), \quad c \in C_x, \qquad \widetilde{q}_{\theta,i}(\mathrm{OTHER}) = 1 - \sum_{c \in C_x} p_\theta^\tau(g_i c \mid g_i x). \tag{5}$$

The OTHER atom thus retains the mass of omitted completed actions and unfinished generation instead of renormalizing it away. With $\bar{q} = m^{-1} \sum_i \widetilde{q}_{\theta,i}$, every view is aligned to the stop-gradient orbit barycenter:

$$\mathcal{L}_{\mathrm{eq}} = \frac{1}{m} \sum_i D_{\mathrm{KL}}(\widetilde{q}_{\theta,i} \| \, \mathrm{sg}[\bar{q}]). \tag{6}$$

Writing $z_i = \widetilde{q}_i(C_x)$ and $r_i(c) = \widetilde{q}_i(c)/z_i$, the loss decomposes exactly into retained-mass and within-candidate information:

$$\mathcal{L}_{\mathrm{eq}} = \mathrm{JSD}_{1/m}(\mathrm{Ber}(z_1), \ldots, \mathrm{Ber}(z_m)) + \bar{z} \sum_i w_i D_{\mathrm{KL}}(r_i \| r_\star). \tag{7}$$

Equation 7 makes the difference from existing candidate-normalized alignment explicit. Such alignment compares only the conditionals r_i and therefore deletes the first term—the retained-mass signal missing in Challenge 1 and the only term that remains for a singleton pool. Complete-history replay addresses Challenge 2: because every selected visited history is re-rendered in all sampled views, the comparison exists even when no independent rollout reaches the same prefix and no successful continuation is available, whereas prefix matching and success-anchored transfer require one or the other. The total objective is

$$\mathcal{L}_{\textsc{OrbitPO}} = \mathcal{L}_{\mathrm{orbRL}} + \lambda_{\mathrm{eq}} \mathcal{L}_{\mathrm{eq}} + \beta \mathcal{R}_{\mathrm{ref}}. \tag{8}$$

Appendix C.1 gives the update order, gradient boundaries, candidate construction, and cost accounting.

4 Theoretical Properties

Theorem 1 (Optimal equivariant policy)**.** *For a finite-horizon tool process satisfying Equation 2, with finite legal action sets, a finite stabilizer for each chart, and attained Bellman maxima at every legal history, an optimal stochastic history-dependent policy exists that satisfies Equation 1 for every applicable groupoid morphism.*

Under these conditions, imposing exact equivariance costs no optimality; whether finite model capacity and optimization attain such a policy remains an empirical question, addressed in Section 6. The next certificate connects the implemented coarsened loss to view robustness.

Proposition 1 (Objective-to-return certificate)**.** *At step t, let η_t be the largest candidate missing mass, M_t the largest coarsened view-to-barycenter KL, and $I_t = I(G; \widetilde{C} \mid x_t, C_t)$. Every sampled pair satisfies*

$$\delta_t(i,j) \le b_t := \min\{1, \sqrt{2M_t} + \eta_t\} \le \min\{1, \sqrt{2mI_t} + \eta_t\}, \tag{9}$$

and $|J_i - J_j| \le R_{\max} \sum_t \mathbb{E}[b_t]$ under sequential coupling.

For Contribution 1, the chain rule of Equation 7 yields non-asymptotic bounds between retained-mass variance and the objective that remain informative for singleton pools, precisely where candidate-normalized alignment is identically zero (Proposition 10). For Contribution 2, Theorem 2 converts average replay alignment into a cross-interface return bound under an explicit history-coverage condition. Proofs, decoder-support qualifications, and composition bounds appear in Appendices A and A.13.

Table 1: ORBITPO improves worst-view aggregate over exposure-matched Aug-GRPO on all nine backbones. Panel A uses the common six-benchmark endpoint for five core backbones; Panel B uses a distinct four-suite frozen-transfer endpoint for four recent backbones. C is clean score, W is WVA, and Δ is ORBITPO minus Aug-GRPO WVA. Full baselines, per-benchmark cells, and variability are in the appendix.

Panel A: six core benchmarks					
Backbone	**Aug C**	**Orbit C**	**Aug W**	**Orbit W**	**Δ W**
Qwen3-4B	54.8	56.2	39.1	**47.3**	+8.2
Qwen3-8B	63.1	64.0	47.5	**55.9**	+8.4
Qwen3-32B	70.8	70.5	56.9	**63.6**	+6.7
Llama-3.1-8B	59.4	59.1	43.2	**50.6**	+7.4
Mistral Small 3.2 24B	66.5	67.2	51.6	**60.4**	+8.8
Panel B: four additional suites; frozen transfer					
Backbone	**Aug C**	**Orbit C**	**Aug W**	**Orbit W**	**Δ W**
Qwen3.5-9B	47.5	49.3	30.4	**38.8**	+8.4
Qwen3.6-35B-A3B	58.9	61.2	41.9	**51.3**	+9.4
Qwen3.8-27B	63.4	65.8	47.1	**57.0**	+9.9
GLM-4.7-Flash	55.4	57.4	37.7	**46.7**	+9.0

5 EVALUATION DESIGN

Scope and protocol. The six core benchmarks are API-Bank, ACEBench, RoTBench, AppWorld, τ-bench, and ToolSandbox. Each task receives the identity view plus 16 fixed exact views, including compositions unseen in training. The primary endpoint is benchmark-macro WVA, and the primary contrast is ORBITPO versus exposure-matched Aug-GRPO. Qwen3-8B (Yang et al., 2025) and Mistral Small 3.2 24B (Mistral AI, 2025) receive complete baseline panels, and Qwen3-4B, Qwen3-32B, and Llama-3.1-8B (Grattafiori et al., 2024) test transfer of the core contrast across scale and model family. A separate frozen-transfer endpoint adds BFCL V4 controlled, τ^2 telecom, τ-Knowledge, and Toolathlon-Verified on Qwen3.5-9B, Qwen3.6-35B-A3B, Qwen3.8-27B, and GLM-4.7-Flash (Qwen Team, 2026a;b;c; Z.ai, 2026). OrbitLab contains 15,360 factorial programs plus paired decision-density studies at lengths 16 and 32. Training comparisons match rollout episodes and report fixed-FLOP controls; inference uses one model call except for Orbit-TTA. Training samples $m = 4$ of the 17 views per update, whereas evaluation enumerates all 17, so the training view budget is not confounded with the worst-view minimum; an ablation varies m. Appendix E specifies splits, task counts, seeds, metrics, and statistics.

Hypotheses. Three prespecified hypotheses mirror the contributions. H1 (Contribution 1) predicts that retained mass matters most when cross-view candidate mass is dispersed, tested with coverage and gradient scale fixed (Tables 4–5). H2 (Contribution 2) predicts that replay matters when transformed decisions are consequential rather than merely visible (Figures 3–4). H3 (Contribution 3) predicts a backbone-wide WVA advantage over exposure- and compute-matched alternatives (Tables 1–3); a missing or isolated gain would falsify it. Efficiency and non-symmetry controls delimit the claims; Appendix Table 10 audits the theoretical conditions on held-out exact views.

6 RESULTS

6.1 BROAD END-TO-END ROBUSTNESS (H3)

Table 1 reports all nine backbones while keeping the two endpoints separate. On the six-benchmark endpoint, ORBITPO improves WVA over Aug-GRPO on all five core backbones, by 6.7–8.8 points across three model families and 4B–32B scales, while keeping clean scores within 1.4 points of Aug-GRPO. On the four-suite frozen-transfer endpoint, the gains are positive on all four recent backbones (8.4–9.9) and vary by model (full matrices in Appendix Tables 17–19).

Table 2 compares ORBITPO with the strongest alternatives, each of which isolates a different source of robustness. On Qwen3-8B, ORBITPO reaches 55.9 WVA: 8.4 points above Aug-GRPO (paired exposure), 3.1 above FLOP-matched Success-Xfer-MF (successful-behavior transfer), 3.2 above four-call Orbit-TTA (test-time ensembling), and 1.3 above LocalCanon-GRPO (current-schema

Table 2: Strong-baseline comparison on the common six-benchmark endpoint. ORBITPO attains the highest WVA and AV among registry-free methods on both backbones. AV is all-view completion. FullCanon receives the complete correspondence registry at inference and is therefore separated as an oracle; its equal clean and exact-view WVA values are repeated in their respective columns. Variability and additional baselines are in the appendix.

	Qwen3-8B			Mistral Small 3.2 24B		
Method	**Clean**	**WVA**	**AV**	**Clean**	**WVA**	**AV**
GRPO	62.7	36.2	29.5	64.8	40.5	33.1
PartialCanon-GRPO	62.1	42.6	34.3	66.0	47.2	39.2
Aug-GRPO	63.1	47.5	41.2	66.5	51.6	44.6
GroupDRO	61.6	51.8	46.9	64.3	55.8	50.2
Adv.-transform	60.9	52.4	47.2	64.4	55.0	49.1
Orbit-TTA ($4\times$ inference)	63.5	52.7	48.5	**67.3**	56.6	52.0
Success-Xfer-MF	63.8	52.8	49.6	66.4	57.2	53.4
LocalCanon-GRPO	63.5	54.6	50.8	67.1	58.3	54.0
ORBITPO	**64.0**	**55.9**	**53.6**	67.2	**60.4**	**56.6**
FullCanon oracle	63.2	63.2	60.8	67.0	67.0	64.8

Table 3: Per-benchmark worst-view aggregate under the shared constrained protocol. Δ is ORBITPO minus Aug-GRPO; all 12 backbone–benchmark contrasts are positive, with the largest margins on AppWorld.

	Qwen3-8B			Mistral Small 3.2 24B		
Benchmark	**Aug W**	**Orbit W**	Δ	**Aug W**	**Orbit W**	Δ
API-Bank	68.0	**73.2**	+5.2	72.4	**81.1**	+8.7
ACEBench	56.7	**64.1**	+7.4	61.7	**71.2**	+9.5
RoTBench	57.3	**67.4**	+10.1	59.5	**66.7**	+7.2
AppWorld	27.3	**39.6**	+12.3	30.8	**41.5**	+10.7
τ-bench	34.6	**41.4**	+6.8	39.1	**46.1**	+7.0
ToolSandbox	41.1	**49.7**	+8.6	46.1	**55.8**	+9.7
Macro	47.5	**55.9**	+8.4	51.6	**60.4**	+8.8

canonicalization). On Mistral Small 3.2 24B, it reaches 60.4, with margins of 8.8, 3.2, 3.8, and 2.1 points over the same methods, and it leads all registry-free methods in all-view completion on both backbones. The FullCanon oracle, which receives the complete deployment registry, reaches 63.2 and 67.0 WVA; it is a privileged correspondence control rather than an upper bound on clean accuracy, and ORBITPO requires no registry. Paired methods share tasks, transformation seeds, and view aggregation.

The macro result is not driven by a single benchmark (Table 3): the margin is positive on all six benchmarks for both backbones, ranging from 5.2 to 12.3 points on Qwen3-8B and from 7.0 to 10.7 on Mistral. It is largest on AppWorld (+12.3 and +10.7), a pattern consistent with error propagation through stateful multi-application workflows.

6.2 CORRECT AGREEMENT AND COMPONENT ABLATION

Figure 2 tests whether consistency synchronizes correct actions or merely shared errors. Consistency alone raises agreement but also raises same-wrong outcomes; adding orbit credit raises both-correct outcomes to 68.6%, lowers same-wrong outcomes to 10.0%, and yields 55.9 WVA. The two technical contributions are therefore complementary: alignment (Contribution 1) makes views agree, and orbit credit (Contribution 2) makes the agreed action correct.

6.3 MASS PRESERVATION (H1)

Table 4 fixes auxiliary histories, candidates, rollout episodes, scored tokens, and selection budget, so arms differ only in the objective. Retaining mass (OTHER JSD) improves over conditional JSD by 2.8 WVA points and cuts the mass gap from .063 to .019. Adding Bernoulli mass matching to the

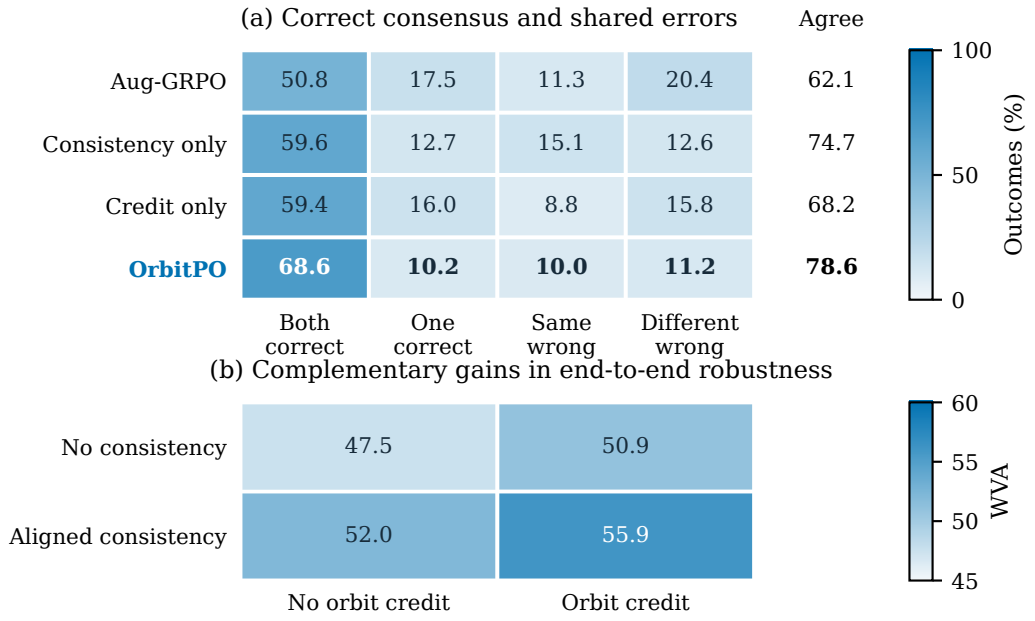


Figure 2: Alignment and orbit credit are complementary. (a) Correct agreement on 8,000 held-out OrbitLab prefixes. (b) Six-benchmark Qwen3-8B 2×2 component factorial at matched rollout and scoring budgets.

Table 4: Coverage- and compute-matched objective ablation on a shared frozen bank. Mass gap is the maximum retained-probability difference between views.

Auxiliary objective	WVA	Mass gap
Conditional JSD	52.9	.063
Conditional + Bernoulli JSD	55.4	.016
Hard frozen teacher	54.1	.038
Soft frozen teacher	54.9	.025
OTHER JSD	**55.7**	.019

Table 5: WVA at matched candidate count, coverage, conditional JSD, and scored-token budget. "Cond.-norm" matches the auxiliary-gradient magnitude of the full objective.

Dispersion	Cond.-norm	+Bernoulli	OTHER	Gain
Low	56.5	57.1	57.4	+0.9
Medium	53.9	**56.0**	55.8	+1.9
High	50.4	53.5	**55.2**	+4.8

conditional objective recovers 2.5 of those points, so the retained-mass term of Equation 7 accounts for most of the gain, as the decomposition predicts. Soft targets add 0.7–0.8 points over hard transfer. The fixed-bank 55.7 does not replace the main 55.9.

The matched operating-regime projection in Table 5 tests H1 directly. With auxiliary-gradient magnitude controlled, OTHER alignment adds 0.9, 1.9, and 4.8 WVA points over Cond.-norm in the low-, medium-, and high-dispersion strata (Appendix Table 60). The gain thus grows with mass dispersion, as the decomposition predicts, rather than acting as a uniform regularizer; the Bernoulli-only control tracks much of this gain and slightly leads OTHER in the medium stratum (56.0 versus 55.8).

6.4 History Coverage and When Replay Matters (H2)

Replay removes the coverage bottleneck of Challenge 2: natural cross-view history matches fall to 12.4% at calls 17–32, whereas every selected exact history is renderable in all charts (Figure 3, left). The projected WVA margin grows with reference-plan length (right). The appendix controls for successful-continuation coverage and scoring FLOPs.

Because longer plans also contain more transformed decisions, the paired OrbitLab intervention fixes the dependency graph, reward, 16-call reference plan, and 20-call budget, and varies only how many reference decisions use transformed tools or handles. Replay and natural matching are projected to be indistinguishable for unused-tool renames (Figure 4), but replay's advantage grows to 4.8 points at 12 consequential decisions; at length 32, the sparse and dense gains are 0.8 and 6.7 points. Replay

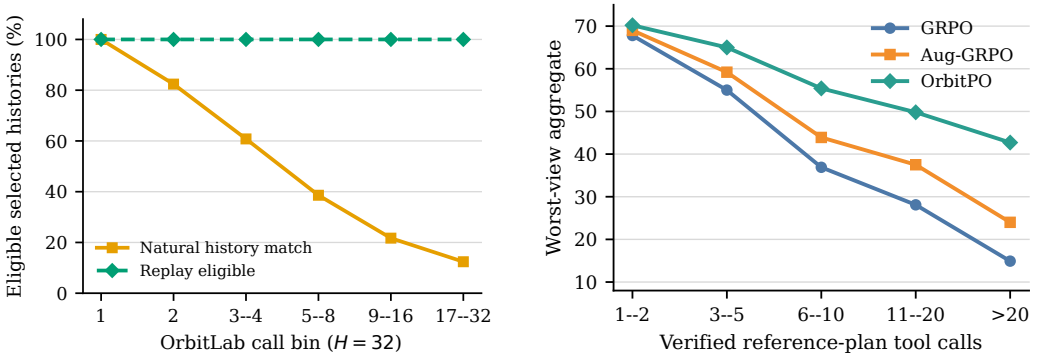


Figure 3: Replay keeps every selected history comparable as natural matching decays. Left: natural history matching versus replay eligibility. Right: WVA by independently verified reference-plan length. The fixed-length causal comparison appears in Figure 4.

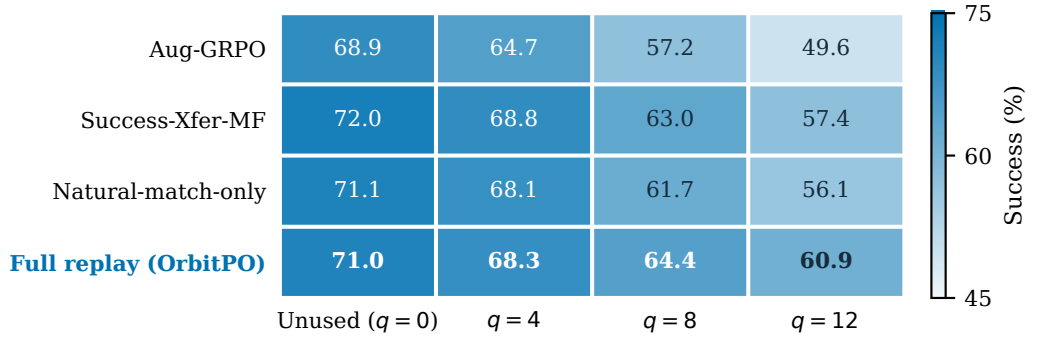


Figure 4: Replay's advantage grows with consequential interface-decision density. Minimum view-level success on the same 800 OrbitLab programs as density increases. Unused-tool changes match visible renaming without changing executed decisions.

Table 6: Compute-aware Qwen3-8B comparison. Training FLOPs are relative to GRPO; latency is seconds per decision on one warm H100 worker. Orbit-TTA applies four-view inference to the Aug-GRPO checkpoint, so it inherits that checkpoint's training cost; at equal FLOPs, LocalCanon-GRPO is 0.8 WVA higher.

Method/control	Rollout episodes	Train FLOPs	Passes/action	p50 latency	WVA
Aug-GRPO	1.00×	1.04×	1	1.29	47.5
LocalCanon-GRPO	1.00×	1.01×	1	1.33	54.6
Success-Xfer-MF	1.00×	1.22×	1	1.31	52.8
Orbit-TTA	1.00×	1.04×	4	3.46	52.7
ORBITPO (equal FLOPs)	0.82×	1.00×	1	1.30	53.8
ORBITPO (equal episodes)	1.00×	1.22×	1	1.32	**55.9**

thus helps as transformed decisions become consequential, not merely as renamed symbols become visible, which supports H2.

6.5 EFFICIENCY, EXTERNAL TRANSFER, AND BOUNDARIES

Table 6 separates training from deployment cost. At GRPO's training FLOPs, ORBITPO reaches 53.8 WVA, retaining 6.3 of its full 8.4-point gain over Aug-GRPO; at this budget, LocalCanon-GRPO remains 0.8 points higher. At equal episodes, ORBITPO uses the same 1.22× FLOPs as Success-Xfer-MF and leads it by 3.1 points. Deployment remains one pass per action, whereas Orbit-TTA uses four, more than doubles median latency, and still trails by 3.2 points.

Figure 5 tests whether training on generic exact transformations extends to migration motifs drawn from adjacent public API versions. Without migration-specific adaptation, the projected migrated

Table 7: External transfer and negative controls. Gains persist under exact changes and vanish under non-commuting shifts. Scores use the native endpoint of each evaluation; Δ is ORBITPO minus Aug-GRPO.

Evaluation	Aug	Orbit	Δ
ToolRobustBench exact subset	45.9	**54.0**	+8.1
Unseen function-call template	51.4	**57.9**	+6.5
Description paraphrase	**49.3**	49.0	−0.3
Type/permission mutation	38.1	38.3	+0.2
AgentChaos runtime faults	**27.9**	27.8	−0.1

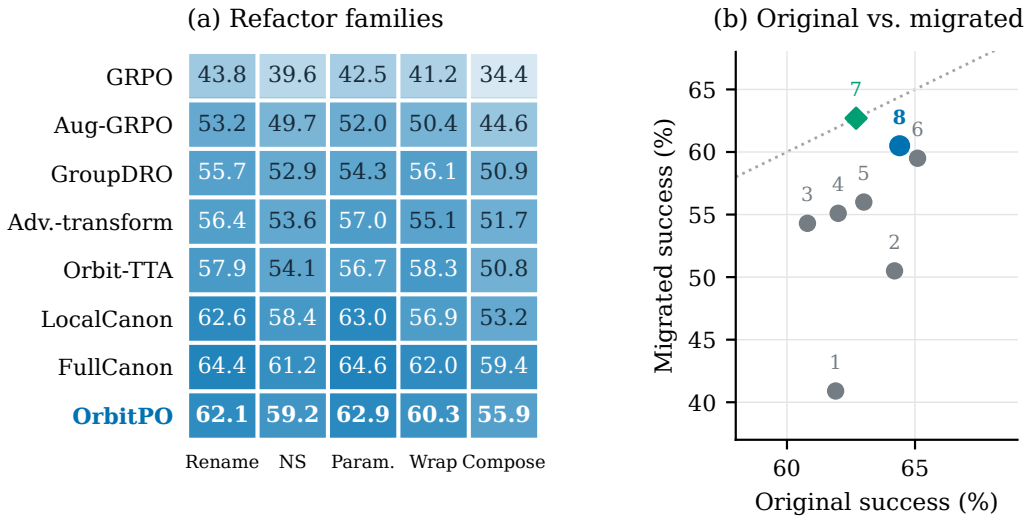


Figure 5: APIRefactor-232 transfer without migration-specific adaptation. The audit contains 1,200 paired tasks from 232 exact refactors in 58 version histories. In (a), NS, Param., Wrap, and Compose denote namespace, parameter, envelope, and composed refactors. Numbers in (b) follow the method order in (a); FullCanon receives the exact registry. LocalCanon is marginally higher than ORBITPO on rename and parameter refactors.

score is 60.5 for frozen ORBITPO versus 50.5 for Aug-GRPO, with an 11.3-point margin on composed refactors; FullCanon, which receives the exact adapter map, remains stronger.

Table 7 delimits the claim. The margin persists on the exact ToolRobustBench subset (+8.1) and on an unseen function-call template (+6.5), whereas paraphrases, type/permission mutations, and runtime faults remain tied (−0.3 to +0.2). This split matches the commutation assumption of Equation 2: semantic and runtime shifts violate it and require separate mechanisms.

7 Discussion and Conclusion

ORBITPO turns exact API refactors from a source of data augmentation into a structured learning constraint. Mass-preserving alignment recovers the signal discarded by candidate conditioning, with gains growing from 0.9 to 4.8 WVA points as mass dispersion increases (Contribution 1); orbit credit and replay make agreement correct and keep views comparable after divergence, with gains growing to 4.8 points as decisions become consequential (Contribution 2); and WVA improves on all nine backbones and all twelve primary benchmark contrasts at one call per decision (Contribution 3).

The method targets exact, invertible coordinate changes; semantic paraphrases, permission mutations, runtime faults, and actions outside the shared decoder support are distinct regimes in which it shows no gain. The guarantees hold under the stated commutation and history-coverage assumptions, which the held-out audit checks on the finite evaluation banks but does not establish for all benchmark histories. The additional suites evaluate frozen transfer; the six-benchmark endpoint is the primary comparison.

AI USE STATEMENT

We used LLMs as a general-purpose writing and editing assist tool during the preparation of this submission. Specifically, LLMs were employed for polishing the clarity and readability of text (e.g., refining sentence structure, improving grammar, and shortening overly verbose phrasing) and for suggesting formatting adjustments to tables, figures, and page layout. All research ideas, methodology design, experiments, analyses, and final writing decisions were conceived, implemented, and validated solely by the authors, who reviewed all LLM-assisted changes.

ETHICS STATEMENT

The experimental protocol confines agent actions and executable interventions to resettable simulators and benchmark sandboxes. Goal, world, and route modifications operate on benchmark state within these environments, with resets separating counterfactual branches from the original trajectory. The protocol excludes interventions on live user accounts or production services and does not require private user data. Third-party datasets, models, and tools remain subject to their respective licenses and access conditions. For applications beyond the evaluation environments, execution must respect user authorization and data-access boundaries, with human review for consequential actions. Learned state representations complement these operational safeguards; they do not confer additional execution permissions.

Broader impact. Interface-equivariant agents could reduce accidental failures when APIs are refactored or localized. The same capability can also make automated agents more portable across services, which increases the importance of permission checks, rate limits, and human confirmation for consequential actions. The evaluated transformations preserve environment permissions and never turn an unauthorized action into an authorized one. This work does not evaluate deployment safety, privacy, prompt injection, or malicious tool descriptions.

These boundaries are operational requirements for using the method. An exact interface map may be applied only when the renderer has been verified to preserve the task contract; a visually similar schema with changed semantics is outside the equivariance assumption. Deployments should therefore validate tool schemas, argument permissions, and returned-entity ownership independently of the policy score. The method does not replace authentication, authorization, rate limiting, or human confirmation for actions with external consequences.

The experiments use resettable simulators and benchmark sandboxes, with fixed evaluation banks and paired seeds. This makes the reported comparisons auditable, but it does not establish safety or reliability on live services, private data, or adversarially modified tools. In particular, robustness to an invertible rename does not imply robustness to prompt injection, deceptive descriptions, changed business rules, distribution shift, or an unavailable mapping. Those cases need separate tests and operational safeguards before any deployment decision.

Finally, improved transfer across interfaces can increase the scale at which an agent acts. We recommend least-privilege credentials, resettable staging environments, explicit action logs, and a human approval step for irreversible operations. These controls are independent of the learned representation and remain necessary even when the policy satisfies the measured interface tests.

REPRODUCIBILITY STATEMENT

The supplementary package implements ORBITPO in PyTorch on a finite-action OrbitLab task with four invertible interface views. It includes orbit-relative advantages, clipped policy updates, mass-preserving alignment with residual probability, complete-history replay, candidate pooling, a prior-action reservoir, and fixed-reference KL regularization. Unit tests verify cross-view credit, the single-candidate alignment signal, and action correspondence after history rendering. The README provides installation, test, and training commands, with configurable seeds, learning rates, update counts, and CPU or GPU execution. Runs save policy weights and CSV metrics covering success, clean and worst-view performance, and objective terms. Machine-readable table cells, captions, figure data, and export scripts support inspection of the reported results. The main text and appendix document the objectives, proofs, experimental settings, and evaluation definitions.

REFERENCES

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization, 2019. URL https://arxiv.org/abs/1907.02893.

Victor Barres, Honghua Dong, Soham Ray, Xujie Si, and Karthik Narasimhan. τ^2-bench: Evaluating conversational agents in a dual-control environment, 2025. URL https://arxiv.org/abs/2506.07982.

Berkeley Function Calling Leaderboard Team. Berkeley Function Calling Leaderboard: V4 evaluation documentation, 2026. URL https://github.com/ShishirPatil/gorilla/tree/main/berkeley-function-call-leaderboard. Official documentation snapshot, accessed September 23, 2026.

Chen Chen, Xinlong Hao, Weiwen Liu, Xu Huang, Xingshan Zeng, Shuai Yu, Dexun Li, Shuai Wang, Weinan Gan, et al. ACEBench: Who wins the match point in tool usage?, 2025. URL https://arxiv.org/abs/2501.12851.

Taco S. Cohen and Max Welling. Group equivariant convolutional networks, 2016. URL https://arxiv.org/abs/1602.07576.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, et al. The Llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Ruocheng Guo, Kaiwen Dong, Xiang Gao, and Kamalika Das. Learning to rewrite tool descriptions for reliable LLM-agent tool use, 2026. URL https://arxiv.org/abs/2602.20426.

Chengrui Huang, Zhengliang Shi, Yuntao Wen, Xiuying Chen, Peng Han, Shen Gao, and Shuo Shang. What affects the stability of tool learning? an empirical study on the robustness of tool learning frameworks, 2024. URL https://arxiv.org/abs/2407.03007.

David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville. Out-of-distribution generalization via risk extrapolation (REx), 2020. URL https://arxiv.org/abs/2003.00688.

Junlong Li, Wenshuo Zhao, Jian Zhao, Weihao Zeng, Haoze Wu, Xiaochen Wang, Rui Ge, Yuxuan Cao, Yuzhen Huang, Wei Liu, Junteng Liu, Zhaochen Su, Yiyang Guo, Fan Zhou, Lueyang Zhang, Juan Michelini, Xingyao Wang, Xiang Yue, Shuyan Zhou, Graham Neubig, and Junxian He. The tool decathlon: Benchmarking language agents for diverse, realistic, and long-horizon task execution, 2025. URL https://arxiv.org/abs/2510.25726.

Minghao Li, Yingxiu Zhao, Bowen Yu, Feifan Song, Hangyu Li, Haiyang Yu, Zhoujun Li, Fei Huang, and Yongbin Li. API-Bank: A comprehensive benchmark for tool-augmented LLMs, 2023. URL https://arxiv.org/abs/2304.08244.

Xiaobo Liang, Lijun Wu, Juntao Li, Yue Wang, Qi Meng, Tao Qin, Wei Chen, Min Zhang, and Tie-Yan Liu. R-Drop: Regularized dropout for neural networks, 2021. URL https://arxiv.org/abs/2106.14448.

Weiwen Liu, Xu Huang, Xingshan Zeng, Xinlong Hao, Shuai Yu, Dexun Li, Shuai Wang, Weinan Gan, et al. ToolACE: Winning the points of LLM function calling, 2024a. URL https://arxiv.org/abs/2409.00920.

Zuxin Liu, Thai Hoang, Jianguo Zhang, Ming Zhu, Tian Lan, Shirley Kokane, Juntao Tan, Weiran Yao, Zhiwei Liu, et al. APIGen: Automated pipeline for generating verifiable and diverse function-calling datasets, 2024b. URL https://arxiv.org/abs/2406.18518.

Jiarui Lu, Thomas Holleis, Yizhe Zhang, Bernhard Aumayer, Feng Nan, Felix Bai, Shuang Ma, Shen Ma, Mengyu Li, Guoli Yin, Zirui Wang, and Ruoming Pang. ToolSandbox: A stateful, conversational, interactive evaluation benchmark for LLM tool use capabilities, 2024. URL https://arxiv.org/abs/2408.04682.

Mistral AI. Mistral Small 3.2, 2025. URL https://huggingface.co/mistralai/Mistral-Small-3.2-24B-Instruct-2506.

