KNOWING MORE, COMPLETING LESS: RELIABILITY LEARNING IN LIFELONG ROBOT FLEETS

Anonymous authors
Paper under double-blind review

ABSTRACT

A robot that stays still may have failed, or it may be waiting for another robot that failed. Learning from such motion is a statistical problem; using what is learned is a coordination problem. We study their connection in lifelong multi-agent path finding. Our central observation is a reversal: under the same planning rule, replacing fleet-average reliability with exact individual reliabilities increases completed tasks by 7.96% in a room but decreases them by 13.09% in a warehouse, with the corresponding sign in all six initial-state averages per setting. To understand this gap, we derive the position-feedback likelihood and characterize exactly which immediate task-value comparisons are identifiable. A task-value difference can be identifiable even when individual reliabilities, or both competing values, are not. We then develop a motion-censored release preference that learns from unambiguous actuator outcomes and refines equally preferred next moves, yielding a 9.76% gain in a two-room evaluation with a frozen method and previously unused seeds. A trajectory decomposition associates the room gain with fewer propagated stalls and the warehouse loss with a growing deficit in proposed moves. A four-robot construction then shows how better immediate task completion, more movement, and fewer robots in cells with at most two neighbors can all lead to fewer tasks after two steps. Together, these results connect what a fleet can learn from motion to the future its planner creates.

1 INTRODUCTION

Suppose a warehouse planner knows exactly which robots are likely to stall. Would giving it this information help the fleet finish more tasks? The natural answer is yes: favor a reliable robot, release the robots behind it, and keep traffic moving. Yet this reasoning skips a crucial step. The robots released now determine the traffic encountered next. A locally attractive move can change a bottleneck, a route, or the next sequence of interactions.

The answer changes sign in paired tests of one fixed planning rule: exact individual probabilities improve task completion by **7.96% in a room** but reduce it by **13.09% in a warehouse**, relative to fleet-average input. The sign repeats in all six initial-state averages per setting and persists at both tested fleet sizes. The planner, failure model, and goal-stream mechanism are shared. This raises two questions for learning-based coordination: *what information should be learned, and at which decision interface should it enter?*

Even the first half of this question is subtle. In lifelong multi-agent path finding (MAPF), robots continuously receive new goals and execute collision-free joint actions. A failed actuator can block an entire line of robots. From the resulting positions, the learner sees motion and nonmotion, not the hidden fault mask. Charging every nonmover with a failure confuses its own reliability with the decisions made around it. A closed cycle is more ambiguous still: all robots move together or all remain, revealing a product of reliabilities rather than a separate label for each robot.

Our approach follows the information through this loop (Figure 1). First, we derive the observation law of legal joint motion and identify the information needed to compare immediate task values. We then use unambiguous labels in a lightweight learned preference. Finally, exact-input interventions and a two-step construction separate the quality of that information from the future created by using it.

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.

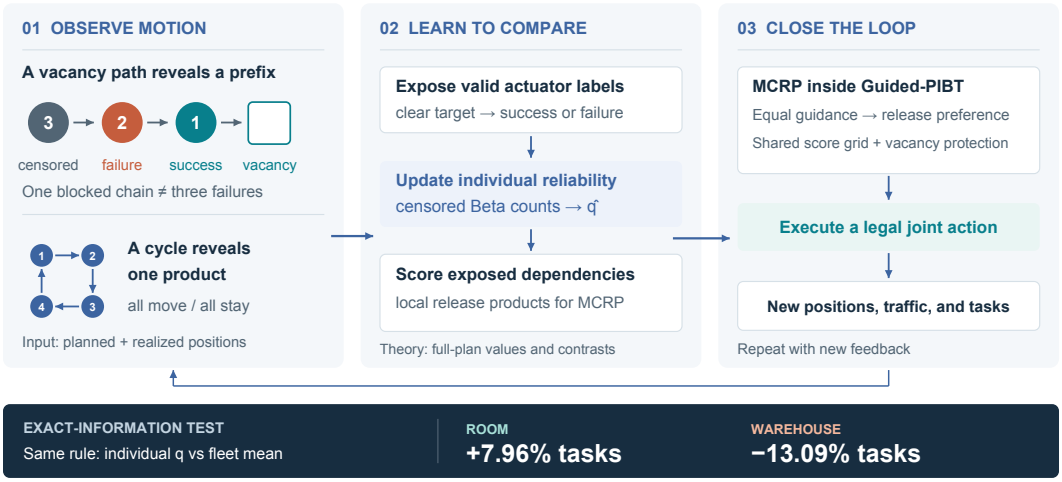


Figure 1: **From observed motion to useful decisions.** Legal joint actions induce vacancy paths and cycles with different feedback. MCRP learns from individually exposed outcomes and uses the resulting estimates within equal-guidance choices. Our theory identifies the products and contrasts needed to evaluate complete plans. The closed loop then determines task throughput: feeding exact identity information to one fixed rule has opposite effects in the prospective room and warehouse tests.

The resulting account has three contributions. **Feedback:** we characterize individual and decision-value identifiability from vacancy-path and cycle observations, including comparisons whose common unidentified terms cancel. **An interface:** we introduce motion-censored release preference (MCRP), which uses unambiguous motion labels to refine equally preferred next moves and improves task completion by 9.76% over its own backbone on previously unused seeds. **A control explanation:** we establish the fixed-rule sign reversal with matched fault and goal streams, decompose the realized motion into proposal activity and execution losses, and construct a legal example in which three favorable one-step criteria reverse in two-step task value. The organizing principle is to distinguish the information needed to score an action from the continuation created by taking it.

2 A FLEET THAT LEARNS FROM ITS OWN MOTION

Execution and feedback. On a four-neighbor grid $G = (V, E)$, n robots are indexed by $i \in \{1, \dots, n\}$. At physical step t, robot i occupies $x_i(t)$ and has goal $g_i(t)$. A joint proposal assigns each robot an adjacent cell or a wait. Legal proposals have distinct targets and no head-on edge swaps. Each proposed mover succeeds with stationary probability q_i, independently across robots and steps. Write $q = (q_1, \dots, q_n)$. A failed robot remains in place; any mover targeting its cell must also remain. Applying this cancellation to a fixed point produces a collision-free realized action. The planner observes the proposal and the next positions. For an online learner, actuator masks and the q_i are hidden.

A task completes when a robot reaches its goal, after which it receives another goal. The objective is the number of completed tasks over $T = 3{,}000$ physical steps. Our backbone is Guided-PIBT (Chen et al., 2024; Okumura et al., 2022): priority inheritance with backtracking (PIBT), using traffic-guided distances to goals. It prioritizes longer time since the last task completion and runs 25 guidance large-neighborhood-search iterations per step. We write Guided25 for this backbone with its inherited preference for empty targets; *equal-guidance* choices have the same guided distance.

Two experimental uses of reliability. The deployable method estimates q_i from past positions. To isolate the effect of identity information, a separate diagnostic supplies either the exact vector q or its exact arithmetic mean $\bar{q}\mathbf{1}$ to one unchanged preference rule. These arms share the same physical fleet; only the planner's input differs. Paired runs share initial positions, initial goals, and every physical fault draw. In the prospective exact-input tests, the kth goal of each robot also agrees

Table 1: **Two information comparisons, two questions.** Within each pair, the rule and physical fleet are shared; only its reliability input differs.

Pair	Reliability input	Question
Learned individual / fleet	$\widehat{q}_i(t)$ / $n^{-1}\sum_j \widehat{q}_j(t)$, from each arm's own censored labels	Does learned identity improve tasks? (Section 5.1)
Exact individual / fleet	True q_i / $n^{-1}\sum_j q_j$, for diagnosis	What does identity change without estimation error? (Section 5.2)

across arms through robot-indexed random streams. Task activation times follow each policy's actual completions.

3 Learning enough to compare plans

3.1 Paths expose labels; cycles expose products

Follow each mover's target to its current occupant. Because a legal proposal has unique targets, its moving robots form disjoint vacancy-terminated paths and directed cycles. Order a path $i_1, \dots, i_L$ outward from its empty endpoint. Its realized moves are a prefix of length K, with

$$\Pr(K = k) = \left(\prod_{j=1}^{k} q_{i_j}\right)(1 - q_{i_{k+1}}), \quad 0 \le k < L, \qquad \Pr(K = L) = \prod_{j=1}^{L} q_{i_j}. \tag{1}$$

The executed prefix gives success labels; the first nonmover gives a failure label. Further upstream, individual outcomes remain hidden behind that failure. In a cycle C, the feedback is instead one all-move bit with probability $Q_C = \prod_{i \in C} q_i$. Independent component likelihoods multiply. This yields the complete position-observation law (Appendix O).

For example, if the robot next to a vacancy moves and the next robot does not, the first succeeded and the second failed. The third robot's unchanged position says nothing about its own actuator. In a four-robot rotation, nonmotion reveals that at least one member failed, but not which ones. These cases call for different statistical updates.

3.2 The information required by a decision

A quantity is *identifiable* if any two reliability vectors inducing the same feedback distribution agree on it. Consider a fixed family of proposals sampled independently of outcomes, each with positive probability. Its *exposure matrix* H contains a unit row $e_i^\top$ for each robot on a vacancy path and an incidence row $\mathbf{1}_C^\top$ for each cycle set C ($\mathbf{1}_C$ marks its members). On $q \in (0,1)^n$, the observation law identifies exactly $H \log q$, with componentwise logarithms. All individual reliabilities are therefore identifiable iff H has column rank n.

A planner often needs less. A release probability is a product over a dependency set; comparing two such products needs only the corresponding contrast of incidence vectors. More generally, fix a state S (positions, goals, and planner history) and a legal joint proposal P, with target P_i for robot i. We call P a complete plan to distinguish it from a partial recursive assignment. For a mover i, let $D_i(P)$ be the path segment from its vacant endpoint through i, or the full cycle containing i. When no current goal is already satisfied, the expected number of goals completed on the next step is

$$r(S, P; q) = \sum_{i: P_i = g_i,\, P_i \neq x_i} \prod_{j \in D_i(P)} q_j = \sum_{D} b_D(P) \exp(\mathbf{1}_D^\top \log q), \tag{2}$$

where $b_D(P)$ counts goal-reaching movers with dependency set D. A dependency traversal evaluates this quantity and the analogous expected-move count in linear time once the dependency arrays are built.

Theorem 1 (Value and contrast identifiability)**.** *For a repeated exogenous proposal design with positive exposure and interior reliabilities, the numerical value $r(S, P; q)$ is structurally identifiable exactly when every $\mathbf{1}_D$ with $b_D(P) > 0$ lies in $\mathrm{row}(H)$. For two plans P, R, their numerical difference is identifiable exactly when this holds for every D with $b_D(P) - b_D(R) \neq 0$.*

Why this holds. In coordinates $z = \log q$, the observation law fixes Hz. Every identified dependency product is therefore a known linear functional of Hz followed by exponentiation. Conversely, a null direction of H preserves the observations. For a sum of distinct exponential monomials to stay constant along every such direction, each term with a nonzero coefficient must lie in the row space. Appendix O gives the full proof.

The cancellation in the second statement matters. Two plans may both contain an unidentified goal-value term that is identical in the two plans. Subtracting their values removes it. A legal five-robot example in the appendix has two unidentified absolute values whose difference is exactly one observed robot's reliability. Similarly, cycle products can identify the value of a goal-reaching rotation while leaving all individual parameters unresolved. For a finite menu of plans, let $\mathcal{V}$ span the incidence vectors of dependency sets whose coefficients $b_D(P)$ vary across the menu. The theorem gives an exact criterion: *all numerical value differences are identifiable iff* $\mathcal{V} \subseteq \mathrm{row}(H)$. Common terms require no identification (Corollary 1).

From identification to estimation. Each mover's execution bit observes the all-success event of its dependency set D. Deduplicating identical sets within a step, $K_D \geq 1$ repeated observations estimate its product by a Bernoulli sample mean. For failure probability $\delta \in (0, 1)$, adaptive proposal selection admits simultaneous radius $\sqrt{\log(2nT^2/\delta)/(2K_D)}$ over all observed sets and steps, with probability at least $1 - \delta$. Weighted sums certify separated immediate-value comparisons (Appendix O). These certificates evaluate complete plans; the implementation below instead estimates individual parameters from exposed labels.

4 Putting reliability into a planner

We now use the path-feedback model in a lightweight controller. MCRP maintains one Beta estimate per robot and applies a local release preference during recursive planning. The complete-plan theory identifies what information a value comparison needs; the implementation uses individually exposed labels and partial-dependency products to refine candidate moves.

Learn only exposed actuator outcomes. Initialize Beta counts $c_i = f_i = 1$. For an attempted acyclic move, the target is clear if it was initially empty or its occupant actually moved. On this event, increment c_i if i moved and f_i otherwise. Censor the remaining nonmoves and all cycle observations. The next-step estimate is $\widehat{q}_i = c_i/(c_i + f_i)$. On an acyclic path, target clearance depends only on downstream actuator outcomes, independently of the focal actuator. Accepted labels therefore retain mean q_i even when past feedback determines the proposal. Each arm learns from its own trajectory. Appendix A formalizes this adaptive-sampling statement; Section 5.4 additionally uses cycle observations.

Refine equal-guidance choices. PIBT assigns actions recursively. For a candidate target, follow committed successor decisions from its occupant and multiply the posterior means of the distinct robots in that committed portion, stopping at an unassigned frontier or a repeated robot. A vacancy has score $\rho = 1$ and a committed wait score zero. This partial-dependency score is a local planning key. It is consulted only when inherited guidance distances tie, leaving task priorities, distance improvements, reservation, swaps, and guidance generation intact.

Share the comparison grid and protect vacancies. For a score $x \in [0, 1)$, use $B = 16$ and

$$Q_U(x) = \lfloor Bx \rfloor + \mathbf{1}\{U < \mathrm{frac}(Bx)\}. \tag{3}$$

Here frac is the fractional part; one keyed pseudorandom $U \in [0, 1)$ is shared throughout a robot's candidate sort, and larger keys are preferred. The unit endpoint has its own key $B + 1$; equal keys fall back to the inherited tie rule. This monotone grid preserves gaps of at least $1/B$ and retains a smaller gap d strictly with probability Bd under uniform U. An empty target always wins.

