COORDINATION SCAFFOLDS: TASK-DEPENDENT WITHDRAWAL OF PRIVILEGED OBSERVATIONS IN COOPERATIVE MULTI-AGENT REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Training-only privileged observations can simplify reinforcement learning, but a deployable policy must eventually act without them. Existing guided-observability methods typically use one withdrawal recipe across tasks. We show that this assumption breaks in cooperative multi-agent reinforcement learning (MARL), where privileged state can scaffold the discovery of coordination itself. Across SMAC and MPE, the same withdrawal schedule can improve one task and degrade another: augmentation prevents bootstrap failure on the hardest map, while linear decay loses up to 0.205 win rate on another and fast decay is consistently poor on low-demand MPE tasks. Schedule-mismatch interventions, dose-matched timing controls, and withdrawal-recovery analyses show that these effects arise along the optimization path; the final removal produces little immediate disruption. We introduce Observation–Control Irreducibility (O_1), a probe-conditioned diagnostic based on the predictive gain from modeling agents' local observations jointly rather than additively. An OCI-guided schedule uses this demand estimate to choose earlier or later withdrawal. It avoids the large fixed-schedule failures on the development tasks and, when frozen before training, is oracle-best on one informative held-out task and within 0.078 of the oracle on another. A QMIX replication preserves the task-dependent diagnosis but requires a different demand-to-schedule calibration. These results identify privileged observations as coordination scaffolds whose withdrawal should be conditioned on the task and learner.

1 INTRODUCTION

Training-only privileged information is widely used to make difficult reinforcement-learning problems easier. In cooperative multi-agent reinforcement learning (MARL), this information appears in several forms. Centralized critics use global state to improve value estimation while actors remain decentralized (Pinto et al., 2018; Lowe et al., 2017; Yu et al., 2022; Amato, 2024). Teacher–student methods first learn with privileged inputs and then distill behavior into a deployable student (Chen et al., 2020; Lee et al., 2020; Kumar et al., 2021). A third route, studied here, injects privileged observations directly into the policy during training and gradually removes them before deployment (Kamienny et al., 2020; Weigand et al., 2021). This route trains one policy through a transition from privileged to decentralized observations.

The transition introduces a design choice that is easy to overlook: *when should the privileged observations be withdrawn?* Direct antecedents use a shared exposure recipe across tasks. Guided observability applies one linear annealing rule and reports little sensitivity to its cutoff on single-agent benchmarks (Weigand et al., 2021); privileged-information dropout replaces explicit timing with one fixed stochastic interface (Kamienny et al., 2020). Cooperative MARL changes the role of the privileged signal. Global state contains information about the other agents, so privileged observations can support the discovery of coordination that local-only learning may fail to bootstrap. The

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.

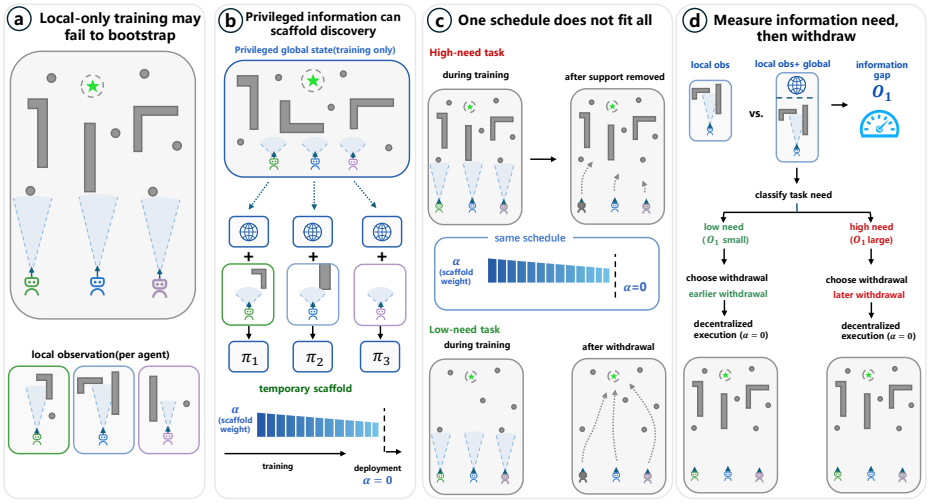

Figure 1: **Privileged observations as coordination scaffolds.** (a) Local-only training can fail to discover coordination. (b) Training-time global state temporarily augments each actor and is fully removed before deployment. (c) The same withdrawal schedule can be too early for a high-demand task and unnecessarily prolonged for a low-demand task. (d) O_1 provides a probe-conditioned demand signal and selects an earlier or later withdrawal; deployment is always decentralized ($\alpha = 0$).

same support can also alter the optimization path for tasks that already admit strong decentralized learning. Withdrawal timing therefore becomes a task-dependent training choice.

We study this problem as an *information curriculum*. The privileged state acts as a temporary coordination scaffold. Its value should depend on how much control-relevant structure is missing from individual local views, while the consequences of a particular schedule depend on how the learner uses that support over training. This perspective leads to three questions. First, does privileged input prevent coordination-discovery failures? Second, can the same withdrawal rule help and hurt across tasks, and where in training does that difference arise? Third, can task demand be measured before full training well enough to condition the withdrawal?

Our experiments answer these questions in sequence. Across three SMAC maps and three MPE tasks, privileged augmentation stabilizes the hardest coordination problem, but no fixed task-agnostic schedule is uniformly favorable. Linear decay helps 5m_vs_6m but loses 0.205 win rate on 10m_vs_11m; fast exponential decay performs best on two SMAC maps and worst on all three low-demand MPE tasks. The same task dependence appears under the published PO-GRL-style episode-mixing interface and under PI-Dropout. Interventions then isolate timing from total exposure. Removing support too early weakens the high-demand map, dose-matched blocks behave differently from smooth decay, and gradual withdrawal produces little immediate terminal disruption. The schedule therefore changes which decentralized strategy is discovered and consolidated during training.

We make this mechanism actionable with Observation–Control Irreducibility (O_1). The diagnostic compares a joint predictor over all agents' local observations with an additive collection of per-agent predictors. The resulting held-out gap quantifies the predictive advantage of cross-agent observation interactions for control-relevant targets. An OCI-guided schedule maps larger O_1 values to later withdrawal. On the development tasks it avoids every large fixed-schedule failure observed in our grid and attains the highest observed worst-seed SMAC floor. We then freeze the diagnostic, thresholds, task list, seeds, and budgets before evaluating held-out tasks. The rule is oracle-best on one informative held-out task and within 0.078 of the oracle on a second. Finally, QMIX reproduces the task-dependent schedule diagnosis but not the MAPPO-calibrated schedule map, separating a task-level demand signal from a learner-specific prescription.

Contributions. Our contribution is a task-conditioned account of privileged-observation withdrawal in cooperative MARL. First, we show that privileged state acts as a coordination scaffold whose benefit and cost vary systematically across tasks, making withdrawal timing a substantive training decision. Second, matched timing interventions and recovery analyses locate the effect in the optimization path and distinguish schedule shape from total information exposure and terminal reliance. Third, we introduce O_1 as a probe-conditioned coordination-demand diagnostic and use it to construct a demand-guided withdrawal rule, with prospective held-out evaluation and a second-learner boundary that clarifies what transfers and what requires recalibration.

2 Related Work

2.1 Privileged Information and Guided Observability

Learning with training-only information originates in learning using privileged information (Vapnik & Vashist, 2009) and enters RL through several interfaces. Privileged critics expose extra state to the value function while the deployed actor always consumes local observations (Pinto et al., 2018; Warrington et al., 2021; Baisero & Amato, 2022; Baisero et al., 2022). Teacher–student methods train a privileged policy and distill it into a deployable student (Rusu et al., 2016; Chen et al., 2020; Lee et al., 2020; Kumar et al., 2021); guider-based methods co-train privileged and decentralized policies (Li et al., 2026a;b). Scaffolder and PIGDreamer place privileged sensing in training-time auxiliary or world-model components (Hu et al., 2024; Huang et al., 2025), and Cai et al. (2024) analyze privileged training theoretically.

The closest setting to ours gives privileged observations directly to the policy and removes them during training. PI-Dropout learns a privileged-information-conditioned noise interface (Kamienny et al., 2020), while PO-GRL transitions the policy from full to partial observability through a shared annealing schedule (Weigand et al., 2021). These approaches establish direct privileged-input training as a practical interface. We study the schedule dimension in cooperative MARL and characterize when a shared withdrawal rule is mismatched to the coordination demands of the task.

2.2 Cooperative MARL and Coordination Information

Centralized critics and value decomposition are standard tools for cooperative MARL (Sunehag et al., 2018; Rashid et al., 2018; Son et al., 2019; Foerster et al., 2018; Lowe et al., 2017; Yu et al., 2022; Kuba et al., 2022; Zhong et al., 2024). Their results also show that additional centralized information is not automatically beneficial: critic centralization can change bias and variance (Lyu et al., 2021; 2023), and independent learners remain competitive on several benchmarks (de Witt et al., 2020). Learned communication provides non-local information at execution time (Foerster et al., 2016; Sukhbaatar et al., 2016; Jiang & Lu, 2018; Das et al., 2019; Wang et al., 2020a; Jaques et al., 2019); our setting keeps execution strictly decentralized and varies training-time information during learning.