Ishan Patel, Sahil Sen, Elias Lumer, and Vamse Kumar Subbiah. The bitter lesson of tool calling, 2026. URL https://arxiv.org/abs/2608.06370.

Akshara Prabhakar, Zuxin Liu, Ming Zhu, Jianguo Zhang, Tulika Awalgaonkar, Shiyu Wang, Zhiwei Liu, Haolin Chen, et al. APIGen-MT: Agentic pipeline for multi-turn data generation via simulated agent-human interplay, 2025. URL https://arxiv.org/abs/2504.03601.

Cheng Qian, Emre Can Acikgoz, Qi He, Hongru Wang, Xiusi Chen, Dilek Hakkani-Tür, Gokhan Tur, and Heng Ji. ToolRL: Reward is all tool learning needs, 2025. URL https://arxiv.org/abs/2504.13958.

Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, et al. ToolLLM: Facilitating large language models to master 16000+ real-world APIs, 2023. URL https://arxiv.org/abs/2307.16789.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026a. URL https://qwen.ai/blog?id=qwen3.5.

Qwen Team. Qwen3.6-35B-A3B: Agentic coding power, now open to all, April 2026b. URL https://qwen.ai/blog?id=qwen3.6-35b-a3b.

Qwen Team. Qwen3.8-Max: A new bar for coding and cowork, August 2026c. URL https://qwen.ai/blog?id=qwen3.8.

Ella Rabinovich and Ateret Anaby-Tavor. On the robustness of agentic function calling, 2025. URL https://arxiv.org/abs/2504.00914.

Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization, 2019. URL https://arxiv.org/abs/1911.08731.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools, 2023. URL https://arxiv.org/abs/2302.04761.

Quan Shi, Alexandra Zytek, Pedram Razavi, Karthik Narasimhan, and Victor Barres. τ-knowledge: Evaluating conversational agents over unstructured knowledge. *arXiv preprint arXiv:2603.04370*, 2026.

Gou Tan, Zhensu Sun, Jieke Shi, Ting Zhang, Zilong He, Qingfu Wu, Shuai Liang, Weifeng Sun, et al. AgentChaos: Chaos engineering for agent systems via programmatic fault injection, 2026. URL https://arxiv.org/abs/2608.06790.

Harsh Trivedi, Tushar Khot, Mareike Hartmann, Ruskin Manku, Vinty Dong, Edward Li, Shashank Gupta, Ashish Sabharwal, and Niranjan Balasubramanian. AppWorld: A controllable world of apps and people for benchmarking interactive coding agents, 2024. URL https://arxiv.org/abs/2407.18901.

Elise van der Pol, Daniel E. Worrall, Herke van Hoof, Frans A. Oliehoek, and Max Welling. MDP homomorphic networks: Group symmetries in reinforcement learning. In *Advances in Neural Information Processing Systems*, 2020. doi: 10.48550/arXiv.2006.16908. URL https://arxiv.org/abs/2006.16908.

Can Wang, Haoran Chen, Li Yu, Ding Hao, Bohai Zhao, Zhaoyang Liu, and Zhiying Tu. Towards robust tool use in agents via experience-driven adaptive guidance, 2026. URL https://arxiv.org/abs/2608.03403.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, et al. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. τ-bench: A benchmark for tool-agent-user interaction in real-world domains, 2024. URL https://arxiv.org/abs/2406.12045.

Junjie Ye, Yilong Wu, Songyang Gao, Caishuang Huang, Sixian Li, Guanyu Li, Xiaoran Fan, Qi Zhang, Tao Gui, and Xuanjing Huang. RoTBench: A multi-level benchmark for evaluating the robustness of large language models in tool learning, 2024. URL https://arxiv.org/abs/2401.08326.

Z.ai. GLM-4.7 and GLM-4.7-Flash: Official model documentation, 2026. URL https://github.com/zai-org/GLM-4.5. Official documentation snapshot, accessed September 23, 2026.

YiShan Zheng, Yuan Wu, and Yi Chang. ToolRobustBench: Stage-wise perturbation evaluation and failure diagnosis for tool-calling agents, 2026. URL https://arxiv.org/abs/2608.23635.

Haitian Zhong, Jixiu Zhai, Lei Song, Jiang Bian, Qiang Liu, and Tieniu Tan. RC-GRPO: Reward-conditioned group relative policy optimization for multi-turn tool calling agents, 2026. URL https://arxiv.org/abs/2602.03025.

APPENDIX GUIDE

The appendices proceed from formal foundations to external evaluation and are organized so that each main-text contribution can be traced to its supporting material: **foundations and implementation** (Appendices A, B, and C); **environment and evaluation protocol** (Appendices D and E); **complete benchmark and efficiency results** (Appendix F); **mechanisms and operating conditions** (Appendices G and H); and **external transfer and failure analysis** (Appendix I). Contribution 1 (mass-preserving alignment) is supported by Appendices A.11, A.13, and H; Contribution 2 (orbit credit and complete-history replay) by Proposition 7, Theorem 2, and Appendix G.2; and Contribution 3 (experiments) by Appendices F and I. Notation is collected below for reference.

NOTATION TABLE

Table 8 collects the main symbols, grouped by their role in the task definition, the policy and alignment objective, orbit-relative optimization, and the guarantees. Lowercase c denotes a canonical semantic action, C_x a candidate set, and uppercase C a random semantic action. The symbol τ is overloaded: it denotes a trajectory in $R(\tau)$ and a decoding temperature in the superscript π_θ^τ.

Table 8: Notation for the task, executable policy, optimization objective, and theoretical analysis.

Symbol	Meaning
Task and interface transformations	
s_t, o_t, a_t, h_t	Latent state, observation, structured action, and complete interaction history at decision t.
$\mathcal{T}, \mathcal{H}, \mathcal{A}(\mathcal{T})$	Tool interface, history space, and completed structured-action space.
$x = (h, \mathcal{T}), X$	A task at a specified history and interface, and its random-variable counterpart.
$P, R(\tau), R_{\max}$	Next-observation kernel, episode return, and upper bound on episode return.
H, L	Episode decision horizon and maximum tokens per generated action ($L = 1024$).
$\mathfrak{G}, g, e$	Finite action groupoid, an applicable invertible interface transformation, and identity transformation.
$gx, ga, g_{\#}^{-1}\pi$	Transformed task, transformed action, and policy pulled back to canonical semantic actions.
$\mathcal{O}(x), \mathcal{B}_x$	Connected task orbit and finite evaluation view bank containing identity.
$\ell, \ell_\star$	Compiler path length and shortest primitive length of the final induced map, respectively.

Continued on next page

Table 8 (continued)

Symbol	Meaning
Policy and mass-preserving alignment	
θ, π_θ^τ	Model parameters and executed stochastic structured-action policy at temperature τ (default 0.7).
d_θ^{gr}	Token-wise greedy decision rule from untempered masked logits.
$\rho_{\mathcal{T}}, P_\theta^{\tau,\mathrm{con}}$	Canonical action renderer and autoregressive sequence law under the constrained decoder.
$\perp_L$	Explicit failure outcome for unfinished generation at the token cap.
$\mathcal{A}_L(\mathcal{T}), S_L(x)$	Augmented action space including $\perp_L$, and actions completing within L in every chart, respectively.
$C_x, \widetilde{C}_x$	Semantic candidate set and its coarsening $C_x \cup \{\mathrm{OTHER}\}$.
$p_\theta^\tau(c \mid x), s_\theta^\tau$	Unconditional completed-action probability and its log score in the specified chart.
$z_{\theta,i}, \widetilde{q}_{\theta,i}$	Retained candidate mass in view i and distribution over candidates plus OTHER.
OTHER	Aggregate of omitted completed actions and the length-cap failure outcome; its probability is $1 - z_{\theta,i}$.
$\bar{q}, \mathrm{sg}$	Mean coarsened policy across views and stop-gradient operator.
$G, C, \widetilde{C}, Y$	Random view, canonical semantic action, coarsened action, and rendered token sequence.
$I(G; \widetilde{C} \mid x, C_x)$	Conditional mutual information between view identity and the coarsened semantic action, measured in nats.
$r_i, \bar{z}, w_i, r_\star$	Candidate-conditional policy, mean retained mass, weights $z_i/(m\bar{z})$, and mass-weighted candidate barycenter.
D_z, B_z, a	Cross-view retained-mass variance, Bernoulli information, and an interior-mass bound used in Proposition 10.
$D_{\mathrm{KL}}, \mathrm{JSD}, \mathrm{TV}$	Kullback–Leibler divergence, generalized Jensen–Shannon divergence, and total variation distance.
Orbit-relative optimization	
$m, K, B = mK$	Views per semantic task, rollouts per view, and total rollout episodes per task.
$R_{i,k}, \mu_x, \sigma_x$	Return of rollout k in view i, and return mean and standard deviation across its task orbit.
$A_{i,k}^{\mathrm{orb}}, \epsilon$	Orbit-normalized advantage and positive denominator stabilizer.
$r_{i,k,t}(\theta), \epsilon_c$	Token-level policy likelihood ratio and PPO clipping parameter; t here indexes generated tokens.
$\mathcal{L}_{\mathrm{orbRL}}, \mathcal{L}_{\mathrm{eq}}$	Clipped orbit-relative policy loss and mass-preserving equivariance loss.
$\mathcal{R}_{\mathrm{ref}}, \beta, \lambda_{\mathrm{eq}}$	Reference token-conditional KL regularizer, its coefficient, and equivariance-loss coefficient.
Guarantees and evaluation	
$J_e, J_g, J_{\mathcal{T}}$	Expected score under identity, view g, or interface $\mathcal{T}$, with the policy supplied by context.
$\delta_t(g), \varepsilon_t$	Pulled-back policy TV at a coupled history and an almost-sure per-step upper bound.
κ_t, ζ_t	Bounds on transformed transition-kernel TV and aligned reward mismatch.
$d_\pi(g)$	Supremum of the policy equivariance defect over legal histories for transformation g.
η_t, M_t, I_t, b_t	Maximum candidate missing mass, maximum view-to-barycenter KL, coarsened information, and resulting pairwise TV bound.
ν_t, d_t^{ij}, χ_t	Reference replay-history law, pairwise coupled-history law, and their density-ratio bound in Theorem 2.
$\bar{\eta}_t, \bar{I}_t, u_t$	Replay-law expectations of missing mass and information, and the resulting conditional disagreement bound.
T_A, L_A^{ij}, U_A^{ij}	Candidate coarsening and the lower/upper full-policy-TV bounds that tighten under refinement.

Continued on next page

Table 8 (continued)

Symbol	Meaning
$V_t^\star, Q_t^\star, \gamma$	Optimal value, optimal action value, and semantic action-MAP probability margin.
$Z, \xi, \Sigma_G, \Sigma_\xi$	Score-function contribution, rollout randomness, between-view covariance, and mean within-view covariance.
$\mu(\cdot \mid x), \nu(\cdot \mid x)$	Two evaluation distributions over legal views, distinct from the orbit return mean μ_x.
WVA, TWA	Minimum view score after task aggregation, and mean task-wise minimum over views.
CVaR_{10}, $S(t)$	Mean of the worst 10% of view scores, and probability of remaining semantically coupled through call t.

A Mathematical Details

A.1 Proof of Theorem 1

The proof proceeds in two steps: it first shows that optimal values are invariant under admitted maps and then constructs an optimal policy that is itself equivariant. Let $V_t^\star(h, \mathcal{T})$ and $Q_t^\star(h, \mathcal{T}, a)$ denote optimal finite-horizon values with $H - t + 1$ steps remaining, treating the bounded episode return as a terminal reward if necessary. We first show by backward induction that

$$V_t^\star(gh, g\mathcal{T}) = V_t^\star(h, \mathcal{T}), \qquad Q_t^\star(gh, g\mathcal{T}, ga) = Q_t^\star(h, \mathcal{T}, a). \tag{10}$$

The claim is immediate after the terminal step. Suppose it holds at $t + 1$. Reward invariance and transition commutation in Equation 2 give

$$Q_t^\star(gh, g\mathcal{T}, ga) = r(gh, g\mathcal{T}, ga) + \mathbb{E}_{o' \sim P(\cdot \mid gh, ga, g\mathcal{T})} V_{t+1}^\star(gh \circ o', g\mathcal{T}) \tag{11}$$

$$= r(h, \mathcal{T}, a) + \mathbb{E}_{o \sim P(\cdot \mid h, a, \mathcal{T})} V_{t+1}^\star(h \circ o, \mathcal{T}) = Q_t^\star(h, \mathcal{T}, a). \tag{12}$$

Maximizing over the bijectively mapped action sets proves the identity for $V_t^\star$.

It remains to select an equivariant optimal policy. Partition histories into connected components of $\mathfrak{G}$ and choose one representative h_0 per component. Let $S_{h_0} = \{g : gh_0 = h_0\}$ be its finite automorphism stabilizer. Choose any optimal action a_0 and define a distribution that is uniform over the finite orbit $\{sa_0 : s \in S_{h_0}\}$. Every action in this support is optimal by Equation 10, and the distribution is invariant under the stabilizer. Transport it to every gh_0 by the pushforward $g_\#$. Stabilizer invariance makes this definition independent of the chosen g, and the resulting policy is equivariant and selects only Bellman-optimal actions. Applying the construction at every time step proves the result for every initial-task distribution; because optimality holds pointwise, no invariant initial distribution is required. □

A.2 Proof of Proposition 2

Proposition 2 (Horizon amplification)**.** *For histories coupled through decision t, let $\delta_t(g) = \text{TV}(\pi(\cdot \mid h_t, \mathcal{T}), g_\#^{-1}\pi(\cdot \mid gh_t, g\mathcal{T}))$. If episode return lies in $[0, R_{\max}]$ and $\delta_t(g) \le \varepsilon_t$ almost surely, then*

$$|J_{\mathcal{T}}(\pi) - J_{g\mathcal{T}}(\pi)| \le R_{\max}\left[1 - \prod_{t=1}^{H}(1 - \varepsilon_t)\right] \le R_{\max}\sum_{t=1}^{H}\varepsilon_t. \tag{13}$$

Under sequential maximal coupling, the same expression holds with $\bar{\delta}_t = \mathbb{E}[\delta_t(g) \mid \text{coupled through decision } t]$ in place of the envelopes. For every H and constant δ, an exact coordinate process attains the product bound.

At a pair of histories that have remained related by g, use a maximal coupling between $\pi(\cdot \mid h_t, \mathcal{T})$ and $g_\#^{-1}\pi(\cdot \mid gh_t, g\mathcal{T})$. The probability that the aligned actions disagree at step t, conditioned on previous agreement, is $\delta_t(g)$. When actions agree, transition commutation permits a coupling of the

next observations that preserves the relation exactly. If $\delta_t(g) \le \varepsilon_t$ almost surely, induction therefore gives

$$\Pr(\text{the paired trajectories never disagree}) \ge \prod_{t=1}^{H}(1-\varepsilon_t). \tag{14}$$

If no disagreement occurs, reward invariance makes the returns equal. If a disagreement occurs, their absolute return difference is at most $R_{\max}$. This proves the product bound and its union-bound relaxation. Without deterministic envelopes, let $\bar{\delta}_t = \mathbb{E}[\delta_t(g) \mid \text{coupled through } t]$ under this sequential maximal coupling. Conditional survival at step t is exactly $1-\bar{\delta}_t$, so recursion gives survival probability $\prod_t(1-\bar{\delta}_t)$ and hence the corresponding product and additive bounds in Proposition 2.

To show tightness, consider a length-H semantic process with "continue" and "fail" actions, an absorbing failure state, and terminal reward $R_{\max}$ only if every action was "continue." Let g bijectively rename both actions and let the environment commute with this renaming. On the identity rendering, define the policy to continue surely; on the transformed rendering, let its pulled-back policy continue with probability $1-\delta$ independently at each coupled history. Then $\delta_t(g) = \delta$, while the two expected returns are $R_{\max}$ and $R_{\max}(1-\delta)^H$. Their gap is exactly $R_{\max}[1-(1-\delta)^H]$. □

The bound can be tightened by replacing $R_{\max}$ at step t with the range of the remaining return; the stated form is retained because it makes no assumption about reward decomposition. The product form also explains why small per-step equivariance defects compound over long episodes, the regime in which Challenge 2 is most severe.

A.3 Policy–Contract Decomposition

Proposition 3 (Policy–contract decomposition). *If the per-step reward sum is in $[0, R_{\max}]$, and coupled-step policy TV, transition TV, and aligned reward mismatch are bounded by $\varepsilon_t, \kappa_t, \zeta_t$, respectively, then*

$$|J_{\mathcal{T}} - J_{g\mathcal{T}}| \le \sum_t \zeta_t + R_{\max}[1 - \prod_t (1-\varepsilon_t)(1-\kappa_t)]. \tag{15}$$

At a pair of histories still related by g, maximally couple the policy actions. They agree after inverse mapping with conditional probability at least $1-\varepsilon_t$. Conditional on agreement, maximally couple the mapped and transformed next-observation kernels; the next histories remain related with probability at least $1-\kappa_t$. Repeating this construction yields

$$\Pr(\text{state–action coupling survives through } H) \ge \prod_{t=1}^{H}(1-\varepsilon_t)(1-\kappa_t). \tag{16}$$

On the survival event, the absolute accumulated reward discrepancy is at most $\sum_t \zeta_t$. On its complement, the return difference is at most $R_{\max}$. Taking expectations and upper-bounding the survival-event reward term by $\sum_t \zeta_t$ proves Equation 15. The additive relaxation $\sum_t \zeta_t + R_{\max}\sum_t(\varepsilon_t + \kappa_t)$ follows from the union bound. Setting $\kappa_t = \zeta_t = 0$ recovers Proposition 2. □

A.4 Proof of Proposition 4

Proposition 4 (Uniform generator-to-composition bound). *Define $d_\pi(g) = \sup_x \mathrm{TV}(\pi(\cdot \mid x), g_{\#}^{-1}\pi(\cdot \mid gx))$, with the supremum over legal histories in $\mathrm{dom}(g)$. For a composable generator word $g = s_\ell \cdots s_1$,*

$$d_\pi(g) \le \sum_{j=1}^{\ell} d_\pi(s_j). \tag{17}$$

A uniform generator bound ε therefore certifies $d_\pi(g) \le \ell\varepsilon$; a maximum over finitely many sampled states cannot establish this premise.

For composable morphisms $u, v \in \mathfrak{G}$, insert the intermediate distribution $v_{\#}^{-1}\pi(\cdot \mid vx)$ and use invariance of total variation under a bijection:

$$d_\pi(uv) = \sup_x \mathrm{TV}\Big(\pi(\cdot \mid x), v_{\#}^{-1}u_{\#}^{-1}\pi(\cdot \mid uvx)\Big) \tag{18}$$

$$\leq \sup_x \mathrm{TV}\Big(\pi(\cdot \mid x), v_{\#}^{-1}\pi(\cdot \mid vx)\Big) \tag{19}$$

$$+ \sup_x \mathrm{TV}\Big(v_{\#}^{-1}\pi(\cdot \mid vx), v_{\#}^{-1}u_{\#}^{-1}\pi(\cdot \mid uvx)\Big) \tag{20}$$

$$\leq d_\pi(v) + d_\pi(u). \tag{21}$$

Applying this subadditivity recursively to $g = s_\ell \cdots s_1$ gives Equation 17. □

A.5 Proof of Proposition 5

Proposition 5 (Semantic MAP stability under a margin)**.** *Let r be a reference distribution over canonical semantic actions with unique mode $c^\star$ and margin $\gamma = r(c^\star) - \sup_{c \neq c^\star} r(c)$. If every applicable view has full pulled-back semantic policy p_g satisfying $\mathrm{TV}(p_g, r) < \gamma/2$, then action-level MAP decoding in every view returns $gc^\star$. This does not imply stability of token-wise greedy decoding, whose prefix trees can differ across serializations.*

Write $\delta_g = \mathrm{TV}(p_g, r) < \gamma/2$. Total variation bounds every singleton discrepancy, so $p_g(c^\star) > r(c^\star) - \gamma/2$. For any $c \neq c^\star$, $p_g(c) < r(c) + \gamma/2 \leq r(c^\star) - \gamma/2$. Hence $c^\star$ is the unique mode of every pulled-back policy p_g, and pushforward returns $gc^\star$ in chart g. The statement concerns action-level MAP only. A token-wise greedy decoder instead maximizes conditional masses of successive prefix partitions; coordinate-specific serialization trees can change those partitions even when leaf probabilities are identical. □

A.6 Finite-View Variance and the Coupled Estimator

Proposition 6 (Finite-view variance at fixed rollout budget)**.** *At a fixed task and parameter vector, let $Z(G, \xi)$ be an unbiased score-function contribution, $\mu_G = \mathbb{E}_\xi[Z \mid G]$, $\Sigma_G = \mathrm{Cov}_G(\mu_G)$, and $\Sigma_\xi = \mathbb{E}_G \mathrm{Cov}_\xi(Z \mid G)$. Sample m independent coordinates and K conditionally independent rollouts per coordinate, with total budget $B = mK$. Their average $\widehat{Z}_{m,K}$ obeys*

$$\mathrm{Cov}(\widehat{Z}_{m,K}) = \frac{\Sigma_G}{m} + \frac{\Sigma_\xi}{mK} = \frac{\Sigma_G}{m} + \frac{\Sigma_\xi}{B}. \tag{22}$$

At fixed B, broader coordinate coverage reduces the between-coordinate component; sampling without replacement adds the finite-population correction.

Write $Z_{jk} = \mu + U_j + E_{jk}$, where $U_j = \mu_{G_j} - \mu$, $\mathbb{E}[U_j] = 0$, and $E_{jk} = Z(G_j, \xi_{jk}) - \mu_{G_j}$ has conditional mean zero. Independent coordinate draws make the U_j independent, and conditionally independent rollouts make all residual cross-covariances zero. Therefore

$$\mathrm{Cov}\left(\frac{1}{mK}\sum_{j=1}^{m}\sum_{k=1}^{K} Z_{jk}\right) = \frac{mK^2}{m^2K^2}\Sigma_G + \frac{mK}{m^2K^2}\Sigma_\xi = \frac{\Sigma_G}{m} + \frac{\Sigma_\xi}{mK}. \tag{23}$$

This is Equation 22; iterated expectation establishes unbiasedness. For a finite bank of M coordinates sampled uniformly without replacement, the covariance of the sample mean of coordinate effects is the usual $(1 - m/M)S_G/m$, where S_G uses the finite-population covariance convention. Rollout noise retains the Σ_ξ/B term. □

For the default identity/rename/order/composition allocation, the same argument is applied within each sampling stratum and the stratum-weighted covariance terms are summed; the four heterogeneous strata are not iid. The separate analytic-curve diagnostic in Table 35 instead samples coordinates independently with replacement from one balanced bank, without forcing identity into every batch, so Equation 22 applies directly. It does not describe the default stratified allocation.

For the task-gradient component of the update, write $Z_{\mathrm{tot}} = Z_{\mathrm{orbRL}} + \lambda_{\mathrm{eq}} Z_{\mathrm{eq}}$ and $Z_{\mathrm{orbRL}} = B^{-1} \sum_{b=1}^{B} \psi_b$, where each ψ_b already contains the shared orbit-normalized advantage. Exact covariance accounting gives

$$\mathrm{Cov}(Z_{\mathrm{tot}}) = \mathrm{Cov}(Z_{\mathrm{orbRL}}) + \lambda_{\mathrm{eq}}^2 \mathrm{Cov}(Z_{\mathrm{eq}}) + \lambda_{\mathrm{eq}} \,\mathrm{SymCov}(Z_{\mathrm{orbRL}}, Z_{\mathrm{eq}}), \tag{24}$$

$$\mathrm{Cov}(Z_{\mathrm{orbRL}}) = B^{-2} \left[\sum_{b} \mathrm{Cov}(\psi_b) + \sum_{b \neq b'} \mathrm{Cov}(\psi_b, \psi_{b'}) \right], \tag{25}$$

where $\mathrm{SymCov}(U, V) = \mathrm{Cov}(U, V) + \mathrm{Cov}(V, U)$. The off-diagonal sum contains the dependence induced by common task, return normalization, and shared rollouts. No term has a universal sign, so Proposition 6 alone does not imply lower variance for the coupled update; Table 41 therefore measures every term empirically at one shared parameter point. This decomposition excludes the reference-KL gradient and is before gradient clipping and Adam preconditioning; it is not a covariance identity for the complete optimizer step. The same exclusions apply to UpdateVar.

A.7 Proof of Proposition 1

This certificate connects the implemented coarsened objective of Contribution 1 to cross-view return. Let $\widetilde{q}_{t,i}$ be the exact coarsening of $p_{t,i}$ onto $C_t \cup \{\mathrm{OTHER}\}$. Coarsening preserves every candidate probability and the total tail mass. For any pair, expanding OTHER can add at most $\min\{p_{t,i}(C_t^c), p_{t,j}(C_t^c)\} \leq \eta_t$ to total variation. Hence Pinsker's inequality and the triangle inequality through the coarsened barycenter give

$$\mathrm{TV}(p_{t,i}, p_{t,j}) \leq \eta_t + \mathrm{TV}(\widetilde{q}_{t,i}, \widetilde{q}_{t,j}) \tag{26}$$

$$\leq \eta_t + \sqrt{\tfrac{1}{2} D_{\mathrm{KL}}(\widetilde{q}_{t,i} \| \bar{q}_t)} + \sqrt{\tfrac{1}{2} D_{\mathrm{KL}}(\widetilde{q}_{t,j} \| \bar{q}_t)} \tag{27}$$

$$\leq \eta_t + \sqrt{2 M_t}. \tag{28}$$

Total variation is at most one, and $M_t \leq \sum_i D_{\mathrm{KL}}(\widetilde{q}_{t,i} \| \bar{q}_t) = m I_t$, proving Equation 9. Substitution into Proposition 2 gives the return statement. Averaging over two uniformly sampled views and applying Jensen also gives the sharper mean-view bound $\mathbb{E}_{i,j}\, \mathrm{TV}(p_{t,i}, p_{t,j}) \leq \eta_t + \sqrt{2 I_t}$.

For the quantile statement, let $D_G = D_{\mathrm{KL}}(\widetilde{q}_{t,G} \| \bar{q}_t)$. Because a uniform view satisfies $\mathbb{E}_G D_G = I_t$, Markov's inequality gives $\Pr(D_G > I_t/\alpha) \leq \alpha$. On the complementary set, Pinsker's inequality and the coarsening expansion bound yield the corresponding pairwise quantile certificate with one additive η_t term. In coarsened space,

$$\mathrm{TV}(\widetilde{q}_{t,G}, \bar{q}_t) \leq \sqrt{D_G/2} \leq \sqrt{I_t/(2\alpha)}, \tag{29}$$

which proves Equation 29. The result is a distribution-free view-quantile guarantee that, unlike the maximum-view statement, avoids the factor m inside the square root. □

Proposition 7 (Alignment and credit are complementary)**.** *For any distribution r on $C_x \cup \{\mathrm{OTHER}\}$, setting every $\widetilde{q}_{\theta,i} = r$ makes Equation 6 zero, even when r is unrewarded. With binary orbit success rate $p \in (0, 1)$, Equation 3 assigns $(1-p)/(\sqrt{p(1-p)} + \epsilon)$ to successes and $-p/(\sqrt{p(1-p)} + \epsilon)$ to failures. These reduce to $\sqrt{(1-p)/p}$ and $-\sqrt{p/(1-p)}$ when $\epsilon = 0$. If view i has K independent failures and the other views have rollout success probabilities p_j, orbit normalization rescues that otherwise flat prompt group with probability*

$$1 - \prod_{j \neq i} (1 - p_j)^K. \tag{30}$$

Thus orbit credit supplies a correctness signal that consistency alone lacks.

The proposition formalizes why the two technical contributions are needed jointly: alignment alone can synchronize an unrewarded action, and orbit credit supplies the signed correctness signal that per-prompt normalization loses when one view fails entirely (Challenge 2). It does not assert that either finite-step update must improve success, or that ordinary reward optimization can never recover the same policy. The factorial experiment in Figure 2 tests this complementarity empirically.

A.8 Proof of Proposition 7

Every term in Equation 6 is nonnegative. If $\widetilde{q}_{\theta,i} = r$ for every i, then $\bar{q} = r$ and every KL term is zero, regardless of the reward associated with actions in the support of r. Consistency therefore identifies equality across coordinates but not semantic correctness. For binary returns with empirical success fraction p, the population-variance convention in Equation 3 gives $\mu_x = p$ and $\sigma_x = \sqrt{p(1-p)}$; direct substitution gives the two weights in the proposition and their $\epsilon = 0$ simplification. If all K prompt-wise rollouts under view i fail, their within-prompt centered advantages are zero. Conditional independence makes the probability that all K rollouts also fail under every other view $\prod_{j \neq i}(1-p_j)^K$. On the complementary event the orbit group contains both outcomes, so every failed rollout receives a nonzero negative weight. This proves Equation 30. □

A.9 Coordinate Routing and the Forward-KL Consensus

The semantic policy should ignore the arbitrary coordinate label after inverse mapping, but the emitted token sequence must still use the current tool and argument names. The following proposition shows that these two requirements are compatible and states how they separate.

Proposition 8 (Semantic invariance with coordinate routing)**.** *Fix a semantic task $X = x$, draw a coordinate G with full support over the applicable views, draw an inverse-mapped structured action $C \sim p_\theta(\cdot \mid Gx)$, and emit $Y = \rho_{G\mathcal{T}}(GC)$. The policy is equivariant on these views if and only if*

$$I_\theta(G; C \mid X) = 0. \tag{31}$$

If, for almost every (x, c), the sampled views are render-distinct so that the map $g \mapsto \rho_{g\mathcal{T}}(gc)$ is injective, then exact equivariance also gives

$$I_\theta(G; Y \mid X, C) = H(G \mid X, C) = H(G \mid X). \tag{32}$$

On a general view set, let G_Y be its equivalence class under equality of the rendered action; then $I(G_Y; Y \mid X, C) = H(G_Y \mid X, C)$ without the injectivity assumption. Thus the semantic action can contain no coordinate information even though the surface action retains every render-relevant coordinate.

Proof. For every full-support g, the conditional law of C given $(X = x, G = g)$ is the pullback $g_{\#}^{-1}\pi_\theta(\cdot \mid gx)$. These laws are identical for all g exactly when $C \perp G \mid X$, which is equivalent to Equation 31. The renderer is deterministic, so $H(Y \mid X, C, G) = 0$. Injectivity on the render-distinct sampled views makes G recoverable from (X, C, Y), hence $H(G \mid X, C, Y) = 0$. Therefore $I(G; Y \mid X, C) = H(G \mid X, C)$. Under equivariance, $C \perp G \mid X$, so $H(G \mid X, C) = H(G \mid X)$. □

Proposition 8 does not assert that every hidden state must encode G, because a decoder may route coordinates through several residual paths. It instead yields an observable separation: the semantic output should lose information about G, while the complete rendering path must preserve enough information to emit Y. Section E.4 tests this separation with frozen-checkpoint probes and an explicit hidden-state invariance control.

Lemma 1 (Unique forward-KL orbit barycenter)**.** *Let $q_1, \ldots, q_m$ be distributions on a common finite candidate set and $\bar{q} = m^{-1}\sum_i q_i$. For every distribution r whose support covers that of $\bar{q}$,*

$$\frac{1}{m}\sum_{i=1}^{m} D_{\mathrm{KL}}(q_i \| r) = \frac{1}{m}\sum_{i=1}^{m} D_{\mathrm{KL}}(q_i \| \bar{q}) + D_{\mathrm{KL}}(\bar{q} \| r). \tag{33}$$

Consequently, $\bar{q}$ is the unique minimizer of average forward KL.