The interface has a local fidelity guarantee. If e_i bounds the error in robot i's estimate, the selected candidate's true partial-release score is within $1/B + \sum_{i \in D_*} e_i + \sum_{i \in D_{\mathrm{sel}}} e_i$ of the best true score. Here D_* and D_{sel} are the committed robot sets of the best and selected candidates. This separates grid resolution from estimation error. Appendix A gives simultaneous confidence radii from accepted-label counts; Appendix B specifies the implementation.

Table 2: **Locked learned-policy evaluation.** Paired task differences over six seed clusters; each map has 18 seed/profile contexts. Intervals use seed-cluster means.

Contrast	Map	Mean task gain	95% interval	Positive rows
MCRP − Guided25	Both	$+249.78$	$[124.01, 375.54]$	30/36
	Room-8	$+325.94$	$[32.24, 619.65]$	14/18
	Room-16	$+173.61$	$[105.47, 241.75]$	16/18
MCRP − learned fleet mean	Both	$+189.67$	$[84.02, 295.32]$	30/36
	Room-8	$+250.72$	$[65.78, 435.67]$	14/18
	Room-16	$+128.61$	$[79.83, 177.40]$	16/18

5 Experiments: follow the information through the loop

We organize the experiments around three questions: does the learned interface improve tasks, what happens when estimation error is removed, and how does immediate execution value relate to continuation? Task performance is the number completed in 3,000 physical steps. Intervals are descriptive 95% Student-t intervals over seed/start means; repeated fault tapes within a start are averaged before forming the across-start interval.

5.1 Learning reliability improves dense-room throughput

Locked evaluation. After fixing MCRP, we evaluated six previously unused seed clusters on MovingAI `room-64-64-8` and `room-64-64-16`, each with 1,000 robots (Stern et al., 2019). Each identity is fragile with probability 0.2. Failure probabilities for fragile/ordinary robots are $(0.50, 0.01)$ in the high profile and $(0.20, 0.005)$ in the low profile; the wide profile draws them from $[0.35, 0.65]$ and $[0.001, 0.019]$. Five methods (MCRP, learned fleet mean, Guided25, and the two HR comparators below), two maps, three profiles, and six seed clusters yield 180 trajectories. A context is a map/profile/seed combination; a seed-cluster mean averages its map/profile contexts. These original evaluations use the same chronological goal-generation law across policies. Appendix C records the frozen protocol and every row.

MCRP gains **249.78 tasks per context**, a macro-average **9.76%**, over Guided25 (Table 2). All six aggregate seed means and 30 of 36 context differences are positive. Against a control that performs the same censored updates but supplies their fleet mean to every robot, the gain is **189.67 tasks**. The information contrast is positive on both maps.

The unchanged author hindrance/regret PIBT preference implementations, with 63/255 resamples (HR63/HR255), (Okumura & Nagai, 2025) provide additional fixed-step comparisons. MCRP completes more tasks in all 36 paired contexts against each. These are fixed-step comparisons; Appendix D reports native call times and separates backbone gains from the learned increment. A feedback-by-ranking development ablation, temporal effects, and fragile/ordinary service accounting are retained in Appendices I and F.

5.2 Exact information reveals a context-dependent reversal

To remove estimation error from the question, we supply exact q_i or exact $\bar{q}$ to the *same* guarded release comparator. Both arms use the low-failure profile above and share the physical fleet, planner binary, and execution contract. Six unused warehouse starts are opened by a prewritten development rule; the room counterpart is fixed before reading warehouse outcomes. Each start has twelve paired fault tapes and robot-indexed goal sequences, verified by independent replay.

Figure 2 shows the result. In room-8 with 1,000 robots, exact identity input gains 347.53 tasks, or 7.96%, with interval $[234.66, 460.39]$. In the warehouse with 1,800 robots, it loses 2588.25 tasks, or 13.09%, with interval $[-3743.77, -1432.73]$. Every room start mean is positive and every warehouse start mean negative. Across individual fault tapes, the corresponding counts are 63/72 positive in room and 54/72 negative in warehouse.

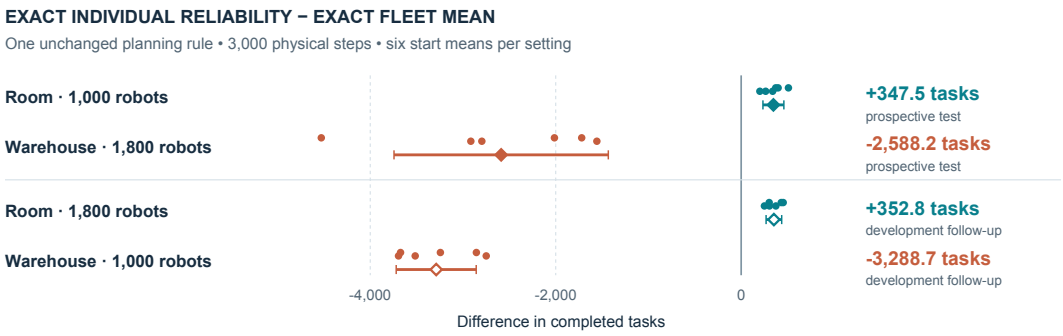


Figure 2: **The same reliability input has opposite task effects across settings.** Exact individual minus exact fleet-mean reliability, under one fixed planning rule. Small circles are six start means, each averaging twelve matched fault tapes; diamonds and bars show the mean and descriptive 95% t_5 interval. Filled diamonds mark prospective tests; open diamonds mark subsequent development cells. The sign persists at both fleet sizes.

Table 3: **Accounting for realized motion.** Exact individual minus fleet mean, in percentage points of robots per tick; $\Delta R = \Delta A - \Delta F - \Delta C$. All 3,000 ticks, six starts and twelve paired tapes per start. Descriptive start-level intervals appear in Appendix Q.

Setting	Proposals ΔA	$-\Delta F$	$-\Delta C$	Realized ΔR
Room / 1,000	-0.248	$+0.262$	$+1.648$	$+1.662$
Warehouse / 1,800	-4.713	$+0.262$	-0.152	-4.603

Does fleet size explain the reversal? We fill the other two map/fleet cells in a subsequent development comparison. Room with 1,800 robots gains 352.82 tasks; warehouse with 1,000 loses 3288.69. All six start means retain the setting's sign in both added cells. The original room/warehouse free-cell occupancy fractions are already close, $0.309/0.316$. The four-cell comparison shows the pattern across both tested fleet sizes (Appendix I).

Does concentration at warehouse stations explain it? A paired task-law intervention keeps the warehouse's obstacles, 1,800 initial robot positions, reliabilities, and fault tapes, but replaces station destinations with free-cell destinations. The identity-input difference becomes -1606.35 tasks, still negative at all six starts. The interaction relative to the station task law is $+981.90$ tasks, with interval $[-282.50, 2246.31]$. A separate geometric intervention opens shelf-top bypasses while matching four arms under a common injected goal stream. Its interaction is $+2191.58$ tasks with interval $[-997.72, 5380.88]$ (Appendix I). These interventions map how the effect changes with tasks and available space.

5.3 Where the motion difference develops

We revisit every tick of the completed exact-input tests in a post-outcome descriptive analysis. Proposed movers A split into direct actuator failures F, successful actuators blocked by failed dependencies C, and realized movers R:

$$R = A - F - C.$$

Table 3 gives this accounting for all 144 trajectory pairs across the two settings. The room gain accompanies fewer propagated stalls: $-\Delta C = +1.648$ percentage points of robots per tick, while proposal activity changes little. In warehouse, $\Delta A = -4.713$ points accounts for most of the realized-motion loss. Direct failures decrease in both settings. Cascade losses decrease in all six room start means; proposal activity decreases in all six warehouse start means.

The warehouse deficit develops over time (Figure 3). At the shared initial state, exact identity information changes approximately 197 of 1,800 robot actions, but changes the proposed-mover count by only -2.67 on average. Its mean proposal deficit grows from 1.55 movers per tick in the

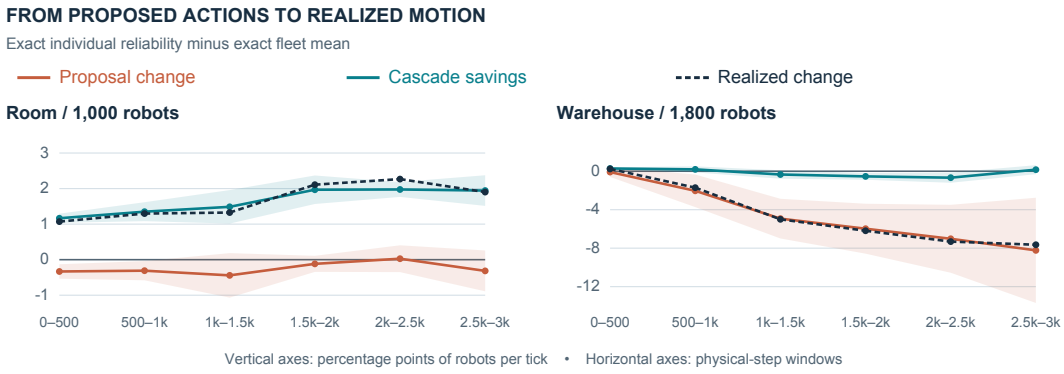


Figure 3: **The two settings diverge along different parts of the loop.** Panels use different vertical scales. Each point averages a fixed 500-step window; shading gives descriptive 95% t_5 intervals for proposals and cascade savings over six start means. Positive cascade savings mean fewer blocked movers. Room sustains cascade savings; warehouse gradually loses proposal activity. This post-outcome accounting describes the evolving trajectories.

first 500-step window to 148.09 in the last. Corresponding task differences change from $+11.9$ to -641.8 per window. Room instead saves cascade losses in every window. Thus the room advantage accompanies improved execution of proposals, whereas the warehouse deficit grows mainly through fewer proposals as the trajectories diverge. Appendix Q records definitions, uncertainty, and the complete readout.

5.4 Better prediction and better control separate

The full feedback law offers an estimator test. For independent incoming Beta marginals, write $M_{i,k} = \mathbb{E}[q_i^k]$ and $Q = \prod_{i \in C} M_{i,1}$. A failed cycle gives exact marginal moments

$$\mathbb{E}[q_i^k \mid C \text{ fails}] = \frac{M_{i,k} - M_{i,k+1} \prod_{j \in C \setminus \{i\}} M_{j,1}}{1 - Q}, \quad k = 1, 2. \tag{4}$$

Moment matching to Beta marginals yields a linear-in-cycle-size approximate filter. A successful cycle increments each success count. This uses both outcomes of feedback that MCRP censors.

On six fixed development trajectories (two initializations each in room, random, and warehouse), the cycle-aware filter reduces early Bernoulli cross entropy against the true reliabilities and fragile-robot estimation error in all six. Replacing only the learner in the controller also reduces own-trajectory cross entropy in all six contexts. Task differences against the original individual learner are nevertheless $+923/-232$ in room, $-454/+3996$ in random, and $-1382/+6640$ in warehouse. This paired prediction/control experiment puts the distinction directly in the learner's loop: the improved estimator changes the decisions and, with them, the states on which future rewards are earned. Complete results and the exact-likelihood benchmark are in Appendix O.

6 Why a better next step can create a worse future

The bridge from prediction to throughput is continuation value: the tasks a policy expects to complete after its next state. Fix a legal proposal. On a vacancy path, let E_i denote success of every robot downstream of i. This both exposes i's own outcome and allows it to change the next state. Let S' be that state, and S^+, S^- its coupled versions with i forced to succeed or fail. For a fixed bounded value function W of observable next-state information, holding all other randomness coupled gives

$$\frac{\partial}{\partial q_i} \mathbb{E}_q[W(S')] = \Pr_q(E_i)\, \mathbb{E}_q[W(S^+) - W(S^-) \mid E_i]. \tag{5}$$

Here the derivative varies the physical success probability under a fixed proposal and value function. The exposure probability controls *how often* this reliability matters; the conditional value difference controls *what its consequence is*. This Bernoulli conditioning identity connects the label-exposure event to the event that makes an actuator pivotal (Appendix A.3).

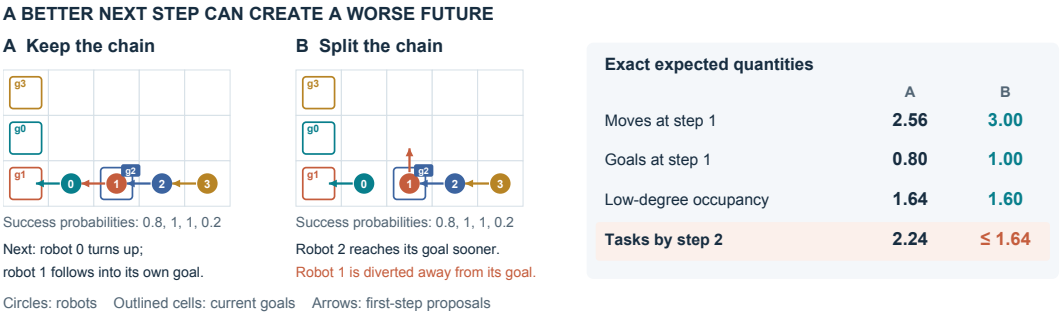


Figure 4: **Three better one-step scores, fewer tasks after two steps.** Four robots initially form a chain toward the left vacancy. Sending robot 1 to a side cell splits the chain and helps robot 2 reach its goal immediately, but puts robot 1 three moves from its own goal. For $q = (0.8, 1, 1, 0.2)$, the edit improves immediate goals, moves, and low-degree occupancy; the original plan with a legal continuation completes more tasks by step two. Arrows show first-step proposals; colored goal markers identify the first three robots' goals.