Coordination measures provide a natural way to condition this training interface. Mutual-information objectives explicitly shape coordination (Kim et al., 2023; Wang et al., 2023), while partial information decomposition and Dec-POMDP theory formalize information that cannot be recovered from one local view (Williams & Beer, 2010; Oliehoek & Amato, 2016; Bernstein et al., 2002). Level of Influence measures interaction through co-player identity and allocates training compute across scenarios (Chen et al., 2024). O_1 measures a joint-over-additive predictive gap in the agents' observations and uses that gap to condition the withdrawal of training-only privileged state.

2.3 Information Curricula

Curriculum learning changes the training distribution over tasks or difficulty (Bengio et al., 2009; Narvekar et al., 2020; Portelas et al., 2020; Florensa et al., 2017; Matiisen et al., 2020), including curricula over team size and scenario difficulty in MARL (Wang et al., 2020b; Long et al., 2020; Chen et al., 2021). Privileged-observation withdrawal defines a different curriculum axis: the environment remains fixed while the actor's information set changes. Our results show that this information cur-

riculum should be conditioned on task demand and, as the QMIX replication indicates, calibrated to the learner that consumes it.

3 METHOD

3.1 PRIVILEGED-OBSERVATION WITHDRAWAL

We consider Dec-POMDPs (Oliehoek & Amato, 2016) under centralized training with decentralized execution. Agent i receives local observation $o_{i,t}$, while the simulator additionally exposes global state s_t during training. We train actors on

$$\tilde{o}_{i,t} = [\, o_{i,t},\ \alpha_t s_t \,],$$

where $\alpha_t \in [0, 1]$ follows a schedule over normalized training progress p. Every withdrawal schedule reaches $\alpha = 0$ by $p = 0.9$, so the final 10% of training is fully decentralized and all primary final evaluations use local observations alone. Within each task, the base learner, network, optimizer, environment threads, seeds, and training budget are fixed across schedules.

The development grid contains four regimes. Standard training uses $\alpha \equiv 0$. Linear decay moves from 1 to 0 over 90% of training and is the deterministic analogue of guided observability's shared linear annealing (Weigand et al., 2021). Exponential decay uses e^{-5p} as a faster task-agnostic withdrawal. The OCI-guided schedule is a falling sigmoid whose midpoint depends on the measured O_1 demand: high-demand tasks retain privileged information longer and low-demand tasks shed it earlier. We additionally use schedule-mismatch interventions and dose-matched exposure blocks to separate timing from total exposure. Full equations and hyperparameters are in App. A.

3.2 OBSERVATION–CONTROL IRREDUCIBILITY

The schedule-selection problem requires a task-level signal of control-relevant joint structure in local views. We estimate this signal from trajectories generated by a pre-specified probe distribution μ_{probe} (App. B). For each control-relevant target y_t—short-horizon return or task events—we fit an additive predictor $f_{\text{add}} = \sum_i f_i(o_{i,t})$ and a joint predictor $f_{\text{joint}}(o_{1,t}, \ldots, o_{n,t})$. The additive model combines independent per-agent predictions; the joint model can represent cross-agent observation interactions. With episode-level splits, we define

$$O_1 = \max_{y \in \mathcal{Y}} \begin{cases} \dfrac{\max\{0,\ R^2_{\text{joint},y} - R^2_{\text{add},y}\}}{R^2_{\text{joint},y} + 0.01}, & R^2_{\text{joint},y} > 0.01, \\ 0, & \text{otherwise.} \end{cases}$$

O_1 is therefore a probe-conditioned estimate of jointly predictable control structure beyond an additive combination of individual views. We use this predictive gap as an empirical coordination-demand proxy: larger values indicate a greater predictive benefit from cross-agent observation interactions under μ_{probe}. The probe distribution matters: behavior that never reaches coordinated states cannot expose the corresponding joint structure. App. B measures this requirement directly.

The six development tasks span O_1 values from 0.000 to 0.081: MPE spread tasks lie at 0.000–0.007, MPE reference at 0.011, 8m_vs_9m at 0.019, 10m_vs_11m at 0.074, and 5m_vs_6m at 0.081. The MPE development runs in Table 1 were selected with a cheaper state-gap screen on the same probe trajectories; both estimators place the MPE tasks in the low-demand regime, with one internal bucket-boundary difference analyzed in App. B.

3.3 DEMAND-GUIDED SCHEDULING AND EVALUATION

The OCI-guided rule maps O_1 to the midpoint of a falling sigmoid. The thresholds and slopes are given in App. A; Algorithm 1 specifies the complete procedure. The mapping is calibrated on the MAPPO development tasks and is frozen for the prospective held-out evaluation. Section 4.6 separately tests which parts of the diagnosis transfer to QMIX.

We evaluate three SMAC maps (5m_vs_6m, 8m_vs_9m, 10m_vs_11m; Samvelyan et al., 2019) and three MPE tasks (Lowe et al., 2017; Mordatch & Abbeel, 2018; Terry et al., 2021). All schedule

Table 1: Final decentralized performance (mean±std, n=5 seeds; seed-paired comparisons). Win rate for SMAC and return for MPE. Bold marks the best mean in each row. The preferred schedule varies across tasks; OCI-guided has the highest observed worst-seed SMAC floor. For the MPE development rows, the OCI-guided column reports runs selected by the original state-gap screen; displayed O_1 values use the unified estimator, and the reference boundary case is analyzed in App. B. Ten-seed extensions of the 5m/10m standard cells (App. H) are $0.510\pm.108$ and $0.753\pm.137$.

Task (O_1)	Standard	Linear	Exponential	OCI-guided
5m_vs_6m (.081)	$0.480\pm.134$	$0.550\pm.042$	$\mathbf{0.575\pm.069}$	$0.565\pm.066$
8m_vs_9m (.019)	$0.840\pm.046$	$0.785\pm.075$	$\mathbf{0.915\pm.034}$	$0.840\pm.068$
10m_vs_11m (.074)	$\mathbf{0.765\pm.160}$	$0.560\pm.181$	$0.725\pm.123$	$0.755\pm.087$
spread_3 (.000)	-60.9 ± 2.0	-61.3 ± 2.4	-62.5 ± 2.3	$\mathbf{-60.6\pm3.0}$
reference (.011)	-36.3 ± 0.5	$\mathbf{-36.1\pm0.6}$	-37.8 ± 0.6	-36.5 ± 0.7
spread_6 (.007)	$\mathbf{-195.8\pm3.1}$	-201.7 ± 5.4	-202.2 ± 4.7	-197.0 ± 2.4
worst SMAC seed	0.225	0.275	0.450	**0.475**

comparisons are paired by environment seed. We report mean±std over five seeds, bootstrap 95% confidence intervals for paired differences, sign-flip permutation tests for within-task effects, and exact task-label permutation tests for cross-task interactions (App. H), following the small-sample recommendations of Agarwal et al. (2021). Final evaluation is always decentralized at $\alpha = 0$; SMAC uses win rate and MPE uses return.

4 Results

4.1 Coordination Scaffolding

Privileged observations first matter by changing whether coordination is discovered at all. On 5m_vs_6m, standard MAPPO reaches 0.480 ± 0.134 over the five development seeds and 0.510 ± 0.108 over the ten-seed extension. One standard seed never bootstraps: its training win rate peaks at 0.179 over 5M steps and finishes at 0.225. Across the forty-five runs of the three augmented schedules, every training peak exceeds 0.4. The augmented conditions also reduce the five-seed standard deviation on this map from 0.134 to 0.042–0.069. On this high-demand task, temporary global state therefore acts as discovery insurance for a coordination strategy that local-only learning does not reliably find.

Table 1 shows that the preferred exposure pattern differs across tasks. Standard training has the highest mean on 10m_vs_11m, while augmented schedules improve 5m_vs_6m. This variation makes exposure timing a central training choice.

4.2 Task-Dependent Withdrawal

The effect of a fixed withdrawal rule changes across tasks. Linear decay improves 5m_vs_6m by 0.070 relative to the five-seed standard mean, but changes 8m_vs_9m by -0.055 and 10m_vs_11m by -0.205. Exponential decay is the strongest method on 5m_vs_6m and 8m_vs_9m and is close to standard on 10m_vs_11m, yet it is the lowest-return schedule on all three low-demand MPE tasks. The schedule that provides useful coordination support in one regime can therefore spend training under an unnecessary privileged input distribution in another.

The same cross-task pattern appears under alternative privileged-information interfaces. A PO-GRL-style episode-mixing schedule (Weigand et al., 2021) gives full augmentation to a linearly decaying fraction of episodes. It reaches 0.505 on 5m_vs_6m and 0.635 on 10m_vs_11m, the latter 0.130 below standard training with four of five paired seeds worse. PI-Dropout (Kamienny et al., 2020), which learns a privileged-information-conditioned noise interface and has no explicit withdrawal time, reaches $0.560 \pm .092$, $0.715 \pm .094$, and $0.680 \pm .170$ on the three SMAC maps. Its effect is again positive on 5m and negative on 10m. These alternatives differ in implementation but preserve the central asymmetry: a single privileged-information recipe does not have a consistent effect across the evaluated coordination regimes.