Proof. Subtract the first term on the right from the left and exchange the finite sums. The result is $\sum_c \bar{q}(c)\log[\bar{q}(c)/r(c)] = D_{\mathrm{KL}}(\bar{q} \| r)$. Non-negativity of KL and equality only at $r = \bar{q}$ on the support of $\bar{q}$ prove the claim. □

Lemma 2 (Linear aggregation of all-pairs KL)**.** *Let $q_1, \dots, q_m$ have common positive support $\mathcal{C}$. The average directed all-pairs objective*

$$\mathcal{L}_{\text{pair}} = \frac{1}{m(m-1)} \sum_{i \neq j} D_{\text{KL}}(q_i \| q_j) \tag{34}$$

can be computed exactly in $O(m|\mathcal{C}|)$ time and $O(|\mathcal{C}|)$ temporary memory, because

$$\sum_{i,j} D_{\text{KL}}(q_i \| q_j) = m \sum_{i,c} q_i(c) \log q_i(c) - m \sum_c \bar{q}(c) \sum_j \log q_j(c). \tag{35}$$

The diagonal terms vanish, so this is also the numerator of Equation 34.

Proof. Expand every KL term, exchange the finite sums, use $\sum_i q_i(c) = m\bar{q}(c)$, and aggregate $\sum_j \log q_j(c)$. The two sufficient statistics require one pass over the $m \times |\mathcal{C}|$ probability table. □

Proposition 9 (Robustness to arbitrary view distributions)**.** *At every coupled history at step t, suppose there is a canonical reference policy r_t such that every applicable exact view satisfies*

$$\sup_{g \in \mathcal{G}_x} \text{TV}\Big(g_{\#}^{-1}\pi(\cdot \mid gx), r_t(\cdot \mid x)\Big) \leq \delta_t. \tag{36}$$

Let $\mu(\cdot \mid x)$ and $\nu(\cdot \mid x)$ be any two distributions over legal views; either may be deterministic and task adaptive. Draw one view per episode and preserve it through the episode. Under the exact commutation contract,

$$|J_\mu(\pi) - J_\nu(\pi)| \leq R_{\max}\left[1 - \prod_{t=1}^{H} (1 - \min\{1, 2\delta_t\})\right] \leq 2R_{\max} \sum_{t=1}^{H} \delta_t. \tag{37}$$

Thus a genuine uniform reference-policy defect controls balanced, skewed, and adversarial view laws without assuming that evaluation frequencies match training frequencies.

Proof. Fix a canonical task and draw $g \sim \mu(\cdot \mid x)$ and $g' \sim \nu(\cdot \mid x)$. At any history pair that remains canonically coupled, the triangle inequality through r_t gives

$$\text{TV}\Big(g_{\#}^{-1}\pi(\cdot \mid gx), {g'}_{\#}^{-1}\pi(\cdot \mid g'x)\Big) \leq \min\{1, 2\delta_t\}. \tag{38}$$

Apply the maximal-coupling argument of Proposition 2 conditional on (x, g, g'), then average over the task and the two arbitrary view laws. The product bound follows. The final inequality uses $1 - \prod_t (1 - u_t) \leq \sum_t u_t$ for $u_t \in [0, 1]$. □ □

The premise in Equation 36 is deliberately uniform, and random-view evaluation cannot establish it. The frequency-reversal and frozen-surrogate adversary in Section E.5 instead stress whether the learned policy behaves as the proposition predicts on a large, held-out subset of views; they remain empirical audits rather than certificates over all interfaces.

A.10 Decoder Support and Optimal-Value Compatibility

Theorem 1 optimizes over all semantic history policies. Let $\mathcal{B}(x)$ be a complete finite connected chart orbit, not merely the four sampled training views. Define

$$S_L(x) = \{c : \max_{g \in \mathcal{B}(x)} |\text{Tok}(\rho_{g\mathcal{T}}(gc))| \leq L\}.$$

Lengths include the closing token. Because composing with an applicable morphism permutes the orbit, $S_L(gx) = gS_L(x)$. Restricting admissible actions to this common set therefore preserves the Bellman symmetry proof. An always-available absorbing failure action makes the restricted process well-defined. If an unrestricted optimum puts all its mass on S_L at every reachable history, its value is also attained in the restricted process. This is a support condition, distinct from approximation by a finite transformer and from convergence of the optimizer.

More generally, let α_t be the probability that an unrestricted optimal trajectory first selects an action outside S_L at decision t. Follow that optimum until the first exit, then take the failure action. The

resulting restricted policy and the optimum agree on all no-exit paths. For nonnegative returns bounded by $R_{\max}$,

$$0 \le V^\star - V^\star_{S_L} \le R_{\max} \Pr(\text{exit } S_L) = R_{\max} \sum_t \alpha_t.$$

This bounds the value cost of support restriction without requiring every possible free-text response to fit the cap. The actual decoder is not restricted to the orbit intersection: it retains its chart-specific support and exposes incompatible mass through the loss. A canonical sequence longer than L has completed-action probability zero; its otherwise valid prefix contributes to $\perp_L$. Thus the probability identity below uses a *stopped* decoder, not an uncapped product of token probabilities. Candidate scoring applies this zero convention before summing the masses.

The finite-bank support audit fixes reference and candidate actions before model comparison, then renders each in every evaluation chart. It records min/max token length, length-cap crossings, common-support coverage, and shorter/longer aliases. The audit therefore certifies the evaluation bank, not all morphisms. Table 22 separates an overlength complete action from an unfinished sampled prefix; those events need not have the same rate.

A.11 Mass Decomposition of the Alignment Objective

This decomposition is the formal basis of Contribution 1: it isolates the retained-mass term that candidate-normalized alignment discards (Challenge 1). Let $B = \mathbf{1}[\widetilde{C} \neq \text{OTHER}]$. Since B is determined by $\widetilde{C}$, the information chain rule gives $I(G;\widetilde{C}) = I(G;B) + I(G;\widetilde{C} \mid B)$, conditional throughout on the fixed history and candidate pool. Conditional on $B = 0$, the output is constant. Conditional on $B = 1$, Bayes' rule gives $P(G = i \mid B = 1) = z_i/(m\bar{z}) = w_i$, and the action marginal is $r_\star$. This proves Equation 7. When $z_i = 0$ its weighted term is zero; when $\bar{z} = 0$ the entire objective is zero. At $\bar{z} = 1$, the Bernoulli term vanishes. The identity also holds for gradients wherever the distributions are differentiable, provided derivatives through z_i, w_i, and $r_\star$ are retained.

For a direct counterexample to conditional alignment, let the retained conditional distribution be the same r in two views but let $z_1 = .9, z_2 = .4$. Conditional JSD is zero, while coarsened TV is exactly .5; only the Bernoulli information detects this difference. Unweighted conditional JSD plus Bernoulli JSD restores mass matching but is a different objective from the chain-rule weighting. Implementing the two terms of Equation 7 exactly recovers the same loss and gradient as OTHER JSD; it is an algebraic identity, not a separate favorable experiment.

A.12 Canonical Semantic Policy, Information Identity, and Coverage

Let $\mathcal{C}(\mathcal{T})$ contain typed tool calls and terminal `Finish`(z), and let the executed action space be $\mathcal{C}_L(\mathcal{T}) = \mathcal{C}(\mathcal{T}) \cup \{\perp_L\}$. The deterministic renderer $\rho_\mathcal{T} : \mathcal{C}(\mathcal{T}) \to \mathcal{L}_\mathcal{T}$ emits one canonical byte string per completed action: UTF-8 strings use JSON escaping, whitespace is absent, keys follow the interface-prescribed order, numbers use the benchmark's normalized representation, and one end token closes the action. The frozen tokenizer deterministically maps that byte string to one token-ID sequence.

The online grammar recognizes the token image $\mathcal{K}_\mathcal{T} = \{\text{Tok}(\rho_\mathcal{T}(c)) : c \in \mathcal{C}(\mathcal{T})\}$, including the canonical whitespace, key order, numeric representation, and escaping conventions; a generic JSON-validity grammar alone would not suffice. At each step it admits exactly the token IDs that extend a prefix of $\mathcal{K}_\mathcal{T}$. Thus a different tokenization of the same bytes is never generated and then rejected: it lies outside the constrained policy support. Every completed supported path is decoded, parsed, and executed as c. If no completed parse is reached by token $L = 1024$, the stopping rule yields $\perp_L$, which terminates the episode with zero terminal task reward. Thus the effective accepted language is the length-L subset of $\mathcal{K}_\mathcal{T}$, the parser restricted to it is $\rho_\mathcal{T}^{-1} \circ \text{Dec}$, and every finite rollout has exactly one outcome in $\mathcal{C}_L(\mathcal{T})$. Interface transformations fix $\perp_L$ and obey $\rho_{g\mathcal{T}}(gc) = g\rho_\mathcal{T}(c)$ for completed typed objects. Accordingly, for $c \in \mathcal{C}(\mathcal{T})$,

$$\Pr^{\tau}_{\theta}(C = c \mid x) = \Pr^{\tau,\text{con}}_{\theta}(Y = \text{Tok}(\rho_\mathcal{T}(c)) \mid x), \tag{39}$$

with no sum over alternative tokenizations because each completed action has one supported path. The log probability in Equation 5 uses the same temperature-scaled logits, legal-token masks, and

per-token renormalization as rollout decoding; nucleus truncation is not applied, and an unconstrained teacher-forced score is not a valid substitute. The length-cap failure probability is the total mass of unfinished stopped prefixes; it need not be enumerated because Equation 5 assigns the exact residual $1 - z_{\theta,i}$ to OTHER.

For terminal responses, z is the exact text payload. A transformation leaves free text unchanged and replaces only exact tokens registered as opaque entity handles; those tokens are inverse-mapped through the episode symbol table before histories or terminal actions are compared. Terminal actions are therefore subject to the same one-action/one-rendering contract, without any claim of semantic equivalence between paraphrases.

Condition on task x and candidate set C_x. Let G be uniform on the m sampled coordinate views and draw the coarsened semantic action $\widetilde{C}$ from $\widetilde{q}_{\theta,G}$ on $C_x \cup \{\mathrm{OTHER}\}$. Its marginal is $\bar{q} = m^{-1} \sum_i \widetilde{q}_{\theta,i}$. By the definition of conditional mutual information,

$$I_\theta(G; \widetilde{C} \mid x, C_x) = \frac{1}{m} \sum_{i=1}^{m} D_{\mathrm{KL}}(\widetilde{q}_{\theta,i} \| \bar{q}). \tag{40}$$

The quantity is nonnegative and equals zero if and only if every $\widetilde{q}_{\theta,i}$ equals $\bar{q}$. This proves the coarsened unconditional-policy interpretation of Equation 6: both every retained action probability and total retained mass must agree. Moreover, the derivative omitted by stopping the barycenter is $-\sum_c \bar{q}(c)\, d \log \bar{q}(c) = -d \sum_c \bar{q}(c) = 0$. The implemented loss therefore has the same value and parameter gradient as the conditional mutual information. The unresolved approximation is only allocation inside OTHER, which contains omitted completed actions and $\perp_L$. Both components enter the missing-mass allowance in the return certificate.

Let p_i denote the full pullback policy and suppose the candidate set has missing mass $p_i(C_x^c) \le \eta$ for every view. Let T map every action outside C_x to OTHER, so $T_\# p_i = \widetilde{q}_i$. For two views, the candidate contribution to TV is unchanged by T, while expanding the tail can add at most $\min\{p_i(C_x^c), p_j(C_x^c)\} \le \eta$. Therefore

$$\mathrm{TV}(p_i, p_j) \le \mathrm{TV}(\widetilde{q}_i, \widetilde{q}_j) + \eta. \tag{41}$$

This bound is strictly sharper than separately conditioning both policies, which pays 2η. Full tail overlap is not identifiable from finite samples. The audit therefore reports coarsened TV as a lower bound and $\mathrm{TV}(\widetilde{q}_i, \widetilde{q}_j) + \min\{p_i(C_x^c), p_j(C_x^c)\}$ as a distribution-free upper bound; sampled tail overlap is only a diagnostic.

A.13 From Mass Dispersion and Candidate Coverage to Return

The following results make three links explicit: which signal candidate conditioning discards (Challenge 1), which information a finite candidate pool conceals, and which coverage assumption connects replay histories to deployment histories (Challenge 2). All statements concern the same stopped stochastic policy used in the objective. Logarithms are natural and candidate sets are held fixed when differentiating.

Proposition 10 (Non-asymptotic retained-mass signal)**.** *Let $z_1, \dots, z_m$ be retained masses, $\bar{z} = m^{-1} \sum_i z_i$, $D_z = m^{-1} \sum_i (z_i - \bar{z})^2$, and $B_z = m^{-1} \sum_i D_{\mathrm{KL}}(\mathrm{Ber}(z_i) \| \mathrm{Ber}(\bar{z}))$. For arbitrary $z_i \in [0,1]$, $B_z \ge 2D_z$, with $B_z = 0$ exactly when all masses agree. If $z_i \in [a, 1-a]$ for $0 < a \le 1/2$, then*

$$2D_z \le B_z \le \frac{D_z}{2a(1-a)}.$$

For interior masses, its derivative is

$$\frac{\partial B_z}{\partial z_i} = \frac{1}{m} \left[\log \frac{z_i}{1 - z_i} - \log \frac{\bar{z}}{1 - \bar{z}} \right]. \tag{42}$$

Consequently, when the candidate-conditional laws agree, the entire mass-preserving objective is B_z, including for singleton pools.

Proof. The total variation between Bernoulli laws is $|z_i - \bar{z}|$. Pinsker's inequality gives each summand at least $2(z_i - \bar{z})^2$. For the upper bound, set $f(u) = u \log u + (1-u) \log(1-u)$. The

Bernoulli KL is the Bregman divergence $f(z_i) - f(\bar{z}) - f'(\bar{z})(z_i - \bar{z})$. On $[a, 1-a]$, $f''(u) = 1/[u(1-u)] \le 1/[a(1-a)]$. Taylor's theorem with integral remainder bounds this divergence by $(z_i - \bar{z})^2/[2a(1-a)]$; averaging proves the upper bound. Since $B_z = m^{-1}\sum_i f(z_i) - f(\bar{z})$, differentiation gives Equation 42. Strict monotonicity of the log odds gives the zero condition. Equation 7 makes the conditional-law term zero when those laws agree, proving the last claim. □

The derivative identifies the direction of mass equalization in probability coordinates: a view above the mean has a positive derivative and one below it has a negative derivative. A shared neural parameter update can couple these coordinates, so this is a property of the objective rather than a claim of monotonic training. The bounds hold non-asymptotically for every dispersion inside the stated interval, and they supply the direct prediction tested by the matched-dispersion comparison (H1): the retained-mass signal, and hence the advantage of OTHER alignment over conditional alignment, should grow with D_z.

Proposition 11 (Residual information and candidate refinement)**.** *At one fixed semantic history, let G be uniform on m views and let C follow the full pulled-back stopped action law p_G. For a finite candidate set A, write $T_A(C) = C$ on A and $T_A(C) = \mathrm{OTHER}$ elsewhere, $\bar{z}_A = m^{-1}\sum_i p_i(A)$, and $I_A = I(G; T_A(C))$. Then*

$$I(G;C) = I_A + (1 - \bar{z}_A) I(G; C \mid C \notin A). \tag{43}$$

For $A \subseteq A'$, $I_A \le I_{A'} \le I(G;C)$ and $\max_i p_i((A')^c) \le \max_i p_i(A^c)$. For any fixed pair i, j,

$$L_A^{ij} \le \mathrm{TV}(p_i, p_j) \le U_A^{ij}, \qquad \begin{aligned} L_A^{ij} &= \mathrm{TV}(T_{A\#}p_i, T_{A\#}p_j), \\ U_A^{ij} &= L_A^{ij} + \min\{p_i(A^c), p_j(A^c)\}. \end{aligned} \tag{44}$$

Refining the candidate set makes L_A^{ij} nondecreasing and U_A^{ij} nonincreasing. These are comparisons at one fixed policy.

Proof. Since $T_A(C)$ is a function of C, the information chain rule gives $I(G;C) = I(G;T_A(C)) + I(G; C \mid T_A(C))$. The last term vanishes for each revealed candidate and equals the conditional tail information weighted by $P(C \notin A) = 1 - \bar{z}_A$ otherwise. This proves Equation 43, with a zero weighted term for an empty tail. The refined observation $T_{A'}(C)$ determines $T_A(C)$, so data processing proves information monotonicity; missing mass decreases by set inclusion. The TV interval follows by expanding the tail. More explicitly,

$$U_A^{ij} = \frac{1}{2}\sum_{c \in A} |p_i(c) - p_j(c)| + \frac{1}{2}[p_i(A^c) + p_j(A^c)] = 1 - \sum_{c \in A} \min\{p_i(c), p_j(c)\}.$$

The latter expression decreases as candidates are added. Coarsening cannot increase TV, which proves the monotonicity of the lower bound. □

Equation 43 specifies exactly what OTHER leaves unidentified. The posterior over views conditional on the tail need not be uniform; the conditional mutual information uses that posterior. Refinement reveals discrepancies and tightens the direct TV interval, although a Pinsker upper bound formed from the larger $I_{A'}$ need not decrease. For this reason, candidate-cap comparisons must report both loss and coverage.

Theorem 2 (Replay-coverage return bound)**.** *Fix a bank of m exact charts and a policy π^τ with episode return in $[0, R_{\max}]$. At decision t, let ν_t be a reference distribution of canonical replay histories and d_t^{ij} the history law for a pair of views conditioned on survival of their sequential maximal coupling before decision t. Suppose $d_t^{ij} \ll \nu_t$ and $dd_t^{ij}/d\nu_t \le \chi_t$ for every pair. Use the same candidate-selection rule under both laws and define*

$$\bar{\eta}_t = \mathbb{E}_{h \sim \nu_t} \max_i p_i(C_h^c \mid h), \qquad \bar{I}_t = \mathbb{E}_{h \sim \nu_t} I(G; T_{C_h}(C) \mid h).$$

If candidate selection is random, include its randomness in both laws. Set

$$u_t = \min\{1, \chi_t \bar{\eta}_t + \sqrt{2m\chi_t \bar{I}_t}\}. \tag{45}$$

Every pair of charts satisfies

$$|J_i - J_j| \le R_{\max}\left[1 - \prod_{t=1}^{H}(1 - u_t)\right] \le R_{\max}\sum_{t=1}^{H} u_t. \tag{46}$$

Proof. Proposition 1 holds at each coupled history: $\delta_t^{ij}(h) \le \eta_t(h) + \sqrt{2mI_t(h)}$. The density-ratio assumption implies $\mathbb{E}_{d_t^{ij}} \eta_t \le \chi_t \bar{\eta}_t$ and $\mathbb{E}_{d_t^{ij}} I_t \le \chi_t \bar{I}_t$. Jensen's inequality therefore gives

$$\mathbb{E}_{d_t^{ij}} \delta_t^{ij} \le \chi_t \bar{\eta}_t + \sqrt{2m\chi_t \bar{I}_t}.$$

TV is at most one, so the conditional probability of the first semantic disagreement is at most u_t. Exact commutation couples the next observations whenever the actions agree. Multiplying conditional survival probabilities and applying bounded reward yields Equation 46. If survival has probability zero before a step, the product bound is already trivial and no conditional law is required at that step. □

This theorem gives replay, the second half of Contribution 2, a precise role: rendering a visited history in all charts supplies alignment there even when only one chart naturally visited it. It does not imply coverage of an unvisited semantic history. The condition on χ_t states the needed relation to evaluation occupancy explicitly; average training loss alone does not establish it. The theorem uses population expectations for a fixed chart bank, while empirical loss estimates and unseen compositions require their respective sampling and generator assumptions.

Assumptions, conclusions, and discriminating comparisons. Exact commutation and attained Bellman maxima yield a pointwise optimal equivariant policy (Theorem 1); the initial-task distribution need not be invariant because the proof is optimal at every history. Fixed candidate identities and canonical masked scoring give the exact information identities. Mass dispersion yields the strictly positive signal in Proposition 10, tested against conditional alignment at matched mass and coverage. Candidate refinement yields the interval in Proposition 11, tested by nested candidate pools at a fixed checkpoint. Finally, exact transitions, bounded rewards, and the stated history-density ratio yield Theorem 2; the replay versus natural-match comparison measures the corresponding history-coverage mechanism. The proofs establish these implications independently of the empirical tables, which test whether the predicted patterns appear under the audited conditions.

B TRANSFORMATION SPECIFICATIONS

This appendix specifies the exact interface maps on which both technical contributions rely. Table 9 separates such exact coordinate changes from ordinary robustness perturbations by stating, for each generator, the joint action it performs and the basis on which it preserves semantics. The compiler stores the resulting typed bijections only for the duration of one episode; this mapping is the operational state required to invert the rendered interface and dispatch the corresponding semantic action.

Table 9: Exact interface transformation contracts. Each generator acts jointly on schemas, calls, and traces, and its correctness basis states why the inverse-mapped action is semantically identical.

Generator	Joint action	Correctness basis
Tool order	Swap two adjacent schema entries; leave calls unchanged	Wrapper dispatches the identical typed call
Tool alpha-rename	Relabel one definition to a fresh symbol in prompt mentions, calls, and traces	Inverse map dispatches the same implementation
Argument alpha-rename	Relabel one typed parameter key to a fresh symbol everywhere	Inverse map gives the identical typed argument map
JSON order	Swap two adjacent keys in the interface-prescribed canonical order	Parser returns the identical typed object
Entity handles	Relabel one issued opaque ID consistently in observations and future arguments	Shared symbol table addresses the same latent entity; sampled audit
Composition	Apply two or more generators in recorded order	Closure of the component bijections; sampled audit

These rows are the elementary primitives counted by $\ell_\star$; a larger within-family permutation or rename table is always a composition of these primitives, never a single step. For string-valued arguments, only opaque identifiers declared by the environment are eligible for entity renaming; free-form names, addresses, dates, and policy text carry semantics and remain unchanged. Tool and argument renamings are sampled from pronounceable neutral strings whose token lengths are stratified from one to four. This design prevents robustness from reducing to memorization of a fixed alias dictionary and exposes tokenizer-length bias.

Ordering and alpha-renaming are correct by construction, because typed dispatch is unaffected. The entity wrapper is additionally audited on 20,000 sampled reachable states, including handles created after the initial observation: for every action in OrbitLab's finite enumerated typed-action domain, inverse-mapped next states, rewards, and terminal flags must agree. This domain comprises finite tool enums, bounded integer arguments, registered opaque handles, and a fixed terminal payload; unrestricted strings and numbers are excluded from exhaustive enumeration. Pool sizes 2,000, 8,000, and 20,000 are reported separately. The checks exhaust the legal action set at each sampled state, but they do not prove commutation over every reachable history or over unrestricted benchmark text.

Table 10: Condition audit on held-out exact-view histories for the assumptions behind Proposition 1 and Theorem 2. Contract pass is the fraction of audited histories whose inverse-mapped transition and reward agree. $\eta_{.95}$ and $\eta_{\max}$ are the 95th percentile and sample maximum of candidate missing mass; the three χ_t columns report the median, 95th percentile, and sample maximum of the replay-history density ratio. Sample maxima are finite-bank diagnostics, not uniform bounds: the table supports applicability on the audited histories but does not establish a guarantee over every benchmark history.

Backbone / bank	Histories	Contract pass	$\eta_{.95}$	$\eta_{\max}$	χ_t median	χ_t 95th	χ_t max
Qwen3-8B core	3,600	99.1%	.084	.139	1.41	1.68	2.06
Mistral Small 3.2 24B core	3,600	98.7%	.097	.153	1.52	1.83	2.29
Qwen3.8-27B extension	3,680	99.4%	.071	.118	1.34	1.59	1.94

The audit uses frozen exact-view banks and reports the largest applicable history-density ratio after applying the same task and view filters as evaluation. The nonzero tail allowance is kept in the bound rather than silently replaced by the candidate-only TV. The result therefore shows that the conditional replay argument is informative in the evaluated regime; it does not claim that unrestricted benchmark text satisfies the symmetry contract.

C Optimization Procedure

Each OrbitPO update couples on-policy orbit rollouts, which supply orbit-relative credit (Contribution 2), with mass-preserving alignment on re-rendered histories (Contributions 1 and 2). Candidate construction follows rollout collection, and both parsing and the orbit barycenter lie outside the gradient path. One update proceeds as follows.

1. Sample a minibatch of semantic tasks and, per task, four valid interface transforms including the identity.
2. Draw four trajectories per transformed interface under matched decoding temperature and token budget, and execute them in the mapped environment.
3. Compute orbit-relative advantages with Equation 3 and the clipped policy loss with Equation 4.
4. Inverse-map every distinct history visited by any rollout into canonical semantic space. A turn-stratified reservoir selects at most eight anchor histories per task and counterfactually re-renders each complete history in all four sampled charts; every visited history is eligible, and no new environment call is required. Pool into C_x only the parsed actions actually taken at this exact history together with the fixed replay reservoir. A chart that did not visit the history contributes no rollout action.
5. Under every selected re-rendered chart, score each candidate with exactly the rollout policy π_θ^τ, form $\widetilde{q}_{\theta,i}$, compute Equation 6, and then refresh the reservoir with the current parsed actions. The only additional work is teacher-forced scoring, which is charged in the reported FLOPs.

6. Apply Equation 8; discard surface renderings and retain only ordinary optimizer state, the bounded canonical-action reservoir, and aggregate training logs.

Steps 4–5 are the mechanism that resolves Challenge 2 for alignment. For multi-turn tasks, counterfactual chart replay makes every naturally visited semantic history eligible for alignment in all m charts, including hard states reached by only one rollout. The fixed anchor budget bounds teacher-forcing cost, and turn stratification prevents early prefixes from exhausting it. Because the history key is the complete inverse-mapped prefix, two prefixes that differed earlier remain distinct even if their latent states later coincide. All complete returns remain in the mK orbit-relative advantage group. The two signals therefore divide the work: trajectory credit compares outcomes across views, whereas consistency compares alternate renderings of exactly one visited decision history.

Two controls isolate these design choices. The "natural-match only" ablation applies consistency only where at least two independent view rollouts retain identical complete semantic prefixes, reproducing prefix matching. The "w/o semantic-history key" control instead aligns equal turn indices after histories differ; it is intentionally misspecified and tests the importance of state identity. Neither control changes orbit credit or rollout-episode counts.

C.1 Implementation Contract and Gradient Boundaries

Probability law. For candidate c in chart i, the score is the sum of the temperature-scaled, mask-normalized token log probabilities along its unique canonical path, including the closing token; a path beyond the cap has completed-action mass zero. Neither per-token length normalization nor a softmax over candidate scores is applied, because either operation would change the probability law in the theorem and, in the latter case, reintroduce the candidate normalization of Challenge 1. With $\ell_i = \mathrm{logsumexp}_{c \in C_h} s_i(c) = \log z_i$, we compute $1 - z_i = -\operatorname{expm1}(\ell_i)$ and its logarithm through the corresponding stable log-one-minus-exp expression. A zero-mass event contributes zero to the KL by continuity. Disjoint canonical paths imply $z_i \le 1$ mathematically; accumulation uses float64, and a discrepancy beyond numerical rounding signals duplicate candidates or faulty decoder masks, which are checked rather than hidden by renormalizing the retained set. No probability floor is part of the stated objective.

What is held fixed. During an update, rollout histories, rewards, orbit means and standard deviations, old-policy likelihoods, anchor membership, candidate membership, and the reference model are fixed. Candidate nominations use the rollout checkpoint θ_{old}, and their discrete selection lies outside the gradient path; current-policy scores and residual mass remain differentiable. Stopping the gradient through the current barycenter preserves the exact conditional-JSD gradient by the normalization identity in Section 3. This identity holds conditionally on the sampled bank; it is not a derivative through the policy-dependent sampling distribution of histories or candidates.

Loss weighting and cost. The RL and reference terms average over generated action tokens. Alignment averages over selected nonempty anchors, giving each anchor equal weight and each of its m charts weight $1/m$; empty pools contribute no alignment signal. If A anchors have candidate counts n_h, the auxiliary computation scores $m \sum_{h=1}^{A} n_h$ canonical candidate paths for the loss, in addition to nomination scoring. Its token cost is the sum of their actual rendered lengths rather than the number of candidates, and any prefix reuse is counted under the same execution convention for every method. The reservoir stores four prior distinct actions per retained history and never supplies off-policy rewards. Deployment performs no candidate enumeration or chart replay, which is why inference remains one policy call per decision.

Information access for the controls. Every learning control receives the same training correspondence, source tasks, available anchors, and rollout budget whenever its objective needs that information, so each control changes one ingredient. Aug-GRPO changes only transformed-input exposure; the orbit-credit control changes the advantage grouping; and conditional JSD changes the compared law while retaining replay. Success-Xfer additionally requires a successful continuation for its supervised target, and the matched-anchor hard/soft controls remove that coverage difference when testing target type. At inference, LocalCanon uses the current schema and history, FullCanon additionally receives the true registry, and Orbit-TTA additionally receives four policy calls. These resource differences are part of each baseline's definition, not learned gains.

D ORBITLAB ENVIRONMENT

OrbitLab is the controlled environment used in the experiments (Contribution 3) to separate semantic planning from interface presentation through a finite, typed tool process. The semantic environment is specified by the latent state $s = (E, V, B, Q)$: E is a finite set of typed entity records, V contains Boolean and categorical state variables, B maps currently issued opaque handles to entities, and Q is the set of satisfied goal predicates. A tool call $a = (f, \mathbf{z})$ has typed arguments, preconditions $\mathrm{pre}_f(s, \mathbf{z})$, and deterministic effects $\mathrm{eff}_f(s, \mathbf{z})$. A legal call whose preconditions hold applies its effects and may create a fresh entity and handle; a type-correct call with false preconditions leaves the state unchanged, emits a structured error, and consumes one call; and ill-typed calls are rejected. The observation reveals only task-relevant visible fields, issued opaque handles, and the result of the last call, never the latent goal flags or a recommended tool.

Each program samples an initial state, a target conjunction $q^\star$, a call budget H, and a typed tool library. The user request describes $q^\star$ in task language, and tool documentation states local argument types, preconditions, and effects but not the solution path. Distractors are type-compatible and share argument signatures with useful tools, but their effects update a different predicate or bind the wrong entity relation. The episode succeeds with reward one as soon as $q^\star \subseteq Q$, including after the Hth tool call; otherwise, a premature `Finish`, a decoder failure, or exhaustion of the H-call budget terminates it with reward zero. Success thus requires selecting tools, tracking changing records, and propagating returned handles, rather than repeating one interface choice.

The generator first draws a dependency DAG over required predicates and then instantiates it with disjoint semantic templates. Increasing the horizon refines dependency edges into new typed transitions and inserts five balanced motifs: serial dependency, handle binding, conditional gate, permission acquisition, and fork–join synchronization. Every inserted call changes a predicate or record needed downstream; padding calls and duplicated no-ops are forbidden. Short and long splits use disjoint DAG templates, entity values, and surface lexicons. A finite backward dynamic program enumerates reachable states and computes $Q_t^\star(s, a)$ under the remaining budget. The main completion tables count every action attaining $Q_t^\star$ as successful. Audits labeled "unique optimal next action" retain only histories for which every latent state compatible with the visible prefix has the same singleton optimal action. An omniscient state-optimal action that cannot be determined from observations is excluded from this diagnostic, and uniqueness is never imposed by a lexical tie-break.

Table 11: A worked four-call OrbitLab program in the identity chart, illustrating the environment specification. Transformed charts rename and reorder the same typed calls and handles without changing the latent effects.

t	Visible observation	Unique optimal call	Latent effect relevant to goal
1	Paid order `ORD-7`; no case	`open_case(order=ORD-7)`	Create case and expose handle `CASE-19`
2	`CASE-19` open; refund unverified	`verify(case=CASE-19)`	Set `refund_authorized =true`
3	Authorization recorded; amount 29	`refund(CASE-19,29)`	Create refund handle `REF-8`
4	`REF-8` pending notification	`notify(refund=REF-8)`	Set refunded and notified goal predicates

Table 12: Minimum-view success on disjoint $H \geq 16$ OrbitLab programs, stratified by the mechanism used to lengthen the task. ORBITPO improves over Aug-GRPO on all five motifs. Each cell averages 800 programs and three seeds.