A complete two-step construction. Figure 4 makes the continuation concrete. On a 5×3 empty grid, robots numbered $0, 1, 2, 3$ at $(1, 0), \ldots, (4, 0)$ move left toward a vacancy. Their goals are $(0, 1), (0, 0), (2, 0), (0, 2)$ and their success probabilities are $(p, 1, 1, 0.2)$, with $p \in (1/2, 1)$. Splitting the chain by sending robot 1 to $(2, 1)$ increases expected moves from $3.2p$ to $p+2.2$, raises immediate expected goal completions from p to 1, and lowers expected occupancy of cells with at most two free neighbors from $1 + 0.8p$ to $0.8 + p$.

Yet the original action has a legal continuation completing $2p + p^2$ expected tasks by step two. After the split, robot 1 is too far from its goal; with robot 2's next goal chosen at least two moves away, *every* continuation completes at most $1 + p^2$. The original plan wins by at least $2p - 1$. At $p = 0.8$, this is 2.24 versus at most 1.64 expected tasks. By continuity, the reversal persists when the two unit reliabilities are slightly reduced. Proposition 3 in Appendix H gives the proof; the evidence archive includes an independent actuator-mask enumeration.

A plan-level empirical test. In an oracle diagnostic, we also compare complete mean-q and individual-q proposals at each state, accepting the latter only if its exact expected current-goal count and move count are no smaller, and its expected occupancy of degree-at-most-two cells no larger. Under a common dual-planner randomization contract, this shield gains 160.17 tasks over its mean-q arm in two exposed room starts but loses 922.50 in two warehouse starts; its development advancement rule is not met. A later single-plan intervention in two selected states uses 64 fresh paired future fault sequences per state and common shield continuation. Its task differences have intervals crossing zero in both settings. These two experiments distinguish a repeated decision rule from a single decision's downstream effect (Appendices J and K).

7 Related Work

Learning under structured feedback. Cascading bandits observe nonclicks followed by the first click, if any, censoring later items (Kveton et al., 2015); network tomography studies latent parameters inferred from path measurements (Chen et al., 2007). Linear partial monitoring formalizes decision-relevant observability (Kirschner et al., 2023). Our observation graph consists of legal simultaneous-motion paths and cycles. The resulting product law gives a row-space criterion for complete-plan task values and permits common unidentified terms to cancel in a comparison. Our confidence results specialize standard concentration and sequential error allocation to these motion observations.

Coordination under uncertain execution. PIBT (Okumura et al., 2022) and traffic-guided lifelong planning (Chen et al., 2024) provide the backbone. Reliability and delay have been studied through stochastic-delay MAPF (Ma et al., 2017), time-independent execution (Okumura et al., 2021), travel-

time learning (Kita et al., 2023), failure attribution (Natan et al., 2023), and robust execution (Liu et al., 2026). FICO explicitly models primary stalls and their secondary propagation along dependencies (Li et al., 2025). Delay-aware planning and execution (Zhang et al., 2025), closed-loop conflict-based search (Li et al., 2026a;b), learned priorities (Zheng et al., 2026), and guidance optimization (Zhang et al., 2024; Zang et al., 2025; Jiang et al., 2026a) address complementary planning interfaces. We focus on learning unknown individual reliability from positions and experimentally separating its predictive value from its effect inside a fixed control rule.

8 Conclusion and limitations

Reliability information has two jobs in lifelong coordination: explain observed motion and support decisions that shape future traffic. This work connects them through an exact feedback model, value-contrast identifiability, a learned release interface, and controlled exact-input tests. The room/warehouse reversal, the trajectory accounting, and the two-step construction make the central lesson tangible: a planner needs both the information that distinguishes its actions and a way to value their continuation.

The confirmed learned-method gain covers two dense room maps; unchanged transfer has negative means on random/warehouse, larger-room replication is inconclusive, and lower-fleet results are mixed. Exact-input tests concern a fixed heuristic, not optimal information value: an optimal policy can always ignore additional information. Settings differ in maps, task configurations, and identity assignments. The interventions and post-outcome motion accounting do not isolate a unique causal mechanism. Failures are independent and stationary. Our confidence certificates concern immediate value and require repeat exposure; in the twelve-trace development panel, fewer than 3% of multi-robot goal terms have twenty prior observations of the same dependency set. The cycle filter is approximate; the tested control extensions have no confirmed cross-setting gain. Reported task comparisons fix physical steps, rather than establish equal-time superiority; direct matched comparisons to FICO and other recent closed-loop methods remain open. These boundaries motivate control objectives that incorporate continuation rather than increasingly precise optimization of a one-step proxy.

Ethics statement

This study uses simulation and contains no human participants or personal data. Real-world deployment requires application-specific safety and service-fairness evaluation.

Reproducibility statement

The appendices provide the algorithm, proofs, complete experimental protocols, paired results, and computational costs. The anonymous combined code/evidence archive contains the frozen learned-method implementation, compact audited trajectories and summaries, later study code, and scripts that reproduce the reported comparisons. A companion motion-audit supplement contains the new paired summaries and their analysis code. The combined archive also includes exact-rational checks of the small constructions. Full raw trajectories are retained in the research archive.

AI use statement

Generative AI tools assisted with hypotheses, experimental design, method implementation and testing, mathematical claims and proofs, result interpretation, conceptual framing, literature discovery, figures and tables, and manuscript drafting and editing. The reproducibility materials document the computational checks performed on the resulting code, experiments, and constructions. The authors have independently verified all content in the paper, item by item.

REFERENCES

Aiyou Chen, Jin Cao, and Tian Bu. Network tomography: Identifiability and fourier domain estimation. In *Proceedings of IEEE INFOCOM*, pp. 1875–1883, 2007. doi: 10.1109/INFCOM.2007.218.

Zhe Chen, Daniel Harabor, Jiaoyang Li, and Peter J. Stuckey. Traffic flow optimisation for lifelong multi-agent path finding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 2024.

He Jiang, Jingtian Yan, Yulun Zhang, Yimin Tang, Tanishq Duhan, Rishi Veerapaneni, Guillaume Sartoretti, and Jiaoyang Li. Search-aided joint agent-environment reinforcement learning for robust lifelong multi-agent path finding with rotations. *arXiv preprint arXiv:2608.05588*, 2026a.

Zixiang Jiang, Yulun Zhang, Rishi Veerapaneni, and Jiaoyang Li. Planning over MAPF agent dependencies via multi-dependency PIBT. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2026b. arXiv:2603.23405.

Johannes Kirschner, Tor Lattimore, and Andreas Krause. Linear partial monitoring for sequential decision making: Algorithms, regret bounds and applications. *Journal of Machine Learning Research*, 24(346):1–45, 2023.

Atsuyoshi Kita, Nobuhiro Suenari, Masashi Okada, and Tadahiro Taniguchi. Online re-planning and adaptive parameter update for multi-agent path finding with stochastic travel times. *arXiv preprint arXiv:2302.01489*, 2023.

Branislav Kveton, Csaba Szepesvári, Zheng Wen, and Azin Ashkan. Cascading bandits: Learning to rank in the cascade model. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 767–776, 2015.

Jiarui Li, Alessandro Zanardi, Federico Pecora, Runyu Zhang, and Gioele Zardini. FICO: Finite-horizon closed-loop factorization for unified multi-agent path finding. *arXiv preprint arXiv:2511.13961*, 2025.

Jiarui Li, Federico Pecora, Runyu Zhang, and Gioele Zardini. Adaptive-horizon conflict-based search for closed-loop multi-agent path finding. *arXiv preprint arXiv:2602.12024*, 2026a.

Jiarui Li, Runyu Zhang, and Gioele Zardini. Certificate-driven closed-loop multi-agent path finding with inheritable factorization. *arXiv preprint arXiv:2604.00428*, 2026b.

Yihao Liu, Xueyan Tang, Wentong Cai, and Jingning Li. Robust and effective multi-agent path execution with timing uncertainty. *Artificial Intelligence*, 358:104586, 2026. doi: 10.1016/j.artint.2026.104586.

Hang Ma, T. K. Satish Kumar, and Sven Koenig. Multi-agent path finding with delay probabilities. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017. doi: 10.1609/aaai.v31i1.11035.

Avraham Natan, Roni Stern, and Meir Kalech. Blame attribution for multi-agent path finding execution failures. In *Proceedings of the 26th European Conference on Artificial Intelligence*, pp. 1763–1770, 2023. doi: 10.3233/FAIA230462.

Keisuke Okumura and Hiroki Nagai. Lightweight and effective preference construction in PIBT for large-scale multi-agent pathfinding. In *Proceedings of the International Symposium on Combinatorial Search*, 2025. arXiv:2505.12623.

Keisuke Okumura, Yasumasa Tamura, and Xavier Défago. Time-independent planning for multiple moving agents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021.

Keisuke Okumura, Manao Machida, Xavier Défago, and Yasumasa Tamura. Priority inheritance with backtracking for iterative multi-agent path finding. *Artificial Intelligence*, 310:103752, 2022. doi: 10.1016/j.artint.2022.103752.

Roni Stern, Nathan Sturtevant, Ariel Felner, Sven Koenig, Hang Ma, Thayne Walker, Jiaoyang Li, Dor Atzmon, Liron Cohen, T. K. Satish Kumar, Roman Barták, and Eli Boyarski. Multi-agent pathfinding: Definitions, variants, and benchmarks. In *Proceedings of the International Symposium on Combinatorial Search*, 2019.

Hongzhi Zang, Yulun Zhang, He Jiang, Zhe Chen, Daniel Harabor, Peter J. Stuckey, and Jiaoyang Li. Online guidance graph optimization for lifelong multi-agent path finding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 14726–14735, 2025. doi: 10.1609/aaai.v39i14.33614.

Yue Zhang, Zhe Chen, Daniel Harabor, Pierre Le Bodic, and Peter J. Stuckey. Concurrent planning and execution in lifelong multi-agent path finding with delay probabilities. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 2025. doi: 10.1609/aaai.v39i22.34506.

Yulun Zhang, He Jiang, Varun Bhatt, Stefanos Nikolaidis, and Jiaoyang Li. Guidance graph optimization for lifelong multi-agent path finding. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pp. 311–320, 2024. doi: 10.24963/ijcai.2024/35.

Han Zheng, Yining Ma, Brandon Araki, Jingkai Chen, and Cathy Wu. Learning-guided prioritized planning for lifelong multi-agent path finding in warehouse automation. *Journal of Artificial Intelligence Research*, 2026. doi: 10.1613/jair.1.20611. arXiv:2603.23838.

A PROOFS OF LOCAL STATEMENTS

A.1 IDENTIFIABLE LABELS

Let $\mathcal{H}_t$ be the observation history and A_t the proposed joint action. Write C_i for target clearance, F_i for own actuator success, and $Y_i = C_i F_i$ for executed motion on an acyclic path. Fix $\mathcal{H}_t, A_t$. On an acyclic dependency path, C_i is determined by the downstream actuator outcomes, not by the focal outcome F_i. Conditional independence therefore gives

$$\mathbb{E}[C_i(Y_i - q_i) \mid \mathcal{H}_t, A_t] = \mathbb{E}[C_i(F_i - q_i) \mid \mathcal{H}_t, A_t] = \mathbb{E}[C_i \mid \mathcal{H}_t, A_t]\mathbb{E}[F_i - q_i \mid \mathcal{H}_t, A_t] = 0.$$

Equivalently, conditional on $C_i = 1$, observed motion equals the focal Bernoulli outcome. A blocked move has $C_i = 0$ and contributes no label. In a cycle, however, observing that the target occupant moved can itself imply the focal robot succeeded; excluding all cycle outcomes avoids that selection.

This is an adaptive-sampling statement, not an i.i.d. statement about accepted times. It remains valid when the controller chooses actions from the complete past, provided own outcomes are stationary and conditionally independent.

For a fixed directed execution cycle, the complete position observation is all-move if every member succeeds and all-wait otherwise. Its Bernoulli parameter is only $\prod_{i \in C} q_i$. A legal four-neighbor grid cycle has at least four movers because edge swaps are excluded. For a four-robot rotation on a 2×2 square, reliability vectors $(0.8, 0.6, 0.9, 0.9)$ and $(0.75, 0.64, 0.9, 0.9)$ both give all-move probability 0.3888 while assigning different reliabilities to two identities. Thus repeated observations of that cycle cannot identify the individual q_i. Other, acyclic proposals may identify them; the claim is confined to cycle-only position feedback.

A.2 FIXED-CHAIN RELEASE

Order a fixed acyclic chain from the robot targeting an empty cell toward the root. Robot i_1 moves exactly when $F_{i_1} = 1$. Inductively, robot i_j moves exactly when its own outcome and every downstream outcome succeed, so its movement indicator is $\prod_{k=1}^{j} F_{i_k}$. Taking expectations and using conditional independence gives $\prod_{k=1}^{j} q_{i_k}$. Summing indicators gives expected executed moves; setting $j = L$ gives root release. The proof holds for a fixed proposal chain, not for candidate-dependent future recursion.

A.3 FEEDBACK EXPOSURE AND VALUE PIVOTALITY

Proposition 1 (Exposure-pivotality identity)**.** *Fix a legal proposal and independent Bernoulli actuator attempts. On a vacancy-terminated path ordered $i_1, \dots, i_L$ from the empty endpoint, let E_h be the event that robots $i_1, \dots, i_{h-1}$ all succeed. For any bounded next-state value W that depends on realized positions, task assignments and position-only planner feedback but not on an unobserved actuator bit, let S^+ and S^- denote next states under coupled other randomness and forced success or failure of robot i_h. Then*

$$\frac{\partial}{\partial q_{i_h}} \mathbb{E}_q[W(S')] = \underbrace{\prod_{j<h} q_{i_j}}_{\Pr(E_h)} \mathbb{E}_q[W(S^+) - W(S^-) \mid E_h].$$

Here E_h is exactly the event on which robot i_h's own actuator outcome is identifiable from realized positions. For a directed execution cycle C, the analogous pivotality factor is $\prod_{j \in C \setminus \{i\}} q_j$, although a failed cycle does not generally reveal which individual failed.