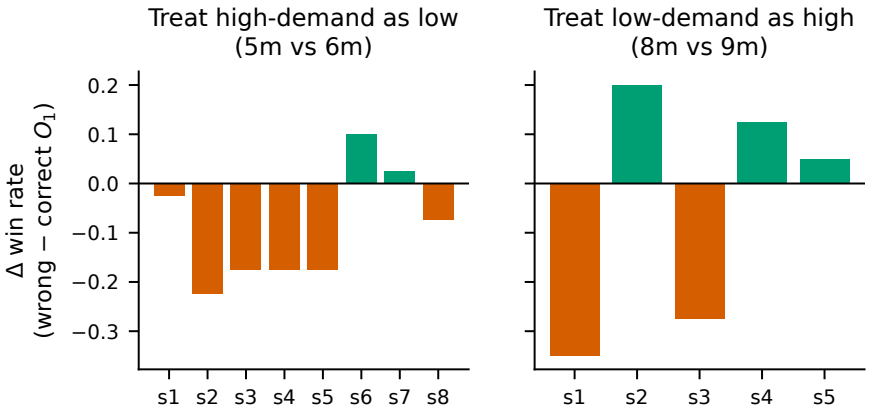

Figure 2: Schedule-mismatch intervention, paired by seed. Assigning the early-withdrawal schedule to the high-demand 5m_vs_6m map reduces performance in six of eight seeds (left). The later-withdrawal schedule on 8m_vs_9m produces mixed paired changes with greater dispersion (right).

Two-stage distillation provides a different compute tradeoff. With a full 5M-step privileged teacher plus a separate 5M-step student, the distilled policy reaches $0.625/0.900/0.683$ on 5m/8m/10m and a 0.500 worst seed. Under the same total 5M-step budget as the withdrawal schedules, splitting 2.5M/2.5M yields $0.492/0.742/0.650$ and a 0.275 floor. Distillation is therefore competitive when a separate teacher budget is available; direct withdrawal uses one training budget and makes the exposure path itself consequential.

Episode-level Bernoulli masking provides a stricter availability-removal control. The masked grid is more variable and its effects are smaller than under amplitude scaling. A pre-specified seed expansion of the discovery-stage exponential-mask 5m–10m interaction does not reproduce on the five independent seeds: the replication slice gives $I = -0.095$ (CI $[-0.325, +0.120]$, $p = 0.532$), while pooling ten seeds gives $I = +0.073$ (CI $[-0.123, +0.255]$, $p = 0.483$). Accordingly, the masked experiment provides evidence of removal-interface sensitivity; the 5m–10m interaction is not stable under independent seed expansion. The complete masked grid and replication analysis are reported in Table 9 and App. H.

4.3 Optimization-Path Mechanism

The mechanism tests separate three possibilities in sequence: whether assigning a different withdrawal time changes learning, whether timing matters beyond total privileged exposure, and whether performance changes at the final removal event.

The first intervention swaps the demand-conditioned schedule. On 5m_vs_6m, assigning the early-withdrawal schedule used for low-demand tasks reduces the mean from 0.547 to 0.456 across eight paired seeds; six seeds worsen and two improve (wrong $-$ correct $= -0.091$, bootstrap CI $[-0.162, -0.012]$, exact sign-flip $p = 0.086$). The later-withdrawal mismatch on 8m_vs_9m primarily increases dispersion. Figure 2 shows the paired effects.

Dose-matched blocks then separate timing from total privileged exposure. Blocks with integrated dose $\bar{\alpha} = 0.2$ are placed early, mid, or late in training. On 8m_vs_9m, final performance is $0.858/0.917/0.850$ ($n = 3$), a spread of 0.067 across placements. On 5m_vs_6m, the same placements reach $0.483/0.533/0.433$, all below the smooth exponential schedule at 0.575. Thus, equal total exposure can lead to different outcomes when its temporal structure changes (Fig. 3).

Recovery dynamics finally test whether the last removal event is the dominant source of the effect. Across ninety smooth-schedule runs from the scaled and masked grids, the last transition to $\alpha = 0$ changes performance by -0.003 on average; the largest immediate dip is 0.064. Eighty-eight of ninety runs re-attain their pre-withdrawal level within 0.05M steps on average, and the remaining two finish within 0.04 of it. The decentralized evaluation curves in Fig. 4 show the same temporal pattern: augmented schedules can lag while support is present and recover after withdrawal,

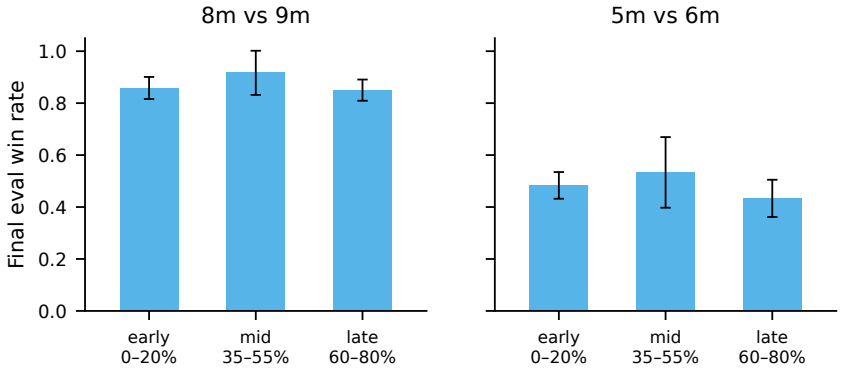

Figure 3: Dose-matched exposure blocks ($\bar{\alpha}$=0.2) placed early, mid, or late (n=3 per placement). On 8m_vs_9m, the three means span 0.067; on 5m_vs_6m, all three block placements remain below the smooth exponential schedule (0.575) with approximately the same integrated dose.

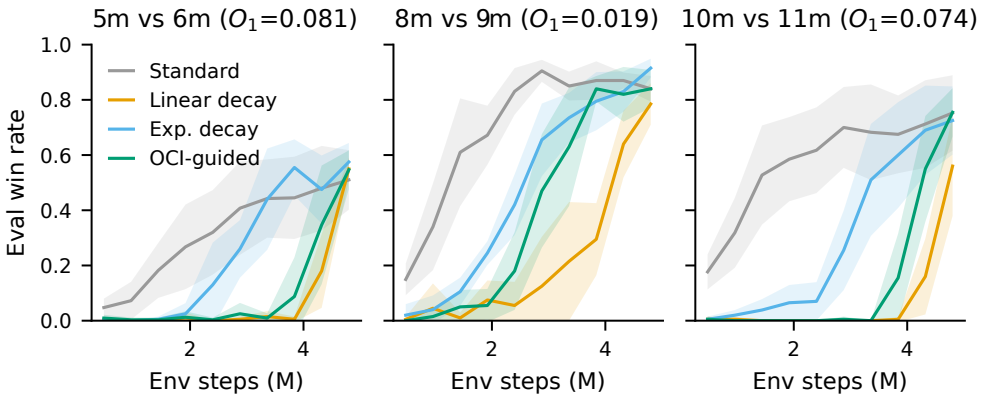

Figure 4: Decentralized evaluation ($\alpha = 0$) throughout training (mean±std). Privileged schedules delay decentralized competence while support is present and recover after withdrawal. Final decentralized performance remains schedule-dependent after withdrawal, while the terminal removal itself produces little immediate disruption.

yet their final decentralized outcomes remain schedule-dependent. Likewise, the late gap between privileged-regime training performance and periodic decentralized evaluation does not predict final performance (within-map correlations -0.04 to -0.19). In contrast, policies trained with $\alpha \equiv 1$ achieve 0.30–0.80 win rate while privilege is present but at most 0.075 after removing it.

Together, the mismatch, dose, and recovery controls support an optimization-trajectory account: changing when support is available changes the trajectory of decentralized learning, while the final transition to local-only input produces little immediate disruption.

4.4 Demand-Guided Scheduling

The preceding results provide two ingredients for schedule selection: withdrawal effects differ across tasks, and timing changes the learning trajectory. The O_1 diagnostic turns this task dependence into a schedule-selection rule. Across the development grid, OCI-guided has a worst SMAC seed of 0.475, compared with 0.450 for exponential decay, 0.275 for linear decay, and 0.225 for standard training. Exponential decay has a slightly higher SMAC mean (0.738 vs. 0.720) and a statistically unresolved floor difference, but it is the lowest-return schedule on all three MPE tasks. OCI-guided withdrawal sheds privilege early on these low-demand tasks and remains near the stan-

Table 2: Prospective held-out schedule grid (mean±std, n=5 seeds; worst seed in parentheses). Bold marks the best mean in each row. The frozen OCI-guided rule is oracle-best on 3s5z and has regret 0.078 on 3s5z_vs_3s6z.

Held-out task	Standard	Linear	Exponential	OCI-guided
3s5z (O_1=.007)	0.590±.167 (0.425)	0.590±.103 (0.475)	0.610±.061 (0.525)	**0.645±.048** (0.575)
3s5z_vs_3s6z (.026)	0.425±.140 (0.250)	**0.538±.236** (0.350)	0.450±.124 (0.325)	0.460±.216 (0.049)

dard baseline. In the development set, it avoids the observed bootstrap failure of standard training, the 10m loss under linear decay, and the consistent MPE cost of exponential decay.

O_1 estimates demand; realized improvement also depends on optimization headroom. The two high-demand maps 5m and 10m have similar O_1 values (0.081 and 0.074) but different baseline headroom. A post-hoc analysis finds that O_1 multiplied by headroom correlates more strongly with realized benefit across the three SMAC maps than O_1 alone ($r = 0.89$ vs. 0.51; Fig. 5, App. B). With only three maps this result is descriptive, but it clarifies the role of the diagnostic: demand indicates where privileged coordination information can matter, while optimization headroom influences how much improvement remains available.