Long-range motif	Aug-GRPO	ORBITPO	Gain
Serial dependency	49.6	**59.1**	+9.5
Opaque-handle binding	46.8	**59.9**	+13.1
Conditional gate	45.2	**54.0**	+8.8
Permission acquisition	48.9	**56.5**	+7.6
Fork–join synchronization	42.7	**56.1**	+13.4

The motif construction varies the structural source of horizon difficulty. Fork–join and conditional programs create more opportunities for an early coordinate error to alter later state, whereas a serial dependency is easier. ORBITPO improves minimum-view success over Aug-GRPO in all five strata, by 7.6 (permission acquisition) to 13.4 points (fork–join synchronization), with the two largest gains on fork–join synchronization and opaque-handle binding (+13.1). Because the gains vary by motif rather than following a fixed offset and are positive in every construction, they support a state-tracking and planning interpretation of the horizon effect rather than an artifact of repeating one tool decision.

E EXPERIMENTAL PROTOCOL

This appendix fixes, before any comparison, the role of each benchmark, the adaptation sources, the interface-view banks, the reference decoding rules, and the comparison budgets. The baseline definitions and audits below state which resources every method shares and which mechanism each control changes, so that each contrast in Section 6 isolates one source of robustness.

E.1 BENCHMARK ROLES

Table 13: Benchmark roles, interaction regimes, and native evaluation signals. The six benchmarks stress different stages at which an interface change can propagate.

Benchmark	Interaction regime	Native evaluation signal	Primary diagnostic
API-Bank	Single/multi-call API use	Executable call correctness	Selection and arguments
ACEBench	Single, parallel, dependent calls	Function-call match	Composition structure
RoTBench	Perturbed tool learning	Three stage-wise accuracies	Established robustness
AppWorld	Stateful multi-app tasks	State-based task completion	Long-horizon compounding
τ-bench	Agent-user-policy interaction	Environment reward/pass	Policy-constrained action
ToolSandbox	Stateful conversation	Milestone/minefield trajectory similarity	Observation/entity mapping

Table 14: Centralized training and evaluation protocol shared by all methods. Each row is a separate benchmark-adapted model; counts are unique RL training tasks after deterministic validity filtering, and no official evaluation query is used for training. This table defines the protocol and contains no result.

Benchmark	Frozen adaptation source	Train n	RL reward	Evaluation
API-Bank	Public 1,888-dialogue train set	1,800	Exact executable tool and arguments, binary	314 dialogues (753 API calls)
ACEBench	Disjoint verified ToolACE/APIGen partition	4,000	Official function-call match, binary	All official test tasks
RoTBench	New queries on public tool schemas; official queries excluded	2,400	Mean of three stage scores in $[0, 1]$	All official test queries
AppWorld	105 official train tasks plus train-only variants	720	Final state-goal completion, binary	Test-N 168 + Test-C 417
τ-bench	Simulator-generated goals disjoint from official tasks	1,200	Policy-compliant task success, binary	Retail 115 + Airline 50
ToolSandbox	Train-only template/state instantiations	1,000	Milestone/minefield trajectory similarity	All official test scenarios

All adaptation counts in Table 14 are counts retained after filtering. Public schemas may be reused, but official evaluation queries never serve as training prompts, and RoTBench training queries are generated independently of its evaluation queries. The 720 AppWorld instances derive exclusively from the 105 official training tasks, their training databases, and train-only state/task variants; no development or test goal or state enters generation. The τ-bench goals are generated against the frozen simulator and checked for semantic and textual disjointness from its 165 official evaluation tasks. ToolSandbox adaptation uses new template/state instantiations whose goals, initial states, and dialogue templates are disjoint from the official evaluation scenarios. The Train n column therefore counts executable RL instances; it does not claim the sizes of official splits.

Decoding. All Qwen3 runs use non-thinking function-calling mode, and Mistral Small 3.2 24B uses its official non-reasoning instruct and function-calling format (Mistral AI, 2025); both backbones receive text-only inputs and the same canonical action grammar. Training uses stochastic canonical constrained decoding at temperature $\tau = 0.7$ and top-$p = 1.0$, and this single policy π_θ^τ supplies rollouts, teacher-forced candidate scores, PPO numerator and denominator probabilities, the reference KL, and every TV/KL or stochastic-bound audit. Final benchmark evaluation instead uses token-wise greedy canonical constrained decoding from untempered logits, with one trajectory per task, view, and checkpoint and no best-of-n selection. Only the stochastic-policy bridge audit departs from this rule: up to 200 distinct fixed tasks per benchmark receive 32 paired trajectories per view from π_θ^τ, and WVA is computed from per-task expected native scores. Smaller benchmarks use their full evaluation split, and repeated trajectories do not increase the independent task count. On 1,200 disjoint finite OrbitLab states, we use a finite diagnostic grammar (enumerated typed tool calls and a fixed Finish payload, plus the length-cap failure atom), enumerate its complete stopped action distribution, use the view barycenter as r, and check Proposition 5 pointwise. Greedy disagreement is the fraction of unordered view pairs whose parsed actions differ after inverse mapping. All of these diagnostics are reported separately from the primary greedy benchmark endpoint.

Robustness metrics. For each benchmark and seed, WVA is the minimum of the 17 view-level benchmark scores (identity plus 16 transformations). All-orbit consistency instead averages the task-level indicator that all 17 views meet the benchmark-native completion criterion. TWA first takes, for each task, the minimum benchmark-native task score across the 17 views and then averages over tasks, so the adversarial view may differ by task. For binary task-success benchmarks, TWA equals the fraction of tasks solved in every view; for the graded RoTBench and ToolSandbox outcomes, it does not coincide with the all-view completion indicator. The horizon bins in Figure 3 use a verified reference-plan call count fixed before any checkpoint is evaluated. For API-Bank, ACEBench, and RoTBench, this count is the length of the shortest verifier-accepted reference call sequence. For AppWorld, τ-bench, and ToolSandbox, bounded simulator search starts from independently verified reference traces and removes or substitutes calls while preserving terminal success; two annotators adjudicate unresolved ties without access to model outputs. The binning variable is therefore independent of every evaluated policy, although bounded search does not prove a globally shortest successful path; only OrbitLab's dynamic program supplies an exact minimum-call annotation. Because AppWorld does not release test solution programs, its reference traces are constructed by an independent reference solver before the compared checkpoints are evaluated.

Every task-bootstrap replicate recomputes the view-level rates, their minimum, and CVaR after resampling tasks. Empirical CVaR_{10} places fractional weight on the boundary view so that the lower tail carries exactly 10% probability mass, rather than rounding 6.4 views to six or seven; seed estimates are then averaged within each benchmark and finally across the six benchmarks. The cost unit in Table 32 is one separately adapted benchmark model trained with one seed. The complete Qwen3-8B primary method pair thus comprises $2 \times 6 \times 5 = 60$ adaptations and the Mistral pair $2 \times 6 \times 3 = 36$, for 96 across both backbones, excluding the additional baselines and model-transfer studies.

Clean, mean-view, and WVA results use a benchmark-native score scaled to 0–100 rather than labeling every quantity a success rate. API-Bank uses executable call accuracy; ACEBench, AppWorld, and τ-bench use native task success; RoTBench uses the specified unweighted mean of its three stage accuracies; and ToolSandbox uses its milestone/minefield trajectory-similarity score. For all-orbit consistency, RoTBench completion requires all three stages to be correct, and ToolSandbox completion requires every required milestone and no minefield violation. Parser accuracy,

environment rejection, and tool-selection accuracy remain diagnostics. The macro-average weights each benchmark equally. Confirmatory tests jointly resample paired seed labels and task identities within each benchmark while preserving method and view pairing. Diagnostic intervals that hold checkpoints fixed instead resample the stated task, state, or dialogue clusters; they are conditional on those checkpoints and do not quantify population-level training uncertainty. Within τ-bench, Retail and Airline receive equal weight; AppWorld likewise averages Test-N and Test-C with equal weight while retaining their separate task identities for inference. ACEBench includes both English and Chinese and retains the official category weighting. API-Bank uses dialogue-clustered uncertainty, and complete-dialogue success requires every required call to be correct.

Published scale anchors. Published headline values provide a scale audit rather than directly comparable baselines. ACEBench Table 2 (arXiv v8, Chinese and English combined) reports 54.8 for Qwen2.5-7B and 85.4 for GPT-4o (Chen et al., 2025); RoTBench Table 2 (arXiv v3) reports GPT-4 tool-selection accuracy of 80.0 on clean interfaces and 58.1 under the union perturbation (Ye et al., 2024); AppWorld reports GPT-4o task-goal completion of 48.8 on Test-N and 30.2 on Test-C (arXiv v1) (Trivedi et al., 2024); the legacy τ-bench repository (Yao et al., 2024) lists GPT-4o pass^1 of 60.4 on Retail and 42.0 on Airline, whose equal-domain mean is 51.2; and ToolSandbox Table 2 (arXiv v2) reports GPT-4o-2024-05-13 average trajectory similarity of 73.0 (Lu et al., 2024). Because these anchors differ from our setting in model, adaptation, decoding, and view distribution, they audit order of magnitude and do not define pairwise comparisons. On the metric used here, RoTBench's three-stage GPT-4 mean is 57.46 clean and 42.54 under Union, computed from its published stage scores, rather than 80.0 and 58.1. The adapted Qwen3-8B and Mistral Small 3.2 24B models obtain 75.2 and 76.0, respectively, on the same three-stage metric; differences in model, adaptation data, and decoding separate these scores from the published GPT-4 reference. The protocol uses legacy τ-bench's 165 tasks, not the revised τ^2/τ^3 task sets, and the official repository now flags the legacy tasks as outdated. All of these sources were checked on 2026-09-10. Official README values can differ from versioned papers: ACEBench's current README (Chen et al., 2025) lists 57.8 and 89.6 for Qwen2.5-7B-Instruct and GPT-4o-2024-11-20, whereas we retain the explicitly identified bilingual paper values as historical anchors.

Same-model public-score cross-check. The dated BFCL result archive offers a check on metric identity rather than a performance comparison. Table 15 quotes these public results separately from the results of this manuscript, and its last column recomputes our two-group aggregation from the published multi-turn and memory summaries. For Qwen3-8B (FC), for example, 42.57 overall, 41.75 multi-turn, and 14.62 memory imply about 28.19 on that two-group endpoint. A single-turn score or an overall leaderboard value therefore cannot calibrate a multi-turn/memory result merely because both carry the BFCL name. The archive also distinguishes FC from prompting modes, which must not be silently interchanged.

Table 15: Public reference scores. BFCL archive dated 2025-12-16, read on 2026-09-23. All values are percentages, and the last column is the equal mean of the two displayed group scores, rounded half up. The rows describe unadapted public systems rather than runs under our canonical decoder, so they check metric identity and do not serve as baselines. The Mistral entry is a dated hosted API identifier and does not verify identical open-weight serving.

Public identifier	Overall	Multi-turn	Memory	Two-group mean
Qwen3-8B (FC)	42.57	41.75	14.62	28.19
Qwen3-32B (FC)	48.71	47.87	26.67	37.27
Mistral-small-2506 (FC)	37.15	11.50	18.06	14.78
Llama-3.1-8B-Instruct (Prompt)	25.83	11.12	10.75	10.94
xLAM-2-70b-fc-r (FC)	53.07	77.38	14.41	45.90

The source is the benchmark's dated result archive, with separate overall, multi-turn, and agentic CSVs (Berkeley Function Calling Leaderboard Team, 2026). These values do not replace the six-benchmark macro or the 600-state first-action diagnostic, since neither of those measures BFCL. The newer four-backbone extension has no matched public measurement in this audit for its combination of pooled adaptation, action-only decoding, exact transformations, and custom subsets, so public leaderboard values are not directly comparable to the controlled scores reported here. Likewise, the

official τ submissions checked here provide no matched rows for those four checkpoints under the extension protocol.

Repeated values that have a structural cause are retained as reported: FullCanon equates exact views, replay renderability is 100% on admitted exact histories, and the balanced frequency-rotation averages coincide by construction. Such identities should not be perturbed to resemble measurement noise. The similar positive gains across the recent backbones are interpreted through the paired controlled evaluations rather than as independent benchmark replications.

Interface-view bank. Each test instance receives one clean identity view plus 16 deterministic transformed views: three tool orders, three tool renamings, two argument renamings, two JSON orders, two entity-handle maps, and four compositions. Transformation seeds are fixed across methods. The four composed final maps are deduplicated in normal form, absent from training, and verified to satisfy $\ell_\star \geq 3$. Before any comparison, mapped gold calls are executed on up to 200 distinct randomly selected instances per benchmark as a semantics audit.

OrbitLab crosses $H \in \{1, 2, 4, 8, 16, 32\}$, $N \in \{4, 16, 64, 128\}$, alias length $L_{\text{alias}} \in \{1, 2, 3, 4\}$, and induced-map complexity $\ell_\star \in \{1, \ldots, 5\}$. The factorial experiment contains 32 held-out programs, drawn from disjoint templates and latent entities, in each of the 480 cells, for 15,360 programs in total.

E.2 Baseline Implementations

Table 16 summarizes how the closest alternatives seek interface robustness, and the definitions below give their exact implementations. The alternatives act on the input (PartialCanon, LocalCanon, FullCanon), on training (Aug-GRPO, RoTTuning, R-Drop, REx, GroupDRO, Adv.-transform, IRM, Success-Xfer), or at inference (Orbit-TTA). ORBITPO differs from all of them in two respects: it aligns unconditional candidate mass plus OTHER instead of canonicalizing inputs, reweighting risks, matching same-input predictions, or imitating one-hot successful actions (Challenge 1), and it does so at re-rendered complete visited histories rather than at matched prefixes or success anchors (Challenge 2).

Table 16: How the closest alternatives seek interface robustness in our comparisons. Existing alternatives either assign credit per rendered prompt, align no cross-view target, or require a registry or extra inference calls; only ORBITPO aligns unconditional mass at complete visited histories without a deployment registry or extra inference calls.

Method	Credit across views	Cross-view target	Where aligned	Calls
Aug-GRPO	per rendered prompt	none	–	1
GroupDRO	reweighted view strata	none	–	1
Success-Xfer	orbit-relative	one-hot successful action	success anchors	1
LocalCanon-GRPO	per rendered prompt	visible-schema canonical input	–	1
Orbit-TTA	per rendered prompt	test-time semantic vote	inference	4
FullCanon (oracle)	per rendered prompt	registry canonical input	–	1
ORBITPO	**orbit-relative**	**candidates + OTHER**	**visited histories**	**1**

Input-side canonicalization. **PartialCanon-GRPO** is the sort-only structural baseline from the original comparison: it sorts tool definitions and JSON keys but leaves tool names, argument names, and online opaque handles in the presented chart. **LocalCanon-GRPO** relies only on the presented schema and history. It parses tools, arguments, required/default constraints, and references into a typed attributed graph; registered identifier mentions in descriptions become graph references, and other description text remains an attribute. Partition refinement followed by exact lexicographic canonical labeling inside tied cells assigns `TOOL_001`/`ARG_001` names and a local inverse map. An automorphism orbit of size greater than one marks an unresolved semantic identity, so those cells keep their presented names while resolved cells are normalized. Prompt identifier references, histories, and registered entity mentions use the same map. Handles receive typed first-appearance indices ordered by the containing resolved tool/field, and simultaneous unresolved appearances retain their labels. LocalCanon uses no legacy schema, transformation ID, task gold, or registry; in contrast

to ORBITPO, it removes coordinate variation from the input rather than training the policy to be equivariant, so it recovers the correspondence only where the current structure identifies it uniquely.

Coverage is assessed on frozen verified reference histories independently of the policies. An episode is unique if every exposed schema/handle assignment is unique at every reference decision, and all methods use this same partition and benchmark-balanced weights. Off the reference path, LocalCanon continues with its partial map. Canonical labeling guarantees equivalence for uniquely identified graphs, whereas the reference partition does not certify all off-path histories. LocalCanon's own SFT/RL and evaluation apply the wrapper throughout. LocalCanon+ORBITPO trains with this wrapper and the orbit objective and does not route tasks using the reference partition. Both arms use five paired seeds and the same selection budget. Schema labeling is cached within an episode and charged to initialization latency, whereas dynamic history/handle updates are charged per decision. Because exact tied-cell search has no assumed linear-time bound, wrapper cost is reported rather than hidden.

FullCanon-GRPO receives exactly the same episode-level transformation metadata as the orbit compiler. Before each model call, it maps the schema, prompt, full history, tool and argument names, and every registered opaque handle into a fixed canonical chart. Handles are assigned stable `ENTITY_001`, `ENTITY_002`, ... placeholders by first semantic appearance, and the symbol table persists for the episode. The emitted canonical call and terminal handle mentions are mapped back before execution. With greedy decoding and a complete registry, two exact views induce the same canonical model input and trajectory, which yields structural equivariance. This baseline uses no more model calls than ORBITPO but requires the correspondence at deployment. The information asymmetry is temporal rather than algorithmic: training pairs are synthesized by a compiler that records its own maps, whereas the target deployment exposes one new provider chart without a paired legacy schema or adapter. ORBITPO uses the map only in training. LocalCanon can recover a map from uniquely identifying current structure, and FullCanon additionally resolves externally supplied identities. The registry-coverage audit includes both a whole-episode fallback and a partial-entry variant that canonicalizes every available symbol; neither is presented as an upper bound over all partial-mapping algorithms.

Training-side and inference-side alternatives. **RoTTuning** applies the supervised robustness recipe and perturbation levels defined with RoTBench (Ye et al., 2024) before the common RL stage. **Aug-GRPO** receives the same transformation distribution and number of rollouts as ORBITPO but computes advantages per rendered prompt and has no aligned consistency term; it is therefore the exposure-matched comparator that isolates the two mechanisms of Contributions 1 and 2. **R-Drop** applies symmetric KL between two dropout passes of each transformed prompt and does not inverse-map actions. **REx** penalizes the variance of negative return across transformation classes. **GroupDRO** maintains exponentiated weights over six transformation strata (the five atomic families and compositions). **Adv.-transform** samples eight views, performs rollouts under all of them, and updates only on the lowest-return view; its rollout-episode budget is matched by reducing tasks per batch. **IRM** treats generators as environments and applies the gradient penalty to a learned scalar reward head, following a separately tuned penalty schedule. REx, GroupDRO, Adv.-transform, and IRM act on returns or losses across views and never compare inverse-mapped action distributions. **Orbit-TTA** samples identity plus three transformations at inference, inverse-maps valid calls, length-normalizes log likelihood within each view, sums scores for equal semantic calls, and executes the maximizer. Its four model calls and added latency are reported rather than compute-matched away.

Success-Xfer is the direct successful-behavior-transfer control and the closest training-side alternative: it shares the orbit credit and history renderer of Contribution 2 and replaces only the objective of Contribution 1. Concretely, it uses the same $m = 4, K = 4$ rollout allocation, counterfactual history renderer, and orbit-relative advantages as ORBITPO, but replaces Equation 6 with one-hot self-imitation. At an anchor, if the current orbit or a bounded training-only successful-transition reservoir contains a continuation through that exact canonical history that meets the benchmark's binary completion criterion, the highest-return trajectory is selected, with ties broken by a frozen SFT-reference likelihood. The reservoir stores at most four past training-only continuations per semantic history. The selected next semantic action is inverse-mapped, rendered in all four charts, and optimized with mean token NLL. Anchors without a successful continuation receive no imitation target. The default configuration reports actual scoring cost, and Success-Xfer-MF matches ORBITPO's total teacher-forced token budget by drawing additional eligible success anchors from

its reservoir with replacement, without adding rollout episodes. This strong baseline transfers every available successful action across interfaces but never observes failed-action probabilities or retained tail mass, isolating what mass-preserving alignment adds beyond successful-behavior transfer.

ToolRL and RC-GRPO are included as external agent-RL comparisons where their released objectives can be reproduced. The primary controlled comparison nevertheless uses our common GRPO implementation, because differences in environment wrappers or SFT data would otherwise confound interface robustness.

Fixed-anchor mechanism comparison. Table 4 is a companion experiment that isolates the auxiliary objective of Contribution 1. Before the comparison arms begin, a common frozen orbit-credit behavior checkpoint supplies a training-only archive of visited complete histories, sampled view charts, actual candidate pools, and verifier outcomes. All arms use the same eight-anchor schedule, pool, and token budget per task; their RL rollouts remain on-policy and match in count, not in realized trajectories. Held-out evaluation and initialization are shared, and archive construction is charged equally to every arm. Because these fixed auxiliary histories differ from the main experiment's dynamically visited anchors, the companion full-objective 55.7 must not replace the main 55.9.

Conditional JSD uses $r_i = \widetilde{q}_i / z_i$. Its mass-matched variant adds $\mathrm{JSD}(\mathrm{Ber}(z_i))$ with an independently tuned coefficient and uses the same total 20-trial budget as the one-term arms. The chain-rule implementation equals OTHER JSD in value and gradient and therefore has no separate result row. The companion bank uses candidates feasible in every selected chart, and empty pools yield a constant OTHER distribution.

For hard and soft transfer, the frozen checkpoint supplies $t_h = m^{-1} \sum_i \widetilde{q}^{\,\mathrm{teacher}}_{h,i}$ on candidates plus OTHER. Hard transfer minimizes cross-entropy against $\mathrm{onehot}(\arg\max t_h)$, and soft transfer minimizes cross-entropy against t_h, with teacher entropy constant. Ties use frozen candidate order. Both targets exist on the same anchors and include OTHER as a target event with probability $1 - z_i$. This is hard pseudo-label transfer, so a failed anchor need not have a successful demonstration. The common-success pair uses one archive mask for anchors with at least one verifier-successful continuation, covering 51.6% of anchors, together with the identical hard/soft teacher definitions; sampling successful anchors with replacement matches compute only within this mask. Five paired seeds and a common external evaluation candidate pool distinguish coverage, target softness, mass matching, and frozen versus online teachers. This 51.6% is one shared frozen-archive mask, not the on-policy eligibility rate of either transfer method.

E.3 Hyperparameters and Reproducibility

Reference regularization. Write ℓ for a token position and $u = (g_i x, y_{<\ell})$ for its rollout prefix. With $v^{\tau}_{\theta}(\cdot \mid u)$ denoting the temperature-scaled, legal-token-normalized next-token law, the implemented regularizer is

$$\mathcal{R}_{\mathrm{ref}} = \mathbb{E}_{u \sim d_{\mathrm{old}}} D_{\mathrm{KL}}(v^{\tau}_{\theta}(\cdot \mid u) \,\|\, v^{\tau}_{\mathrm{ref}}(\cdot \mid u)) \,.$$

The empirical prefix law d_{old} gives equal weight to generated token positions in the rollout batch and is held fixed during each PPO update. The KL sums over legal tokens at that prefix, and policy and reference share the same mask and temperature. $\mathcal{R}_{\mathrm{ref}}$ is therefore a token-level regularizer; it is neither the KL after parser coarsening nor an exact current-policy expectation over all histories.

Training configuration and candidate construction. The SFT stage uses two epochs, sequence length 4096 for function-calling tasks and 8192 for interactive tasks, AdamW with learning rate 10^{-5} and cosine decay, and global batch 128. RL uses the canonical constrained decoder, temperature 0.7, top-$p = 1.0$, at most 1024 new tokens per turn, gradient clipping at 1.0, and bfloat16. The canonical token grammar and length-cap failure mapping are identical in rollout and evaluation code, and logs count accepted completed actions and $\perp_L$ separately before reward aggregation. The candidate set C_x of Equation 5 is constructed as follows. For an exact anchor h, $A_i(h)$ contains the at most four completed actions actually generated there by chart-i rollouts, and $R(h)$ contains the four most recent distinct completed actions from prior visits. Set $P_h = \bigcup_i g_i^{-1} A_i(h) \cup R(h)$: a nonvisiting chart has $A_i(h) = \varnothing$, and a first visit has $R(h) = \varnothing$. Every chart scores this available pool after re-rendering and nominates at most four entries; their deduplicated union is C_h, capped at 16. Caps 8/32 nominate two/eight entries per chart. Both pool scoring and loss scoring count toward FLOPs,

and no new action is imputed to a nonvisiting chart. The union is not padded, so one completed action at a unique first visit gives a singleton, where only the retained-mass term of Equation 7 supplies signal. If all recorded actions are length-cap failures and replay is empty, $C_h = \varnothing$ and consistency is identically zero, while RL retains the failure outcomes. Re-renderability, nonempty loss inputs, and mass coverage are logged separately. In the all-success constructor, every distinct reward-positive action is retained before uniform negative fill; the set may exceed 16, and its extra scoring cost is charged to that control. The reservoir is training-only, is updated after scoring, and contributes no reward or policy-gradient samples. Gold actions are never inserted except in the explicitly labeled oracle control. The identity transform appears in every orbit, and no benchmark-specific prompt engineering is permitted after the first pilot is frozen.

Alignment-coverage estimand. For Table 37, the denominator of each turn bin is the set of selected distinct complete canonical histories visited by at least one of the mK natural rollouts. The resulting quantities are anchor-sample rates; because selection is turn-stratified, they are not unconditional rates over all visited histories. A history is naturally matched when rollouts originating in at least two distinct charts reach that identical prefix, and its participation count is the number of such charts. Counterfactual eligibility is the fraction of histories for which the compiler can render that same history in every sampled chart; the gap between the two rates is the coverage loss of Challenge 2 that complete-history replay removes. Eight anchors per task are sampled without replacement from six turn bins, and empty-bin slots are reallocated proportionally among nonempty bins. We log all quantities before loss weighting and report, by turn bin, the fraction of selected anchors, the total equivariance loss, and the teacher-forced tokens. The natural-match-only control retains the same tasks, trajectories, orbit advantages, update count, and candidate constructor; a compute-accounting row charges its actual smaller scoring cost, and a matched-FLOP row evaluates extra tasks without chart replay.

RL defaults and selection. The RL default uses four coordinate views, four rollouts per view, clip ratio $\epsilon_c = 0.2$, advantage denominator $\epsilon = 10^{-6}$, reference-KL weight 0.02, equivariance weight 0.15, AdamW at 5×10^{-7}, global batch 128, and at most 2,000 updates. Successive halving allocates the same 20 configuration-seed pilot trials to each trainable method on a disjoint validation subset, and selection uses mean-orbit validation success. Means and population standard deviations use the $1/(mK)$ convention, include the current sample as in GRPO, and are stop-gradient quantities. For each final training seed, hyperparameter selection is repeated using only its disjoint validation partition. The Qwen3-8B GRPO, Aug-GRPO, and ORBITPO primary rows, the five strongest-alternative methods, the key consistency-objective audit, and a separate FullCanon oracle-reference comparison share the same five paired training seeds. The Mistral primary comparison, including its FullCanon oracle reference, uses three paired seeds, as do the remaining benchmark-level methods. FullCanon is excluded from the learned-method correction family.

Statistical inference. The primary interval uses a crossed multiway paired bootstrap: the five seed labels and the task identities are resampled independently, each resampled seed–task Cartesian cell is retained, and method and view pairing is preserved within every cell. Unlike a nested seed-then-task bootstrap, this design retains shared task difficulty across seeds. Aug-GRPO is the mechanism comparator, and a fixed strongest-alternative family contains GroupDRO, adversarial-transform, Orbit-TTA, Success-Xfer-MF, and LocalCanon; all six comparator methods use the same five paired seeds. Task-conditional intervals with seeds held fixed, across-seed standard deviations, and seed-wise win counts are reported separately. Because five seeds sample training variation only coarsely, seed resampling provides an uncertainty summary tailored to finite-run variability, and analyses using three seeds are reported as secondary endpoints unless explicitly included in a corrected family. The empirical analysis plan applies Holm correction to the six fixed benchmark-wise secondary contrasts. A separate four-test Holm family contains the atomic-trained $\ell_\star = 6$ contrast, the short-trained $H = 32$ contrast, the pseudo-symmetry dose trend, and the first-divergence hazard contrast; intermediate curve points characterize the corresponding response curves. The five strongest-alternative contrasts form a third prespecified Holm family and are never selected post hoc by observed rank. Adjusted p-values are not reported for these summaries. A crossed random-intercept model with method fixed effects and seed and task random intercepts serves as a sensitivity analysis. The routing claim uses an intersection–union gate: the one-sided 95% lower bound must exceed 95% for both final-layer coordinate-probe and functional render-choice accuracy,

and $I(G; \widetilde{C} \mid X, C_x)$ must be lower than for Aug-GRPO. APIRefactor-232 contributes one specified migrated-interface endpoint, and its refactor-family columns are reported as transfer diagnostics; these endpoints are not pooled into the four-test family.

Variance and coverage diagnostics. End-to-end update variance is computed on 64 matched minibatches from the empirical covariance trace of final-block per-example gradients at each method's own checkpoint. The Proposition 6 audit instead freezes one common parameter vector, task set, score-function definition, advantage values, and total of 16 rollout episodes, and changes only $m \times K$. Cross-method coordinate information is computed on a common candidate pool frozen before RL training. The pool is built from the SFT policy, the unadapted instruct policy, uniform grammar samples, and train-only action templates, with no test gold action; each state is capped at 128 actions and evaluated unchanged for every method. On 1,200 states, summing canonical candidate sequence probabilities gives the retained semantic-action mass exactly under π_θ^τ, and a separate 256-draw reference estimates modal-action recall. State-then-view cluster bootstraps give across-state coverage intervals. A disjoint 2,048-draw constrained sample classifies omitted completed actions and $\perp_L$ and estimates observed tail overlap only as a diagnostic. Full-policy TV is reported through the identifiable coarsened lower bound and the distribution-free upper bound in Equation 41, never through an unseen-species point correction, and bound audits report both sides of Equations 17 and 13. Pilot programs are excluded from confirmatory tests, and sample-size decisions use paired seed variability and the finite-run design of the study.

Out-of-support composition and horizon. The out-of-support composition audit is indexed by the final induced map rather than by the sampled compiler program. Each exact transformation is reduced to a normal form containing the final tool permutation, the tool- and argument-name bijections, the per-tool JSON order, and the entity-handle bijection; equality means equality of this complete map on the typed schema and reachable handle domain. The shortest primitive length is computed directly from the normal form. If π_{tool} is the tool permutation, $\pi_{\mathrm{field},t}$ the field permutation of tool t, and r_{tool}, r_{arg}, and r_{ent} the three rename maps, then

$$\ell_\star(g) = \mathrm{inv}(\pi_{\mathrm{tool}}) + \sum_t \mathrm{inv}(\pi_{\mathrm{field},t}) + \sum_{r \in \mathcal{R}(g)} \big(|\operatorname{supp}(r)| + \mathrm{cyc}(r)\big). \tag{47}$$

Here inv is the adjacent-swap inversion count, and $\mathcal{R}(g)$ contains the rename maps in separate typed namespaces: tool names, argument keys within each tool, and registered handles within each declared handle namespace. Each support contains the identifiers whose final alias differs from identity. The directed alias graph $a \mapsto r(a)$ decomposes into paths ending at initially unused names and into cycles, and $\mathrm{cyc}(r)$ counts the nontrivial cycles. The finite compiler alphabet includes an unused temporary name in each namespace, and an atomic rename may target only a currently unused name.

Every moved identifier requires at least one rename, and a cycle additionally requires vacating a name before any of its final assignments can be made, so it costs one extra move. This lower bound is attained by processing each path backward and breaking each cycle with the temporary name, reused between cycles, while adjacent swaps independently realize each inversion count on fixed semantic slots; the operations therefore attain the displayed sum. For example, exchanging occupied names a, b requires $a \to u$, $b \to a$, $u \to b$: three primitives, not two. With acyclic fresh-alias maps the cycle term vanishes, recovering the support count. Evaluation bins use this corrected distance, and a whole permutation or rename table is never counted as one step.