Proof. Hold all other actuator outcomes and task-generator randomness fixed. The expectation over F_{i_h} is affine in q_{i_h}, so differentiating yields the expectation of the forced-success minus forced-failure value. If any downstream member fails, robot i_h and all upstream members remain blocked regardless of F_{i_h}; realized positions, feedback and induced tasks therefore agree under the two forced outcomes. Otherwise its target clears independently of F_{i_h}, its bit is individually exposed, and the value difference may be nonzero. Independence gives $\Pr(E_h) = \prod_{j<h} q_{i_j}$. In a cycle, the two forced outcomes differ only when every other cycle member succeeds. □

For a fixed additive one-step movement reward $W(S') = \sum_{k=1}^{L} w_k \mathbf{1}\{i_k \text{ moves}\}$, the identity reduces to

$$\frac{\partial}{\partial q_{i_h}} \mathbb{E}_q W(S') = \Big(\prod_{j<h} q_{i_j}\Big) \sum_{k=h}^{L} w_k \prod_{h<j\le k} q_{i_j}.$$

Taking w_k as the indicator of a currently targeted goal yields a nonnegative immediate-goal derivative. Taking W to include subsequent planner and task value need not preserve that sign: changing the first position can change later routes, assignments and goals. The exposure factor bounds the magnitude of a one-step value sensitivity by $\mathrm{osc}(W)\Pr(E_h)$ but does not determine its sign. This identity does not turn passive exposure into a sufficient long-run planning rule. Differentiating an expectation affine in one Bernoulli parameter is elementary; the motion-specific point is that its pivotal event on a vacancy path equals the event on which position feedback reveals that robot's own actuator outcome, while cycles retain only a product observation.

A.4 Shared-grid stability

For $x, x' \in [0,1)$, $Q_U(x)$ changes under a perturbation to x' of magnitude ϵ only when a grid boundary lies in the swept interval. A uniformly shifted grid of width $1/B$ hits that interval with probability at most $\min(1, B\epsilon)$. A union bound yields $\min(1, 2B\epsilon)$ for two scores. For products, replace factors one at a time:

$$\left|\prod_{i=1}^{L} q_i - \prod_{i=1}^{L} \widehat{q}_i\right| \le \sum_{j=1}^{L} |q_j - \widehat{q}_j| \prod_{i<j} q_i \prod_{i>j} \widehat{q}_i \le \sum_{j=1}^{L} |q_j - \widehat{q}_j|.$$

The protected unit endpoint is handled separately and never collides with a nonunit key.

A.5 Adaptive finite-sample fidelity

The acceptance indicator for a focal acyclic attempted move is determined by the current downstream failures and the pre-step action, so it is independent of that robot's current Bernoulli actuator outcome conditional on the past. The accepted labels can consequently be coupled, in acceptance order, to a Bernoulli(q_i) sequence; whether its kth member is observed by T can depend on earlier labels. For each *latent* prefix $k \ge 1$, Hoeffding's inequality and a union bound over the nT robot/count pairs give, simultaneously for every i and $1 \le k \le T$ with probability at least $1-\delta$,

$$\left|\frac{S_{i,k}}{k} - q_i\right| \le \sqrt{\frac{\log(2nT/\delta)}{2k}},$$

where $S_{i,k}$ is the number of successes in the first k coupled labels. The bound applies whenever that prefix has been observed, without conditioning on the adaptive realized count. Beta$(1,1)$ smoothing gives

$$\left|\frac{1+S_{i,k}}{2+k} - \frac{S_{i,k}}{k}\right| = \frac{|k-2S_{i,k}|}{k(k+2)} \le \frac{1}{k+2}.$$

At $k=0$, the estimate is $1/2$ and its absolute error is at most $1/2$. Thus for $N_i \ge 1$ accepted labels set $e_i = \sqrt{\log(2nT/\delta)/(2N_i)} + 1/(N_i+2)$, and set $e_i = 1/2$ for $N_i = 0$. These radii bound all posterior-mean errors simultaneously through step T.

For any fixed exposed candidate chain P, the telescoping product bound above limits the difference between its estimated and true partial-release scores by $\sum_{i\in P} e_i$. For two estimated nonendpoint scores separated by at least $1/B$, a shared grid assigns strictly ordered keys for *every* shift U; the protected empty target also always receives the maximum key. Thus if P_* has the largest true score and P_{sel} wins the local key comparison, their estimated-score gap is less than $1/B$. Adding the two score-estimation errors proves the stated local regret bound. This statement conditions on a fixed candidate list and exposed recursion state; it does not bound which complete PIBT action will survive backtracking or its continuation value.

A.6 REPEATED-SERVICE ACCOUNTING

For proposal rate α and conditional execution rate σ, realized movement is $\mu = \alpha\sigma$. Adding and subtracting the two symmetric cross-terms gives

$$\alpha_I\sigma_I - \alpha_G\sigma_G = \frac{\sigma_I + \sigma_G}{2}(\alpha_I - \alpha_G) + \frac{\alpha_I + \alpha_G}{2}(\sigma_I - \sigma_G).$$

This is an exact two-factor decomposition. In the stylized unit-service case, one attempt assigned to robot i has expected reward q_i; linearity of expectation gives $\epsilon(q_i - q_j)$ after moving assignment mass ϵ from j to i. The argument stops at service reward. It does not cover multi-step tasks, policy-dependent eligibility, dependency closures, or traffic-state effects.

B ALGORITHM AND IMPLEMENTATION CONTRACT

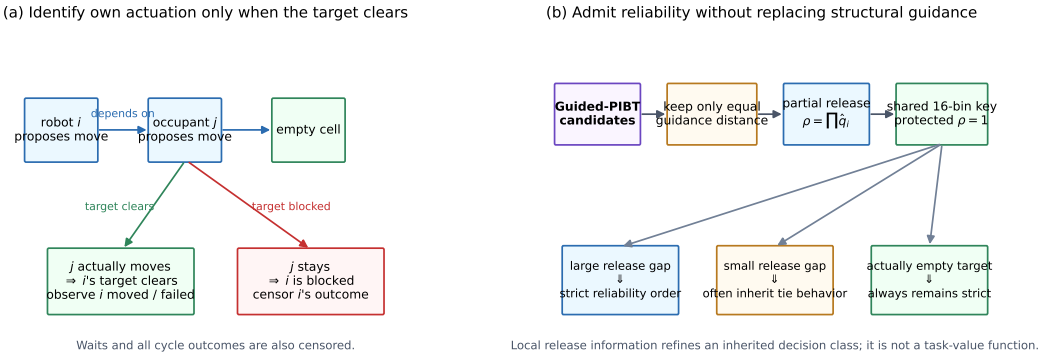


Figure 5: MCRP keeps unbiased individual Beta labels from vacancy paths and refines equal-guidance candidates with a coarsened partial-release score. Other Guided-PIBT logic is unchanged.

```
Initialize c_i = f_i = 1 for each robot i
for physical step t = 0, 1, ... do
  From the previous proposal and realized motion:
    for each acyclic attempted move i -> v:
      clear <- (v was empty) or (its previous occupant moved)
      if clear: increment c_i if i moved, otherwise f_i
    censor blocked moves, waits, and all cycle outcomes
  qhat_i <- c_i / (c_i + f_i)
  Run the unchanged Guided-PIBT recursion
  whenever candidate guidance distances tie:
    rho <- product of qhat along exposed committed successors
    key <- 17 if target is empty, else shared-Q_U(rho), B = 16
    compare keys; on equality use inherited tie behavior
  Execute the proposal through fixed-point cascading aborts
end for
```

The shared random value is keyed by the physical step, robot, and candidate-sort invocation. It is shared across candidates within that invocation. The method does not modify task-age priority, guidance distance, candidate generation, recursion, reservation, swap handling, task generation, or the physical executor.

C FINAL PROTOCOL AND COMPLETE PRIMARY ROWS

Matrix. The final matrix crosses two maps, six seed clusters (100–105), three profiles, and five methods. Every trajectory lasts 3,000 physical steps. This yields $2 \times 6 \times 3 \times 5 = 180$ trajectories and 540,000 physical steps. Before the matrix, ten 100-step integration smokes used development seed 943; their task counts are excluded. There was one final matrix and one audit.

Pairing. Within a seed/map/profile context, methods share initial positions, initial goals, robot identities and reliability probabilities, and the complete tickwise actuator-success tape. Profiles and maps are repeated paired contexts within seed clusters, not independent replicates. Subsequent goals come from the same chronological task law but are not forced to match after policies complete different numbers of tasks.

Estimands. The primary estimand is individual minus Guided25, equally weighting the six map/profile contexts inside each seed. Per-map effects are secondary. The information estimand is individual minus fleet mean. External contrasts retain both candidate-minus-HR and Guided25-minus-HR so the family effect is visible. Descriptive 95% Student-t intervals use the six seed-cluster means and five degrees of freedom. They describe replication variability; they are not thresholded gates or corrections for multiple comparisons.

Table 4 contains every primary final row. It explains why the overall positive result coexists with the negative room-8 seed-101 aggregate: that cluster contains two negative profiles and one positive profile.

Table 4: All final individual-minus-Guided25 completed-task differences.

	room-8			room-16		
Seed	high	low	wide	high	low	wide
100	+388	+282	+184	+317	+347	+82
101	-185	+296	-319	+115	+453	+182
102	+682	+108	+691	+217	+58	+184
103	+563	-79	-4	+150	+483	-102
104	+230	+138	+649	-34	+254	+72
105	+437	+1129	+677	+28	+191	+128

Table 5: Seed-cluster means for the two primary information contrasts.

Contrast	Map	100	101	102	103	104	105
Individual − Guided25	room-8	284.67	-69.33	493.67	160.00	339.00	747.67
Individual − Guided25	room-16	248.67	250.00	153.00	177.00	97.33	115.67
Individual − mean	room-8	359.67	-14.00	289.00	93.67	319.00	457.00
Individual − mean	room-16	216.67	108.67	85.33	101.00	127.33	132.67

D TASK COUNTS AND COMPUTATION

Table 6: Final mean completed tasks and attribution of the external-baseline gap. Each cell averages 18 seed/profile contexts.

Map	HR63	HR255	Guided25	Individual	Ind.-Guided	Guided-HR63
room-8	2220.00	2237.94	3233.89	3559.83	+325.94	+1013.89
room-16	1644.06	1657.00	2115.11	2288.72	+173.61	+471.06

HR63 is approximately 3.7–5.6 times faster per native call than the proposed method. HR255 has similar cost on room-8 and is faster on room-16. The fixed 3,000-step experiment cannot support equal-time superiority or Pareto dominance.

E COMMON-CONTRACT PIBT-D DEVELOPMENT COMPARISON

The authors' clean PIE-D commit is run with mapfPlanner3, delayPolicy3, commit length three, persistent within-window delay state, and four revealed future tasks. Guided25, individual and PIBT-D share the map, 1,000-agent start, fixed per-agent future-task tapes, complete actuator tape, and 300-step physical horizon. Planning overruns are measured rather than converted into extra physical

Table 7: Native call time in milliseconds (mean across final trajectories). These are measured costs, not deadlines or a continuous compute frontier.

Map	Guided25	Individual	Fleet mean	HR63	HR255
room-8	42.50	42.52	42.36	11.46	42.58
room-16	66.23	66.56	67.17	11.82	43.74

wait ticks. Table 8 is therefore a matched fixed-step development comparison, not an equal-wall-clock or untouched-timing result.

Table 8: Completed tasks in the audited common-contract development runs.

Map	Profile/seed	Guided25	Individual	PIBT-D	Ind.–PIBT-D
room-8	high/943	431	496	193	+303
room-8	low/943	698	722	399	+323
room-8	wide/943	437	463	211	+252
room-16	high/943	512	527	232	+295
room-16	low/943	684	714	363	+351
room-16	wide/943	533	520	241	+279
room-8	high/101	436	474	228	+246
room-16	high/101	512	514	227	+287

Individual and Guided25 exceed PIBT-D in all eight contexts. PIBT-D consumes 15.2–22.1 times individual's summed planner time. A descriptive exact decomposition of tasks = moves × tasks/move attributes 72–118% of the individual–PIBT-D gap to the movement-count term; conversion per realized move is comparable. The policies visit different states, so this is arithmetic attribution rather than causal mediation. These known development seeds and the shorter fixed-task-tape contract are not pooled with the reserved final confirmation.

F Repeated-service and one-action diagnostics

For identity group g and policy π, let α_g^π be proposed moves per robot per tick, σ_g^π their execution fraction, and $\mu_g^\pi = \alpha_g^\pi \sigma_g^\pi$ realized moves. Then

$$\mu_g^I - \mu_g^G = \frac{\sigma_g^I + \sigma_g^G}{2}(\alpha_g^I - \alpha_g^G) + \frac{\alpha_g^I + \alpha_g^G}{2}(\sigma_g^I - \sigma_g^G). \tag{6}$$

This exact accounting identity separates proposal allocation from conditional execution, but not causal mediation because each policy induces its own trajectory. In a unit-service bottleneck, shifting attempt mass ϵ from success probability q_j to q_i changes expected service by $\epsilon(q_i - q_j)$; MAPF violates that model's equal-value and fixed- eligibility assumptions, so this does not imply long-run task optimality.

The repeated-service analysis was fixed after final task and cohort results were known but before movement-allocation statistics were read. It uses all 36 accepted individual–Guided25 pairs, samples movement on ticks $0, 10, \ldots, 2990$, and reads every task completion. Reconstructed total and fragile task counts match the frozen audit in all 72 trajectories. Table 9 gives the group effects.

Table 9: Post-outcome repeated-service diagnostic. Stable movement reallocation does not imply a stable task shift in late room-16.

Map	Window	Proposal DiD	Move DiD	Move interval	Task DiD / interval
room-8	first	.0647	.0379	[.0349,.0409]	.1076 / [.0672,.1481]
room-8	late	.0723	.0402	[.0338,.0466]	.1239 / [.0315,.2163]
room-16	first	.0779	.0408	[.0325,.0491]	.1416 / [.1112,.1721]
room-16	late	.0895	.0296	[.0190,.0403]	.0209 / [-.0228,.0646]

The table reports ordinary-minus-fragile differences-in-differences: movement is per robot per tick and task completion is per robot per window. Intervals cluster profiles within the six seeds.