4.5 Prospective Held-Out Tasks

We freeze the O_1 estimator, thresholds, task list, seeds, and budgets before running three held-out tasks: SMAC 3s5z ($O_1 = 0.007$), SMAC 3s5z_vs_3s6z ($O_1 = 0.026$), and SMACv2 zerg_5_vs_5 ($O_1 = 0.100$) (Ellis et al., 2023). All use the same feedforward $[128, 128]$ architecture. The first two tasks yield informative schedule grids (Table 2); the third remains at zero under every tested schedule and even under fully privileged evaluation, placing it beyond the learnability boundary of the frozen protocol.

On 3s5z, the frozen rule selects the oracle-best schedule, with mean 0.645 and worst seed 0.575. On 3s5z_vs_3s6z it ranks second, paying 0.078 regret to linear decay, and one OCI-guided seed collapses to 0.049. Across the two informative held-out tasks, the selector is competitive without task-specific schedule tuning. The 0.049 seed on 3s5z_vs_3s6z falls below the development-set floor. Full protocol and per-seed outcomes are in App. F.

4.6 Learner Dependence

We repeat the three-map SMAC grid with QMIX (Rashid et al., 2018), using the same budgets and evaluation protocol (three seeds per cell). The task-level diagnosis transfers: fixed schedules again change effect across maps, and the map where linear decay is harmful under MAPPO is also harmed under QMIX (-0.205 under MAPPO and -0.292 under QMIX on 10m_vs_11m). The MAPPO-calibrated OCI sigmoid does not transfer: it is the weakest QMIX schedule and includes collapsed seeds. The task diagnostic and schedule calibration therefore occupy different levels. O_1 measures structure in the task and probe distribution, while the mapping from that demand to a safe withdrawal trajectory depends on the learner that consumes the information.

5 Discussion

The experiments support a simple view of direct privileged-input training in cooperative MARL: privileged observations are temporary coordination scaffolds, and their withdrawal defines an information curriculum. This view explains why a shared schedule can reverse effect across tasks. A high-demand task may need the scaffold long enough to discover and consolidate coordinated behavior; a lower-demand task can spend the same training time under an input distribution that contributes little to the final decentralized policy. The schedule is therefore coupled to what the task requires and to how the learner moves through its optimization landscape.

The intervention results sharpen this account. Total exposure alone is insufficient: equal-dose blocks and smooth decay produce different outcomes on the high-demand map. Gradual schedules show almost no terminal cliff and typically recover immediately, placing the lasting difference earlier in

the training trajectory. Never-withdraw policies illustrate the limiting case: they can perform with global state and become unusable without it, whereas gradual withdrawal gives the learner time to consolidate a decentralized strategy. This distinction turns withdrawal from a deployment heuristic into part of the training dynamics.

O_1 provides a task-level handle on this process through a joint-over-additive predictive gap under a probe distribution. Final gain additionally depends on optimization headroom, and the QMIX experiment shows that the same probe-conditioned signal can require a different schedule map for a different learner. A practical scheduler should therefore combine a task diagnostic with learner-specific calibration and, eventually, uncertainty about the probe estimate. The held-out results show that a frozen demand-based rule can remain competitive without per-task schedule search.

Taken together, the results suggest a broader design principle for CTDE. Training-time information should be treated as a resource with a task- and learner-dependent exposure policy. The design question is how long the actor should consume global state before its information set matches deployment. This framing connects privileged learning to curriculum design and provides a concrete route for making training-time centralization conditional on the coordination structure of the task.

6 LIMITATIONS

The empirical scope covers six development tasks from SMAC and MPE, two informative prospective held-out tasks, and one held-out task that the frozen training protocol could not learn. MAPPO provides the primary grid and QMIX tests learner transfer. The mechanism experiments localize the effect to the optimization path but do not distinguish whether exploration, representation learning, or credit assignment carries the change. The Bernoulli-mask discovery interaction did not reproduce on the independent seed expansion. The masked grid therefore characterizes sensitivity to the removal mechanism. Finally, O_1 depends on probe trajectories that contain minimally competent behavior; on the SMAC tasks, random or 10%-budget probes are insufficient. These boundaries make cheap probe acquisition and learner-specific calibration the main requirements for extending automatic schedule selection.

7 CONCLUSION

Training-only privileged observations can make decentralized coordination easier to discover, but their value depends on how and when they are withdrawn. Across cooperative MARL tasks, fixed withdrawal rules exhibit task-dependent benefits and costs, and timing interventions show that these differences are created along the optimization path, while the final removal step produces little immediate disruption. Observation–Control Irreducibility provides a probe-conditioned signal for withdrawal timing: an OCI-guided schedule avoids the large fixed-schedule failures in the development grid and remains competitive under a frozen held-out evaluation, while the QMIX results reveal the need for learner-specific calibration.

The broader implication is that privileged observations should be designed as coordination scaffolds with task-conditioned exposure. Once the actor itself consumes training-only information, the schedule becomes part of the learning problem. Conditioning that information curriculum on task demand offers a principled way to retain the discovery benefits of centralized information while still producing decentralized policies for deployment.

REPRODUCIBILITY STATEMENT

Code, configurations, and raw results are provided in the supplementary material. The appendix documents the experimental setup, O_1 diagnostic, and statistical procedures, and reported values are automatically checked against the raw results.

AI USE STATEMENT

Generative AI tools were used to assist with literature search, provide feedback on experimental design, support result interpretation, assist with code implementation and debugging, and support

manuscript drafting, translation, revision, and grammar editing. The research hypotheses and core conceptual framing were developed and refined by the authors. All AI-assisted outputs incorporated into the work were reviewed and verified by the authors, who take responsibility for the final content.

REFERENCES

Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron C Courville, and Marc Bellemare. Deep reinforcement learning at the edge of the statistical precipice. In *Advances in Neural Information Processing Systems*, 2021.

Christopher Amato. An introduction to centralized training for decentralized execution in cooperative multi-agent reinforcement learning. *arXiv preprint arXiv:2409.03052*, 2024.

Andrea Baisero and Christopher Amato. Unbiased asymmetric reinforcement learning under partial observability. In *International Conference on Autonomous Agents and Multiagent Systems*, 2022.

Andrea Baisero, Brett Daley, and Christopher Amato. Asymmetric DQN for partially observable reinforcement learning. In *Uncertainty in Artificial Intelligence*, 2022.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *International Conference on Machine Learning*, 2009.

Daniel S Bernstein, Robert Givan, Neil Immerman, and Shlomo Zilberstein. The complexity of decentralized control of Markov decision processes. *Mathematics of Operations Research*, 27(4): 819–840, 2002.

Yang Cai, Xiangyu Liu, Argyris Oikonomou, and Kaiqing Zhang. Provable partially observable reinforcement learning with privileged information. In *Advances in Neural Information Processing Systems*, 2024.

Dian Chen, Brady Zhou, Vladlen Koltun, and Philipp Krähenbühl. Learning by cheating. In *Conference on Robot Learning*, 2020.

Jiayu Chen, Yuanxin Zhang, Yuanfan Xu, Huimin Ma, Huazhong Yang, Jiaming Song, Yu Wang, and Yi Wu. Variational automatic curriculum learning for sparse-reward cooperative multi-agent problems. In *Advances in Neural Information Processing Systems*, 2021.

Yuxin Chen, Chen Tang, Thomas Tian, Chenran Li, Jinning Li, Masayoshi Tomizuka, and Wei Zhan. Quantifying interaction level between agents helps cost-efficient generalization in multi-agent reinforcement learning. *Reinforcement Learning Journal*, 4:1950–1964, 2024.

Abhishek Das, Théophile Gervet, Joshua Romoff, Dhruv Batra, Devi Parikh, Mike Rabbat, and Joelle Pineau. TarMAC: Targeted multi-agent communication. In *International Conference on Machine Learning*, 2019.

Christian Schroeder de Witt, Tarun Gupta, Denys Makoviichuk, Viktor Makoviychuk, Philip HS Torr, Mingfei Sun, and Shimon Whiteson. Is independent learning all you need in the StarCraft multi-agent challenge? *arXiv preprint arXiv:2011.09533*, 2020.

Benjamin Ellis, Jonathan Cook, Skander Moalla, Mikayel Samvelyan, Mingfei Sun, Anuj Mahajan, Jakob Foerster, and Shimon Whiteson. SMACv2: An improved benchmark for cooperative multi-agent reinforcement learning. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2023.

Carlos Florensa, David Held, Markus Wulfmeier, Michael Zhang, and Pieter Abbeel. Reverse curriculum generation for reinforcement learning. In *Conference on Robot Learning*, 2017.

Jakob Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. Counterfactual multi-agent policy gradients. In *AAAI Conference on Artificial Intelligence*, 2018.

Jakob N. Foerster, Yannis M. Assael, Nando de Freitas, and Shimon Whiteson. Learning to communicate with deep multi-agent reinforcement learning. In *Advances in Neural Information Processing Systems*, 2016.

Edward S Hu, James Springer, Oleh Rybkin, and Dinesh Jayaraman. Privileged sensing scaffolds reinforcement learning. In *International Conference on Learning Representations*, 2024.

Dongchi Huang, Jiaqi Wang, Yang Li, Chunhe Xia, Tianle Zhang, and Kaige Zhang. PIGDreamer: Privileged information guided world models for safe partially observable reinforcement learning. In *International Conference on Machine Learning*, pp. 25859–25875, 2025.

Natasha Jaques, Angeliki Lazaridou, Edward Hughes, Caglar Gulcehre, Pedro Ortega, DJ Strouse, Joel Z Leibo, and Nando De Freitas. Social influence as intrinsic motivation for multi-agent deep reinforcement learning. In *International Conference on Machine Learning*, 2019.

Jiechuan Jiang and Zongqing Lu. Learning attentional communication for multi-agent cooperation. In *Advances in Neural Information Processing Systems*, 2018.