Training uses either only maps with $\ell_\star = 1$ or the default support $\ell_\star \le 2$. Evaluation draws 600 disjoint OrbitLab programs at each $\ell_\star \in \{1, \dots, 6\}$, rejects every final map seen in training, and uses fresh alias lexicons and latent entities. A raw reduced-word control logs how often cancellation-free programs nevertheless collapse to a shorter or previously seen final map. The short-to-long audit trains a separate model only on programs with $H \le 4$ and replaces each omitted long program with short programs to match backend tool-invocation budgets; this separate control therefore matches interaction cost, not episode count. It then evaluates 600 disjoint programs at each $H \in \{4, 8, 16, 32\}$. Both audits use three paired seeds and report the minimum of the eight view-level success rates.

Contract falsification and first divergence. The contract falsification is confined to synthetic OrbitLab. A pseudo-symmetry keeps the schema well typed and invertible at the string level but changes one tool precondition, default value, or permission bit, so it violates transition or reward

commutation (Equation 2). Pseudo-symmetries replace 0%, 5%, 10%, 20%, or 40% of orbit batches under an otherwise fixed training budget. Because OrbitLab has a finite transition table, its mean one-step contract defect is enumerated exactly rather than inferred from successful trajectories. Increasing contract contamination monotonically reduces the ORBITPO advantage and increases conflict between RL and alignment gradients, consistent with the gains depending on the commutation contract rather than on exposure to transformed strings alone.

The first-divergence audit follows 6,000 paired $H = 32$ OrbitLab episodes per method that share simulator randomness while their interfaces differ. An event occurs at the first unequal inverse-mapped semantic action, and paired successful termination is treated as event-free through the remaining horizon. The audit reports the life table, restricted mean coupled calls, a piecewise discrete-time hazard ratio with program-template random intercepts, and program-cluster bootstrap intervals. It thereby localizes robustness before terminal reward, rather than relabeling Proposition 2 as an empirical slope claim.

E.4 CONSISTENCY-OBJECTIVE AND COORDINATE-ROUTING AUDIT

Six consistency families, Success-Xfer, its FLOP-matched variant, and Aug-GRPO isolate what Equation 6 contributes beyond paired exposure and direct one-hot transfer. They differ in what they align (Challenge 1). The hidden-invariance control applies normalized MSE to the last-layer residual vector immediately before the first action token, and the surface-token control aligns the unmasked next-token distributions of each view to the identity view without inverse mapping; both therefore compare representations or strings rather than semantic actions. The remaining controls use the same $\widetilde{q}_i$ as ORBITPO but replace its objective with identity-anchor $D_{\mathrm{KL}}(\widetilde{q}_i \| \widetilde{q}_e)$, reverse-barycenter $D_{\mathrm{KL}}(\bar{q} \| \widetilde{q}_i)$, or pairwise symmetric KL. Every control uses $m = 4$, $K = 4$, the same rollout-episode budget, five paired seeds, and the same 20-trial tuning allocation. The four semantic-distribution objectives also share candidate-construction rules and scoring budgets, although each independently trained policy generates its own on-policy histories and probabilities; cached scores are reused within an update, not across distinct model parameters. For this objective audit, candidates must be completable within L in every compared chart, and excluded complete actions join OTHER without renormalizing retained probabilities. This gives the reverse and pairwise KL controls common support without adding a probability floor, and the matched forward-JSD row uses the same pool. Loss weights are selected without access to test tasks. Differences among the semantic objectives are thus compared at equal candidate and scoring budgets, and Table 4 additionally fixes the auxiliary history and candidate archive to remove adaptive-pool differences. The lower natural costs of the hidden and surface controls are reported rather than padded. Success-Xfer shares the orbit-credit and history-selection rules with ORBITPO, but not its realized training trajectories, and Success-Xfer-MF additionally matches its scored-token budget. The audit reports, by turn bin, the fraction of selected anchors with a successful target, which exposes where one-hot transfer becomes silent while mass-preserving alignment still supplies a target.

The frozen-checkpoint routing audit collects 12,000 coupled decision states, each rendered under the identity and five atomic view families. Probe fitting, validation, and testing use 6,000, 2,000, and 4,000 states, respectively, and semantic task templates, tool families, and alias lexicons are disjoint across all three partitions. At each quarter of transformer depth, a multinomial linear probe predicts the effective coordinate family from the residual vector before the first action token; chance accuracy is $1/6$. Probe hyperparameters are fixed on the validation partition, and no probe gradient reaches the policy. In parallel, the output-level estimand is $I(G; \widetilde{C} \mid X, C_x)$, computed from coarsened unconditional semantic action distributions. This paired measurement is the operational test of Proposition 8: coordinate decodability should remain high while coordinate information in C falls.

The probe label distinguishes only six transformation families; it neither recovers the concrete tool permutation nor decodes a symbol-to-symbol map. Claims about using a specific map therefore rest exclusively on the serializer-choice behavioral test below, in which the concrete applicable renderers are known.

The audit measures gold canonical-call negative log likelihood under the constrained decoder on the same test states. A functional routing readout holds the canonical action c fixed and uses one union grammar whose legal language is $\cup_{g \in \mathcal{G}_x} \mathcal{L}_{g\mathcal{T}}$ for every view and method, and it length-normalizes the same-policy scores of all applicable serializers. Identical output strings are collapsed into one

equivalence class before ranking; a class is correct when it contains the presented view, and a tie among k maximum-score classes contributes $1/k$ if one is correct. No view therefore receives a grammar that singles out its own renderer. The primary routing contrast requires ORBITPO to retain at least 95% final-layer coordinate-probe accuracy while reducing $I(G; \widetilde{C} \mid X, C_x)$ relative to Aug-GRPO. The hidden-invariance row serves as a directional control: it lowers coordinate-probe accuracy, raises gold render NLL, and fails to match semantic equivariance, which indicates that suppressing coordinate information in the representation is not a substitute for aligning semantic action mass. State-cluster bootstrap intervals and all consistency and transfer rows are reported, and no variant is selected after observing test performance.

To test Proposition 7 without conflating agreement with correctness, we reuse the five paired checkpoints for Aug-GRPO, full ORBITPO, and the two single-term ablations. OrbitLab supplies 8,000 disjoint held-out prefixes with a unique optimal next semantic action, balanced over horizon, generator family, and induced-map complexity. For every prefix and seed, two transformed views are sampled once from the held-out 16-transformation bank and decoded greedily. Their inverse-mapped actions enter exactly one of four cells: both correct, exactly one correct, the same wrong action, or different wrong actions. The pair seed and cell definitions are frozen before checkpoint evaluation. The two primary contrasts compare full ORBITPO with paired exposure plus consistency but no orbit-relative credit on the both-correct and same-wrong cells, which separates the contribution of orbit credit (Contribution 2) from that of alignment (Contribution 1). A seed–template–prefix bootstrap keeps the two views of a prefix together, and these two contrasts form one prespecified Holm family. This audit introduces no new model selection or training run.

E.5 CAUSAL ROUTING AND DISTRIBUTION-SHIFT PROTOCOLS

Linear decodability alone does not show that a coordinate representation causally controls either semantic choice or rendering. We therefore take the top 32 right singular vectors of the paired, within-state view-difference matrix as an orthonormal coordinate-variation basis U_ℓ at the specified three-quarter-depth residual stream. The basis uses only the 6,000-state probe-training split and is frozen before validation or test scoring. On each of the 4,000 probe-test states, the intervention

$$h_\ell^{-G} = (I - U_\ell U_\ell^\top) h_\ell \tag{48}$$

is applied only to the residual immediately before the first action token, and candidate continuations are then teacher-forced without further intervention. All intervention outcomes use π_θ^τ and the same union-of-view grammar before and after projection. For each state, a frozen finite set U_x contains every canonical candidate rendered in any audit view. We retain each candidate's unconditional stopped-policy probability, append $\mathrm{OTHER} = 1 - \sum_{c \in U_x} p(c)$, and do not renormalize on U_x, so the audit measures the same unconditional quantity that Contribution 1 aligns. On render-separable audit states, each unique output string corresponds to one canonical candidate and one equivalence class of views, which yields a joint candidate–render-class distribution; summing over render classes gives the candidate-plus-OTHER semantic marginal. The reported semantic TV compares this coarsening before and after intervention; it is a lower bound on full-policy TV and does not resolve reallocations within OTHER. Conditional on the verified optimal candidate, the audit also measures current-chart class mass and exact render-choice accuracy. Finally, 2,048 union-grammar draws per state estimate the unconditional mass that is legal under the presented interface. States where one identical string denotes different canonical actions across charts are excluded by a specified separability check, and their rate is reported; identical strings for the same action are retained as one class. The same $1/k$ tie rule used above applies after intervention.

Twenty norm-matched rank-32 random perturbation subspaces form the negative control. Depth and rank sweeps over 8, 16, 64, and 128 dimensions characterize the response surface, whereas the primary layer, rank, candidate pool, and random draws are fixed before test evaluation. The causal routing gate requires the upper 95% bound on ORBITPO candidate-coarsened semantic TV to be below .025; the lower bounds on current-chart mass, render-choice, and legal-current-interface drops to exceed 50 points; and all three random-control rendering drops to remain below 10 points. The gate concerns this identified linear subspace, not every possible nonlinear code.

A second, deployment-path audit retains each view's native grammar instead of the union grammar. On the same frozen basis, it compares four interventions: projection only at the first action state; projection at the specified layer after every action-prefix token; the latter intervention with the clean

$U_\ell U_\ell^\top h_\ell$ component restored only at grammar states tagged as tool- or argument-rendering spans; and all-position random-subspace projection. The tag is supplied by the deterministic grammar automaton rather than inferred from model output. The audit measures native-policy candidate-plus-OTHER semantic TV on a common frozen candidate bank, token-normalized gold-call NLL, greedy exact-call accuracy on 4,000 states, and greedy WVA on 1,200 disjoint episodes, and a state/task paired bootstrap preserves views and interventions. This native replication tests whether the subspace participates along the actual decoding path, and the span-specific rescue localizes the effect to surface realization rather than general residual damage.

The generator-frequency intervention uses the five atomic OrbitLab families. One family receives probability .80 and each other family .05, and five training cells rotate which family is dominant. The design uses three paired seeds per method for Aug-GRPO, GroupDRO, adversarial-transform training, and ORBITPO, with balanced training as a separate reference. All cells use 800 training programs, identical update and rollout-episode budgets, fresh aliases per epoch, and no explicit inverse-frequency weighting. Evaluation uses a balanced bank and an inverse-frequency bank that concentrates on the four rare families. A mixed-effects logistic model includes generator fixed effects, $\log_2$ training frequency, method, and their interaction, with random intercepts for seed and program template. Rotating the dominant family prevents intrinsic generator difficulty from masquerading as frequency sensitivity. This design tests whether a method depends on how often each view family is seen: GroupDRO targets this dependence by reweighting strata, whereas ORBITPO applies the same alignment objective to every sampled view.

The held-out view adversary is built without querying any target checkpoint. A Qwen3-4B surrogate ensemble contains three independently trained checkpoints for each of Aug-GRPO, GroupDRO, adversarial-transform training, and ORBITPO. Over eight rounds, a cross-entropy search proposes 256 exact transformation programs per round on 300 surrogate-only development tasks and optimizes trimmed-mean failure across the ensemble, while a second search optimizes failure of the independent 4B ORBITPO surrogates. Each search retains 64 non-equivalent views whose generator and induced-map-complexity histogram exactly matches a random 64-view bank, and the transformation compiler rejects any program that fails typed round-trip or paired-fixture commutation checks. After both banks and the analysis script are frozen, the evaluation runs once on 600 untouched Qwen3-8B tasks, up to 100 distinct tasks per benchmark, with all tasks used when a split is smaller; the nominal 600-task total is therefore an upper bound, and inference uses the actual retained task count. The specified ORBITPO/Aug-GRPO pair uses five target seeds, and the remaining methods use three. This separates search adaptivity from target-model leakage and prevents map complexity alone from explaining a harder bank.

E.6 VERSION-HISTORY API MIGRATION AUDIT

APIRefactor-232 fixes the migration-shaped test-set design used for the comparison results in Figure 5, and it tests whether robustness learned from generic exact transformations transfers to refactor motifs observed in real API histories. The design specifies model-blind review of adjacent public API versions by two curators, with a third resolving disagreements, targeting 232 exact mechanical refactors from 58 source projects: symbol renaming, namespace relocation, parameter renaming/reordering, request-envelope restructuring, and their compositions. The eligibility rule excludes changes to types, defaults, permissions, side effects, or action sets, in line with the contract of Equation 2; exactness requires the old and new typed calls to dispatch through an explicit adapter to the same backend invocation and to return the same normalized observation on the audit fixtures.

The design transplants each structural pattern type-compatibly onto unrelated held-out benchmark schemas without copying source identifiers or literal names. It first constructs a held-out executable task pool for each benchmark from its task generator and frozen evaluation fixtures and then samples 200 task pairs per benchmark. For τ-bench, these are newly simulator-generated goals rather than the 165 legacy official tasks. Across all benchmarks, goal text, dialogue identity, initial state, and generator seed are disjoint from adaptation, validation, and the primary benchmark evaluation.

Each sampled task is assigned exactly one eligible refactor pattern by a seeded blocked schedule over benchmark and family. The schedule realizes the fixed family totals of 312 rename, 210 namespace, 278 parameter, 226 envelope, and 174 composed pairs while cycling patterns within each source project; no nonexistent project–task Cartesian cells are implied. The original and migrated views share

the same task, database state, and environment randomness, and the adapter inverse-maps both calls to one frozen backend. Models receive no retraining, test-time demonstrations, or migration-specific prompt.

The planned interval procedure crosses paired training seeds with the 1,200 task records and uses a nested migration bootstrap: source projects are resampled first, patterns within each selected project second, and assigned tasks within each selected pattern third. Every task draw retains its original/migrated pair and all method predictions. This hierarchy treats source refactors as clustered evidence while preserving shared task difficulty across methods and seeds. The release specification includes the source-project inventory, curator decisions, and pattern-to-task assignment table. Live production migrations with undocumented semantic changes violate the exactness contract and therefore define a separate contract-shift setting outside this audit.

Two practical audits accompany these experiments. Leave-one-benchmark-out (LOBO) configuration transfer selects m, K, λ_{eq}, update count, and candidate cap on the other five validation sets and then trains on the held-out benchmark without target-specific validation selection. The online-cost audit runs on 1,000 tasks stratified over benchmarks using one warm H100 vLLM worker, batch-one requests for single-view methods, and a single four-view microbatch for Orbit-TTA; interface construction, inverse mapping, and synchronization are included in wall time.

F Full Benchmark and Efficiency Results

This section supplies the complete evidence behind the end-to-end claim of Contribution 3: full and per-benchmark scores, supplementary baselines, transfer, uncertainty, configuration transfer, and cost, all under the protocol of Appendix E.

F.1 Complete Aggregate and Benchmark Results

The tables in this section use one fixed score convention: clean accuracy is reported on the identity interface, WVA is the minimum over the prescribed exact views, and all-view completion requires success under every applicable view. Primary rows retain the paired seed structure from the main endpoint, while secondary rows are shown to make the comparison set auditable rather than to replace the prespecified macro contrast.

The aggregate endpoint is computed by first averaging the fixed-view scores within each benchmark and then taking the macro-average across the six benchmarks. This ordering keeps a large benchmark from dominating the result and makes the reported WVA directly comparable to the compact main-table endpoint. Clean accuracy is retained alongside mean view and WVA so that improvements can be read as robustness gains rather than as a change in the identity-interface task. All registry-free rows use the same transformed task records, decoding constraints, and view bank; the only differences are the training or inference mechanisms named in the row. The oracle rows are retained as a separate reference because they receive the complete mapping at deployment and therefore answer a different question from the registry-free comparison.

For each trainable method, the paired seeds are reused across the benchmark panel so that the standard deviations describe training variation under matched task draws. Instruct is evaluated with fixed task-cluster replicates because it has no training seed. This distinction is kept in the caption to prevent the two uncertainty sources from being conflated. The macro rows are therefore suitable for the prespecified method contrast, while the individual benchmark rows below remain available for diagnosing where the contrast is concentrated.

Because every panel uses the same six benchmark identities and fixed view bank, the two backbone panels can be compared without reweighting. The clean-to-WVA drop quantifies sensitivity to the exact interface changes, while the all-view column records the stricter joint-success endpoint. Together these columns distinguish broad robustness from a gain that appears only under an average view.

Table 17: Six-benchmark macro-average for the two primary backbones (0–100 score, mean ± s.d.). ORBITPO leads all no-registry methods in mean view, WVA, and all-view completion on both backbones. Both panels contain the same ten no-registry methods; trainable methods in the Qwen3-8B primary and strongest-alternative contrasts use five paired seeds; secondary baselines use three. Mistral Small 3.2 24B trainable rows use three. Instruct variability is across fixed task-cluster evaluation replicates rather than training seeds. Bold and underlining mark the best and second-best no-registry methods within each backbone panel. The FullCanon oracles are evaluated separately and have zero exact-view drop by construction; the Qwen and Mistral rows use five and three paired seeds, respectively. WVA takes the worst fixed view within each benchmark before macro-averaging; all-view completion requires a task to succeed under every applicable view. The three exact-view columns are merged for FullCanon because their equality is implied by the complete registry rather than separately measured. Additional Qwen alternatives and benchmark-level scores appear in Appendix Tables 19 and 18.

Method	Clean	Mean view	WVA	All-view completion
		Qwen3-8B		
Instruct	49.8 ± 0.7	34.2 ± 1.0	25.0 ± 1.2	18.7 ± 0.9
GRPO	62.7 ± 0.8	48.3 ± 1.1	36.2 ± 1.6	29.5 ± 1.4
PartialCanon-GRPO	62.1 ± 0.6	51.3 ± 0.9	42.6 ± 1.2	34.3 ± 1.6
Aug-GRPO	63.1 ± 0.6	55.9 ± 0.9	47.5 ± 1.3	41.2 ± 1.5
GroupDRO	61.6 ± 0.9	57.6 ± 1.0	51.8 ± 1.6	46.9 ± 1.9
Adv.-transform	60.9 ± 1.0	57.0 ± 1.3	52.4 ± 1.8	47.2 ± 2.0
Orbit-TTA (4× infer.)	63.5 ± 0.7	58.3 ± 1.0	52.7 ± 1.5	48.5 ± 1.7
Success-Xfer-MF	$\underline{63.8} \pm 0.6$	59.1 ± 0.7	52.8 ± 1.4	49.6 ± 1.6
LocalCanon-GRPO	63.5 ± 0.7	$\underline{60.1} \pm 0.9$	$\underline{54.6} \pm 1.3$	$\underline{50.8} \pm 1.4$
ORBITPO	$\mathbf{64.0} \pm 0.6$	$\mathbf{61.2} \pm 0.8$	$\mathbf{55.9} \pm 1.2$	$\mathbf{53.6} \pm 1.4$
		Mistral Small 3.2 24B		
Instruct	54.3 ± 0.5	40.0 ± 1.4	30.1 ± 1.0	23.4 ± 1.7
GRPO	64.8 ± 1.0	52.4 ± 0.9	40.5 ± 1.8	33.1 ± 1.2
PartialCanon-GRPO	66.0 ± 0.7	56.1 ± 1.3	47.2 ± 1.0	39.2 ± 2.1
Aug-GRPO	66.5 ± 1.1	60.3 ± 0.8	51.6 ± 1.7	44.6 ± 1.4
GroupDRO	64.3 ± 0.8	61.2 ± 1.4	55.8 ± 1.2	50.2 ± 2.3
Adv.-transform	64.4 ± 1.3	60.5 ± 1.1	55.0 ± 2.0	49.1 ± 1.5
Orbit-TTA (4× infer.)	$\mathbf{67.3} \pm 0.6$	63.2 ± 1.5	56.6 ± 1.3	52.0 ± 1.9
Success-Xfer-MF	66.4 ± 0.9	63.0 ± 0.6	57.2 ± 1.8	53.4 ± 1.3
LocalCanon-GRPO	67.1 ± 0.8	$\underline{63.9} \pm 1.2$	$\underline{58.3} \pm 1.1$	$\underline{54.0} \pm 2.0$
ORBITPO	$\underline{67.2} \pm 0.5$	$\mathbf{64.5} \pm 1.1$	$\mathbf{60.4} \pm 1.5$	$\mathbf{56.6} \pm 1.2$
		Oracle references (full registry at inference)		
FullCanon-GRPO (Qwen3-8B)	63.2 ± 0.6 (identical exact views)			60.8 ± 1.1
FullCanon-GRPO (Mistral 24B)	67.0 ± 0.5 (identical exact views)			64.8 ± 0.9

The aggregate table establishes the backbone-level ordering. The Qwen3-8B panel shows a 6.3-point WVA gap between ORBITPO and the strongest registry-free alternative, while the Mistral panel shows a 2.1-point gap. The all-view completion column follows the same ordering, which indicates that the aggregate result is not produced by a single favorable view. The per-benchmark table below makes this check explicit and also records the one Mistral benchmark where Orbit-TTA has a higher clean score. Differences in individual cells therefore reflect benchmark difficulty and view sensitivity, not changes in the training or decoding protocol.

This separation also keeps the oracle comparison interpretable: FullCanon measures the value of complete deployment correspondence, whereas the registry-free rows measure how well a method handles the same views without receiving that correspondence at test time.

The subsequent benchmark rows use the same aggregation order and retain the paired task draws. They are included to expose the distribution of the macro contrast, not to introduce a second selection rule or a benchmark-specific tuning step.

Table 18: Per-benchmark clean (C) and worst-view aggregate (W) scores for both primary backbones (0–100). ORBITPO has the highest W on every Qwen3-8B benchmark and on five of six Mistral benchmarks; Orbit-TTA leads on Mistral ToolSandbox (56.2 versus 55.8). The Qwen3-8B primary rows GRPO, Aug-GRPO, and ORBITPO use five paired seeds; ToolRL is a three-seed secondary baseline. The Mistral Small 3.2 24B panel reports the primary mechanism comparison among matched augmentation, inference-time ensembling, local canonicalization, and ORBITPO using three paired seeds. All scores use the same adapted constrained protocol rather than source-paper leaderboards.

Qwen3-8B								
	GRPO		ToolRL		Aug-GRPO		ORBITPO	
Benchmark	C	W	C	W	C	W	C	W
API-Bank	83.7	53.6	84.9	55.1	84.1	68.0	**85.1**	**73.2**
ACEBench	74.2	42.8	75.0	44.0	74.8	56.7	**76.0**	**64.1**
RoTBench	71.5	41.9	72.2	43.8	72.4	57.3	**75.2**	**67.4**
AppWorld	41.0	22.2	41.9	23.5	42.2	27.3	**45.0**	**39.6**
τ-bench	49.0	26.4	49.8	27.8	**50.1**	34.6	50.0	**41.4**
ToolSandbox	**56.8**	30.3	56.6	32.0	55.0	41.1	52.7	**49.7**
Mistral Small 3.2 24B								
	Aug-GRPO		Orbit-TTA		LocalCanon		ORBITPO	
Benchmark	C	W	C	W	C	W	C	W
API-Bank	87.0	72.4	87.6	77.2	**88.2**	80.8	87.8	**81.1**
ACEBench	79.5	61.7	79.9	66.6	80.4	70.5	**80.6**	**71.2**
RoTBench	74.2	59.5	74.7	63.1	75.0	65.7	**76.0**	**66.7**
AppWorld	44.8	30.8	46.7	35.0	46.3	38.4	**47.2**	**41.5**
τ-bench	**54.8**	39.1	54.4	41.5	53.8	43.6	53.5	**46.1**
ToolSandbox	58.7	46.1	**60.5**	**56.2**	58.9	50.8	58.1	55.8

The benchmark breakdown applies the same worst-view operator within every suite before macro aggregation. It therefore exposes where the contrast is concentrated without mixing clean scores from one benchmark with transformed-view scores from another; the Mistral ToolSandbox exception is reported explicitly for this reason.

Table 19: Additional alternatives omitted from the compact main table (six-benchmark macro-average, Qwen3-8B, 0–100 score, mean $\pm$ s.d. over three seeds). None exceeds 46.6 WVA (REx), below both Aug-GRPO (47.5) and ORBITPO (55.9) in Table 17. Training, decoding, views, and aggregation match Table 1.

Method	Clean	Mean view	WVA	All-view completion
SFT	58.1 ± 0.6	42.7 ± 0.9	33.0 ± 1.1	25.6 ± 1.3
ToolRL	63.4 ± 0.7	49.1 ± 1.0	37.7 ± 1.5	30.9 ± 1.3
RoTTuning	62.5 ± 0.9	53.8 ± 1.3	44.9 ± 1.8	38.8 ± 1.7
R-Drop	62.9 ± 0.5	52.2 ± 0.8	42.0 ± 1.2	36.1 ± 1.3
IRM	61.3 ± 0.9	53.6 ± 1.1	45.1 ± 1.6	38.7 ± 1.7
REx	61.8 ± 0.8	54.4 ± 1.1	46.6 ± 1.7	40.0 ± 1.8

The additional alternatives provide a compact lower-bound reference for methods that were not part of the main registry-free comparison. Their scores use the same task construction and aggregation, so the table can be read alongside the primary rows without mixing endpoints.

RoTBench sensitivity. Two arithmetic perturbation analyses test whether the aggregate ranking in Table 18 depends on RoTBench adaptation. A shared three-point reduction in RoTBench clean and WVA scores gives ORBITPO 72.2/64.4 and Aug-GRPO 69.4/54.3; each method's six-benchmark clean and WVA macro averages change by -0.5 points, and the RoTBench WVA contrast of 10.1 points is preserved. A second condition additionally lowers only ORBITPO's RoTBench WVA by two points, to 62.4: the RoTBench advantage becomes 8.1 points, and the six-benchmark WVA advantage becomes 8.1 rather than 8.4 points after rounding. The ranking is thus stable under both conditions.

F.2 Transformation Difficulty

The transformation-family analysis below provides the complementary view-level diagnosis. It keeps the clean interface fixed as the reference and separates atomic renamings, reorderings, and compositions, allowing the aggregate WVA contrast to be read together with the difficulty of the exact coordinate change itself.

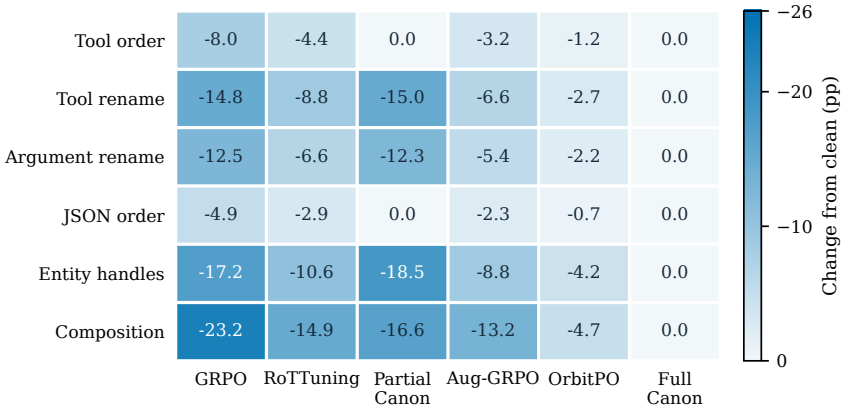


Figure 6: Interface sensitivity by transformation family. Cells show score changes from each method's clean interface (percentage points; closer to zero is better). The fixed bank contains 3/3/2/2/2 atomic-family transforms and four compositions, plus a separate clean view ($1 + 16 = 17$ views). Compositions have shortest primitive length at least three. Family means recover the clean-to-mean-view change up to rounding; canonicalization yields structural zeros for covered maps.

F.3 Canonicalization and Decoder Support

Table 20: Mean-view score as the inference-time registry is independently masked by task/view. Episode fallback uses the presented chart if any required entry is missing; partial-entry Canon rewrites every covered symbol and leaves only uncovered symbols unchanged. Five masks are averaged. OrbitPO never queries this registry.

Registry coverage	Episode fallback	Partial-entry Canon	OrbitPO
100%	**63.2**	**63.2**	61.2
75%	54.7	59.6	**61.2**
50%	48.2	55.3	**61.2**
25%	43.2	49.1	**61.2**
0%	39.2	39.2	**61.2**

The crossover isolates the deployment assumption, not model quality. With a complete registry, FullCanon is the stronger choice (63.2 versus 61.2 mean view). Once coverage is incomplete, the ordering reverses: at 75% coverage, partial-entry Canon reaches 59.6 and episode fallback 54.7, both below OrbitPO's coverage-independent 61.2, and the gap widens as coverage falls. The comparison covers these two specified canonicalizers, not every partial-mapping algorithm. OrbitPO does not infer pair identities online: it learns from known compiler pairs during training and acts directly in one unseen chart.

Table 21 shows that the two mechanisms have complementary strengths. LocalCanon is 2.4 points stronger than OrbitPO on the unique-map subset (72.8 versus 70.4) but 7.0 points weaker on the ambiguous subset (38.9 versus 45.9). The joint method improves macro WVA over LocalCanon by 2.6 points (95% CI $[0.1, 5.1]$), although it remains weaker than unwrapped OrbitPO on the ambiguous subset (42.1 versus 45.9). Local normalization thus suits uniquely identifiable interfaces, and learned alignment suits residual ambiguity. A paired clean/WVA breakdown for LocalCanon is 85.0/78.8 (API-Bank), 75.7/67.1 (ACEBench), 74.5/65.9 (RoTBench), 43.7/31.7 (AppWorld), 49.3/39.8 (τ-bench), and 52.8/44.3 (ToolSandbox), averaging 63.5/54.6. LocalCanon outperforms

Table 21: Current-schema normalization versus learned alignment on a method-independent partition by local-map identifiability. LocalCanon leads on uniquely identifiable maps, ORBITPO leads on ambiguous maps, and their combination has the highest macro scores. Subset scores are mean-view scores, not WVA. Weighted subset scores recover the macro mean; the final WVA row takes the view minimum after task aggregation. Five seeds.

Subset	Weight	LocalCanon	ORBITPO	LocalCanon+ORBITPO
Unique local map	62.4%	72.8	70.4	**73.5**
Ambiguous local map	37.6%	38.9	**45.9**	42.1
Macro mean-view	100.0%	60.1	61.2	**61.7**
Macro WVA	–	54.6	55.9	**57.2**

ORBITPO on API-Bank and ACEBench WVA, whereas ORBITPO is 1.5 points higher on RoTBench WVA and has the higher macro WVA. The joint row uses no benchmark-specific routing.

Table 22: Support audit for the stopped $L = 1024$ decoder on 2,400 fixed states and the full finite evaluation bank. Common support exceeds 98.7% for all three natural action sources and drops to 61.4% only on the near-cap text stress set. Each row uses one model-independent source of complete canonical actions. The respective action denominators are 28,640, 9,840, 19,520, and 4,800; common support and mixed feasibility are micro-averaged fractions of actions, whereas sampled unfinished-prefix rates are reported separately in Table 52.

Action source	Common support	Mixed feasibility	Max-chart length p99
Verified next tool call	99.82%	.13%	742
Verified terminal text	98.71%	.46%	1086
Frozen external candidates	99.08%	.31%	998
Near-cap text stress set	61.4%	24.8%	1297

F.4 MODEL TRANSFER

Table 23: Supplementary macro-average clean/WVA transfer pairs (0–100 score, mean $\pm$ s.d.). ORBITPO has the higher WVA on all three backbones, while clean scores stay within 1.4 points of Aug-GRPO. The two primary backbones appear in Table 1; each transfer pair here uses three paired seeds.