The allocation term in Eq. 6 explains 86.3%/79.1% of the room-8 first/late move DiD and 82.9%/91.3% on room-16. Sampled individual posteriors already separate ordinary from fragile identities in the first window (approximately .978 versus .599) and remain separated late (approximately .990 versus .601). These are policy-dependent total effects, not identified mediators or a fairness guarantee.

The one-action diagnostic uses 12 development states. Both arms share a learned-mean prefix through tick 999, apply one individual-versus-mean proposal at tick 1000, reset planner tie randomness after fixing that proposal, and return to Guided25 through tick 1199. Four common suffix tapes per state give 96 trajectories and 48 audited pairs. Room task effects are -1.46 at tick 1100 and $+1.46$ at tick 1200; random-map effects are $+2.04$ and $+14.96$. Every map/horizon interval crosses zero, state signs are mixed, and overall exact-release/task correlations are near zero. This development intervention does not support treating expected released movers as a reliable long-run task-value proxy in these tested states; it neither revises nor enlarges the frozen final scope.

G EXACT IMMEDIATE VALUE AND LIFELONG CONTROL BOUNDARY

For a decision state with no already satisfied outstanding goal, each valid complete proposal's unique-target dependency graph consists of vacancy-terminated paths and cycles: a waiter already targets its own cell, so another robot cannot target it. On a path reaching an initially empty cell, robot i moves precisely when all actuator draws from the endpoint through i succeed. On a cycle, all members move precisely when all cycle draws succeed. Summing each current-goal mover's release probability gives Eq. 2. The movement indicators share failures but expectation remains additive. Owner lookup, dependency traversal, and release-product propagation take expected $O(N)$ time with hash lookup for N robots, or linear time after the dependency arrays are built. The deployed score uses the motion-censored posterior; true reliability appears only in retrospective diagnostics.

On six already exposed random-map trajectories at seeds 107 and 110, a same-state selector compared the same three fixed candidate actions at every tick. Individual posterior means gained 30.91–41.49 true conditional expected current-goal completions over the fleet-mean posterior per 3,000 states. The six rows are three generating policies at each of only two seed clusters, not independent replications. A physical follow-up on room, random, and warehouse seeds 106 and 108 used three route candidates per tick with identical candidate generation and compute in its individual and pooled arms. Same-state individual-information gains were +13.09 to +32.04 true conditional expected goals per 3,000 states across six map/seed contexts, but the individual arm finished with more tasks in only three. This isolates one-step selection on fixed candidates, not the effect of changing an entire candidate generator or the expected lifelong policy value.

A narrower physical rule chose dependency-closed components of an alternate action when a posterior-screened goal-value gain and move-value nonloss were predicted. For two feasible complete proposals A, B, connect agents whenever their cross-proposal targets coincide or one proposal makes an agent target the other's current cell. Selecting A or B for each whole connected component preserves target injectivity, mover-into-waiter legality, and swap exclusion: any new conflict would connect the two agents, forcing the same valid source proposal. A failure-dependency path or cycle cannot cross components, so conditional expected current-goal and move values add by component. The graph has $O(N)$ edges; hash-based construction, traversal, and release scoring are expected linear time. The implemented union-find variant has practical near-linear cost, without a strict worst-case linear claim for that implementation.

The selected rule changes only 10.37–53.51 proposals per selected event on average by context. Of 570 executed selected events on six exposed map/seed trajectories, 547 had positive true conditional immediate goal gain; the six trajectory sums were all positive (+7.54 to +29.87). The continuing controller nevertheless gained tasks against its equal-compute control in only two of six contexts. The component construction preserves legal actions and exact additivity of one-step release values, but neither property bounds continuation value.

Finally, a single-action intervention used the same control history through the first eligible selected component, executed it once, then returned to the control's baseline action rule. Both arms computed both candidate actions throughout, shared initial states and tickwise fault tapes, and passed indepen-

dent physical, learning, selector, task and path audits. Table 10 reports every preselected exposed case.

Table 10: One changed joint action: exact true conditional immediate goal gain versus *realized* 3,000-step task difference. The two quantities have different expectation and horizon status; these rows do not determine the sign of the expected lifelong policy effect.

Map	Seed	Pulse tick	Changed robots	Immediate gain	Task difference
room-8	106	115	3	+.095	-57
room-8	108	102	17	+.117	-374
random	106	71	5	+.150	-930
random	108	38	5	+.178	-3354
warehouse	106	50	12	-.012	-6826
warehouse	108	55	18	+.335	-577

The five rows with positive immediate expected gains also have negative realized lifelong differences. The warehouse-106 row is negative on both quantities and is not counted among those five. The shared chronological task generator can assign different later goals to identities after the policies diverge. These development rows therefore provide observed closed-loop sensitivity, not a causal decomposition into congestion, planner memory, or task-draw effects. No reserved final seed was opened for these follow-ups.

H VACANCY SEGMENTATION: LOCAL DOMINANCE AND DEVELOPMENT PILOT

Consider a valid complete proposal with a vacancy-terminated dependency path $i_0, \dots, i_{L-1}$, ordered from the initially empty target upstream. For an internal pivot i_d, $1 \le d < L-1$, suppose its proposed target is not its current goal and an adjacent traversable cell c is initially empty and untargeted by the complete proposal. Replacing only i_d's target with c leaves the joint action legal: the target remains adjacent and unique, creates no swap or mover-into-waiter conflict, and turns the original path into two vacancy-terminated paths.

Proposition 2 (Fixed-state vacancy segmentation)**.** *Under conditionally independent actuator successes q_i, this edit weakly increases every affected robot's conditional movement probability and the expected number of realized moves and currently assigned goals completed next step. Its expected-move increment is*

$$\left(1 - \prod_{h=0}^{d-1} q_{i_h}\right) \sum_{k=d}^{L-1} \prod_{h=d}^{k} q_{i_h}.$$

Proof. For $k < d$, the release product remains $\prod_{h=0}^{k} q_{i_h}$. For $k \ge d$, it becomes $\prod_{h=d}^{k} q_{i_h}$, weakly larger because omitted factors lie in $[0, 1]$. Summing the differences yields the displayed increment. Other components are unchanged. Every original goal-targeting mover retains its target and weakly greater release probability; the pivot's original target was not its goal. The pivot's new target can only add an immediate goal. Distinct untargeted vacancies allow multiple disjoint-path edits under the same argument. The result is conditional on the current state and action; it makes no comparison of future positions, goals or policy value. □

Proposition 3 (Two-step reversal despite three one-step improvements)**.** *There is a legal vacancy split that strictly increases immediate expected moves and current-goal completions, and strictly decreases expected occupancy of static-degree-at-most-two cells after the step, yet has fewer expected completed tasks by step 2 than the original action followed by a legal continuation.*

Construction and proof. The missing continuation implication holds even for *expected* task counts, not merely for individual observed paths. Consider an empty 5×3 grid with vacancy $(0, 0)$ and robots at $(1, 0), (2, 0), (3, 0), (4, 0)$, numbered $0, 1, 2, 3$. Their current goals are $(0, 1), (0, 0), (2, 0), (0, 2)$. The baseline proposal moves them respectively into $(0, 0), (1, 0), (2, 0), (3, 0)$. The edit sends non-goal internal pivot 1 to the empty, untargeted $(2, 1)$ instead. Let their independent stationary success probabilities be $(p, 1, 1, 0.2)$ with $p \in (1/2, 1)$. If the learned pooled estimate is at most 0.8, its fourth-power path gate is below $1/2$; robot 2 is goal-targeting and robot 3 is an endpoint, leaving pivot 1 as the eligible split. This condition requires no true-reliability input. Baseline one-step expected

moves and current-goal completions are $3.2p$ and p; the edit strictly raises them to $p + 2.2$ and 1. In this 5×3 grid, the initial low-degree occupants consist only of robot 3 at the corner $(4, 0)$. Robot 0 enters the other corner $(0, 0)$ with probability p in both proposals, while robot 3 leaves its corner with probability $0.2p$ under the baseline and 0.2 under the edit. All other source and target cells of movers have static degree at least three. Consequently expected post-step low-degree occupancy falls from $1 + 0.8p$ to $0.8 + p$ under the edit, a strict decrease of $0.2(1 - p)$.

Give robot 2 a next goal at least two moves away after it completes its current goal, so it cannot finish another task in step 2; robot 3's goal also remains too distant to reach by then. After a successful first baseline step, robot 0 can move from $(0, 0)$ to its goal $(0, 1)$ while robot 1 follows into its own goal $(0, 0)$; after a failed first step, the original chain can retry. These are legal proposals. By the end of step 2, robot 2 has finished with probability $1 - (1 - p)^2$, while robots 0 and 1 each finish with probability p^2. The baseline therefore has expected $2p + p^2$ completions under this legal continuation. Under the edited first step, robot 2 finishes immediately; robot 1 at $(2, 1)$ is three grid moves from its goal and cannot finish by step 2, and robot 0 can finish only if it succeeds on both steps. Thus *any* continuation after the edit has at most $1 + p^2$ expected completions. The baseline advantage is at least $2p - 1 > 0$ despite all three strict one-step improvements. If success probabilities must lie strictly below one, the reversal persists for robots 1 and 2 sufficiently close to one by continuity. This example establishes a two-step value reversal, not a claim about the magnitude or cause of the empirical long-horizon losses below. □

The development controller considered paths whose learned release product was below $1/2$, chose the best non-goal pivot by expected-move increment within each path, and took at most four paths per step. One arm used the motion-censored, posterior-screened identity vector for gate and ranking; another used its contemporaneous fleet mean. Both computed the same guide, two PIBT candidates, learner and selectors as a no-split control. Table 11 gives the full frozen four-context pilot on already exposed low-fault seeds. All nine new trajectories and three reused seed-106 controls passed independent physical, posterior, selector and task audits. Reused seed-106 controls have exact action-prefix parity but their older binary omits split-selector overhead, so their CPU is not an exact cost match.

Table 11: Completed tasks over 3,000 physical steps in the full exposed segmentation pilot. The identity-specific arm failed its preregistered advancement rule; none of these rows is independent confirmation.

Map	Seed	No split	Pooled split	Identity split
room-8	106	4092	4536	4554
random	106	35694	34031	34943
random	107	19322	34396	25489
warehouse	106	19384	22372	20326

The identity arm beat pooled segmentation in only two of four rows; its macro mean log task ratio to the pooled arm was -0.0913, below the frozen $+0.01$ threshold. It therefore stopped before any further identity-arm development or unopened-seed confirmation. Pooled segmentation exceeded no-split control in three rows but lost 1,663 tasks on random-106. Its selection for the subsequent complete exposed-seed study was adaptive to this pilot and cannot turn these four rows into independent evidence. The locally dominant immediate action property and these realized long-run contrasts address different estimands.

The pooled arm was then selected *adaptively* for an unchanged-rule, complete exposed-development matrix: all six already exposed seeds on each of room-8, random and warehouse, paired to the same no-split computation. The four pilot pooled trajectories and four accepted no-split controls were reused; 28 new full trajectories passed independent physical, posterior, selector and task audits. Paired fault tapes and input hashes matched, and every no-split task count matched its earlier Guided25 reference. Table 12 gives every task outcome; the reused older seed-106 no-split binary omitted split-scoring overhead, so CPU comparisons use the five fully matched seed-107–111 pairs on each map.

Table 12: Complete adaptive exposed-development study of unchanged pooled vacancy segmentation, 3,000 physical steps per pair.

Map	Seed	No split	Pooled split	Relative Δ (%)
room-8	106	4092	4536	+10.85
	107	3969	4761	+19.95
	108	4768	4200	-11.91
	109	4900	4633	-5.45
	110	4075	4076	+0.02
	111	5116	4860	-5.00
random	106	35694	34031	-4.66
	107	19322	34396	+78.01
	108	34739	34387	-1.01
	109	34981	35308	+0.93
	110	19926	34806	+74.68
	111	33775	23468	-30.52
warehouse	106	19384	22372	+15.41
	107	23339	21474	-7.99
	108	24955	20282	-18.73
	109	21164	18399	-13.06
	110	20509	21367	+4.18
	111	20225	21346	+5.54

The prespecified advancement rule required a positive mean log task ratio on each map, overall mean at least $+0.03$, at least four positive pairs per map, no individual loss below -15%, and mapwise matched native CPU ratio below 1.5. Observed map means are $+0.0085/+0.1203/-0.0321$, with only three of six positive pairs on *each* map. The overall mean $+0.0322$ and matched CPU ratios $1.016/1.130/1.016$ pass, but the map-sign, win-count and loss-cap criteria fail; random-111 loses 30.52% and warehouse-108 loses 18.73%. The arm stopped without unopened-seed confirmation.

The equal-compute ratios isolate the incremental cost of *executing* the split because the no-split arm still computes both PIBT actions and both split selectors. For practical context, accepted historical one-pass Guided25 runs on the same 18 map/seed/fault-tape contexts averaged 134.3/107.5/222.5 native CPU seconds (room/random/warehouse), versus 163.9/143.9/286.7 for pooled split: ratios of means 1.22/1.34/1.29. Their task-window sums and fault digests match the no-split controls. This cross-binary cost reference is not the precommitted matched-computation gate or an equal-wall-clock performance comparison.

The random-map effect is nonuniform: the split avoids no-split late collapses in seeds 107 and 110 ($+15{,}074/+14{,}880$ tasks) but has its own late collapse in seed 111 ($-10{,}307$). In the first 500 steps, split-minus-control is $+210$ tasks for seed 107 and $+209$ for seed 111; early reward alone does not distinguish these observed outcomes. A post-outcome sample of exact static goal-distance progress likewise gives no established selector: its mean is negative early on the winning random-107 trajectory and positive throughout the losing warehouse-108 trajectory. These are trajectory diagnostics, not estimates of expected long-run causal effect or grounds for an adaptive success claim.