Pierre-Alexandre Kamienny, Kai Arulkumaran, Feryal Behbahani, Wendelin Boehmer, and Shimon Whiteson. Privileged information dropout in reinforcement learning. *arXiv preprint arXiv:2005.09220*, 2020.

Woojun Kim, Whiyoung Jung, Myungsik Cho, and Youngchul Sung. A variational approach to mutual information-based coordination for multi-agent reinforcement learning. In *International Conference on Autonomous Agents and Multiagent Systems*, pp. 40–48, 2023.

Jakub Grudzien Kuba, Ruiqing Chen, Muning Wen, Ying Wen, Fanglei Sun, Jun Wang, and Yaodong Yang. Trust region policy optimisation in multi-agent reinforcement learning. In *International Conference on Learning Representations*, 2022.

Ashish Kumar, Zipeng Fu, Deepak Pathak, and Jitendra Malik. RMA: Rapid motor adaptation for legged robots. In *Robotics: Science and Systems*, 2021.

Joonho Lee, Jemin Hwangbo, Lorenz Wellhausen, Vladlen Koltun, and Marco Hutter. Learning quadrupedal locomotion over challenging terrain. *Science Robotics*, 5(47), 2020.

Yueheng Li, Guangming Xie, and Zongqing Lu. Guided policy optimization under partial observability. In *International Conference on Learning Representations*, 2026a.

Yueheng Li, Guangming Xie, and Zongqing Lu. Multi-agent guided policy optimization. In *International Conference on Learning Representations*, 2026b.

Qian Long, Zihan Zhou, Abhinav Gupta, Fei Fang, Yi Wu, and Xiaolong Wang. Evolutionary population curriculum for scaling multi-agent reinforcement learning. In *International Conference on Learning Representations*, 2020.

Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. In *Advances in Neural Information Processing Systems*, 2017.

Xueguang Lyu, Yuchen Xiao, Brett Daley, and Christopher Amato. Contrasting centralized and decentralized critics in multi-agent reinforcement learning. In *International Conference on Autonomous Agents and Multiagent Systems*, 2021.

Xueguang Lyu, Andrea Baisero, Yuchen Xiao, Brett Daley, and Christopher Amato. On centralized critics in multi-agent reinforcement learning. *Journal of Artificial Intelligence Research*, 77:295–354, 2023.

Tambet Matiisen, Avital Oliver, Taco Cohen, and John Schulman. Teacher-student curriculum learning. *IEEE Transactions on Neural Networks and Learning Systems*, 31(9):3732–3740, 2020.

Igor Mordatch and Pieter Abbeel. Emergence of grounded compositional language in multi-agent populations. In *AAAI Conference on Artificial Intelligence*, 2018.

Sanmit Narvekar, Bei Peng, Matteo Leonetti, Jivko Sinapov, Matthew E Taylor, and Peter Stone. Curriculum learning for reinforcement learning domains: A framework and survey. *Journal of Machine Learning Research*, 21(181):1–50, 2020.

Frans A Oliehoek and Christopher Amato. *A concise introduction to decentralized POMDPs*. Springer, 2016.

Lerrel Pinto, Marcin Andrychowicz, Peter Welinder, Wojciech Zaremba, and Pieter Abbeel. Asymmetric actor critic for image-based robot learning. In *Robotics: Science and Systems*, 2018.

Rémy Portelas, Cédric Colas, Lilian Weng, Katja Hofmann, and Pierre-Yves Oudeyer. Automatic curriculum learning for deep RL: A short survey. In *International Joint Conference on Artificial Intelligence*, 2020.

Tabish Rashid, Mikayel Samvelyan, Christian Schroeder de Witt, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. QMIX: Monotonic value function factorisation for deep multi-agent reinforcement learning. In *International Conference on Machine Learning*, 2018.

Andrei A Rusu, Sergio Gomez Colmenarejo, Caglar Gulcehre, Guillaume Desjardins, James Kirkpatrick, Razvan Pascanu, Volodymyr Mnih, Koray Kavukcuoglu, and Raia Hadsell. Policy distillation. In *International Conference on Learning Representations*, 2016.

Mikayel Samvelyan, Tabish Rashid, Christian Schroeder de Witt, Gregory Farquhar, Nantas Nardelli, Tim GJ Rudner, Chia-Man Hung, Philip HS Torr, Jakob Foerster, and Shimon Whiteson. The StarCraft multi-agent challenge. In *International Conference on Autonomous Agents and Multiagent Systems*, 2019.

Kyunghwan Son, Daewoo Kim, Wan Ju Kang, David Earl Hostallero, and Yung Yi. QTRAN: Learning to factorize with transformation for cooperative multi-agent reinforcement learning. In *International Conference on Machine Learning*, pp. 5887–5896, 2019.

Sainbayar Sukhbaatar, Arthur Szlam, and Rob Fergus. Learning multiagent communication with backpropagation. In *Advances in Neural Information Processing Systems*, 2016.

Peter Sunehag, Guy Lever, Audrunas Gruslys, Wojciech Marian Czarnecki, Vinicius Zambaldi, Max Jaderberg, Marc Lanctot, Nicolas Sonnerat, Joel Z Leibo, Karl Tuyls, and Thore Graepel. Value-decomposition networks for cooperative multi-agent learning based on team reward. In *International Conference on Autonomous Agents and Multiagent Systems*, 2018.

J K Terry, Benjamin Black, Nathaniel Grammel, Mario Jayakumar, Ananth Hari, Ryan Sullivan, Luis S Santos, Clemens Dieffendahl, Caroline Horsch, Rodrigo Perez-Vicente, Niall L Williams, Yashas Lokesh, and Praveen Ravi. PettingZoo: Gym for multi-agent reinforcement learning. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2021.

Vladimir Vapnik and Akshay Vashist. A new learning paradigm: Learning using privileged information. *Neural Networks*, 22(5-6):544–557, 2009.

Jiangxing Wang, Deheng Ye, and Zongqing Lu. Mutual-information regularized multi-agent policy iteration. In *Advances in Neural Information Processing Systems*, 2023.

Tonghan Wang, Jianhao Wang, Chongyi Zheng, and Chongjie Zhang. Learning nearly decomposable value functions via communication minimization. In *International Conference on Learning Representations*, 2020a.

Weixun Wang, Tianpei Yang, Yong Liu, Jianye Hao, Xiaotian Hao, Yujing Hu, Yingfeng Chen, Changjie Fan, and Yang Gao. From few to more: Large-scale dynamic multiagent curriculum learning. In *AAAI Conference on Artificial Intelligence*, 2020b.

Andrew Warrington, Jonathan W Lavington, Adam Scibior, Mark Schmidt, and Frank Wood. Robust asymmetric learning in POMDPs. In *International Conference on Machine Learning*, 2021.

Stephan Weigand, Pascal Klink, Jan Peters, and Joni Pajarinen. Reinforcement learning using guided observability. *arXiv preprint arXiv:2104.10986*, 2021.

Paul L Williams and Randall D Beer. Nonnegative decomposition of multivariate information. *arXiv preprint arXiv:1004.2515*, 2010.

Chao Yu, Akash Velu, Eugene Vinitsky, Jiaxuan Gao, Yu Wang, Alexandre Bayen, and Yi Wu. The surprising effectiveness of PPO in cooperative multi-agent games. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2022.

Yifan Zhong, Jakub Grudzien Kuba, Xidong Feng, Siyi Hu, Jiaming Ji, and Yaodong Yang. Heterogeneous-agent reinforcement learning. *Journal of Machine Learning Research*, 25(32): 1–67, 2024.

A Training Setup and Withdrawal Schedules

MAPPO. We use the HARL implementation (Zhong et al., 2024) with shared actor parameters. Common hyperparameters across all schedules and tasks: learning rate 5×10^{-4} (actor and critic, Adam, $\epsilon = 10^{-5}$), discount 0.99, GAE $\lambda = 0.95$, PPO clip 0.2, entropy coefficient 0.01, value-loss coefficient 1.0, max gradient norm 10, huber loss ($\delta = 10$), orthogonal initialization, feature normalization. SMAC: 20 parallel environments, episode length 200, 5M environment steps; 5m_vs_6m and 10m_vs_11m use feedforward $[128, 128]$ actors, 8m_vs_9m uses a recurrent $[64, 64, 64]$ actor (one GRU layer, data chunk length 10). MPE: 8 parallel environments, 500k steps, feedforward $[64, 64]$, PPO epochs 5. Within each map, all schedules share every one of these settings and the environment seed list; the schedule $\alpha(p)$ is the only difference. Architectures differ *across* maps (following the tuned per-map configurations of the underlying library), but the primary schedule comparisons are within-map, where architecture and optimizer are fixed. The mismatch interventions of Sec. 4.3 are also within-map. App. H additionally reports an 8m control with the feedforward $[128, 128]$ actor used on 5m and 10m (standard, exponential, and OCI-guided; seeds 1–3), removing the remaining cross-map architecture variation.

Schedules. With training progress $p \in [0, 1]$: standard $\alpha = 0$; linear $\alpha = \max(0, 1 - p/0.9)$; exponential $\alpha = e^{-5p}$; OCI-guided $\alpha = 1 - \sigma\big(k\,(p - p_{\text{mid}})\big)$, a falling sigmoid whose midpoint is set by the task's measured demand: $p_{\text{mid}} = 0.55$ for $O_1 > 0.05$ (k=10), 0.40 for $O_1 \in (0.02, 0.05]$ (k=8), 0.30 for $O_1 \in (0.01, 0.02]$ (k=8), and 0.25 otherwise (k=10). All schedules are clamped to 0 for $p \geq 0.9$, so the final 10% of training is always fully decentralized.