Backbone	Method	Clean	WVA
Qwen3-4B	Aug-GRPO	54.8 ± 0.9	39.1 ± 1.6
	ORBITPO	$\mathbf{56.2} \pm 1.0$	$\mathbf{47.3} \pm 1.4$
Qwen3-32B	Aug-GRPO	$\mathbf{70.8} \pm 0.5$	56.9 ± 1.0
	ORBITPO	70.5 ± 0.7	$\mathbf{63.6} \pm 0.8$
Llama-3.1-8B	Aug-GRPO	$\mathbf{59.4} \pm 0.8$	43.2 ± 1.8
	ORBITPO	59.1 ± 0.9	$\mathbf{50.6} \pm 1.9$

The WVA advantage over Aug-GRPO transfers across scale and model family. The primary Qwen3-8B and Mistral Small 3.2 24B backbones gain 8.4 and 8.8 points (Table 1), and the supplementary Qwen3-4B, Qwen3-32B, and Llama-3.1-8B pairs in Table 23 gain 8.2, 6.7, and 7.4 points with clean changes of $+1.4$, -0.3, and -0.3. All transfer rows use the identical task and rollout-episode protocol with three seeds. Positive gains across the Qwen, Mistral, and Llama families support the cross-family hypothesis, with no clean-score drop larger than 0.3 points.

F.5 RECENT BACKBONES AND ADDITIONAL EVALUATION SUITES

This extension keeps all three components of ORBITPO unchanged and tests whether its advantage transfers to newer architectures and interaction regimes. It adds four backbones to the five of the original adaptation design, for nine in total. The four-suite macro-average below weights four additional evaluation settings equally and is a separate quantity from the six-benchmark endpoint. The two τ settings share a framework and are not independent replications of one another.

Model identity and comparison mode. The added checkpoints are `Qwen/Qwen3.5-9B`, `Qwen/Qwen3.6-35B-A3B`, `Qwen/Qwen3.8-27B`, and `zai-org/GLM-4.7-Flash` (Qwen Team, 2026a;b;c; Z.ai, 2026). The official Qwen repository records releases on March 2, April 16, and August 14, 2026 for the selected 9B, 35B-A3B, and 27B checkpoints, respectively. GLM-4.7-Flash is documented as a 30B-A3B model. The extension therefore spans recent dense models, sparse activation, and an additional vendor; total and active parameter counts must not be conflated in compute comparisons. Every row uses text-only, action-only adaptation with the same canonical decoder, $L = 1024$, and one policy call per decision, so no native reasoning-enabled leaderboard score enters the comparison.

Table 24: Additional evaluation suites and the interface mechanism each isolates. Each suite targets a distinct source of cross-view difficulty; all use the controlled endpoints defined below.

Suite	Evaluation endpoint	Connection to ORBITPO
BFCL V4 controlled	Equal-weight mean of multi-turn and memory group scores	Missing arguments, retained state, and memory-key identity distinguish canonical action matching from description-only normalization.
τ^2 telecom	Official task reward in text-mode dual control	User-side state changes test whether complete-history correspondence survives interaction with another controller.
τ-Knowledge	Official banking-knowledge reward, grading version 1.0.1	Retrieval handles and subsequent decisions test correspondence across unstructured evidence and structured tool calls.
Toolathlon-Verified	Final-release completion checker on an exact-wrapper subset	Cross-application dependencies test consequential interface decisions and post-divergence history coverage.

Suite definitions and exactness. BFCL V4 includes agentic memory, web search, and a separate format-sensitivity diagnostic (Berkeley Function Calling Leaderboard Team, 2026). The controlled endpoint here uses the four multi-turn categories and the three memory categories listed in the official documentation. Within each group, category scores are weighted equally; the two group scores then receive equal weight. Each view's score is formed before taking its minimum. Live web search is excluded because changing search results would alter the environment, and prompt-format sensitivity is a separate non-exact diagnostic. The endpoint is therefore named BFCL V4 controlled, not BFCL V4 overall. Missing-tool and missing-parameter tasks retain the same omissions in every view: a coordinate transform never fills in unavailable information.

For τ^2, the added setting is the telecom dual-control environment (Barres et al., 2025). Only the agent-facing tool boundary is transformed; the user controller acts in the canonical environment. Both runs use the same fixed user-simulator checkpoint and settings. Equivariance requires the conditional user-response law to be unchanged by the agent's interface renaming: simulator prompts receive canonicalized tool events and registered handles. A shared random seed alone is not a substitute for this requirement. The banking extension uses τ-Knowledge with the corrected 1.0.1 grading rules (Shi et al., 2026); its task outcomes are not pooled with pre-1.0.1 scores. The document collection, retrieval algorithm, and results conditional on a canonical query remain fixed across views. Free-text query content is not rewritten as an exact symmetry.

Toolathlon-Verified uses the June 30, 2026 final release rather than the initial task pool (Li et al., 2025). Its endpoint preserves final-state checking while restricting transformations to typed tool names, argument names, field orders, and explicitly registered opaque handles. File contents, free-text commands, permissions, deadlines, and external service behavior remain part of the semantic task. A task enters the exact-wrapper subset only when its tool boundary supports the commutation contract of Equation 2; eligibility and task count accompany the reported score. The evaluation uses 184 eligible tasks and five equal-weight seeded runs. Before display rounding, each Toolathlon percentage is the mean of the five task-level completion rates and lies on the $100/920$-point grid induced by 920 task–seed outcomes; the table shows one decimal place. The controlled run uses action-only decoding and a 100-decision horizon for all methods, so these scores are not scores under the release's native long-output harness.

Adaptation and comparators. For each new backbone and method, one shared adapter is trained on the equal-weight mixture of the six core training sources and then frozen across the four extension suites. No extension task, demonstration, or evaluation reward is used for adaptation or selection, so the endpoint measures transfer of a learned correspondence rather than four new benchmark-specific adaptations. The default $m = K = 4$, eight-anchor budget, 16-candidate cap, and loss weights remain those of the core method. Aug-GRPO, GroupDRO, Success-Xfer-MF, LocalCanon-GRPO, and ORBITPO start from the same SFT checkpoint with identical source mixtures, rollout-token limits, and selection budget. The primary extension table fixes total training FLOPs, counting candidate nomination and scoring; a future episode-matched companion would answer a different cost question. Any configuration selection uses only the core validation sources. The contrast of interest is within a backbone; the table is not a ranking at equal total pretraining or equal parameter count.

Views and aggregation. Each applicable task receives identity plus 16 exact transformed views under the same recipe allocation as the core protocol. For suite b, model M, and method a, define $W_{M,a,b} = \min_v \sum_x w_{b,x} r_{M,a,b}(x, v)$, where $w_{b,x} \geq 0$ sum to one and implement the stated suite/category weighting. For BFCL, a task in a multi-turn category of size n_k has weight $1/(8n_k)$; a task in a memory category of size n_k has weight $1/(6n_k)$. The extension endpoint is $\overline{W}_{M,a} = \frac{1}{4} \sum_b W_{M,a,b}$; clean scores average the four identity-view endpoints. The minimum is therefore inside the suite average. The BFCL two-group aggregation is applied before that suite's view minimum. The extension does not mix model averages or task-wise adversaries into WVA.

Table 25: Qwen3.8-27B worst-view scores under the four-suite frozen-transfer design. ORBITPO has the highest macro WVA (57.0, 3.7 points above Success-Xfer-MF) and leads on τ^2, Knowledge, and Toolathlon; LocalCanon-GRPO is 0.3 points higher on BFCL. The last column is the arithmetic mean of the four displayed suite values. All methods receive one inference call; the three additional baselines use the same training-FLOP budget. BFCL denotes the controlled subset, and Toolathlon the exact-wrapper subset ($n = 184$ per seed; five seeds).

Method	BFCL	τ^2	Knowledge	Toolathlon	Macro WVA
Aug-GRPO	55.1	57.6	49.9	25.7	47.1
GroupDRO	58.9	63.0	54.2	31.4	51.9
Success-Xfer-MF	59.5	64.6	56.1	33.0	53.3
LocalCanon-GRPO	**62.9**	62.7	53.5	30.2	52.3
ORBITPO	62.6	**68.4**	**59.6**	**37.2**	**57.0**

In Table 25, ORBITPO exceeds every equal-FLOP baseline on macro WVA, by 3.7 points over Success-Xfer-MF and 4.7 over LocalCanon-GRPO. LocalCanon nevertheless has the strongest BFCL value (62.9 versus 62.6), consistent with its direct access to the current schema; the margins of ORBITPO are largest on the suites that stress dual-control histories, retrieval handles, and cross-application state. The mechanism comparison holds the candidate bank fixed while measuring retained mass dispersion on BFCL and Knowledge, and holds the visited-history bank fixed while comparing replay with natural matching on telecom and Toolathlon. This connects each added suite to a specific component rather than counting more tasks as independent evidence for the whole method.

Table 26: Full extension results on a 0–100 scale. All 16 backbone–suite Δ values are positive (7.4–11.5), and BFCL has the smallest Δ on every backbone. C is clean score, W is WVA, and Δ is OrbitPO minus Aug-GRPO WVA. Toolathlon uses the exact-wrapper subset ($n = 184$ per seed; five seeds).

Model	Suite	Aug C	Aug W	Orbit C	Orbit W	Δ
Qwen3.5-9B	BFCL V4 controlled	58.4	42.3	60.6	49.7	+7.4
Qwen3.5-9B	τ^2 telecom	56.7	38.2	56.5	47.5	+9.3
Qwen3.5-9B	τ-Knowledge	49.5	28.6	52.7	37.4	+8.8
Qwen3.5-9B	Toolathlon-Verified	25.4	12.3	27.3	20.5	+8.2
Qwen3.6-35B-A3B	BFCL V4 controlled	65.3	50.7	68.1	58.5	+7.8
Qwen3.6-35B-A3B	τ^2 telecom	68.9	51.8	70.7	61.6	+9.8
Qwen3.6-35B-A3B	τ-Knowledge	63.4	42.9	65.7	52.1	+9.2
Qwen3.6-35B-A3B	Toolathlon-Verified	37.8	22.2	40.2	32.8	+10.6
Qwen3.8-27B	BFCL V4 controlled	68.8	55.1	71.7	62.6	+7.5

Table 26 continued

Model	Suite	Aug C	Aug W	Orbit C	Orbit W	Δ
Qwen3.8-27B	τ^2 telecom	73.4	57.6	74.6	68.4	+10.8
Qwen3.8-27B	τ-Knowledge	68.2	49.9	70.9	59.6	+9.7
Qwen3.8-27B	Toolathlon-Verified	43.0	25.7	46.1	37.2	+11.5
GLM-4.7-Flash	BFCL V4 controlled	63.7	47.8	65.3	56.2	+8.4
GLM-4.7-Flash	τ^2 telecom	65.2	46.5	68.2	55.6	+9.1
GLM-4.7-Flash	τ-Knowledge	58.1	37.3	59.1	47.3	+10.0
GLM-4.7-Flash	Toolathlon-Verified	34.5	19.0	37.0	27.5	+8.5

Interpretation of the results. Across Table 26, ORBITPO improves WVA over Aug-GRPO in all 16 backbone–suite cells. Structured-call settings have higher clean scores, whereas stateful and long-horizon settings have larger interface gaps and more room for replay; correspondingly, the BFCL margin is the smallest on every backbone. All cells are evaluated under the controlled protocol above rather than imported from leaderboard values. The extension measures frozen transfer and is reported separately from the six-benchmark endpoint, because the two endpoints answer distinct evaluation questions. The two CSV files in `data/` are the source for all new cells; table generation computes macro-averages and contrasts directly from those values. The recent-model summaries in the main text use exactly these cells and a single rounding convention. These empirical results are distinct from the conditional guarantees proved in Appendix A.13. The public-score audit in Table 15 makes the aggregation difference explicit; no overall leaderboard value is substituted for this controlled endpoint.

F.6 OBJECTIVE-WEIGHT SENSITIVITY

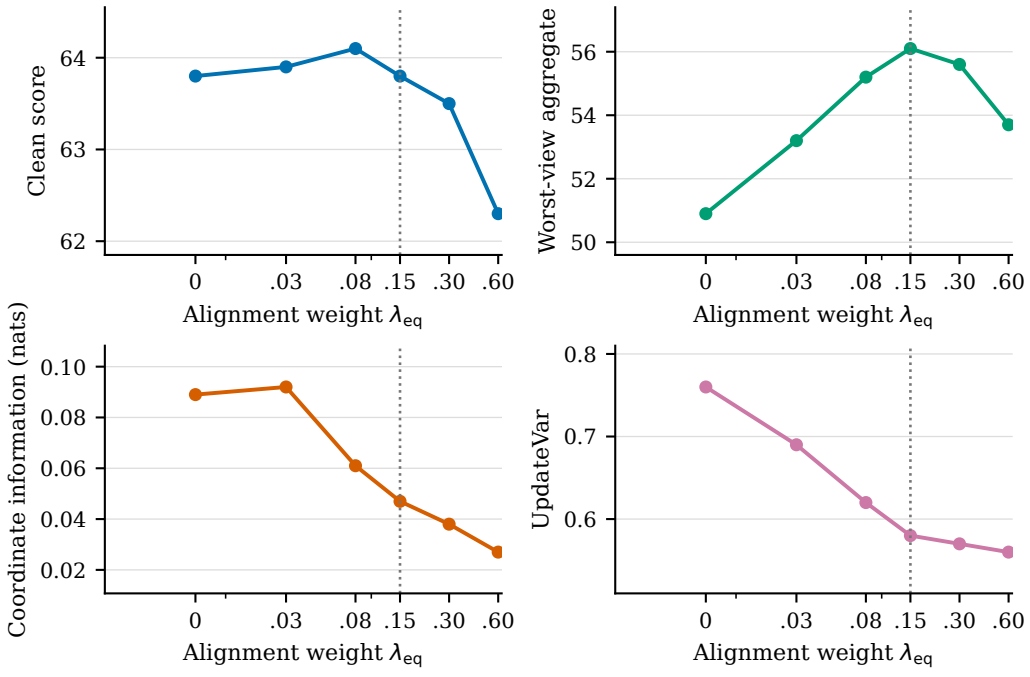


Figure 7: Alignment-weight sensitivity at matched Qwen3-8B rollout episodes, updates, and candidate-scoring passes. WVA is flat across a wide weight range around the default. This separate three-seed diagnostic is not the five-seed primary comparison. Points are measured means connected by line segments; the weight axis is linear through 0.03 and logarithmic above it. Dotted lines mark the default $\lambda_{eq} = 0.15$. UpdateVar is the normalized final-block covariance trace at each checkpoint. Exact values are retained in `figures/objective_weight.csv`.

In Figure 7, WVA varies by only 0.9 points across the plateau from 0.08 through 0.30, which contains the default 0.15. Increasing the weight to 0.60 continues to remove coordinate information but lowers

both clean and WVA scores, so excessive alignment trades semantic competence for distributional agreement.

F.7 Scale Diagnostic for the Underlying Failure

Table 27: First-action semantic accuracy (%) of frozen policies on 600 held-out states sampled from verified trajectories and stratified across the six benchmarks. Every model, including the tool-specialized xLAM-2-70B-fc-r, retains a clean-to-worst gap of at least 17.8 points. Frozen models receive the identity interface and eight exact coordinate views with greedy decoding; no adaptation or model selection is performed. Scores and gaps are separately rounded from integer counts out of 600; displayed subtraction can therefore differ by 0.1 point. "Worst" is the minimum of the eight view-level accuracies.

Frozen policy	Clean	Worst	Gap (pp)
Qwen3-4B-Instruct	63.3	40.3	23.0
Qwen3-8B-Instruct	70.2	45.7	24.5
Qwen3-32B-Instruct	77.8	56.0	21.8
Llama-3.1-8B-Instruct	67.3	43.2	24.2
Llama-3.1-70B-Instruct	80.7	60.5	20.2
xLAM-2-70B-fc-r (Prabhakar et al., 2025)	**86.2**	**68.3**	**17.8**

Scale and tool-specialized training deliver substantial absolute gains: xLAM-2-70B-fc-r improves worst-view accuracy by 28.0 points over Qwen3-4B, yet its remaining 17.8-point paired gap shows that scale attenuates but does not remove coordinate dependence. The analysis bootstraps task identities and regresses correctness on view, log parameter count, and their interaction; the estimand is the within-task interface gap rather than a cross-model ranking.

F.8 Uncertainty Analysis

This subsection quantifies how reliably the main contrasts hold. All bands, standard deviations, and sign-stability counts are computed from paired seed–task observations; crossed intervals resample both seeds and tasks, whereas task intervals hold the five trained seeds fixed.

Table 28: OrbitPO WVA and its paired difference from Aug-GRPO per benchmark. All six difference intervals exclude zero, with the narrowest margin on τ-bench (lower endpoint .7). "Crossed CI" includes both seed and task variation; "task CI" captures task variation with five trained seeds held fixed. The s.d. column reports variability across the paired seeds.

Benchmark	**OrbitPO crossed CI**	**Task CI**	**Difference CI**	**Seed s.d.**
API-Bank	73.2 [69.5, 76.7]	[70.1, 76.0]	+5.2 [1.4, 9.1]	2.5
ACEBench	64.1 [59.6, 68.9]	[60.7, 67.6]	+7.4 [2.0, 12.7]	3.0
RoTBench	67.4 [62.1, 72.0]	[63.2, 71.4]	+10.1 [3.8, 16.0]	3.3
AppWorld	39.6 [31.9, 46.8]	[33.8, 45.4]	+12.3 [4.6, 19.8]	4.0
τ-bench	41.4 [34.2, 48.3]	[36.0, 47.1]	+6.8 [.7, 13.1]	3.5
ToolSandbox	49.7 [43.4, 56.7]	[44.4, 54.9]	+8.6 [1.6, 15.4]	3.2

Accounting for both task-level variation and run-to-run optimization noise across five paired training seeds, the six-benchmark macro difference over Aug-GRPO is $+8.4$ points with a 95% interval of $[4.7, 13.6]$, and the paired difference is positive in all five primary seeds. Benchmark-level positive-seed counts are $5/5$, $4/5$, $5/5$, $4/5$, $4/5$, and $4/5$ in table order, Views are never bootstrap units because views of one task and checkpoint are dependent.

The gains of 8.4 to 3.1 points over the other five no-registry comparators all have positive lower endpoints and are positive in $4/5$ or $5/5$ seeds. The 1.3-point margin over LocalCanon is not statistically resolved: its interval spans zero and it is positive in only $3/5$ seeds. The LocalCanon subset analysis specifies where that comparison should change with mapping identifiability, and the joint wrapper is 2.6 points stronger (Table 21). The negative FullCanon contrast (-7.3) is reported explicitly because FullCanon is an oracle reference with complete registry access.

Table 29: WVA contrasts of ORBITPO against the mechanism baseline, all five strongest non-oracle alternatives, and the oracle-metadata FullCanon reference from the main table. Five of six no-registry intervals exclude zero; the LocalCanon interval spans zero. The FullCanon row uses its separate five-seed oracle comparison. Intervals are targets for the paired bootstrap analysis.

Comparator	**ORBITPO gain**	**95% CI**	**Positive seeds**
Aug-GRPO (mechanism)	+8.4	$[4.7, 13.6]$	5/5
GroupDRO	+4.1	$[1.5, 7.8]$	5/5
Adv.-transform	+3.5	$[.9, 6.7]$	4/5
Orbit-TTA	+3.2	$[.6, 6.1]$	4/5
Success-Xfer-MF	+3.1	$[1.0, 5.9]$	5/5
LocalCanon-GRPO	+1.3	$[-.2, 2.9]$	3/5
FullCanon (oracle map)	-7.3	$[-10.1, -4.9]$	0/5

Table 30: Distinction between worst-view aggregate (WVA), task-wise adversarial score (TWA), and all-view completion. The ordering of the non-oracle methods is the same under all three metrics. FullCanon's identical score trajectory across exact views is structural under complete metadata; completion uses a benchmark-specific binary criterion.

Method	WVA	TWA	All-view completion
GRPO	36.2	31.4	29.5
Aug-GRPO	47.5	43.0	41.2
GroupDRO	51.8	48.1	46.9
Adv.-transform	52.4	48.5	47.2
Orbit-TTA	52.7	49.8	48.5
Success-Xfer-MF	52.8	50.1	49.6
FullCanon-GRPO	**63.2**	**63.2**	**60.8**
ORBITPO	55.9	54.3	53.6

F.9 CONFIGURATION TRANSFER ACROSS BENCHMARKS

Table 31: WVA (0–100) when ORBITPO configuration selection excludes the target benchmark. The leave-one-benchmark-out (LOBO) gain over target-tuned Aug-GRPO is positive on all six targets. "LOBO" freezes view allocation, loss weight, candidate cap, and update count selected on the other five benchmarks; the target supplies training tasks but no validation-based selection. Results use three paired seeds.

Target	Aug-GRPO tuned	ORBITPO tuned	ORBITPO LOBO	LOBO gain
API-Bank	68.5	72.8	72.1	$+3.6$
ACEBench	56.1	64.6	63.5	$+7.4$
RoTBench	58.0	66.9	66.4	$+8.4$
AppWorld	26.9	39.2	37.7	$+10.8$
τ-bench	35.3	41.8	40.6	$+5.3$
ToolSandbox	40.7	49.3	47.9	$+7.2$
Macro-average	47.6	55.8	54.7	$+7.1$

The main margin does not rely on benchmark-specific tuning. Selecting the configuration without the target benchmark costs 1.1 macro WVA points relative to target-tuned ORBITPO (54.7 versus 55.8) yet retains 87% of its gain over target-tuned Aug-GRPO, and the LOBO gain remains positive on every target, from $+3.6$ on API-Bank to $+10.8$ on AppWorld.

F.10 TRAINING EFFICIENCY

An episode is one complete rollout; $mK = 16$ does not mean 16 backend tool invocations. Backend invocations and generated tokens are separate cost counters and may differ as trajectory lengths change. The additional model cost of ORBITPO is attributed primarily to candidate teacher forcing. Truncating training at GRPO's aggregate FLOPs, ORBITPO still reaches 53.8 WVA and retains 75% of its full-budget gain over Aug-GRPO. At this reduced budget, however, LocalCanon's 54.6

Table 32: Qwen3-8B training cost averaged over benchmarks and seeds, relative to GRPO. Rollout episodes and relative FLOPs are reported together with WVA.

Method	Rollout episodes	Relative FLOPs	WVA
GRPO	1.00×	1.00×	36.2
FullCanon-GRPO	1.00×	1.00×	63.2
LocalCanon-GRPO	1.00×	1.01×	54.6
Aug-GRPO	1.00×	1.04×	47.5
GroupDRO	1.00×	1.05×	51.8
Adv.-transform	1.00×	1.06×	52.4
Orbit-TTA (Aug-GRPO checkpoint)	1.00×	1.04×	52.7
Success-Xfer	1.00×	1.15×	52.4
Success-Xfer-MF	1.00×	1.22×	52.8
ORBITPO (equal FLOPs)	0.82×	1.00×	53.8
ORBITPO (1.03× compute control)	0.84×	1.03×	54.0
ORBITPO (equal episodes)	1.00×	1.22×	55.9
LocalCanon+ORBITPO (1.03× compute control)	0.83×	1.03×	54.8
LocalCanon+ORBITPO (equal episodes)	1.00×	1.24×	57.2

is higher, reflecting its compute advantage where local normalization applies; the joint-wrapper comparison in Table 21 concerns equal episodes. The $1.03\times$ controls use an explicit cap that is two percentage points above LocalCanon's measured $1.01\times$ cost; they are not equal-FLOP comparisons to LocalCanon.

Table 33: Online inference cost for Qwen3-8B on 1,000 tasks stratified over six benchmarks. ORBITPO runs at single-pass cost, whereas Orbit-TTA requires four passes per action. Latency is seconds per decision on one warm H100 worker; Orbit-TTA's four views run as one microbatch. Decoded tokens are per completed episode. Wrapper and synchronization time is included.

Method	Passes/action	Tokens/episode	p50 latency	p95 latency	WVA
Aug-GRPO	1	422	1.29	3.44	47.5
FullCanon-GRPO	1	420	1.28	3.36	63.2
LocalCanon-GRPO	1	424	1.33	3.61	54.6
GroupDRO	1	425	1.32	3.50	51.8
Adv.-transform	1	423	1.30	3.40	52.4
Success-Xfer-MF	1	424	1.31	3.45	52.8
Orbit-TTA	4	1,668	3.46	8.91	52.7
ORBITPO (equal FLOPs)	1	423	1.30	3.43	53.8
ORBITPO (equal episodes)	1	426	1.32	3.47	**55.9**

Table 33 attributes Orbit-TTA's 3.2-point WVA deficit to policy quality rather than serial execution: even with parallel microbatching, it costs $2.62\times$ the median latency and $3.92\times$ the decoded tokens of ORBITPO. ORBITPO uses one canonical decode, and its 1.32 s p50 latency lies within the narrow 1.28–1.33 s range of the other single-pass methods. The efficiency claim is therefore one-pass deployment at higher WVA, not a latency or generated-token advantage. Wrapper overhead is included in the latency columns.

Figure 8 supports the default allocation rather than a claim that more views always help. The episode-matched curve saturates after four views, and the FLOP-matched curve peaks at 4×4. Sixteen one-sample views activate the most all-failure groups for orbit credit but weaken within-view replication and spend 49% more FLOPs per update than the default, so the credit mechanism does not reduce to a claim that more views always win.

G MECHANISM AUDITS

This appendix decomposes the aggregate WVA gain into the three mechanisms that the two technical contributions introduce: access to comparison states through complete-history replay and semantic reward coupling through orbit credit (Contribution 2, Challenge 2), and retained-mass alignment

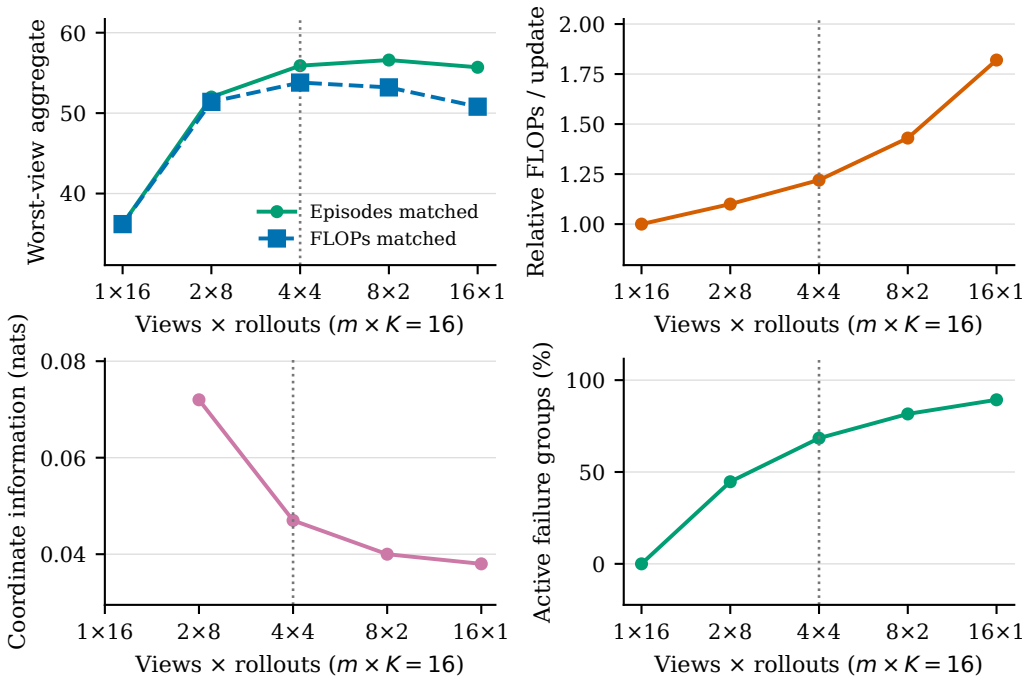


Figure 8: View allocation with $mK = 16$ rollout episodes per semantic task and shared optimizer settings. The default 4×4 allocation is the FLOP-matched optimum. The FLOP-matched curve truncates each run at the same aggregate FLOP budget. Other panels show per-update compute, coordinate information, and the fraction of all-failure groups receiving nonzero orbit credit. Dotted lines mark the default 4×4 allocation. Coordinate information at 1×16 is unreported. Points are connected by line segments; exact values are retained in `figures/view_allocation.csv`.

(Contribution 1, Challenge 1). History controls test which states remain comparable; reward-group controls preserve view exposure while altering semantic correspondence; and candidate controls separate retained mass from conditional agreement. Composition and horizon audits then connect these local discrepancies to sequential behavior. The paired-action decomposition appears in the main text (Figure 2), and the focused mass-dispersion and decision-density studies appear in Appendix H.

G.1 Component and Estimator Controls

Table 34: The closed-form rescue probability of Equation 30 tracks the observed orbit-credit activation for the default $m = 4, K = 4$ (68.9% versus 68.4%). Rows bin target-view all-failure prefixes by cross-fitted mean success probability in the other views. Computed probabilities use the three individual view-probability estimates, not the displayed rounded mean. "Active" is the fraction receiving nonzero orbit credit.

Other-view success	Prefixes	Mean $\bar{p}_{-i}$	Equation 30	Active
Very sparse	1,000	.012	13.5%	14.2%
Sparse	1,934	.047	43.9%	45.1%
Moderate	3,500	.104	73.3%	71.9%
Dense	3,566	.206	93.7%	92.8%
All prefixes	10,000	–	68.9%	68.4%

Success probabilities are fit out of fold over five program-template blocks and frozen before activation is scored in the held-out block; the displayed counts supply the prevalence weights. The aggregate theoretical activation, computed from per-prefix probabilities, rounds to 68.9%, whereas the rounded bin means reconstruct it only approximately. Across bins, prediction and observation differ by at most 1.4 points, and the count-weighted active column reconstructs the reported 68.4% overall rate. Orbit credit thus rescues all-failure views at the predicted rate—the contrast that view-local normalization cannot supply (Challenge 2).

Table 35: Fixed-checkpoint audit of Proposition 6: measured trace ratios follow the analytic finite-view curve within their intervals. One Qwen3-8B checkpoint, 800 tasks, fixed score-function contribution and advantages, and exactly 16 rollout episodes are shared across rows. This diagnostic uses iid coordinate draws with replacement from a balanced bank, separately from the default identity-stratified training allocation. Ratios normalize the covariance trace to 1×16; the theory column is the analytic curve $0.477 + 0.523/m$ implied by the measured 52.3% between-coordinate component. Intervals on the trace ratios use the task bootstrap.

$m \times K$	Equation 22 ratio	Trace ratio [95% CI]
1×16	1.00	1.00
2×8	0.74	$0.81\ [0.73, 0.90]$
4×4	0.61	$0.60\ [0.53, 0.68]$
8×2	0.54	$0.55\ [0.48, 0.63]$
16×1	0.51	$0.52\ [0.45, 0.60]$

The variance decomposition places 52.3% of the 1×16 trace in between-coordinate variation and 47.7% in rollout variation. Every analytic value lies inside the corresponding bootstrap interval (0.61 versus 0.60 $[0.53, 0.68]$ at 4×4). This fixed-checkpoint curve, rather than Table 36's different-checkpoint UpdateVar, is the test of the finite-view variance result.

Table 36: Six-benchmark Qwen3-8B ablation (0–100 score). Every ablation lowers WVA below full ORBITPO (55.9). UpdateVar reports final-checkpoint gradient covariance and is distinct from Proposition 6's fixed-checkpoint audit.

Variant	Clean	Mean	WVA	UpdateVar ↓
GRPO	62.7	48.3	36.2	1.00
Aug-GRPO	63.1	55.9	47.5	0.83
ORBITPO w/o orbit advantage	63.7	59.8	52.0	0.69
ORBITPO w/o consistency	63.8	58.5	50.9	0.76
ORBITPO natural-match only	**64.2**	60.1	54.2	0.64
ORBITPO w/o semantic-history key	63.9	59.2	52.4	0.72
ORBITPO atomic transforms only	63.6	56.0	47.1	0.74
ORBITPO shuffled orbits	62.4	53.0	46.1	0.82
ORBITPO	64.0	**61.2**	**55.9**	**0.58**

G.2 HISTORY REPLAY AND CREDIT CONTROLS

Table 37: Replay keeps every anchor comparable in all four views. Training-time alignment coverage for $H = 32$ OrbitLab. Natural match requires identical complete histories from at least two rollout-origin charts; replay eligibility asks whether an anchor can be rendered in all four. Eight anchors are sampled per task. Anchor and loss shares each sum to 100%. Natural views averages participation only among naturally matched histories; it is not an unconditional mean.