I Development evidence and negative boundaries

Development selected the final interface and scope. On six room-8 development clusters, individual minus Guided25 was $+335.78$ tasks, 18/18 rows positive, with descriptive interval $[226.39, 445.17]$ and 10.73% pooled gain. Unchanged transfer to room-16 was $+198.70$ $[72.10, 325.20]$, 16/18 rows and 6/6 cluster means positive. These observations motivated the locked evaluation but are not pooled with it.

Feedback and ranking controls. A separate six-seed room-8 development replication crossed motion-censored versus naive attempted-nonmovement feedback with raw versus shared-grid, protected-vacancy release ranking, and retained the censored fleet-mean arm. The same three fault profiles, 3,000-step horizon and Guided25 backbone were used in every arm. Table 13 reports

all five learned arms against the matched backbone, clustering the three profiles by seed. All new trajectories passed physical and learner replay; this is development evidence, not part of the frozen final effect estimate.

Table 13: Six-seed room-8 development feedback/ranking controls, completed tasks minus the same Guided25 backbone; descriptive 95% t_5 intervals.

Learned arm	Mean Δ tasks	95% t_5 interval
Motion-censored, shared-grid protected (MCRP)	+335.78	[226.39, 445.17]
Motion-censored, raw release score	+199.11	[-15.67, 413.89]
Naive feedback, shared-grid protected	+159.67	[-36.23, 355.56]
Naive feedback, raw release score	+207.39	[66.24, 348.54]
Motion-censored, protected fleet mean	+20.56	[-162.98, 204.09]

The candidate-minus-censored-raw and candidate-minus-naive-protected means are $+136.67$ $[-69.91, 343.24]$ and $+176.11$ $[-57.35, 409.57]$; the feedback-by-ranking interaction is $+184.39$ $[-90.98, 459.76]$. Thus this sample does not establish separate task benefits for censoring, coarsening/endpoint protection, or their interaction. The candidate's offline own-reliability mean-squared error was 0.000423, versus 0.11821 for naive-protected feedback. This supports calibration of the censored estimate, not a causal claim that calibration explains the task effect. The preceding two-seed pilot even ranked some controls above the candidate; it was not removed or pooled into this six-seed table.

Reliability-to-identity alignment control. An earlier, separately frozen eight-seed room-8 development control supplied the *true* per-robot reliabilities to the same ranking rule, then permuted their assignment to identities while preserving the reliability multiset and paired physical failure tapes. Correct assignment minus the single fixed-per-context permutation averaged $+403.83$ tasks, with a descriptive 95% t_7 interval $[238.20, 569.47]$ and all eight seed-cluster means positive. This oracle-information control supports the importance of identity alignment in the selected room regime; one permutation per context and development-only selection preclude a general or deployed-method claim.

Room-scope exact-reliability controls. An earlier eight-cluster room-8 development matrix with high, low, wide and homogeneous profiles found identity true q minus exact fleet-mean q to average $+230.8$ tasks $[92.4, 369.3]$ under the same upper-known preference; all eight seed-cluster means were positive. Correct identity assignment minus one fixed permutation of the same true values averaged $+239.2$ $[116.4, 361.9]$, with seven of eight seed means positive. These are secondary selected-development contrasts, with one permutation per context; they do not establish a map-population effect. The compact E133/E135 audits and a direct recomputation are in the study supplement.

Perfect reliability is not a policy oracle. On the separately selected room-16 six-cluster development cohort, an otherwise identical release comparator received either the physical true q_i, their fleet mean, or a deterministic permutation of their identities. The original online censored learner and Guided25 rows were joined on matched seeds, profiles and fault tapes after the three oracle arms passed complete physical and estimator replay. Table 14 reports the paired seed-cluster contrasts; intervals are descriptive t_5 intervals and do not repair development selection.

Table 14: Room-16 development contrasts with the same guarded Guided25 preference, six seed clusters and three fixed profiles.

Contrast	Mean task Δ	95% t_5 interval
True identity q − Guided25	$+131.3$	$[19.0, 243.6]$
True identity q − exact fleet-mean q	$+59.1$	$[-12.1, 130.2]$
True identity q − permuted true q	$+79.5$	$[-2.8, 161.8]$
Online learner − true identity q	$+67.4$	$[1.3, 133.5]$
Fixed-count true-q shrinkage − fixed true q	$+45.0$	$[-19.3, 109.3]$
Online learner − fixed-count shrinkage, first 1,000 ticks	$+64.8$	$[5.1, 124.5]$

The shrinkage policy supplied $(1 + n_i q_i)/(2 + n_i)$, with n_i its own identifiable-exposure count; it had overall reliability MSE 0.000221 against 0.000428 for the online learner on these contexts. These policies induce different trajectories and exposure opportunities, so the task contrast is a total policy effect, not a same-state value of estimation error. An unrestricted controller given true q could ignore it and emulate the learner; the fixed plug-in contrast does not show that extra information lowers optimal policy value. An additional development replay held the accepted learner's complete time-indexed exposure-count schedule fixed across two arms, replacing its realized posterior ratios by their fixed-count true-q expectations. The realized-minus-expected task difference was -1.4 $[-52.2, 49.3]$ across the same six clusters. This does not prove equivalence; the source schedule is itself a path-dependent footprint and future policies still diverge. The matched protocol and audit are retained in the companion study artifacts.

Perfect-input transfer diagnostic. In a separate protocol on already exposed random64 and warehouse low-profile contexts, we supplied the physical q_i to the same one-PIBT upper-known preference with the online feedback update disabled. The matched Guided25 and censored-learner controls share the start, first goals, physical probability vector and complete tickwise fault tape; simulator, gateway and map hashes also match. Future goals can diverge after task completions. All four 3,000 step oracle trajectories passed independent physical and task replay. Table 15 reports every prespecified row. In the two warehouse contexts true q still loses to Guided25, by 7.3% and 42.5%. Random signs disagree. This is a fixed-controller diagnostic with two development seeds per map, not a deployable algorithm or population estimate.

Table 15: Exposed-development perfect-input task counts, 3,000 ticks. The oracle arm has the physical q as a privileged input.

Map	Seed	Guided25	Censored	True q	Δ Guide	Δ Censored
random64	106	35,694	36,067	35,311	-383	-756
random64	108	34,739	25,590	35,329	$+590$	$+9,739$
warehouse	106	19,384	19,738	17,966	$-1,418$	$-1,772$
warehouse	108	24,955	14,044	14,343	$-10,612$	$+299$

After seeing the two warehouse losses above, we froze an adaptive extension to the four other already exposed warehouse low-profile seeds. No parameters or control rule changed. All four new 3,000-step trajectories passed complete physical and task replay, with matched starts, first goals, fault tapes, simulator, gateway and map hashes against their accepted Guided25 controls. Table 16 retains the positive seed109. Across all six warehouse seeds, true q loses to Guided25 in five and gains in one; the mean is -4075.3 tasks with descriptive t_5 interval $[-8960.8, 810.2]$. The extension was chosen after two losses and all seeds were previously exposed, so this pattern is not independent confirmation.

Table 16: Adaptive perfect-input warehouse extension on four exposed seeds.

Seed	Guided25	True q	True q − Guided25
107	23,339	14,129	$-9,210$
109	21,164	21,604	$+440$
110	20,509	17,761	$-2,748$
111	20,225	19,321	-904

For all six warehouse rows and both random rows in the original diagnostic, we located the first differing proposals at their still-matched tick-zero state. The true-q action increases the exact conditional expected number of moved robots by 0.52–9.08 in every row, while expected immediate current-goal completions are unchanged. This illustrates that higher local executable-move value need not predict lifelong tasks. It does not attribute the later task gap to this one action: both persistent planners and their future goal streams diverge. The eight exact calculations, full-row audits and protocol are in the companion code/evidence bundle.

The original six warehouse rows each used one physical-fault tape. To test whether the two largest losses were stable to that randomness, V2-O3 fixed twelve new tapes for each of the previously

selected seeds 107/108 and ran both full policies on each tape. Before reading these task outcomes, V2-O4 fixed a third, privileged exact-fleet-mean-q arm. It uses the identical upper-known comparator, shared quantizer, planner and physical setting as the identity-true-q arm; only the supplied vector changes. All 72 new 3,000-step trajectories passed complete physical and task replay. The three-arm audit checked equal initial states/goals, reliability profiles, simulator and native hashes, and all 24×3000 common fault masks. Table 17 reports descriptive intervals over fault-tape pairs *within each selected fixed context*; it does not estimate a population effect over maps, starts or task streams.

Table 17: Privileged-input contrasts on two adaptively selected warehouse-low starts, 12 new paired fault tapes each. Entries are mean task difference and descriptive two-sided 95% t_{11} interval over the twelve tapes.

Contrast	Seed 107	Seed 108
Mean q − Guided25	-199.2 $[-2524.3, 2126.0]$	-1268.1 $[-3840.9, 1304.7]$
Identity q − mean q	-3249.5 $[-5408.9, -1090.1]$	-989.7 $[-3647.6, 1668.3]$
Identity q − Guided25	-3448.7 $[-5574.6, -1322.8]$	-2257.8 $[-4829.1, 313.6]$

Identity q trails mean q in nine of twelve seed-107 pairs, with its descriptive interval below zero. Seed 108 is less decisive: seven of twelve identity-minus-mean differences are positive, but four large losses make its mean negative and the interval crosses zero. Mean q does not establish an advantage over Guided25 on either selected start. In the first *shared* state, identity q changes 130/282 proposals relative to mean q and increases exact expected executed moves by $1.349/4.097$ in seeds 107/108; expected immediate current-goal completions are unchanged. This first-split calculation was written after outcomes, and each context's twelve tapes share the same initial action pair. The local gain and the seed-107 long-run loss thus describe two different value levels, not a causal effect of that single action. Future goals are endogenous to each trajectory. An unrestricted policy given true q could emulate the mean-q policy, so the fixed plug-in contrast is not negative optimal value of information. Both contexts and all preexisting six-row warehouse evidence are exposed development data; no held-out seed was opened.

V2-O5 then locked the four *remaining* exposed warehouse-low starts, seeds 106/109/110/111, before any of their new outcomes. It used the same twelve-tape three-arm protocol without modifying either earlier O4 start. All 144 new full trajectories passed independent physical and task replay; the three-arm contract regenerated every Bernoulli fault mask and checked common starts, physical profiles and input identities over $48 \times 3000 = 144{,}000$ matched triplet ticks. Combining both phases now covers all six previously exposed warehouse-low starts, twelve new tapes per start. Table 18 shows the fixed-comparator identity contrast and exact first common action value under physical q.

Table 18: Identity true q minus exact fleet-mean q under the same fixed upper-known preference on all six exposed warehouse-low starts. Task entries are mean difference and descriptive 95% t_{11} interval over 12 common fault tapes per fixed start; no population or multiplicity-adjusted inference. The last column values one *distinct* tick-zero action pair per start.

Seed	Task difference [interval]	Positive pairs	First $\Delta\mathbb{E}$[moves]
106	-3668.3 $[-6080.6, -1256.1]$	$2/12$	$+7.277$
107	-3249.5 $[-5408.9, -1090.1]$	$3/12$	$+1.349$
108	-989.7 $[-3647.6, 1668.3]$	$7/12$	$+4.097$
109	-3694.8 $[-6621.4, -768.1]$	$2/12$	$+2.094$
110	-1329.8 $[-4108.0, 1448.4]$	$3/12$	$+7.398$
111	-4007.8 $[-7402.8, -612.7]$	$3/12$	$+1.077$

Identity q has a negative mean task contrast in all six starts; four within-start tape intervals are below zero and two cross it. The equal-start descriptive mean is -2823.3 tasks, with 52/72 negative individual tape differences. Mean q minus Guided25 remains heterogeneous: its seed109 mean is $+3294.3$ $[1661.8, 4926.7]$, while the other five within-start intervals cross zero. True q minus Guided25 has six negative context means, but only the previously selected seed107 interval is wholly below zero. At every first shared state identity q increases exact expected executed moves, while expected immediate current-goal completions are equal; each seed's twelve tapes share only one distinct local action pair. The V2-O5 first-split addendum was written after launch but before task

outcomes were viewed; it is a secondary diagnostic, not an independent causal test. Actual moves and tasks per 1,000 moves under identity q are also lower than under mean q in all six starts, but these are downstream closed-loop outcomes, not mediators. The extension addresses the earlier choice of two loss-making starts *within the exposed warehouse panel*; the map and profile were themselves selected in development. An unrestricted policy could ignore identity information, and no held-out seed or deployable controller was tested here. All per-tape results, costs, gate audits and the 144,000-tick contract are in the companion V2 study bundle.

One identity-input decision under mean-q continuation. V2-O6 was written after viewing O5 but before any new run. On the same six exposed starts and 12 retained fault tapes each, a new native uses identity true q only at tick zero, then exact fleet-mean q from tick one through tick 2999. The unchanged always-mean and persistent-identity arms are reused. With the switch disabled, the isolated native reproduced both old 120-tick controls exactly; all 72 new full trajectories passed independent physical/task replay. A separate 216,000-tick pair contract matched every fault mask and starting state to the mean arm, and every new tick-zero proposal and realized action to the identity arm. The only source difference is the tick-one input-vector replacement under a flag.

Table 19: V2-O6 policy-schedule decomposition on the same six exposed warehouse-low starts. Task contrasts and descriptive 95% paired t_{11} intervals over 12 common physical-fault tapes per fixed start; no population or multiplicity-adjusted inference. The one-step regime uses identity true q at tick zero and exact fleet-mean q thereafter.