Algorithm 1 OCI-guided withdrawal. The O_1-to-schedule map is calibrated for the learner family; Sec. 4.6 tests transfer to QMIX.

collect probe trajectories on the target task (App. B)
fit f_{add}, f_{joint}; compute the normalized held-out R^2 gap O_1
set sigmoid midpoint p_{mid} from O_1 (higher demand $\rightarrow$ later withdrawal; thresholds above)
for each training iteration at progress p **do**
    $\alpha \leftarrow 1 - \sigma\big(k(p - p_{\text{mid}})\big)$, clamped to 0 for $p \geq 0.9$
    train on $\tilde{o}_i = [\, o_i,\ \alpha s \,]$ with the unmodified learner
end for
deploy and evaluate at $\alpha = 0$

The PO-GRL baseline (Sec. 4.2) follows Weigand et al. (2021): each episode is granted full augmentation (α=1) with probability $\max(0, 1 - p/0.5)$, resampled per environment thread at episode boundaries, and no augmentation otherwise; per-seed results in Table 6. Block schedules set $\alpha = 1$ inside a window (0–20%, 35–55%, or 60–80% of training) and 0 elsewhere, matching the exponential schedule's integrated dose $\bar{\alpha} \approx 0.2$.

B O_1 Diagnostic and Probe Requirements

The quantity used to set p_{mid} is the normalized held-out R^2 gap in Sec. 3: an additive probe $\sum_i f_i(o_{i,t})$ versus a joint probe $f(o_{1,t}, \ldots, o_{n,t})$, fit on episode-held-out splits. Targets y_t are short-horizon return and task events (kills / episode outcome). The SMAC values (5m_vs_6m 0.081, 10m_vs_11m 0.074, 8m_vs_9m 0.019) come from this joint-versus-additive probe and are the numbers that selected the schedules in Table 1.

Probe data provenance. Probe trajectories are independent of the curriculum training runs they condition. The frozen SMAC values were computed from episodes mixing a random policy, a

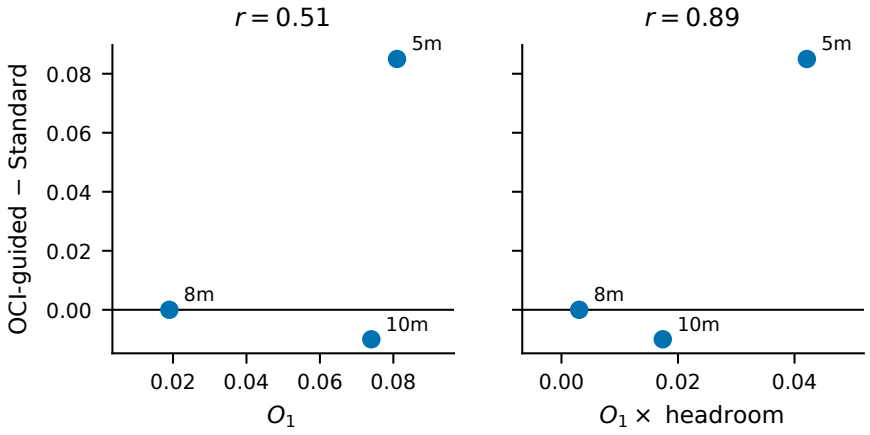

Figure 5: Post-hoc relation between the realized OCI-guided benefit over standard training and O_1 alone (left) or $O_1 \times$ baseline headroom (right), across the three SMAC development maps.

scripted attack-weakest heuristic, and policy checkpoints from earlier unrelated projects; all predate this study and every curriculum run. Under this protocol the estimate is dominated by task identity: across three probe seeds per map, between-task variance exceeds within-task variance (intraclass correlation 0.73), the six-map ordering is exactly reproduced (Spearman $\rho = 1.0$ against the frozen means), and the modal per-seed bucket matches the frozen bucket on four of six maps, with the two exceptions sitting on bucket boundaries.

How much behavior quality does the probe need? We audited two cheaper protocols, both fixed before their results were seen. (i) Checkpoint-free episodes alone (random + scripted heuristic, no training, three seeds per map): the estimate collapses (most seeds return exactly 0, ICC drops to 0.15, and only one of six modal buckets matches), because behavior that never reaches coordinated states leaves the joint probe nothing to explain beyond the additive one. (ii) Probe policies trained from scratch for 10% of the main budget (500k of 5M steps, matching the 50k-of-500k protocol that suffices on MPE): on SMAC this also fails: ICC 0.27, ordering uncorrelated with the frozen values ($\rho = -0.5$), zero of five modal buckets match (this protocol covers the five SMAC maps; no from-scratch probe was trained for zerg_5_vs_5). Competent play is therefore a prerequisite of the diagnostic on long-horizon SMAC tasks, and a 10% training budget does not reach it there, while it does on the short-horizon MPE tasks. These results establish a probe requirement: O_1 is an estimate under μ_{probe}, and that distribution must contain minimally competent play. Obtaining such trajectories is therefore the dominant selection cost when competent behavior is not already available. On MPE the probe costs 10% of one training run, measured once per task and reused by every seed (a 2% amortized overhead over the five seeds trained here). On SMAC the frozen values reuse pre-existing checkpoints; the tested 10%-budget from-scratch probe is insufficient.

The MPE values in that table ($0.000/0.007/0.011$ for spread_3 / spread_6 / reference) come from the same joint-versus-additive estimator, applied to trajectories from 50k-step probe policies (three seeds per task). Schedule selection on MPE, however, was originally run with a cheaper ridge state-gap screen $R^2(o, s \to y) - R^2(o \to y)$ on those trajectories, which gave $0.002/0.004/0.009$ (cross-seed std ≤ 0.011). Both estimators place all three MPE tasks in the low-demand region below 0.02 and agree on the spread buckets, but they disagree on reference, which straddles the internal 0.01 boundary: the screen prescribes p_{mid}=0.25, the schedule whose runs Table 1 reports, while the unified estimator prescribes p_{mid}=0.30, measured next.

Boundary sensitivity. We ran the p_{mid}=0.30 schedule that the unified estimator prescribes on reference. Across five paired seeds, final return changes from -36.52 ± 0.73 (p_{mid}=0.25) to -37.70 ± 1.19 (delta -1.18, CI $[-2.66, 0.30]$, exact $p = 0.25$; three seeds worsen). The coarse low-demand classification is stable across estimators, while the hard bucket boundary introduces sensitivity near $O_1 = 0.01$. This motivates continuous or uncertainty-aware schedule mappings.

C Development-Set Results and Timing Diagnostics

Tables 3 and 4 contain the per-seed runs behind Table 1. They are generated directly from the raw result files by `gen_appendix_tables.py`.

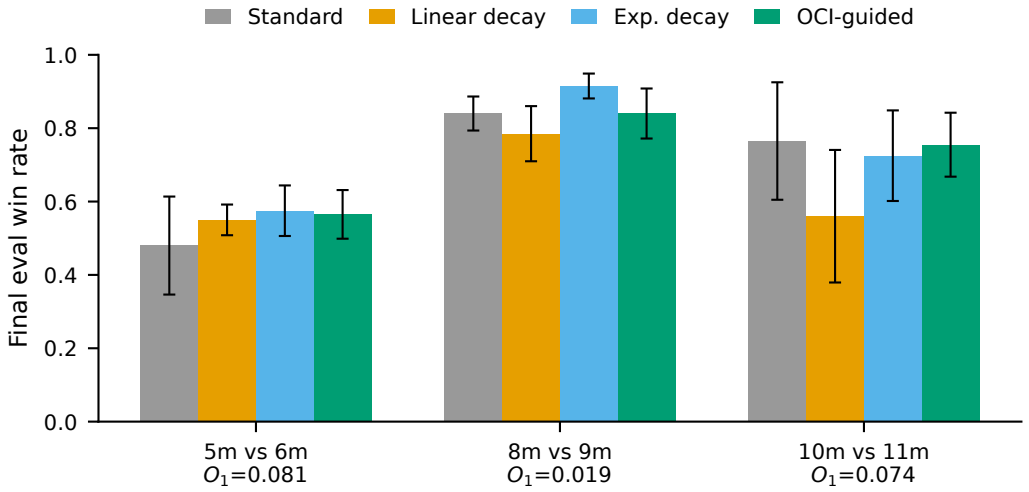

Figure 6: Bar-chart view of Table 1: final decentralized SMAC win rate (mean±std). The schedule ranking changes across maps, and standard training has the widest dispersion on 5m_vs_6m and 10m_vs_11m.

Table 3: Per-seed final decentralized win rates, SMAC (MAPPO). Every number underlying Table 1.

Map	Schedule	s1	s2	s3	s4	s5
5m_vs_6m	Standard	0.500	0.600	0.225	0.500	0.575
	Linear	0.525	0.625	0.550	0.550	0.500
	Exponential	0.450	0.650	0.575	0.575	0.625
	OCI-guided	0.650	0.600	0.500	0.600	0.475
8m_vs_9m	Standard	0.925	0.800	0.825	0.800	0.850
	Linear	0.775	0.675	0.750	0.825	0.900
	Exponential	0.900	0.875	0.975	0.900	0.925
	OCI-guided	0.900	0.750	0.925	0.775	0.850
10m_vs_11m	Standard	0.825	0.875	0.800	0.875	0.450
	Linear	0.450	0.800	0.650	0.275	0.625
	Exponential	0.675	0.575	0.700	0.725	0.950
	OCI-guided	0.850	0.750	0.700	0.625	0.850

Timing controls

Every augmented schedule delays the availability of a deployable policy: while $\alpha > 0$, the decentralized greedy evaluation lags. Averaging decentralized evaluation over all of training (AUC), standard training leads on every map (e.g. 8m_vs_9m: standard 0.694, exponential 0.476, OCI-guided 0.390, linear 0.225; 5m_vs_6m: 0.295 / 0.247 / 0.098 / 0.076). These AUC values quantify the intermediate-deployability cost visible in Fig. 4.