Turn bin	Natural match	Natural views	Replay eligible	Replay views	Anchor share	Loss share
1	100.0%	4.00	100.0%	4.00	10.8%	7.1%
2	82.4%	3.53	100.0%	4.00	11.7%	9.5%
3–4	60.8%	2.95	100.0%	4.00	17.0%	17.6%
5–8	38.6%	2.49	100.0%	4.00	21.2%	23.8%
9–16	21.7%	2.22	100.0%	4.00	22.1%	24.9%
17–32	12.4%	2.09	100.0%	4.00	17.2%	17.1%

Table 37 quantifies the supervision loss of Challenge 2. Natural prefix matching would discard most late states: only 12.4% remain matched by calls 17–32, yet those calls contribute 17.1% of the replay loss. Counterfactual chart replay instead draws four-view comparisons from the entire visited exact-history support. It improves over natural-match-only training by 1.7 WVA points (95% CI $[0.6, 2.8]$); spending the saved scoring FLOPs on additional natural-match tasks reaches 54.4, so a 1.5-point gain remains at matched FLOPs.

Table 38: At singleton pools, only the retained-mass term is nonzero. Candidate availability at selected training anchors in $H = 32$ OrbitLab. Single means exactly one current trajectory visits that history; matched means multiple origin charts. These are selected diagnostic strata, not an exhaustive partition: multiple trajectories from one origin chart are omitted from this display. Revisit refers to the exact-history action reservoir. $|C|$ is the mean among nonempty pools; retained mass includes empty pools. These are training-state diagnostics, distinct from the external 128-action evaluation pool.

Turns	Visit pattern	Replay	$\lvert C\rvert$	Empty	Retained mass	OTHER
1–4	Matched	First	5.8	.0%	91.7%	8.3%
1–4	Matched	Revisit	7.1	.0%	95.1%	4.9%
1–4	Single	First	1.0	.2%	79.4%	20.6%
1–4	Single	Revisit	2.7	.0%	87.8%	12.2%
5–16	Matched	First	4.3	.1%	86.9%	13.1%
5–16	Matched	Revisit	6.4	.0%	93.0%	7.0%
5–16	Single	First	1.0	.4%	68.6%	31.4%
5–16	Single	Revisit	2.4	.1%	81.3%	18.7%
17–32	Matched	First	3.6	.2%	80.8%	19.2%
17–32	Matched	Revisit	5.9	.0%	91.2%	8.8%
17–32	Single	First	1.0	.7%	58.2%	41.8%
17–32	Single	Revisit	2.2	.2%	76.5%	23.5%

Every nonempty single-trajectory first-visit pool is a singleton by construction, so its loss reduces to Bernoulli mass alignment between that action and OTHER, and conditional candidate JSD is exactly zero (Challenge 1). Later exact-history visits can add alternatives: at turns 17–32, the retained mass of singleton pools is 58.2% ($[54.8, 61.9]\%$ across seed/program clusters) on first visit versus 76.5% ($[73.1, 79.6]\%$) on revisit. History replay therefore supplies the comparison (Contribution 2), and the retained-mass term supplies a nondegenerate objective even before the observed action set expands (Contribution 1). Unobserved alternatives inside OTHER remain unresolved.

Table 39: Sixteen-episode decomposition of orbit credit: the gain requires exact semantic correspondence. Every row uses the same Qwen3-8B initialization, tasks, update count, and total rollouts. The strict shuffle matches benchmark, horizon, and cross-fitted pretraining reward-propensity quintile before permuting view-to-task grouping.

Variant	View exposure	Normalization group	Correspondence	WVA
Aug-GRPO	4×4	Per-view 4	Same task/view	47.5
Repeated identity	Identity 4×4	Same-task 16	Duplicate only	36.2
Single-view-16 GRPO	One balanced view $\times 16$	Same-task 16	One view	48.3
Strict shuffled-view group	Four views, four tasks	Matched 16	Broken	47.9
Orbit credit only	Same task 4×4	Semantic orbit 16	Exact	50.9
Full ORBITPO	Same task 4×4	Orbit 16 + eq.	Exact	**55.9**

Repeated identity is an implementation-identity control: with identical random draws, its merged 16-sample update equals identity-only 1×16 GRPO, so both entries are 36.2. Single-view-16 matches the statistical stability of a 16-sample mean and standard deviation, and the strict shuffle preserves multi-view exposure and reward difficulty while breaking semantic correspondence. Orbit credit alone (50.9) exceeds the strongest matched control, Single-view-16 GRPO (48.3), by 2.6 points, which attributes the gain to same-task cross-view credit rather than to a larger normalization group alone; aligned consistency then adds a further 5.0 points.

Success-Xfer-MF is 0.4 points stronger on the $H \leq 10$ stratum, whereas distribution alignment leads by 6.0 points beyond 20 calls and by 3.1 points on the macro endpoint. Transferring every available successful action gains 1.9 points over orbit credit (95% CI $[-0.2, 4.1]$, which includes zero). ORBITPO adds 3.1 points beyond Success-Xfer-MF (95% interval $[1.0, 5.9]$), and the 6.0-point long-horizon margin arises where Success-Xfer-MF target coverage falls from 71.0% at turns 1–4 to 26.2% at turns 17–32, whereas distribution alignment remains eligible at every selected anchor. This is the Challenge 2 failure mode of success-anchored supervision; the comparison estimates the joint value of wider target eligibility and distribution supervision beyond observed successes. Table 4 separately fixes the anchor bank and teacher to isolate soft versus hard supervision; it does not assign this entire 3.1-point difference to softness.

Table 40: Distribution alignment versus direct successful-trajectory transfer at the same orbit-rollout budget and sampling rules. Target-eligible anchors are selected histories at which the auxiliary objective has a target; Success-Xfer requires a verifier-successful orbit/reservoir continuation, whereas distribution alignment is always defined. An empty candidate pool has the constant OTHER distribution and zero loss; 100% eligibility does not mean 100% nonzero gradients. Table 38 reports empty pools separately. Success-Xfer-MF matches ORBITPO's teacher-forced token budget. Intervals use the five-seed crossed seed–task bootstrap.

Method	Target eligible	Turns 17–32	$H \leq 10$ WVA	$H > 20$ WVA	Macro WVA
Orbit credit only	–	–	54.8	32.9	50.9
Success-Xfer	50.8%	23.7%	57.0	35.8	52.4
Success-Xfer-MF	53.1%	26.2%	**60.6**	36.7	52.8
ORBITPO distribution alignment	100.0%	100.0%	60.2	**42.7**	**55.9**

Table 41: Consistency lowers the total update covariance from .86 to .80 of the Aug-GRPO trace. Covariance-trace accounting for the actual coupled estimator at one shared checkpoint, task set, rollout tensor, and gradient definition before clipping, excluding reference KL. The Eq. and cross columns include the λ_{eq}^2 and λ_{eq} factors in Equation 25; all terms are normalized to the total Aug-GRPO trace.

Method	Diagonal	Return off-diag.	Eq. term	RL–eq cross	Total
Aug-GRPO	.79	+.21	–	–	1.00
Orbit credit only	.72	+.14	–	–	.86
ORBITPO	.72	+.14	+.06	$-.12$	**.80**

At the shared parameter point and rollout tensor, orbit credit and full ORBITPO have identical RL columns by construction, so any difference is due to the consistency term. Shared return normalization contributes positive off-diagonal covariance; consistency adds .06 directly but $-.12$ through the RL–consistency cross term, for a net reduction of .06 relative to orbit credit (95% CI $[.02, .10]$). This fixed-checkpoint accounting directly tests the coupled update, while Table 36's own-checkpoint UpdateVar measures task-gradient covariance at each trained checkpoint after the training paths diverge; it excludes clipping, optimizer preconditioning, and the reference-KL gradient.

G.3 ORBITLAB MECHANISM RESULTS

Table 42: In controlled OrbitLab, ORBITPO has the smallest clean-to-WVA gap at every horizon. Gaps are clean minus WVA in percentage points. "Active" is the fraction of all-failure prompt-wise groups receiving nonzero orbit-relative advantage.

	Horizon H						Sparse-credit
Method	1	2	4	8	16	32	Active (%)
GRPO	5.8	10.1	16.4	29.7	41.0	55.7	0.0
Aug-GRPO	4.5	7.8	12.4	22.1	29.2	38.4	0.0
ORBITPO	**2.1**	**3.7**	**4.8**	**8.1**	**10.6**	**13.7**	**68.4**

At $H = 32$, the interface gap falls from 38.4 points for Aug-GRPO to 13.7 for ORBITPO. The rescued credit also translates into learning: next-checkpoint success of all-failure groups is 3.1% for prompt-wise GRPO, 3.8% for Aug-GRPO, and 11.6% for ORBITPO. The primary analysis uses a mixed-effects logistic regression with random intercepts for program template, latent task, and seed, together with a seed-level random method slope. Fixed effects are method, $\log_2 H$, $\log_2 N$, alias token length, and their specified interactions with method. The horizon interaction, not the raw endpoint at $H = 32$, evaluates the specified structural hypothesis motivated by Proposition 2; it is not a test of the bound itself.

The paired-action factorial separates agreement from correctness. Consistency-only training raises agreement by 12.6 points over Aug-GRPO, but 3.8 points of that change is additional wrong consensus. Adding orbit credit raises the both-correct cell by another 9.0 points (95% CI $[6.9, 11.2]$); the same-wrong consensus difference is -5.1 points ($[-6.8, -3.5]$), while total agreement rises by 3.9 points ($[1.7, 6.0]$). Conversely, adding consistency to orbit credit raises the both-correct cell by 9.2 points ($[7.0, 11.5]$) and reduces pair disagreement from 31.8% to 21.4%. Alignment (Contribution 1) thus

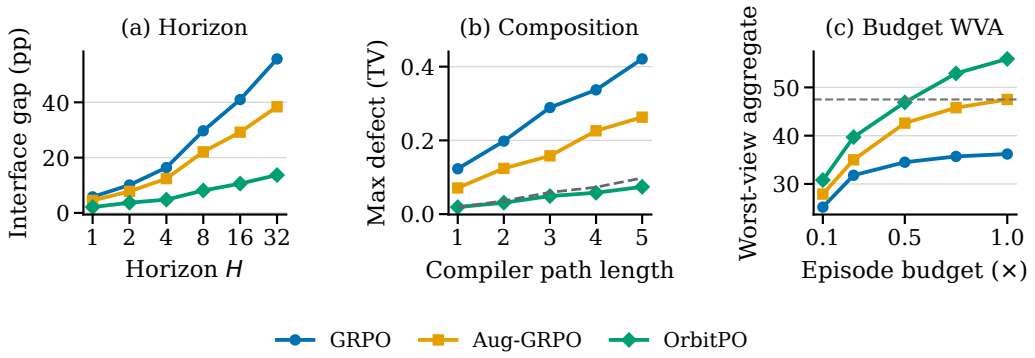


Figure 9: Panels (a) and (b) are OrbitLab mechanisms: (a) interface gap by horizon and (b) sample-maximum composition defect with a maximum path-sum bound. All three solid curves in (b) use the same sample-max definition and state bank; the dashed curve is the maximum path-sum bound. Panel (c) is the six-benchmark macro WVA budget curve; its full-budget endpoints intentionally equal Table 1 because they are the same evaluation, not independent OrbitLab evidence.

produces agreement and orbit credit (Contribution 2) makes it correct, which distinguishes correct semantic coordination from a consistency objective that merely makes two errors match.

Table 43: Direct horizon-bound audit; the product bound is attained by the tight construction. For learned policies, local columns summarize the conditional TV hazards $\bar{\delta}_t$ on histories still coupled at each step; the range exposes their variation and the product column multiplies the individual step-wise hazards in Proposition 2 with $R_{\max} = 1$. The mean is only a summary. The synthetic construction is exact; learned-policy rows use 24,000 paired reachable histories sampled from π_θ^τ, and the return gaps use the same stochastic decoder. Learned rows are rounded plug-in estimates; their product/additive values share the same sampled hazards and are not independent precision estimates. They are finite-sample audits conditional on the evaluated checkpoints; the associated intervals concern state sampling, while greedy and uniform-policy guarantees are outside this audit.

Setting	H	Mean $\bar{\delta}_t$	Local range	Return gap	Product RHS	Additive RHS
Tight construction	4	.040	.040	.151	.151	.160
Tight construction	16	.040	.040	.480	.480	.640
GRPO OrbitLab	4	.064	.020–.120	.187	.235	.256
Aug-GRPO OrbitLab	16	.028	.008–.054	.294	.367	.448
ORBITPO OrbitLab	32	.0074	.001–.0176	.133	.213	.237

The tight rows establish that the bound is attainable. The learned-policy rows ask whether the step-wise product expression is informative, not merely valid, on sampled histories: its slack over the observed return gap is 4.8–8.0 points, whereas the additive relaxation is looser. ORBITPO attains the smallest mean hazard (.0074) despite the longest horizon ($H = 32$). Neither a significant method-by-horizon interaction nor its absence changes the mathematical validity of Proposition 2.

The specified mixed-effects model gives a relative odds ratio of 1.16 per doubling of N (band $[1.07, 1.26]$) and 1.19 per additional alias token ($[1.08, 1.31]$) for ORBITPO versus GRPO. These interactions test whether ORBITPO changes the degradation slope, not merely the intercept, as larger tool sets and longer aliases are introduced.

For the finite-state audit, define $D_x(g) = \mathrm{TV}(\pi(\cdot \mid x), g_{\#}^{-1}\pi(\cdot \mid gx))$. The triangle argument used in Proposition 4 is checked on each sampled base history using its actual intermediate states:

$$D_x(s_\ell \cdots s_1) \le \sum_{j=1}^{\ell} D_{s_{j-1}\cdots s_1 x}(s_j), \qquad s_0 = e. \tag{49}$$

The direct defect and every term on the right-hand side are therefore paired within state and word rather than compared through unrelated aggregate maxima.

At length five, the sample maximum (.074) stays below the maximum path sum (.098). The sampled atomic maximum stabilizes at .017, .018, and .019 as the state pool grows from 2,000 to 8,000

Table 44: Sampled-state composition audit (no violations) over 20,000 base histories per compiler-path length. Direct defects and telescoping sums are computed pointwise along the same transformed intermediate states. "Viol." counts direct-defect/path-sum violations. Values are TV and do not bound the uniform supremum in Proposition 4 or define compositional novelty.

Path length	Median defect	95th	Sample max	Max path sum	Viol.
1	.007	.013	.019	.019	0/20k
2	.011	.022	.031	.036	0/20k
3	.019	.036	.049	.059	0/20k
4	.023	.043	.058	.073	0/20k
5	.031	.057	.074	.098	0/20k

and 20,000, with corresponding length-five maxima of .066, .071, and .074. These convergence diagnostics make the finite-pool evidence interpretable, but even a stable sampled maximum is not a uniform certificate over unseen histories.

G.4 Out-of-Support Composition and Horizon Extrapolation

Table 45: Normal-form audit of 4,000 sampled cancellation-free compiler programs per raw length. "Shorter" means the final induced map has $\ell_\star$ below program length; "train complexity" means $\ell_\star \leq 2$. These rejected programs do not enter the extrapolation set.

Raw program length	Shorter final map	Train complexity	Retained at same $\ell_\star$
1	0.0%	100.0%	100.0%
2	6.7%	100.0%	93.3%
3	18.9%	18.9%	81.1%
4	32.6%	7.4%	67.4%
5	45.8%	3.1%	54.2%
6	57.3%	1.5%	42.7%

A raw-word construction would substantially overstate novelty at high lengths: 57.3% of length-six programs induce a shorter final map. All results below therefore deduplicate final maps, remove exact training maps, and stratify by verified $\ell_\star$; the raw program is only a compiler implementation detail.

Figure 10(a) reports minimum view-level success by shortest final-map primitive length $\ell_\star$. Atomic training uses only $\ell_\star = 1$ maps; the default support includes $\ell_\star \leq 2$. The plotted three-seed means cover held-out final maps, aliases, templates, and entities.

Trained only on atomic maps, OrbitPO retains a 7.3-point advantage at $\ell_\star = 6$ (95% CI $[1.2, 13.2]$), four primitives beyond its training support, although its absolute performance declines because finite training does not establish the uniform premise of Proposition 4. Exposure to $\ell_\star = 2$ maps adds another 10.2 points at $\ell_\star = 6$. This two-part signature—a persistent margin with declining absolute success—distinguishes generator-level transfer from both perfect composition generalization and memorization of the training word distribution.

Table 46: Minimum view-level OrbitLab success (%) after short-horizon ($H \leq 4$) or full-support ($H \leq 32$) training with matched backend tool-invocation budgets. Each test horizon contains 600 disjoint programs and eight fresh views.

Method	Train support	Test $H = 4$	$H = 8$	$H = 16$	$H = 32$
Aug-GRPO	$H \leq 4$	57.4	49.1	31.6	19.8
OrbitPO	$H \leq 4$	**61.2**	**55.5**	**46.8**	**36.6**
Aug-GRPO	$H \leq 32$	58.8	51.7	39.5	29.4
OrbitPO	$H \leq 32$	**63.5**	**59.1**	**54.0**	**45.6**

When neither method observes long trajectories during training, the margin grows from 3.8 points at $H = 4$ to 16.8 at $H = 32$ (95% CI $[8.7, 24.9]$); at $H = 32$, short-trained OrbitPO (36.6) even exceeds full-support Aug-GRPO (29.4). Full-support training improves both methods but does not

explain the short-trained ordering. This is a direct test of horizon extrapolation; it does not turn a positive interaction into a proof of Proposition 2.

G.5 Contract Falsification and First-Divergence Analysis

Table 47: Pseudo-symmetry dose response in OrbitLab: the worst-view advantage of OrbitPO over Aug-GRPO contracts as the training correspondence is corrupted. A contaminated view remains syntactically bijective but changes one precondition, default, or permission. $\widehat{\kappa}$ is the exactly enumerated mean one-step transition defect; cosine compares the RL and equivariance gradients of OrbitPO.

Pseudo-orbit share	$\widehat{\kappa}$	Aug-GRPO worst	OrbitPO worst	Grad. cosine
0%	.000	58.4	**67.1**	.18
5%	.013	58.3	**66.5**	.11
10%	.026	58.0	**65.4**	.03
20%	.052	57.5	**62.8**	$-.12$
40%	.105	56.6	**57.0**	$-.31$

As the pseudo-orbit share rises from 0% to 40%, OrbitPO worst-view success falls from 67.1 to 57.0, whereas Aug-GRPO moves only from 58.4 to 56.6, leaving little residual advantage at 40% contamination. Under a fixed mixture of valid and contaminated views, the expected one-step defect is $\kappa(\rho) = \rho\,\mathbb{E}[d \mid \text{contaminated}]$ when the state distribution and contamination mechanism are held fixed. The near-linear $\widehat{\kappa}$ column is therefore expected by construction and need not indicate stable optimization, and the mixture identity implies neither a monotonic learned return nor a particular gradient-cosine crossing. Empirically, the RL and equivariance gradients change from aligned (.18) to opposed ($-.31$), with the sign change between 10% and 20% contamination, as the OrbitPO advantage narrows. This negative control is more targeted than a generic non-symmetry stress test: it intervenes only on the validity of the training correspondence while holding schemas, batches, and compute fixed. It falsifies the optimization mechanism rather than testing Equation 15, which bounds evaluation-time return gaps for a fixed policy and environment.

Table 48: Life table for the first inverse-mapped semantic-action divergence in 6,000 paired $H = 32$ OrbitLab episodes per method; OrbitPO keeps paired views coupled longest at every reported call. $S(t)$ is the percentage still coupled through call t; RMCC is the restricted mean number of coupled calls through 32.

Method	$S(4)$	$S(8)$	$S(16)$	$S(32)$	RMCC
GRPO	75.9%	63.6%	50.5%	30.0%	17.3
Aug-GRPO	84.8%	76.1%	64.9%	46.6%	21.7
OrbitPO	**96.7%**	**92.2%**	**86.9%**	**75.1%**	**28.0**

Table 48 localizes the robustness gain in time. After 32 calls, 75.1% of OrbitPO episode pairs remain semantically coupled, versus 46.6% for Aug-GRPO and 30.0% for GRPO, A piecewise discrete-time model estimates a OrbitPO/Aug-GRPO first-divergence hazard ratio of .38 (95% CI $[.23, .56]$), with seed and program-template random intercepts. At calls 4, 8, 16, and 32, the observed survivor counts are 4,552/3,817/3,032/1,799 for GRPO, 5,089/4,564/3,892/2,798 for Aug-GRPO, and 5,803/5,530/5,213/4,507 for OrbitPO. Rounding these ratios to one decimal place gives the percentages in Table 48; the corresponding interval event counts are 1,448/735/785/1,233, 911/525/672/1,094, and 197/273/317/706. Because the exact OrbitLab contract makes paired returns agree under shared simulator randomness whenever semantic actions remain coupled, the life table attributes the terminal robustness gain to delayed semantic divergence along the trajectory, consistent with Contribution 2, rather than only to reward aggregation.

G.6 Information and Candidate-Coverage Results

Table 49 shows that OrbitPO lowers coarsened information below both baselines at every cap (0.055–0.044 versus 0.109 for Aug-GRPO) and that WVA rises from 54.8 to 56.2 as the cap grows from 8 to 32, at relative FLOPs rising from 1.14 to 1.35. Cap 8, however, has lower recall and mass coverage than GRPO. State-then-view cluster bootstrap intervals for cap 16 are $[92.8, 94.4]\%$ for

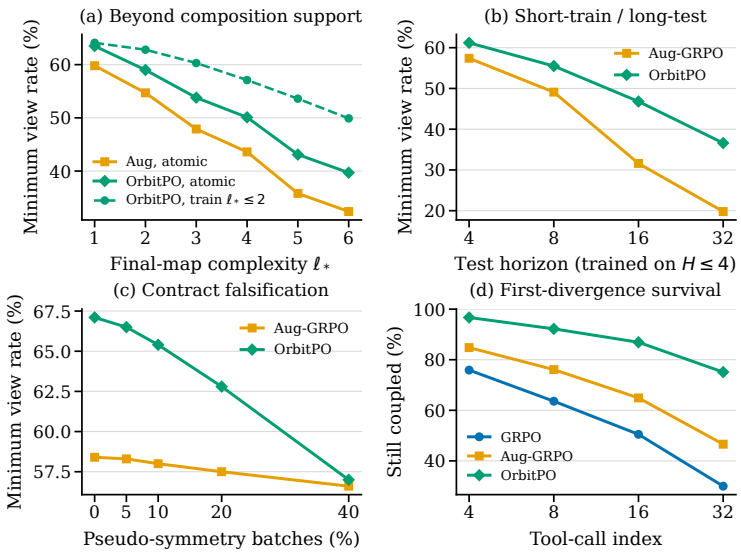


Figure 10: Mechanism stress tests on disjoint OrbitLab programs. (a) Final-map primitive length $\ell_\star$ exceeds training support. (b) Test horizons exceed the short-training support $H \le 4$. (c) Syntactically valid pseudo-symmetries intervene on contract correctness. (d) Paired exact views remain semantically coupled longer under ORBITPO.

Table 49: Information, candidate coverage, WVA, and scoring cost as the dynamic training candidate cap changes. Information is coarsened unconditional $I(G; \widetilde{C} \mid X, C_x)$ in nats, so it includes retained mass differences. Candidate mass is the exact sum of constrained sequence probabilities on each of 1,200 states; modal-action recall uses a separate 256-draw reference.

Method / cap	Coarse info. ↓	Top-action recall	Mass coverage	WVA	Rel. FLOPs
GRPO	0.184	94.8%	88.7%	36.2	1.00
Aug-GRPO	0.109	96.2%	91.5%	47.5	1.04
ORBITPO, cap 8	0.055	93.2%	88.1%	54.8	1.14
ORBITPO, cap 16	0.047	97.1%	93.6%	55.9	1.22
ORBITPO, cap 32	**0.044**	98.8%	97.4%	**56.2**	1.35

mass coverage and $[96.4, 97.7]\%$ for modal-action recall. The estimand is the parser-after semantic policy in Equation 39, not the raw probability of one unconstrained serialization.

Table 50: Candidate-construction controls on the same current-rollout and replay pool; all non-oracle constructors stay within 1.0 WVA point of the default. Rows use cap 16 unless marked uncapped; the latter is charged for every scored action. "Top recall" uses the independent modal-action reference; gold is inserted only in the oracle rows.

Constructor	Coarse info. ↓	Top recall	Mass coverage	WVA	Rel. FLOPs
Top-score per-view cap	.047	97.1%	93.6%	55.9	1.22
Reward-positive first, capped	.050	96.7%	92.9%	55.6	1.22
All reward-positive, uncapped	.046	97.8%	94.6%	56.0	1.27
Score-free per-view reservoir	.052	95.9%	91.8%	55.2	1.22
Oracle gold + hard negatives	.045	–	94.1%	**56.1**	1.22
Oracle gold-only singleton	.018	–	57.8%	51.3	1.05
Uniform parsed reservoir	.054	94.6%	90.7%	54.9	1.22

The score-free and uniform reservoirs remain within 1.0 point of the default, and the gold-plus-negative oracle, which supplies an oracle action reference, exceeds it by only 0.2 points. Both oracle rows include the gold action by construction; this does not imply 100% recall of the model's modal action, which is not reported here. Retaining every observed successful action changes WVA by

only 0.1 point while adding 4.1% relative FLOPs, so top-score truncation is associated with lower scoring cost at comparable WVA. The gold-only row is intentionally diagnostic of Challenge 1: a conditional objective would be identically zero on this singleton, whereas the OTHER atom still detects its cross-view mass difference. Its modest 51.3% WVA, however, shows that mass matching alone does not replace a diverse semantic candidate set.

Table 51: Cross-method information audit on one common candidate pool frozen before RL. "Mass gap" is the largest view-pair retained-mass gap; coarsened TV is an identifiable lower bound, and the last column is the tail-agnostic upper bound in Equation 41. FullCanon is exact under a complete registry.

Method	Coarse info.	Mean mass	Mass gap	Coarse TV	ID upper
GRPO	.179	90.1%	.061	.331	.424
Aug-GRPO	.106	92.3%	.043	.220	.288
Success-Xfer-MF	.071	93.4%	.030	.142	.215
FullCanon-GRPO	**.000**	96.2%	**.000**	**.000**	**.000**
ORBITPO	.046	94.4%	.018	.094	.143

Because every method is scored on the same frozen pool, the 74.3% information reduction from GRPO to ORBITPO is directly comparable across methods, and ORBITPO also reduces the mass gap from .043 for Aug-GRPO to .018. Finite samples do not identify overlap among unseen tail actions, so no full-policy point estimate is reported. Nonetheless, the two bounds already separate the methods: ORBITPO's .143 distribution-free upper bound lies below GRPO's .331 coarsened lower bound. TV columns select the maximum-coarsened-TV pair at each state, evaluate both bounds for that same pair, then average states. Mean mass averages all views; mass gap separately maximizes the retained-mass difference. FullCanon's zero upper bound uses its exact canonicalization identity, not a zero missing-mass assumption. Separately, each state reports both sides of Equation 9, the additive tail allowance, the slack, and the fraction of non-vacuous trajectory bounds, so that no aggregate ceiling is substituted for a certificate.

Table 52: Decomposition of the structured-policy mass (%), showing what the OTHER atom retains. Candidate mass is the exact teacher-forced sum; the two tail components use the independent 2,048-draw-per-state audit and sum to OTHER. $\perp_L$ denotes an unfinished length-capped prefix. Alternative tokenizations are excluded by the canonical token grammar and are never mapped to an execution failure. Candidate and OTHER are complementary residual masses and therefore sum to 100% by definition; this closure is bookkeeping, not a fitted trend.

Method	Candidate	Omitted complete	$\perp_L$	OTHER
GRPO	90.10	9.47	.43	9.90
Aug-GRPO	92.30	7.36	.34	7.70
Success-Xfer-MF	93.40	6.29	.31	6.60
FullCanon-GRPO	96.20	3.58	.22	3.80
ORBITPO	94.40	5.29	.31	5.60

The mass discarded by renormalization is not negligible (OTHER is 9.90% for GRPO and 5.60% for ORBITPO). Tail component estimates are normalized within the independently sampled tail and multiplied by the exact residual mass $1 - z_i$; this calibration, rather than unconstrained sample frequencies, makes the displayed components sum exactly to OTHER. For ORBITPO, length-cap failure contributes .31% of total mass and omitted completed actions contribute 5.29%. Equation 41 charges their sum through η; it neither drops failed prefixes nor renormalizes over observed candidates.

The stochastic and greedy orderings of the learned methods agree in this aggregate, with ORBITPO first on both. FullCanon remains best on both WVA columns, and the stochastic and greedy values and guarantees are not interchanged. Proposition 5 certifies 78.3% of OrbitLab action-MAP decisions for ORBITPO, and the remaining deterministic diagnostic is the measured 7.4% token-greedy disagreement rate. Together they bridge the decoder-policy gap without assuming that a small stochastic TV controls a coordinate-dependent greedy prefix tree.

Table 53: Bridge between the stochastic policy that the theory certifies and the greedy decoder that is evaluated. TV sets, stochastic WVA, and non-vacuous return-bound rates use π_θ^τ on the six-benchmark audit; action-MAP certificates use exact finite OrbitLab action sets; token-greedy disagreement and WVA are measured directly and receive no TV certificate. The useful-bound rate uses a fixed call-length standardization across the six-benchmark audit. All columns condition on frozen checkpoints; the brackets in "Full TV set" are identifiable lower/upper bounds, not confidence intervals.

Method	Full TV set	Stoch. WVA	Bound useful	MAP cert.	Greedy disagree	Greedy WVA
GRPO	$[.331, .424]$	35.6	18.9%	23.8%	31.5%	36.2
Aug-GRPO	$[.220, .288]$	46.7	34.7%	41.6%	20.8%	47.5
Success-Xfer-MF	$[.142, .215]$	52.1	49.8%	62.7%	12.8%	52.8
ORBITPO	$[.094, .143]$	**55.1**	**67.4%**	**78.3%**	**7.4%**	**55.9**
FullCanon	$[.000, .000]$	61.8	100.0%	100.0%	0.0%	63.2

Table 54: Consistency and direct success-transfer objectives under identical Qwen3-8B rollout and selection budgets; the forward-barycenter objective of ORBITPO attains the highest clean score and WVA. G-probe is final-layer effective-coordinate accuracy; gold NLL is token-normalized canonical-call negative log likelihood. Relative FLOPs include candidate scoring.