Seed	One-step − mean q [interval]	Identity q − one-step [interval]	One-step positive pairs
106	-1352.0 $[-3129.5, 425.5]$	-2316.3 $[-4928.8, 296.1]$	$4/12$
107	$+637.1$ $[-2215.6, 3489.8]$	-3886.6 $[-6876.0, -897.1]$	$8/12$
108	$+1145.2$ $[-1282.7, 3573.1]$	-2134.8 $[-4713.0, 443.3]$	$8/12$
109	-2777.9 $[-4631.9, -923.9]$	-916.8 $[-3794.6, 1961.0]$	$2/12$
110	$+1211.7$ $[-1682.7, 4106.1]$	-2541.5 $[-5650.4, 567.4]$	$7/12$
111	$+413.5$ $[-3209.7, 4036.7]$	-4421.2 $[-6582.4, -2260.1]$	$8/12$

The one-step-minus-mean equal-start task mean is -120.4, with four positive and two negative context means and 37/72 positive individual tape differences. Only seed109's within-start interval is wholly below zero; the five others remain uncertain. Persistent identity minus the one-step regime has six negative context means, two wholly negative within-start intervals, 54/72 negative tape differences and equal-start mean -2702.9. Per pair and in equal-start means, the two contrasts sum exactly to O5's identity-minus-mean contrast (-2823.3 up to rounding). This is an algebraic decomposition of input-schedule *regimes*, not a mediator attribution: tick-zero input changes realized states, endogenous goals, planner priorities and random-stream state before mean-q continuation begins. The one-step input does not reproduce the broad persistent-policy loss across all six starts, but the uncertainty cannot exclude a substantial first-step effect in five. Seed109 provides one negative selected-context estimate despite its higher exact first-step expected moves. Equal-start mean realized moves are 3,240,229/3,233,336/2,974,532 for mean/one-step/ persistent identity; their mean native CPU seconds are 219.7/219.8/212.1. These are downstream descriptions, not causal mediators. All new per-tape differences, gate records and audits are in the companion V2 study bundle.

Robot-indexed future-goal-stream robustness. The preceding policies use the inherited single chronological generator: after task completions diverge, later draws can be assigned to different robots. V2-G1 was written before building its isolated simulator or viewing any new outcome. It keeps the weighted task law, map, physical fault tapes, unchanged native upper-known rule and six exposed warehouse-low starts, but draws each robot's goals from its own seeded stream. Thus each robot's accepted ordinal-goal prefix is identical across the two policy arms. This also changes the *initial* goals relative to V2-O5, so comparing the two study estimates does not isolate a chronological-stream effect. A postoutcome first-tick check on rep0 of four O5 extension starts kept the same positions and fault mask but found 1,741–1,748 of 1,800 initial goal assignments changed. All 144 new trajectories passed independent physical and task replay. Across 72 pairs, a separate audit matched all 216,000 fault-mask ticks, initial states/goals, and 1,416,467 shared robot goal ordinals. The unmodified control simulator rebuilt to the pinned original binary hash under the new build settings; the source patch changes only KivaSystem goal-sampling RNG ownership.

Table 20: V2-G1 identity true q minus exact fleet-mean q with robot-indexed goal streams. Six already exposed warehouse-low starts, 12 matched physical fault tapes per start. Intervals are descriptive within-start paired t_{11} intervals, not population or multiplicity-adjusted inference. The last column evaluates one distinct first shared action pair per start.

Start	Task difference [interval]	Positive pairs	First $\Delta\mathbb{E}[\text{moves}]$
106	-3297.7 $[-4854.7, -1740.6]$	$1/12$	$+3.855$
107	-1468.3 $[-3544.1, 607.4]$	$3/12$	$+2.727$
108	-4881.9 $[-6664.8, -3099.1]$	$0/12$	$+0.919$
109	-1625.2 $[-4130.7, 880.4]$	$4/12$	$+2.882$
110	-1355.4 $[-5138.7, 2427.9]$	$6/12$	-0.310
111	-2950.3 $[-6134.5, 233.9]$	$2/12$	$+2.611$

The six start means remain negative, with equal-start mean -2596.5 tasks (56/72 negative tape differences). A descriptive t_5 interval over these *selected* starts is $[-4050.3, -1142.6]$; it does not become an unselected warehouse population interval. All six equal-start 500-tick window differences are negative, from -23.8 in the first window to approximately -578 in the last. First-action expected current-goal completions are equal in all six, while exact expected moves favor identity q in five and mean q in one. Mean realized moves are 3.024/3.258 million and tasks per 1,000 moves are 5.754/6.133 for identity/mean q, downstream descriptions rather than identified mediators. The goal-stream change preserves the fixed-rule loss direction in these exposed contexts. It does not identify a unique traffic mechanism or a negative optimal value of individual information, and the differences between V2-O5 and V2-G1 are not a matched causal task-generator intervention.

Independent warehouse and prospective room fixed-rule tests. Before G1 complete task outcomes were read, V2-G2 fixed a conditional advancement rule: at least five negative G1 start means and an equal-start loss of at least 1,000 tasks. G1 passed with six negative means and -2596.5, opening previously unused warehouse starts 120–125. Each of those six starts received twelve paired actuator tapes and both exact identity-q and exact fleet-mean-q inputs to the unchanged upper-known preference for 3,000 ticks. Before viewing *any complete G2 task outcome*, V2-G3 also froze the full corresponding room-8/1,000-robot low-fault matrix, without a G2-dependent launch rule. Both studies use the same robot-indexed goal simulator and native planner binaries, low-fault generation law and fault-tape seed formula; the map, robot count and induced identity/start/task configurations differ. The six numeric seed labels are the same, but are not a paired causal map treatment.

Table 21: V2-G2/G3 complete true identity q minus exact fleet-mean q start contrasts under one fixed preference. Each entry is the mean and descriptive 95% paired t_{11} interval over twelve matched physical tapes for that fixed start. The final column gives positive tape pairs; there is no multiplicity adjustment or claim of arbitrary-context transfer.

Setting	Start	Task difference [interval]	Positive/12
Warehouse, 1,800	120	-1719.8 $[-3896.5, +457.0]$	3
	121	-1556.2 $[-3961.1, +848.6]$	5
	122	-4527.9 $[-7190.7, -1865.1]$	2
	123	-2915.2 $[-5293.4, -537.1]$	2
	124	-2013.8 $[-4979.1, +951.6]$	4
	125	-2796.6 $[-4588.2, -1005.0]$	2
Room-8, 1,000	120	$+509.6$ $[+267.5, +751.7]$	11
	121	$+201.5$ $[-91.5, +494.5]$	9
	122	$+398.4$ $[+250.5, +546.3]$	11
	123	$+339.3$ $[+136.6, +542.0]$	11
	124	$+372.3$ $[+88.6, +656.0]$	11
	125	$+264.0$ $[-47.3, +575.3]$	10

All 288 full trajectories passed independent physical/task replay, and each study's 72 pair contracts independently regenerated 216,000 tickwise fault masks. G2 matched 1,398,882 shared robot-goal ordinals and G3 matched 436,006, with equal within-pair initial states and goals. All launches and audits completed without failure. Across six new warehouse starts, identity minus mean q is -2588.25

tasks (-13.09% of mean-q tasks); the descriptive six-start t_5 interval is $[-3743.77, -1432.73]$, with 54/72 negative tape pairs. Across six prospectively specified room starts it is $+347.53$ tasks ($+7.96\%$), descriptive t_5 interval $[+234.66, +460.39]$, with 63/72 positive tapes. Three warehouse within-start t_{11} intervals are wholly negative, four room intervals wholly positive; the others cross zero. The twelve tapes share a start and ordinal-goal stream, so the six starts rather than 72 tapes underlie each t_5 interval. G1's selected starts remain separate from G2; G3 was proposed after earlier room development evidence and is not pooled with it.

The warehouse's equal-start first 500-tick task difference is $+11.9$ before five negative windows ($-163.6, -499.3, -625.7, -669.7, -641.8$); the room's six windows are all positive ($+58.7, +44.0, +44.0, +59.0, +75.8, +66.1$). This is temporal description, not a mediator decomposition. Free-cell occupancies are 0.3158/0.3094 for warehouse/room, but their maps, fleet sizes, exact reliability identity draws and task configurations are not matched; no single map-causal interaction follows. The earlier selected room-8/low oracle control had 7/8 positive seed means and mean $+405.1$ under the inherited task stream, but is neither a new confirmation sample nor combined with V2-G3. The fixed-rule sign boundary does not constrain an optimal policy allowed to ignore individual reliability. Source, complete positive/negative rows, costs and independent contract audits are included in the companion study bundle; the large raw traces have separately verified server archives.

Developmental map/fleet and task-law interventions. Before either missing-cell complete task result, V2-G4 froze the same six-start, twelve-tape, 3,000-tick exact-input comparison on warehouse with 1,000 robots and room-8 with 1,800 robots. These fill a two-by-two descriptive map-family/fleet-count table, keeping the native controller, simulator, fault law and robot-indexed goal mechanism fixed. All new trajectories passed independent physical/task replay, both sets of 72 within-cell contracts matched all 216,000 fault ticks, and cross-map fault masks agreed in all 72 seed/tape pairs at each fixed fleet count. The table below gives the equal-start identity-minus-mean effect and a descriptive six-start t_5 interval; signs count positive start means and tape contrasts.

Table 22: G2/G3 fixed-rule tests and G4 developmental missing cells. The six numeric start labels are not identical physical states across map or fleet cells; contrasts between cells are descriptive.

Map family	Robots	Δ tasks [95% t_5 interval]	Relative	Positive starts/tapes
Room-8	1,000	$+347.5$ $[+234.7, +460.4]$	$+7.96\%$	$6/6; 63/72$
Room-8	1,800	$+352.8$ $[+268.7, +436.9]$	$+11.40\%$	$6/6; 64/72$
Warehouse	1,000	-3288.7 $[-3720.1, -2857.3]$	-18.99%	$0/6; 13/72$
Warehouse	1,800	-2588.3 $[-3743.8, -1432.7]$	-13.09%	$0/6; 18/72$

The room sign is positive at both fleet counts and the warehouse sign negative at both. The room 1,800-minus-1,000 difference is $+5.3$ tasks with descriptive numeric-seed t_5 interval $[-97.9, +108.5]$; the warehouse difference is $+700.4$ $[-604.4, +2005.3]$. No physically matched fleet-size effect follows. In the two new cells, identity q changes realized moves by $+96{,}794$ (room/1,800) and $-326{,}157$ (warehouse/1,000) per trajectory on average. It also changes tasks per 1,000 realized moves in the respective same directions; these co-moving closed-loop outcomes are not identified mediators.

V2-G5 was separately frozen before its complete outcomes. It holds the warehouse/1,800 obstacles, physical q, initial robot positions, fixed preference and 72 tickwise fault tapes equal to G2, but converts endpoint and 40-workstation destinations to the simulator's free-cell task law. Initial goals, future goal streams and task difficulty consequently change. A cross-law audit verified all 72 matched starts, q vectors, obstacle maps and fault digests. Every G5 trajectory passed full physical/task replay; 72 within-law contracts checked 216,000 fault ticks and 1,025,798 common robot-ordinal goal assignments. The identity-minus-mean effect is -1606.35 tasks (-11.81%), descriptive six-start t_5 $[-2109.79, -1102.91]$, with six negative start means and 15/72 positive tapes. Relative to G2's 40-station law, the within-seed/tape difference in identity effects averages $+981.90$ tasks, descriptive six-start interval $[-282.50, +2246.31]$; one of six start interaction means is negative. The free-cell task law therefore leaves a persistent negative fixed-rule effect in this panel, while the interaction estimate does not isolate service-location concentration as its cause. A prewritten follow-up gate required the free-law identity effect to exceed $+100$ tasks and be positive in at least five starts. It failed,

so no contingent W400 experiment was run. These studies are developmental diagnostics, separate from G2/G3's prospective fixed-rule tests and from E189's deployable-controller final evaluation.

Controlled warehouse bypass intervention. V2-T1 pre-specified an additional four-arm geometry intervention after G5: original warehouse versus a copy in which the top row of each two-by-ten shelf block is opened (2,000 cells), crossed with exact identity versus exact fleet-mean q. The 40 workstations and 2,540 endpoints retain their locations and labels. Each of six already opened starts 120–125 has six paired fault tapes, the same 1,800 robot positions, physical q, and first goals across maps and input arms. A robot-indexed generator couples future goals within each robot across all four arms. This is an *injected-goal* stream: loading G2's first goal does not consume the generator's first draw, so the next goal repeats the first task class before ordinary alternation. It is therefore not numerically interchangeable with G2's standard task stream. An upstream loader initially assigned a fixed orientation to these injected goals despite rotation being disabled; its invalid 120-step gate was retained, and an isolated one-line repair gave all four arms the same unconstrained goal orientation before any full T1 outcome was read.

All 144 complete trajectories passed independent physical/task replay; 36 four-way contracts checked 108,000 shared physical-fault ticks and 763,826 shared per-robot ordinal goals, with zero full-run failures. The six-start contrasts in Table 23 show a mean bypass-minus-original interaction of $+2191.6$ tasks, but only four of six start interactions are positive and its descriptive t_5 interval crosses zero. The bypass-map mean identity contrast is $+863.9$ tasks, likewise positive in only four starts with an interval crossing zero. The prewritten standard-stream follow-up rule required at least five positive start means for either the bypass contrast or a > 1500-task interaction; neither gate passed, so no further T2 runs were made. These are exploratory paired interactions on opened starts, not a confirmed geometry-dependent selector.

Table 23: T1 true-identity-minus-fleet-mean completed-task contrasts under the common injected-goal stream; six fault tapes per start. The interaction is the bypass contrast minus the original-map contrast. Intervals are descriptive over six start means.

Start	Original map	Bypass map	Interaction
120	-1714.0	$+2900.3$	$+4614.3$
121	$+1541.0$	-1022.3	-2563.3
122	$+1084.3$	$+547.3$	-537.0
123	-1300.7	$+2750.7$	$+4051.3$
124	-4217.3	-1340.0	$+2877.3$
125	-3359.3	$+1347.5$	$+4706.8$
Equal-start mean	-1327.7	$+863.9$	$+2191.6$
95% t_5 interval	$[-3751.6, +1096.3]$	$[-1039.9, +2767.8]$	$[-997.7, +5380.9]$

The intervention changes traversable cells from 5,699 to 7,699 and the degree-two-cell fraction from 43.24% to 3.69%. In the fixed first-goal input panel, mean shortest-path distance changes by only 0.02–0.03 cells per start, while the initial fraction of robots with no adjacent empty cell falls from about 4–6% to 1–2%. New cells are used: sampling every 50th tick in bypass trajectories gives mean occupancy 383.7 (identity input) versus 355.1 (fleet mean). These occupancy counts follow the policy and do not identify a causal mediator. Geometry, free area, robot density per free cell, local vacancies and route availability change together. Mean native CPU rises from about 220 to 304 seconds per trajectory across the two input arms. Full paired rows, source identities, the invalid gate and the complete audits are preserved in the companion T1 study evidence.