D Teacher–Student Distillation Baseline

The teacher is a MAPPO policy trained with $\alpha \equiv 1$ (augmented observations throughout; checkpoints saved at the end of training). The teachers double as the never-withdraw endpoint of Sec. 4.3:

Table 4: Per-seed final returns, MPE (MAPPO).

Task	Schedule	s1	s2	s3	s4	s5
spread_3	Standard	-64.70	-60.06	-58.83	-59.80	-61.07
	Linear	-62.32	-62.97	-58.05	-58.95	-64.40
	Exponential	-62.65	-61.66	-62.15	-59.42	-66.62
	OCI-guided	-59.50	-61.11	-55.97	-61.06	-65.23
reference	Standard	-36.42	-35.96	-36.04	-35.69	-37.15
	Linear	-35.17	-35.88	-36.94	-36.03	-36.38
	Exponential	-38.51	-37.72	-38.31	-36.82	-37.57
	OCI-guided	-35.98	-35.55	-36.59	-36.82	-37.66
spread_6	Standard	-190.61	-196.92	-199.67	-197.45	-194.34
	Linear	-196.07	-203.05	-195.16	-209.82	-204.18
	Exponential	-196.25	-200.44	-208.91	-198.95	-206.30
	OCI-guided	-199.53	-194.02	-195.34	-200.03	-195.98

Table 5: QMIX replication grid (final decentralized win rate, mean$\pm$std, n=3 seeds).

Map	Standard	Linear	Exponential	OCI-guided
5m_vs_6m	0.258 ± 0.031	0.290 ± 0.045	0.254 ± 0.202	0.113 ± 0.145
8m_vs_9m	0.457 ± 0.220	0.358 ± 0.174	0.592 ± 0.092	0.274 ± 0.179
10m_vs_11m	0.625 ± 0.162	0.333 ± 0.263	0.459 ± 0.224	0.250 ± 0.241

evaluated with the privileged input they win 0.30–0.80 of episodes (5m $0.300/0.400/0.550$, 8m $0.800/0.700/0.700$, 10m $0.450/0.650/0.425$ across seeds), but evaluated decentralized their win rate is 0.0 on every map and seed except one 8m seed at 0.075. The student shares the actor architecture but sees only o_i (global slots zeroed). Distillation is on-policy behavior cloning: the student acts in the environment, the teacher labels the visited states, and the student minimizes cross-entropy to the teacher's action distribution (2 epochs, 4 minibatches per batch). The $2\times$-budget variant trains the teacher for 5M steps and distills for 5M steps; the matched-budget variant splits the 5M-step budget 2.5M/2.5M. Per-seed results in Table 7.

E PI-DROPOUT BASELINE

PI-Dropout (Kamienny et al., 2020) conditions training-time multiplicative noise on privileged information so that no withdrawal schedule is needed; at execution the privileged branch is ignored. Our adaptation to the shared-parameter MAPPO actor preserves this contract exactly. The deployed computation path consumes only the local observation: the privileged tail of the input is sliced off before the actor trunk, so privileged information cannot influence the deployed policy's forward pass under any evaluation condition (verified numerically on trained checkpoints: actor features are bit-identical under any value of the privileged tail once noise is disabled). During training, a two-layer noise network maps the global state to per-unit Gaussian-dropout rates $p \in (0, 0.5]$ on the actor features ($\sigma^2 = p/(1-p)$, reparameterized so the PPO objective trains the noise network jointly), with an information-dropout KL pressure toward larger noise at coefficient $\beta = 10^{-3}$. What is adapted: the original method is single-agent and actor-critic; we apply the noise to the shared MAPPO actor trunk and leave the centralized critic unchanged, as in every other baseline in the paper. What is matched: training budget (5M steps), optimizer, PPO hyperparameters, and evaluation protocol are identical to the schedule grids. Actors are feedforward $[128, 128]$ on all three maps; on 8m this matches the feedforward architecture control of App. H rather than the recurrent development actor. $\beta = 10^{-3}$ (a conventional information-dropout scale) and the p-cap of 0.5 (bounding the off-policyness the resampled noise introduces into PPO) were fixed before the full runs, shared across all three maps and all seeds, and held constant across tasks.

Per-seed deployment win rates (α=0, greedy, the same protocol as every other run): 5m 0.575, 0.725, 0.525, 0.525, 0.450 (mean 0.560); 8m 0.550, 0.800, 0.800, 0.675, 0.750 (mean 0.715); 10m 0.450, 0.775, 0.850, 0.825, 0.500 (mean 0.680). The learned noise converges to small magnitudes

Table 6: Guided-observability baseline (PO-GRL sample mixing, linear ramp to 50% cutoff), per-seed final win rates (n=5 on 5m and 10m; n=3 on 8m).

Map	s1	s2	s3	s4	s5	mean±std
5m_vs_6m	0.400	0.500	0.575	0.325	0.725	0.505±0.139
8m_vs_9m	0.700	1.000	0.850	—	—	0.850±0.122
10m_vs_11m	0.625	0.725	0.625	0.525	0.675	0.635±0.066

Table 7: Distillation baselines, per-seed final win rates (n=3). 2× budget: 5M-step teacher + 5M-step student; matched: 2.5M + 2.5M.

Map	Variant	s1	s2	s3
5m_vs_6m	2× budget	0.725	0.625	0.525
	matched budget	0.275	0.625	0.575
8m_vs_9m	2× budget	0.875	0.825	1.000
	matched budget	0.850	0.600	0.775
10m_vs_11m	2× budget	0.725	0.825	0.500
	matched budget	0.825	0.600	0.525

by the end of training (mean $\sigma^2 \approx 0.05$ on visited states for the 5m seed-1 checkpoint), so the regularizer's effect is concentrated earlier in training, where the privileged conditioning shapes which policy is found, consistent with the path-dependence mechanism of Sec. 4.3.

F Prospective Held-Out Protocol

The written protocol (included with the code as `heldout_protocol.md`, with recorded freeze dates and one amendment) fixed, before any curriculum run on these tasks existed: the task list, the frozen probe values (O_1=0.007/0.026/0.100, all from pre-existing result files predating this study), the unchanged $O_1 \to p_{\text{mid}}$ thresholds, seeds 1–5, and budgets (2M steps for 3s5z, whose baseline is stronger; 5M for the other two). All three tasks use feedforward $[128, 128]$ actors and the MAPPO configuration of App. A. Scoring by selection regret was recorded in the protocol before any grid completed. The zerg_5_vs_5 amendment (truncation to one seed per schedule after observing win rate 0.0 at every training and evaluation point, including fully privileged evaluation) is dated in the same file. The reported held-out set therefore follows the frozen task list; Table 8 lists the per-seed outcomes.

Table 8: Held-out per-seed final decentralized win rates.

Task	Schedule	s1	s2	s3	s4	s5
3s5z	Standard	0.425	0.450	0.875	0.525	0.675
3s5z	Linear	0.500	0.475	0.550	0.725	0.700
3s5z	Exponential	0.550	0.650	0.525	0.650	0.675
3s5z	OCI-guided	0.675	0.575	0.600	0.700	0.675
3s5z_vs_3s6z	Standard	0.250	0.625	0.325	0.550	0.375
3s5z_vs_3s6z	Linear	0.350	1.000	0.475	0.475	0.390
3s5z_vs_3s6z	Exponential	0.475	0.675	0.325	0.350	0.425
3s5z_vs_3s6z	OCI-guided	0.550	0.650	0.049	0.450	0.600
zerg_5_vs_5	all schedules	0.000	(truncated per amendment)			

G QMIX Learner Replication

Feedforward agent networks have two 128-unit layers. We use a standard hypernetwork mixer (embedding 32), double-Q targets, a replay buffer of 200k transitions, batch 256, learning rate $5 \times$

Table 9: Bernoulli-masked schedules (mean±std; worst seed in parentheses). Cells marked † pool ten seeds after the pre-specified expansion in App. H; all other cells use the five discovery seeds.

Schedule (masked)	5m_vs_6m	8m_vs_9m	10m_vs_11m
Standard (no privilege)	$0.510\pm.108$ $(0.225)^{\dagger}$	$0.840\pm.046$ (0.800)	$0.753\pm.137$ $(0.450)^{\dagger}$
Linear-mask	$0.515\pm.093$ (0.400)	$\mathbf{0.890\pm.034}$ (0.850)	$0.740\pm.103$ (0.550)
Exponential-mask	$0.568\pm.094$ $(0.325)^{\dagger}$	$0.710\pm.214$ (0.375)	$0.737\pm.144$ $(0.500)^{\dagger}$
OCI-guided-mask	$\mathbf{0.585\pm.064}$ (0.500)	$0.850\pm.032$ (0.800)	$\mathbf{0.766\pm.130}$ (0.550)

10^{-4}, ϵ-greedy annealed $1.0 \rightarrow 0.05$ over the first 10% of training, hard target sync every 2500 gradient updates, one gradient update per two vectorized environment steps. Augmentation enters the agent network input exactly as in the MAPPO setup; transitions are stored un-augmented and re-scaled by the α at sampling time. Full grid in Table 5.