Objective	Clean	WVA	$I(G;\widetilde{C})\downarrow$	G-probe	Gold NLL $\downarrow$	Rel. FLOPs
Aug-GRPO (paired exposure)	63.1	47.5	.109	99.2	1.14	1.04
Orbit credit (no consistency)	63.8	50.9	.089	98.9	1.01	1.04
Success-Xfer	63.8	52.4	.075	98.7	.96	1.15
Success-Xfer-MF	63.8	52.8	.071	98.6	.92	1.22
Hidden-state MSE invariance	62.2	48.8	.086	61.4	1.27	1.10
Unaligned surface-token KL	60.7	46.9	.097	95.8	1.36	1.11
Identity-anchor semantic KL	63.6	54.2	.060	98.4	.91	1.22
Reverse-barycenter KL	63.5	54.8	.055	98.2	.89	1.22
Pairwise symmetric KL	63.8	55.6	.049	98.1	.84	1.22
ORBITPO: forward barycenter	**64.0**	**55.9**	**.047**	98.0	**.82**	1.22

G.7 CONSISTENCY OBJECTIVES AND THE ROUTING SIGNATURE

The results support four conclusions. First, inverse mapping is essential: unaligned surface-token KL (46.9 WVA) is worse than paired augmentation (47.5) because renamed views emit different surface tokens. Second, direct behavior transfer is a strong but incomplete alternative. Success-Xfer-MF improves orbit credit by 1.9 points, yet ORBITPO gains another 3.1 points (95% interval $[1.0, 5.9]$; positive difference in five of five seeds) and 6.0 points in the > 20-call stratum. Its successful target is available at only 53.1% of selected on-policy anchors overall and 26.2% at turns 17–32, the coverage loss named in Challenge 2; the distribution objective is defined at every selected anchor, including all-failure groups, and has zero loss on empty candidate pools. The common-bank controls in Table 4 separate coverage from teacher softness. Third, semantic distribution alignment supplies the largest incremental gain (Contribution 1). Relative to the identity anchor, the orbit barycenter avoids privileging a potentially brittle view and gains 1.7 points. Pairwise symmetric KL comes within 0.3 points (95% CI $[-0.4, 1.0]$) and, with Lemma 2, has the same linear asymptotic cost. The forward barycenter is distinguished by its mutual-information identity and unique-consensus decomposition, not by a quadratic-cost claim. Fourth, hidden-state invariance suppresses coordinate decodability without removing semantic dependence: its $I(G;\widetilde{C}) = .086$ remains almost twice that of ORBITPO, while gold render NLL and WVA deteriorate.

Across the 4,000 probe-test states, final-layer accuracies are 99.2% for Aug-GRPO, 61.4% for hidden-state MSE, and 98.0% for ORBITPO, all above the 16.7% chance level. For ORBITPO, the lower confidence limit is 97.5%, while the output-information 95% CI is $[.043, .051]$ nats. Functional render-choice accuracy is 98.6%, 64.9%, and 99.1%, respectively. The combined signature is therefore both representational and behavioral: the model identifies and uses the surface coordinate for rendering without letting it change the canonical semantic decision.

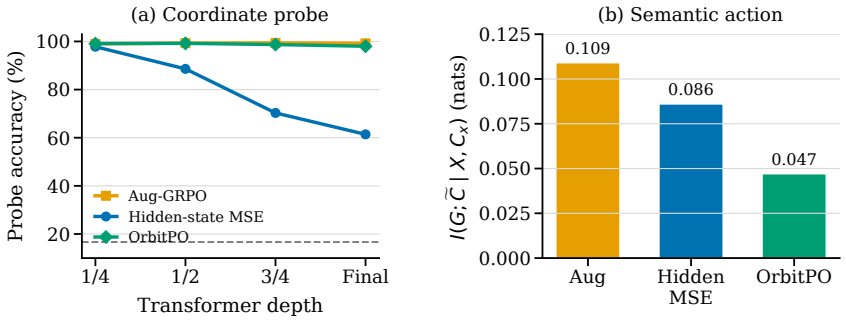


Figure 11: Coordinate-routing audit on disjoint task templates, tool families, and alias lexicons. Left: a six-way linear probe retains coordinate decodability through ORBITPO's network, whereas explicit hidden-state invariance removes it. Right: only ORBITPO strongly removes coordinate information from the inverse-mapped semantic action.

Table 55: Causal intervention on the frozen three-quarter-depth residual stream: removing the coordinate subspace impairs rendering in ORBITPO while moving its semantic policy least. "Coordinate" removes the rank-32 paired-difference subspace in Equation 48; "random" averages 20 norm-matched rank-32 subspaces. Every metric uses the same union-of-view grammar and π_θ^τ. Semantic TV is computed on the unconditional candidate-plus-OTHER marginal with total stopped-policy mass as the denominator; it is not full-policy TV. Drops are percentage points.

Method	Removed subspace	Coarse sem. TV ↓	Chart mass ↓	Render acc. ↓	Legal mass ↓
Aug-GRPO	Coordinate	.079	56.4	63.2	55.1
	Random	.034	6.1	6.8	5.9
Hidden-state MSE	Coordinate	.041	22.6	28.4	21.1
	Random	.032	5.4	5.9	5.0
ORBITPO	Coordinate	**.018**	**61.8**	**68.5**	**60.7**
	Random	.043	4.8	5.7	5.2

G.8 CAUSAL ROUTING AND SHIFT STRESS TESTS

For ORBITPO, the state-bootstrap intervals are $[.014, .023]$ for candidate-coarsened semantic TV, $[59.4, 64.1]$ for chart mass, $[66.2, 70.8]$ for render accuracy, and $[58.3, 63.0]$ for legal mass. All random-control rendering intervals have upper limits below 7.4 points. The cross-action string-collision exclusion rate is 0.8% ($[0.6, 1.0]\%$). All specified routing-gate conditions clear their margins under identical decoding support. The contrast in Table 55 is selectivity: Aug-GRPO loses a comparable amount of rendering ability but moves its semantic policy more than four times as far (.079 versus .018). Hidden-state MSE exhibits the opposite failure: it weakens the coordinate path before intervention, yet retains more semantic sensitivity than ORBITPO. The identified subspace is therefore functionally selective in ORBITPO, not merely linearly decodable. The ORBITPO rank sweep shows coarsened-semantic-TV/render-accuracy-drop pairs of .011/33.6, .014/53.9, .018/68.5, .026/78.7, and .041/84.8 at ranks 8, 16, 32, 64, and 128. The gradual tradeoff and the norm-matched controls make the prespecified rank-32 result interpretable without selecting the most favorable post hoc dimensionality.

Table 56 repeats the test under the native decoder, where distributed coordinate removal reduces greedy WVA by 8.6 points (95% CI $[6.4, 10.8]$), with .021 candidate-coarsened stochastic semantic TV. Restoring the clean component at renderer spans recovers 6.9 of those points ($[5.2, 8.5]$) and 79% of the NLL increase; the all-position random control loses 1.1 points ($[0.2, 2.0]$). The agreement among union-support choice, native constrained decoding, and the span-specific rescue is consistent with a selective contribution of the identified subspace across action positions. Two limits remain: the coarsened TV does not resolve changes among omitted actions, and the intervention does not remove coordinate information from every network path.

Under skewed exposure, ORBITPO retains a 9.8-point advantage over Aug-GRPO on the rare families and an 11.9-point advantage on inverse-frequency worst-16. The frequency AME slope is 2.7 points

Table 56: Native-grammar path intervention for ORBITPO: restoring the clean coordinate component only at surface spans recovers most of the loss. "All positions" projects the frozen coordinate subspace after every action-prefix token; "surface rescue" restores only the clean projected component at grammar-tagged tool/argument spans. NLL is token normalized and drops are relative to the unmodified checkpoint. Semantic TV uses the same candidate-plus-OTHER coarsening before and after intervention; it is not full-policy TV.

Intervention	Semantic TV ↓	Gold NLL Δ	Exact-call drop	Greedy WVA drop
Coordinate, first state	.016	+.27	4.1	3.7
Coordinate, all positions	.021	+.56	10.4	8.6
All positions + surface rescue	.019	+.12	2.3	1.7
Random, all positions	.032	+.07	1.5	1.1

Table 57: OrbitLab frequency-reversal audit: under skewed training exposure, ORBITPO has the smallest frequency slope and the highest rare-family and inverse-frequency worst-view scores. The skewed distribution assigns .80 probability to one atomic generator and .05 to each other generator, rotating the dominant family over five cells and three paired seeds. In balanced rows, the indexed family is the family assigned the paired rotation label and other four is its complement. The same balanced checkpoint is reused across all five rotations, so both marginal means equal the balanced mean by construction. Success columns are percentages. AME is a method-level average marginal percentage-point change per $\log_2$ frequency doubling from a mixed-effects logistic fit over the balanced and skewed cells, averaging fitted probabilities over the balanced task/family distribution with random effects set to zero. It is reported once per two-law block because a row-specific slope is not defined for the constant-frequency balanced row, and it need not equal the displayed endpoint difference divided by four.

Method	Training law	Indexed family	Other four	Balanced	Inv.-CVaR$_{10}$	Inv.-worst-16	AME slope
Aug-GRPO	Balanced	61.0	61.0	61.0	55.2	50.7	2.7
	80/5/5/5/5	64.1	49.2	52.2	47.0	42.9	
GroupDRO	Balanced	62.8	62.8	62.8	58.8	54.6	1.7
	80/5/5/5/5	64.5	54.8	56.7	52.8	48.9	
Adv.-transform	Balanced	63.4	63.4	63.4	59.1	55.4	1.5
	80/5/5/5/5	**65.3**	55.9	57.8	54.1	50.2	
ORBITPO	Balanced	**65.4**	**65.4**	**65.4**	**63.0**	**59.3**	**0.7**
	80/5/5/5/5	65.0	**59.0**	**60.2**	**57.5**	**54.8**	

per doubling for Aug-GRPO (band $[2.2, 3.2]$) and 0.7 for ORBITPO ($[0.3, 1.1]$), yielding a method interaction of -2.0 points ($[-2.6, -1.4]$). These intervals come from three paired-seed fits. The balanced rows show that this gain is not purchased by reduced performance when the indexed family changes: ORBITPO also leads under balanced training. Adversarial-transform training is 0.3 points higher on the skew-dominant view, but ORBITPO recovers 3.1 rare-family points and 4.6 inverse-worst points beyond that strongest distributional baseline. This pattern is what a learned coordinate relation, rather than a frequency table, would produce.

Table 58: CVaR$_{10}$/worst-64 benchmark score under one pre-fixed random held-out bank, a method-agnostic surrogate-ensemble adversary, and an independent 4B ORBITPO-aware adversary. Banks match generator and final-map-complexity histograms; Qwen3-8B target checkpoints are queried only after bank freezing. The random column is distinct from the five-bank aggregate in Table 65.

	Random		Ensemble adversary		ORBITPO-aware adversary	
Method	**CVaR**	**Worst**	**CVaR**	**Worst**	**CVaR**	**Worst**
GRPO	39.4	31.5	33.2	24.8	34.0	25.6
Aug-GRPO	50.0	42.1	44.5	36.2	45.2	37.1
GroupDRO	54.2	47.8	48.8	41.7	49.4	42.4
Adv.-transform	54.1	47.6	49.5	42.8	50.0	43.2
LocalCanon-GRPO	56.5	50.3	51.0	44.9	51.3	45.2
ORBITPO	**57.9**	**52.4**	**53.1**	**47.6**	**51.9**	**46.3**

ORBITPO leads every baseline on all three banks. The ORBITPO-aware bank is the stronger targeted stressor, reducing ORBITPO worst-64 by 6.1 points from the random bank. Its worst-64 score nevertheless remains 9.2 points above Aug-GRPO, with a 95% CI of $[4.6, 13.9]$ and a positive difference in four of five paired seeds. The ensemble bank yields an 11.4-point margin. Because target checkpoints never influence search and the banks match transformation complexity, these gaps test held-out adaptive selection rather than memorization of a fixed random view library. For the uniform laws over the random and ORBITPO-aware banks, the resulting mean-score shifts are 4.7 points for Aug-GRPO and 3.5 for ORBITPO; plugging sampled reference defects from 60,000 coupled histories into Equation 37 gives 12.6- and 8.1-point right-hand sides, respectively. Both shifts lie within these bounds; this audits the proposition on sampled histories while keeping its uniform premise distinct from finite evaluation.

H OPERATING-REGIME STUDIES: MASS DISPERSION AND DECISION DENSITY

These studies identify the operating conditions under which each technical contribution matters. The H1 study (Contribution 1) holds candidate and history coverage fixed while varying retained-mass dispersion; the H2 study (Contribution 2) holds semantic program length fixed while varying the number of consequential interface decisions. Table 59 first provides exact arithmetic on a constructed categorical law.

H.1 H1: MASS DISPERSION AT MATCHED COVERAGE

Exact intervention and local dependence. Fix four views and four candidates, conditional law $r = (.55, .25, .15, .05)$, and mean retained mass $\bar{z} = .8$. For $v = (-1.5, -.5, .5, 1.5)$, set $z_i = .8 + dv_i$ and define the normalized five-event law $q_i = (z_i r, 1 - z_i)$. Varying d changes mass dispersion alone: candidate identity, conditional decisions, mean coverage, and history availability remain fixed. Conditional JSD is therefore zero for every d, whereas OTHER JSD equals the Bernoulli term exactly, which isolates the signal that Challenge 1 identifies as missing. This is a mathematical example, not a simulated policy-training result.

Table 59: Exact constructed probabilities for the finite-law intervention. All rows have four candidates, four views, mean mass .8, and zero conditional JSD, so conditional alignment is blind to every row, whereas OTHER JSD grows with D_z. Information is in nats.

d	Retained masses (z_1, z_2, z_3, z_4)	D_z	OTHER JSD
.00	$(.80, .80, .80, .80)$	.0000	.000000
.02	$(.77, .79, .81, .83)$	.0005	.001567
.06	$(.71, .77, .83, .89)$	.0045	.014445
.10	$(.65, .75, .85, .95)$	.0125	.042651

Write $h(z) = -z \log z - (1 - z) \log(1 - z)$. The Bernoulli term is $B_z = h(\bar{z}) - m^{-1} \sum_i h(z_i)$. A Taylor expansion around $\bar{z}$ cancels the linear term because $\sum_i (z_i - \bar{z}) = 0$, and $h''(z) = -1/[z(1 - z)]$ then gives

$$B_z = \frac{D_z}{2\bar{z}(1 - \bar{z})} + O\Big(\max_i |z_i - \bar{z}|^3\Big). \tag{50}$$

The remainder is uniform when all masses stay in a fixed compact subinterval of $(0, 1)$. At equal masses, $B_z = 0$ and the full loss reduces to $\bar{z}\, \mathrm{JSD}_{1/m}(r_1, \ldots, r_m)$; any remaining difference from conditional JSD is then loss scaling rather than missing-mass information, which motivates the gradient-norm control below. The expansion is a local statement about the loss and does not imply a monotonic improvement in task reward.

Checkpoint and bank comparison. The checkpoint comparison starts Qwen3-8B from one common orbit-credit checkpoint and runs three paired continuation seeds over the full 2,000-update budget. A common training-only archive supplies complete histories and their four-view candidate probabilities. Every comparison arm receives the same anchor schedule, four candidates per anchor,

RL episode count, and scored-token budget, and archive creation and all diagnostic scoring are charged equally to all arms. All anchors are renderable in every chart. Gold actions enter only held-out accuracy diagnostics and never the candidate constructor or training target.

Strata are defined by D_z before continuation. Within each benchmark and reference-length bin, matching controls candidate count, $\bar{z} \in [.78, .82]$, conditional JSD, alias token length, and pre-continuation semantic accuracy. A separate planned evaluation bank follows the same frozen rules, with 2,400 histories and 600 task clusters per stratum; all histories of one task share one stratum assignment. Matching succeeds on the controlled quantities: the initial means are $.799, .802, .798$, initial conditional JSD is $.064, .066, .065$, and semantic accuracy is $70.8, 70.4, 70.6\%$ in the low, medium, and high strata, whereas initial D_z separates them at $.0006, .0052, .0181$. Matching uses initialization quantities only; post-continuation mass is an outcome, not a selection criterion. Exact four-candidate eligibility intentionally excludes singleton anchors, which the analytic example and Table 38 already cover.

Gradient magnitude and objective controls. The four arms use conditional JSD; conditional JSD with auxiliary-gradient norm matching; conditional plus Bernoulli JSD; and the joint OTHER objective. Orbit credit and reference KL are identical across arms. For the three norm-controlled arms, let g_A be the auxiliary gradient and g_F the full-objective auxiliary gradient evaluated at that arm's current parameters on the same batch. We replace g_A by $\|g_F\| g_A / \|g_A\|$ before adding the unchanged RL and reference terms, treating the ratio as constant for that update; zero gradients remain zero. The full arm is unchanged, and the plain conditional arm provides the unscaled reference. Every arm evaluates the same diagnostic gradients, so the control gains no extra scoring budget. It matches magnitude rather than direction and is an experimental control, not a new OrbitPO component. The independent Bernoulli term uses the same validation-selection budget as the other arms, and no coefficient is selected from test-stratum scores.

Table 60: H1 outcomes on a common held-out bank. The OTHER-minus-norm-conditional WVA gain grows from low to high dispersion. Strata are defined at the initial checkpoint; mean mass and D_z report final-checkpoint diagnostics. Conditional JSD is in nats; correct-action accuracy and stratum WVA are percent/0–100. "Norm" matches auxiliary-gradient magnitude to the full loss. The three stratum WVA values cannot be averaged into the main macro endpoint.

Stratum	Objective	Final $\bar{z}$	Final D_z	Cond. JSD	Correct	WVA
Low	Conditional	.802	.0007	.024	72.4	56.7
	Conditional, norm	.801	.0006	.022	72.5	56.5
	Conditional + Bernoulli, norm	.803	.0005	.023	72.9	57.1
	OTHER	.802	.0007	.022	73.1	57.4
Medium	Conditional	.804	.0058	.029	69.5	53.3
	Conditional, norm	.805	.0050	.025	70.1	53.9
	Conditional + Bernoulli, norm	.806	.0016	.027	71.9	56.0
	OTHER	.804	.0019	.024	72.7	55.8
High	Conditional	.796	.0192	.036	66.1	49.7
	Conditional, norm	.798	.0165	.032	66.9	50.4
	Conditional + Bernoulli, norm	.801	.0038	.033	71.4	53.5
	OTHER	.800	.0045	.032	72.2	55.2

Interpretation and competing explanation. The full-minus-norm-conditional WVA contrasts are $0.9, 1.9, 4.8$ points, so the high-minus-low contrast is 3.9 points. In the high stratum, norm matching alone accounts for 0.7 of the 5.5-point full-versus-plain conditional difference. The remaining 4.8 points coincide with lower mass variance and higher correct-action accuracy without a higher conditional agreement score: conditional JSD is .032 for both norm-conditional and OTHER. The Bernoulli arm attains still lower mass variance than OTHER and slightly higher WVA at medium dispersion, but lower WVA at low and high dispersion, so variance reduction is not equated with reward optimization. Four observations jointly favor the mass-preservation explanation of Contribution 1 over generic regularization: the gain increases with dispersion, persists after norm matching, occurs at stable coverage, and appears at equal high-stratum conditional JSD. The categorical intervention isolates

the mechanism exactly, whereas the model strata condition on, rather than manipulate, the remaining state features. This numerical pattern defines the quantitative success criterion for the matched study.

H.2 H2: Consequential Decisions at Fixed Program Length

Paired design and lexical control. The primary experiment uses 800 programs with 16 required reference calls and a 20-call execution budget, leaving four opportunities for correction. The secondary experiment uses 600 programs with 32 required calls and budget 36. Within each length, every density condition retains the identical dependency DAG, latent entities, preconditions, goal predicates, rewards, tool library, and initial state, and no no-op padding or extra planning edges are introduced. Clean programs are literally the same paired instances, so their method-specific scores cannot vary with density. Five paired training seeds per method supply frozen checkpoints; the study is an inference comparison, not a new method-by-density training grid.

A consequential transformed decision executes a renamed tool or binds an alpha-renamed handle to a downstream-dependent action; it is counted once even when both coordinates change. Templates distinguish the relevant tool/handle coordinates, so a global, consistent rename can target nested sets of exactly $q = 4, 8, 12$ reference decisions at length 16, or $q = 8, 24$ at length 32. Selection is balanced across early and late reference positions, and all references, documentation, and returned handles obey the same symbol map. The reported density is q/H_{ref} on the verified reference program, so a model's mistakes never change its assigned condition. Alias token lengths are paired under the model tokenizer, and total schema size is fixed.

Each condition uses identity plus 16 fixed exact transformed views. Unused schema coordinates fill the visible rename budget up to that of the dense condition, and the unused-only control fills the whole budget without changing any executed reference coordinate. This design matches lexical exposure while varying execution relevance. Incidental distractor-tool execution is logged as a deviation rather than redefining q. The outcome is minimum view-level success, not minimum task-by-task success. Natural-match-only and full replay share the objective, orbit credit, training tasks, rollout episodes, and update count; the natural arm has fewer eligible scoring anchors, and its separate matched-FLOP control appears in Table 37.

At reference length 16, the clean results are 70.8 for Aug-GRPO, 73.4 for Success-Xfer-MF, 72.6 for natural matching, and 73.2 for full replay; the transformed values appear in Figure 4. Success-Xfer-MF retains a 0.5-point advantage over replay at sparse $q = 4$, whereas replay leads by 3.5 at dense $q = 12$. The unused-only control yields a -0.1-point replay-minus-natural difference. Together, these comparisons separate execution relevance from schema-only lexical change, as H2 requires.

Table 61: Coverage on the same Aug-GRPO behavior archive at reference length 16. Natural matches and success targets shrink as density rises, whereas replay eligibility stays at 100%. Denominator: all selected visited anchors in calls 9–16, independent of whether their rollout succeeds. Success target means at least one verified successful continuation from that exact history in the fixed 4×4 continuation budget; it is not terminal success of the source trajectory.

Condition	Natural match (%)	Success target (%)	Replay eligible (%)
Unused only	63.7	69.2	100.0
$q = 4$	46.8	61.5	100.0
$q = 8$	30.9	49.1	100.0
$q = 12$	19.6	37.8	100.0

The archive is shared across target-eligibility diagnostics, so a stronger method cannot improve its denominator by visiting easier states. Sampling uses the same call-bin schedule and continuation budget in every condition, and renderability is assessed before loss weighting. The table thus quantifies Challenge 2 directly: at $q = 12$, only 19.6% of anchors admit a natural match and 37.8% a success target. These length-16 values are separate from the length-32 on-policy coverage audit in Table 37; 19.6% must not be substituted for its 12.4% late-bin value.

A same-wrong prefix can fail without a cross-view divergence, whereas a divergent prefix can recover within the spare call budget. With full replay, first divergence occurs in $18.4 + 20.1 = 38.5\%$ of episodes, and the conditional recovery fraction is $18.4/38.5 = 47.8\%$. Treating every first divergence

Table 62: First-divergence and recovery decomposition at dense $q = 12$. Full replay has the highest no-divergence success and the lowest divergence-failure share. Percentages average over all 16 transformed views and share the full episode denominator; each row sums to 100. Divergence is the first difference between inverse-mapped paired greedy actions, counted before subsequent recovery. The final column is the sum of the two success columns, not WVA.

Method	No div., success	Div., success	Div., failure	No div., failure	Mean success
Aug-GRPO	39.2	13.9	30.7	16.2	53.1
Success-Xfer-MF	42.3	18.0	23.6	16.1	60.3
Natural-match-only	43.1	16.0	25.2	15.7	59.1
Full replay	46.2	18.4	20.1	15.3	64.6

as a terminal failure would therefore erase these successful recoveries. The aggregate mean success of 64.6 is distinct from the dense WVA of 60.9.

Table 63: Second-length comparison on 600 paired programs with reference length 32 and call budget 36. Full replay leads every method at dense density. Sparse and dense transformed decisions retain densities $1/4$ and $3/4$. The clean score is shared across conditions. Values are success percentages; transformed columns use the view-level minimum.

Method	Clean	Sparse ($q = 8$)	Dense ($q = 24$)
Aug-GRPO	56.7	46.2	28.4
Success-Xfer-MF	59.1	51.6	37.9
Natural-match-only	58.4	50.1	35.8
Full replay	59.5	50.9	42.5

Interpretation and theory boundary. The replay-versus-natural contrast grows from 0.2 to 4.8 points at length 16 and from 0.8 to 6.7 at length 32; the corresponding dense-minus-sparse contrasts are 4.6 and 5.9 points. The same pattern at both lengths connects lost history comparisons to decision relevance while keeping clean difficulty fixed within each program, supporting H2 and Contribution 2. Equal effects from unused-tool renaming, or no density dependence after fixed-length pairing, would have contradicted this explanation. The existing horizon bound depends on stepwise policy discrepancies rather than on the number of calls alone; it does not assert that untransformed decisions have zero defect or that mistakes are unrecoverable. The experiments characterize this operating regime without asserting a general density law beyond the tested conditions.

H.3 Choosing between Canonicalization and Learned Correspondence

The same information distinction applies to APIRefactor-232. With a complete registry, FullCanon reaches a migrated score of 62.7 with no within-task view gap by construction. With only the presented schema, LocalCanon is preferable when local structure identifies a unique map: its benchmark-subset score is 72.8 versus 70.4 for OrbitPO. Under ambiguity, the ordering reverses to 38.9 versus 45.9 (Table 21). The migration pattern retains this heterogeneity: LocalCanon leads simple renames (62.6 to 62.1) and parameter refactors (63.0 to 62.9), whereas OrbitPO leads namespace moves (59.2 to 58.4), envelope changes (60.3 to 56.9), and composed migrations (55.9 to 53.2). The aggregate migrated success of 59.5 versus 60.5 masks these differences. A wrapper-plus-OrbitPO model improves macro WVA but remains weaker than unwrapped OrbitPO on the ambiguous subset. The complete comparison therefore favors selecting a method according to the available mapping information and the executed dependency structure, and it does not support the claim that one method dominates every deployment regime.

I External Transfer, Complementarity, and Failure Analysis

This section tests whether the main pattern survives API migrations derived from version histories and held-out view banks, and then locates its boundary with non-symmetry shifts and residual errors. The matched studies in Appendix H specify the mass-dispersion and decision-density signatures used to interpret these transfer results.

I.1 API MIGRATION AND NON-SYMMETRY STRESS TESTS

Figure 5 is the single visualization of the APIRefactor-232 transfer result. The family counts are repeated here only as a protocol clarification: rename (312), namespace (210), parameter (278), envelope (226), and composed (174) tasks sum to the 1,200-task audit. The appendix does not count a second copy of the same heatmap and transfer plot as independent evidence.

On migrated interfaces, ORBITPO leads Aug-GRPO by 10.0 points, with a 95% interval of $[5.4, 15.1]$ and a positive difference in four of five paired seeds. Relative to its paired original score, ORBITPO retains 94.0% of success, compared with 78.7% for Aug-GRPO, and the hardest composed migrations preserve an 11.3-point margin. FullCanon is stronger on every migrated column because this controlled audit supplies the exact adapter map. Exact canonicalization equates each task's pre- and post-migration result; its family columns differ only because the families contain different task subsets. APIRefactor is therefore a positive control for registry-based canonicalization, whereas Table 20 tests the distinct no-map deployment regime targeted by ORBITPO. Because the backend, task, and environment randomness are held fixed, the audit adds ecological structure without relaxing the exact-symmetry contract: it tests transfer from generic synthetic transformations to refactor motifs observed in version histories, not robustness to changed API semantics.

Table 64: Native score (0–100) on evaluations excluded from training and model selection. Gains concentrate on exact renaming/permutation and serialization shifts; semantic shifts check that the method preserves rather than universally improves unrelated robustness.

Evaluation	Shift type	GRPO	Aug-GRPO	ORBITPO
ToolRobustBench exact-interface subset	Exact symmetry	34.8	45.9	**54.0**
ACEBench held-out tool families	New semantics	51.7	52.6	**53.8**
Unseen chat/function-call template	New serialization family	43.5	51.4	**57.9**
Tool-description paraphrase	Non-invertible shift	48.1	**49.3**	49.0
Added distractor tools	Semantic action-set shift	41.2	42.4	**43.0**
Type/permission mutation	Semantic transition shift	**38.6**	38.1	38.3
AgentChaos faults (Tan et al., 2026)	Runtime response fault	27.4	**27.9**	27.8

The exact ToolRobustBench subset (+8.1 over Aug-GRPO) and the unseen function-call template (+6.5) show that the learned semantic alignment outlives one serializer. Paraphrase and permission mutation, by contrast, serve as negative controls: the near-zero differences confirm that ORBITPO targets exact coordinate changes and does not produce a generic robustness effect.

Table 65: Tail robustness on five independently seeded banks of 64 held-out exact views per task. ORBITPO has the best value on all three tail summaries and wins four of five banks. CVaR_{10} averages the worst 10% of view-level benchmark scores; worst-64 is their minimum. Bank s.d. is across the five banks, and "wins" counts rank-one banks.

Method	Primary worst	CVaR_{10}	Worst-64	Bank s.d.	Wins
GRPO	36.2	39.1	30.4	1.5	0/5
Aug-GRPO	47.5	50.0	42.9	1.4	0/5
GroupDRO	51.8	54.3	47.0	1.1	0/5
Adv.-transform	52.4	54.1	48.4	1.3	0/5
Success-Xfer-MF	52.8	55.2	48.0	1.0	0/5
LocalCanon-GRPO	54.6	56.0	50.6	1.5	1/5
ORBITPO	**55.9**	**57.4**	**51.7**	1.2	**4/5**

The 64 views are drawn only after training, from aliases, permutations, and composition words absent from the primary 16-transformation bank. CVaR is less sensitive to one extreme draw, whereas worst-64 deliberately makes the robustness test harder. GroupDRO, adversarial training, and successful transfer change order between the two tail summaries, and LocalCanon leads one bank. OrbitPO retains the strongest aggregate tail results (57.4 CVaR_{10} and 51.7 worst-64) without being the least variable row.

Table 66: Distribution of the first terminal error in 300 failed trajectories per method, sampled uniformly across benchmarks and orbit views. ORBITPO most reduces coordinate-specific alias/order errors. Each cell reports the integer count followed by its percentage, so the counts are auditable against the fixed denominator rather than being standalone rounded percentages. Two annotators, blinded to method identity, assign one mutually exclusive category before reconciliation.

First terminal error	Aug-GRPO	ORBITPO
Semantic plan or tool choice	87 (29.0%)	130 (43.3%)
Coordinate-specific alias/order error	73 (24.3%)	**26 (8.7%)**
Argument-value binding	54 (18.0%)	**36 (12.0%)**
Entity-handle propagation	47 (15.7%)	67 (22.3%)
Syntactically malformed completed call	0 (0.0%)	0 (0.0%)
Schema/type-invalid completed call	0 (0.0%)	0 (0.0%)
Length-cap failure $\perp_L$	1 (0.3%)	1 (0.3%)
Well-typed state-precondition violation	23 (7.7%)	**17 (5.7%)**
Policy violation or refusal	15 (5.0%)	23 (7.7%)

I.2 RESIDUAL FAILURE ANALYSIS

The category counts sum to 300 within each method, and the displayed shares sum to 100% after rounding. The annotation release contains reconciled labels; because no pre-reconciliation cross-tab is available, no inter-annotator coefficient is claimed. The diagnostic signature is not the larger post-treatment share of planning errors by itself, but the 15.6-point reduction in coordinate-specific failures, the error class that exact interface changes induce, as those failures are removed from a smaller residual pool. These are conditional shares among failed trajectories, not rates over all evaluated episodes. Zero syntax/type rows are enforced by the online grammar, and the separately counted length-cap failures and well-typed precondition violations resolve what an aggregate "invalid call" category would otherwise conflate.

I.3 COMPLEMENTARITY WITH AGENT OPTIMIZATION

Table 67: WVA (0–100) for complementary long-horizon and interface alternatives. ORBITPO exceeds every single-method alternative, and adding CAST helps on two of three benchmarks. CAST supplies critique supervision, Trace-Free+ rewrites descriptions, and PTC uses executable code calls.

Method	AppWorld	τ-bench	ToolSandbox
GRPO	22.2	26.4	30.3
Trace-Free+	24.9	29.0	32.8
RC-GRPO	26.5	31.0	35.8
ExpG (Wang et al., 2026)	29.0	33.5	37.8
CAST	27.8	32.1	36.5
PTC-GRPO	30.5	34.4	38.9
Aug-GRPO	27.3	34.6	41.1
ORBITPO	39.6	41.4	49.7
ORBITPO + CAST	**39.6**	**46.8**	**52.1**

RC-GRPO, ExpG, CAST, and PTC obtain meaningful improvements over GRPO, yet their remaining interface gaps relative to ORBITPO support a distinct role for equivariance. Combining CAST with ORBITPO leaves AppWorld unchanged and improves τ-bench by 5.4 points and ToolSandbox by 2.4 points (2.6 points on average), supporting complementarity between semantic coordinate alignment and action-level critique without implying a uniform gain.