H Stochastic Removal and Statistical Protocol

Bernoulli-masked grid

The Bernoulli gate removes the privileged state at the episode level; the scaled schedules instead vary its magnitude continuously. Its discovery-stage effects are smaller and more variable than those of the scaled schedules. The pre-specified seed expansion below tests the strongest 5m–10m interaction from this grid and does not reproduce it on the independent seeds. Accordingly, the masked experiment characterizes sensitivity to the removal mechanism, with the complete grid retained in this appendix.

Architecture-matched control

The development grid uses a recurrent actor on 8m and feedforward actors on 5m and 10m (App. A). Re-running 8m with the same feedforward $[128, 128]$ actor as 5m and 10m (n=3) preserves the cross-task reversal: exponential decay helps 5m ($+0.095$) and hurts feedforward 8m (-0.092), giving interaction $I = +0.187$, CI $[+0.037, +0.350]$, and exact permutation $p = 0.143$. Architecture also moderates the within-map ranking on 8m: exponential is strongest with the recurrent actor (0.915) and weakest with the feedforward actor (0.650 vs. standard 0.742). Task demand and training architecture therefore jointly determine how a withdrawal trajectory is realized.

Statistical protocol

All cross-schedule comparisons pair runs by environment seed. Reported intervals are bootstrap 95% CIs over seed-paired deltas (10,000 resamples). Within-map schedule effects are tested with sign-flip permutation tests on the paired deltas (exact for $n = 5$: 2^5 sign patterns); cross-map interaction contrasts use the exact task-label permutation test defined below. Primary inference uses paired and permutation-based analyses because individual map comparisons at $n = 5$ are noisy. Cross-task interactions are evaluated directly with the task-label permutation test below.

Task–schedule interactions

For a schedule S and maps A, B, the per-map treatment effect is $\Delta_m = \frac{1}{n}\sum_k \left(J^{(k)}_{m,S} - J^{(k)}_{m,\mathrm{std}}\right)$ over the seeds k paired in both cells (n=5 for the discovery grid, n=10 for the expanded 5m/10m exponential-mask cells), and the interaction is $I = \Delta_A - \Delta_B$. The CI bootstraps seeds independently within each map (10,000 resamples). The p-value is an exact task-label permutation test of the interaction null $H_0 : \Delta_A = \Delta_B$: the pooled per-seed deltas are reassigned to the two maps in all $\binom{n_A+n_B}{n_A}$ ways (252 for two five-seed cells, 56 for the five-versus-three architecture control, 184,756 for the ten-seed expansion), and p is the fraction of reassignments whose $|\bar{d}_A - \bar{d}_B|$ reaches the observed $|I|$. This tests effect equality directly: a schedule effect common to both maps does not by itself produce rejection. We treat this p as the primary inferential statistic and the bootstrap CI as an effect-size summary; with few seeds the permutation null is discrete (the smallest attainable p is

$2/252 \approx 0.008$ for five-seed pairs and $2/56 \approx 0.036$ for the architecture control), so a bootstrap CI excluding zero can coexist with p above 0.05, as it does for the architecture control.

Table 10 reports the complete discovery-stage matrix: every pairwise contrast for every schedule under both mechanisms, plus a per-row omnibus permutation test (statistic $\sum_m n_m(\bar{\Delta}_m - \bar{\Delta})^2$; 10^5 random reassignments of the pooled deltas to maps), to place each discovery-stage interaction in the complete schedule matrix. At n=5 seeds per cell no contrast reaches $p < 0.05$ (the strongest, exponential-mask 5m–8m, has p=0.056) and no omnibus test reaches significance; given the replication outcome below, we treat every entry as discovery-stage evidence. The evidential weight lies in the direction pattern (the schedule effect on 5m is positive in all six schedule–mechanism combinations, on 10m negative or null in all six, while 8m flips sign with schedule and mechanism) and in the converging intervention, learner-replication, and architecture-control results.

Table 10: Complete task×schedule interaction matrix over the five discovery seeds. Δ_m: mean seed-paired effect of the schedule against standard training on map m. $I_{a,b} = \Delta_a - \Delta_b$ with exact task-label permutation p; p_{omni}: omnibus permutation test of equal effects across the three maps. Scaled 8m entries use the development (recurrent) 8m grid; the architecture-matched control above uses a feedforward 8m actor. No entry reaches $p < 0.05$, and the pre-specified expansion below did not replicate the exponential-mask 5m–10m contrast; all entries are discovery-stage evidence.

Schedule	Mech.	$\Delta_{5\text{m}}$	$\Delta_{8\text{m}}$	$\Delta_{10\text{m}}$	$I_{5,8}$	p	$I_{5,10}$	p	$I_{8,10}$	p	p_{omni}
Linear	scaled	$+0.070$	-0.055	-0.205	$+0.125$	0.183	$+0.275$	0.119	$+0.150$	0.357	0.133
Exponential	scaled	$+0.095$	$+0.075$	-0.040	$+0.020$	0.889	$+0.135$	0.413	$+0.115$	0.492	0.551
OCI-guided	scaled	$+0.085$	$+0.000$	-0.010	$+0.085$	0.302	$+0.095$	0.532	$+0.010$	0.968	0.649
Linear	mask	$+0.035$	$+0.050$	-0.025	-0.015	0.833	$+0.060$	0.381	$+0.075$	0.111	0.322
Exponential	mask	$+0.130$	-0.130	-0.110	$+0.260$	0.056	$+0.240$	0.167	-0.020	0.937	0.187
OCI-guided	mask	$+0.105$	$+0.010$	$+0.001$	$+0.095$	0.159	$+0.104$	0.508	$+0.009$	0.992	0.663

Pre-specified seed expansion (masked exponential, 5m–10m). Because the matrix above was computed on the seeds that suggested its contrasts, we expanded the cells {5m, 10m} × {standard, exponential-mask} to seeds 6–10 (20 new runs, identical configuration). The protocol (cells, seeds, primary pooled analysis, and the commitment to run no further seeds regardless of outcome) was frozen before any expansion run launched; an amendment fixing the interpretation matrix, including that seeds 6–10 alone constitute the independent replication evidence (seeds 1–5 participated in selecting the contrast) and the paper wording for every possible outcome, was frozen after the runs landed but before any result was inspected. The protocol, the amendment, and a SHA-256 manifest of the raw result files ship with the code (`seed_expansion_protocol.md`, `analyze_seed_expansion.py`).

Outcome: the discovery seeds gave $I = +0.240$ (CI $[-0.054, +0.470]$, p=0.167); the five new seeds alone give $I = -0.095$ (CI $[-0.325, +0.120]$, exact p=0.532; $\Delta_{5\text{m}} = -0.015$, $\Delta_{10\text{m}} = +0.080$); pooling all ten seeds gives $I = +0.073$ (CI $[-0.123, +0.255]$, p=0.483). Under the frozen wording matrix this is the non-replication branch: the discovery-set interaction did not replicate on independent seeds. One deviation from the frozen protocol is disclosed: the protocol specified the sign-flip permutation inherited from the discovery analysis (p=0.129 discovery, 0.500 replication, 0.471 pooled), which a pre-submission audit found to target the joint null of no effect on either map. The corrected task-label permutation instead targets $\Delta_A = \Delta_B$. The values above use the corrected task-label permutation; both tests select the same pre-specified non-replication branch, so no interpretive language changed. Cell means at n=10: 5m standard 0.510 ± 0.108, 5m exponential-mask 0.568 ± 0.094, 10m standard 0.753 ± 0.137, 10m exponential-mask 0.737 ± 0.144; the pooled per-map effects are $+0.057$ (5m) and -0.015 (10m). The pooled per-map signs match the discovery grid but are carried largely by the discovery seeds; the replication slice alone reverses both. The sharp discovery-set magnitudes do not survive, and the main text reports the masked grid accordingly. Figure 7 shows the three slices side by side.

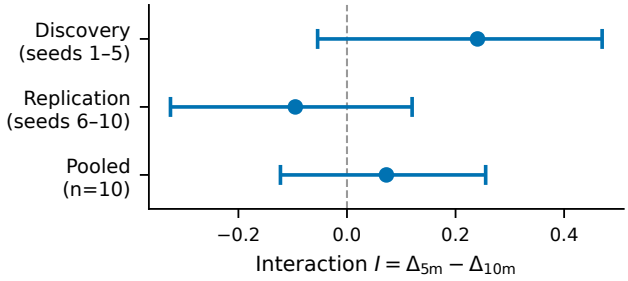

Figure 7: Forest plot of the masked exponential 5m–10m interaction $I = \Delta_{5\mathrm{m}} - \Delta_{10\mathrm{m}}$ across the three pre-specified analysis slices (points: estimates; bars: bootstrap 95% CIs). The discovery estimate does not reproduce on the independent seeds, and no slice excludes zero.

I Compute and Reproducibility

Experiments ran on a shared server (Intel Xeon Platinum 8462Y+ CPUs; NVIDIA RTX 6000 Ada GPUs, with CPU fallback during a temporary hardware outage; policies are small enough that the device does not affect results, only wall-clock time). A full SMAC run (5M steps) takes 6–10 hours; the study comprises over 200 completed training runs. Every number in the paper is checked against the raw result files by `verify_paper_numbers.py`, which exits nonzero on any mismatch; the full check suite runs clean at submission time.