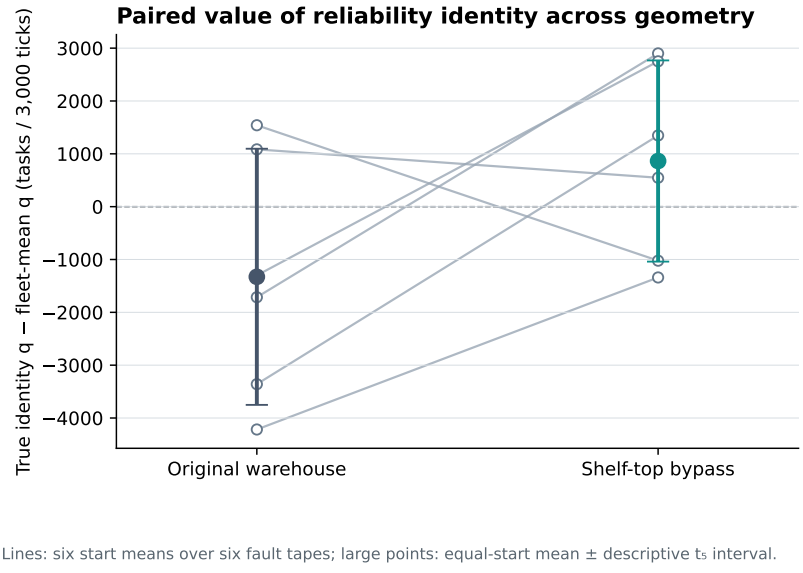


Figure 6: The six paired T1 start means (thin lines) and equal-start means with descriptive t_5 intervals (large points). Heterogeneity prevents a precise sign conclusion for the bypass interaction.

J Complete-plan one-step admission: a development falsification

V2-PS1 pre-specified a stronger exact-information control interface after the local and geometric development results. At a common current state, two independent persistent Guided-PIBT planners propose complete legal actions M and I using exact fleet-mean and exact individual q, respectively. They receive the same realized feedback and use a common deterministic per-tick tie seed, while retaining their own guidance states. Under exact individual q, the shield chooses I only when its conditional expected current-goal completions and movers weakly exceed those of M, and its expected next-step count of robots in static graph-degree-at-most-two cells does not exceed that of M. Otherwise it chooses M. A small fixed floating-point tolerance handles numerical equality, without tuning on task outcomes. Exact q is privileged, so this is a mechanism probe rather than a deployable reliability learner.

For each proposed mover i, let D_i contain the dependency prefix to a vacancy or all members of its directed cycle, and put $p_i = \prod_{j \in D_i} q_j$. Linearity of expectation gives the three scores $\sum_{i:A_i=g_i} p_i$, $\sum_i p_i$, and $L(S) + \sum_i p_i[c(A_i) - c(x_i)]$, where $c(v)$ indicates degree at most two and $L(S)$ is the current number of robots in such cells. The selector therefore satisfies these three conditional inequalities at the *same* state, up to its fixed 10^{-9} numerical comparison tolerance. It has no inequality between the distinct future states reached by the closed-loop policies. The four-robot construction in Appendix H satisfies all three inequalities strictly yet reverses expected two-step task value.

The missing quantity has an exact finite-horizon description. Augment the Markov state with both planners' guidance memories, goal-stream state and tie-seed state, and let $V_t^M(s)$ be expected remaining completed tasks under the always-mean-plan policy. At each state where the shield admits a distinct identity plan I, denote its expected current-goal gain over the mean plan by $\Delta r_t(s) \geq 0$, and the difference in expected V_{t+1}^M after their respective next states by $\Delta V_t(s)$. The ordinary Bellman telescoping identity gives

$$J_T(\text{shield}) - J_T(M) = \sum_{t<T} \mathbb{E}_{s_t \sim \text{shield}} \left[\mathbf{1}\{I \text{ admitted at } s_t\} (\Delta r_t(s_t) + \Delta V_t(s_t)) \right].$$

This is exact under the shared transition law, not a new policy-improvement guarantee. The guard controls Δr; its movement and occupancy conditions do not bound ΔV. A negative expected throughput effect would require the aggregate continuation term to outweigh the nonnegative immediate-goal term.

Two 120-step, three-arm engineering gates passed independent full physical and task replay, common fault and robot-ordinal goal-prefix contracts, and independent recomputation of every proposed score and shield choice. Distinct identity plans were admitted in 45/120 room and 41/120 warehouse gate ticks, above the prewritten 1% activity minimum. The complete prewritten panel contains room-8/1,000 and original warehouse/1,800, already exposed starts 106 and 108, three matched fault tapes per start, and three 3,000-step arms (M, I, shield): 36 trajectories. All 36 passed independent full physical/task replay; twelve triplet contracts matched 36,000 physical fault ticks, initial conditions and common robot-ordinal goal prefixes. All 84 compact launch, audit and contract process records returned zero with empty stderr. A separate scan of all 108,000 planner input states found no already satisfied current goal, satisfying the immediate-goal score's stated prerequisite on this panel. The original raw and isolated executable are retained in the companion evidence. After outcomes, an independent read-only selector audit also recomputed both complete-plan scores and the choice on every one of the 36,000 shield ticks. All saved proposals, scores, admission flags and selected actions agreed; the largest absolute numerical score discrepancy was 2.28×10^{-13}. This verifies execution of the stated guard throughout the full panel, not its long-horizon value.

Table 24: PS1 exposed-start complete-plan development. Tasks are equal-start means over two starts and three tapes each; start columns give shield minus its matched mean-q arm. These are not fresh confirmation estimates.

Setting	Mean	True	Shield	Shield–mean	Start 106	Start 108
Room-8, 1,000	4571.0	4816.3	4731.2	$+160.2$	-97.7	$+418.0$
Warehouse, 1,800	21800.2	14264.8	20877.7	-922.5	-1712.0	-133.0

A secondary measurement rule was written while complete trajectories were running, before complete task outcomes were read: sample realized low-degree occupancy at ticks $0, 50, \ldots, 2950$ on every trajectory. The mean/true/ shield fractions are $0.09407/0.09400/0.09274$ in room and $0.61449/0.64384/0.62970$ in warehouse. Warehouse shield-minus-mean is $+0.01520$ and positive in both starts. This does not violate the common- state conditional guard; the two arms' future state distributions differ. It is descriptive rather than an identified mediator. For the mean/true/shield arms, equal-start averages of each trajectory's task-per-million-realized-move ratio are $3219/3310/3261$ in room and $6396/5265/6275$ in warehouse, likewise descriptive.

A retrospective same-state score readout finds that distinct plans were proposed on every shield tick. Of its average 922.5 room and 820.2 warehouse identity-plan admissions per 3,000 ticks, only 29.4% and 17.1%, respectively, strictly raised expected immediate current-goal completions beyond 10^{-7} numerical tolerance. Nearly all admitted plans strictly raised expected movers; 90.9% of warehouse admissions strictly reduced expected low-degree occupancy. Conditional immediate-goal score gains summed to $+108.2$ in room and $+57.0$ in warehouse on the shield's own visited states. These are post-outcome descriptive proposal margins, not additive causal contributions to the matched-arm task differences or proof that a particular traffic variable caused the warehouse loss.

The shield's warehouse gain of $+6612.8$ tasks relative to this study's own unrestricted true-q arm must not be compared numerically to earlier exact-input studies, which use a different planner randomness schedule. The shield falls below its *matched* mean-q arm in warehouse, including both start means. Mean native CPU seconds per trajectory are $285.0/285.5/284.2$ in room and $484.3/431.6/462.2$ in warehouse for mean/true/shield. Only two of the five prewritten advancement conditions pass: the average room gain exceeds 100 tasks and CPU stays within 3 times the old single-planner comparator. The requirement that both room starts improve, and both warehouse gain and nonnegative-start requirements, fail. No learned port, post-hoc admission threshold, or fresh-start test followed.

The stopped task-age direction is also retained as a boundary. In a fresh four-seed development replication, combined failure-censored age plus learned reliability had positive point means versus Guided25 on room, random, and warehouse maps ($+93.17$, $+152.58$, $+376.50$), but only 9/12, 7/12, and 6/12 rows were positive and every seed-cluster interval crossed zero. Component and factorial contrasts were mixed; two predeclared mechanism associations were near zero and sign-unstable. Fragile-agent completions decreased on all three maps while ordinary completions rose. We therefore do not include this priority modification in the method or claim that its positive map means are a replicated effect.

Table 25: Preserved development boundaries for the unchanged interface. These are not final tests and differ in seed counts and, for room-32, evaluation stage.

Setting	Mean Δ	Interval	Interpretation
random64, 1,000 robots	-1103.83	[-7145.63, 4937.97]	Two-seed development result; no transfer support.
warehouse, six clusters	-1120.89	[-2645.70, 403.92]	Negative mean; individual also loses to fleet mean.
room-32, combined	+23.50	[-974.10, 1021.10]	Positive first subset reversed in extension; not final.
room-8, 200 robots	+0.11	[-101.38, 101.60]	Dense-room result collapses at lower population.
room-16, 200 robots	+129.50	[29.85, 229.15]	A positive lower-density exception, not a universal rule.

A separate two-seed mechanism split replaces every mutable heuristic-tie draw with a deterministic key indexed by seed, tick, agent, sort invocation, and comparison invocation. This exactly prevents earlier draw consumption from shifting a later tie while preserving comparison-level bit variation. On the random map, individual reliability minus its matched keyed-vacancy control is $-12{,}047/-15$ tasks; keyed individual minus Guided25 is $-12{,}706/+6{,}243$ (mean $-3{,}231.5$, a 10.15% mean loss). Both frozen advance conditions fail despite positive room effects. The random-map loss is late-emerging and accompanied by a $-573{,}464$ realized-move difference on one seed. We therefore reject RNG-phase stabilization as a cross-domain repair and do not include the keyed policy in the method.

The combined evidence does not yield a validated scope selector. Geometry, density, estimated reliability accuracy, and one-step dependency statistics are plausible moderators, but available interventions do not identify any one as the causal mediator of long-run throughput.

K One complete-plan pulse under a common shield continuation

V2-PP1 follows PS1 with a single-action causal probe. Two initial states were selected *after* inspecting PS1 outcomes and tick-zero proposals: room-8/1,000/start 106 and original warehouse/1,800/start 108, both in the low-failure profile. Their exact individual-q and fleet-mean-q planners proposed distinct complete plans that the PS1 three-metric shield would admit. In each state, 64 new paired physical fault tapes run for 3,000 ticks. At tick zero only, the two arms force the mean or identity plan; at every later tick both use the unchanged PS1 shield, with the same dual-planner setup, per-tick tie schedule and robot-indexed goal streams. The initial positions, goals, identity reliabilities and fault tape match within each pair; future goal *ordinal* prefixes match, though goal activation times can diverge. This is a conditional effect for two exposed, outcome-selected states, not a map-population contrast or a new controller.

At tick zero, identity minus mean exact expected current-goal completions are zero in both states: both absolute scores are zero in room, and both are about 0.995 in warehouse. Identity minus mean expected movers are $+3.6714/+0.9192$ and expected degree-at-most-two occupancy drifts are $-3.5899/-1.0040$ (room/warehouse). The plans differ on 167/1,000 and 221/1,800 robot targets. These are treatment descriptions, not future task outcomes.

All 256 full trajectories passed independent physical/task replay, and all 128 pair contracts matched their 384,000 physical fault ticks, initial conditions, first candidate plans and common robot-ordinal goal prefixes (403,255 room and 1,405,819 warehouse ordinals). An independent read-only scorer recomputed both candidate scores, the admission flag and the selected complete plan on every one of the 768,000 PP1 ticks; all matched, with maximum absolute score discrepancy 2.28×10^{-13}. All input hashes remained unchanged. A compact input auditor's first attempt mistakenly required the room and warehouse *map* hashes to agree; it stopped at the first warehouse row. The corrected check requires each map's hash to be stable within its setting and all non-map source hashes to agree across settings. It passes all 256 trajectories; the engineering correction is retained in the supplement. The prewritten summary was recomputed byte-identically from all 128 compact pair rows.

Table 26: PP1 single-plan pulse, two exposed selected states. Paired identity-minus-mean task differences use 64 new fault tapes per state; intervals are descriptive t_{63} intervals, not map-level inference.

State	Mean-plan	Identity-plan	Difference	95% interval	Positive/negative
Room-8/1,000, start106	4983.4	4883.0	-100.4	$[-269.6, +68.7]$	29/35
Warehouse/1,800, start108	20724.3	20390.2	-334.1	$[-1204.8, +536.6]$	31/33

The full paired differences range from -1551 to $+1396$ room tasks and -8336 to $+7996$ warehouse tasks. Six prespecified 500-tick window means and cumulative intervals are in the compact supplement; the windows are secondary timing descriptions, not separate success tests. The prewritten single-plan effect sign is unresolved in *both* states: both primary intervals cross zero. PP1 therefore neither attributes PS1's sustained warehouse loss to one accepted action nor excludes harmful or beneficial individual admissions. It also does not estimate the always-mean-policy continuation term in the Bellman decomposition above, because PP1 uses the *shield* for every later action. Mean dual-planner native CPU seconds per trajectory are $286.6/284.5$ in room and $485.7/483.5$ in warehouse (mean/identity pulse arms). Raw traces, binary identities and all compact rows are retained separately.

L Route-load interventions: transfer and timing

Policy and comparison contract. These experiments retain the Guided-PIBT planning backbone and its 25-iteration search budget. For the preceding joint proposal, draw an edge from mover i to the occupant of its target cell. Let b_i count reverse-reachable movers, including i, that depend on it. The pooled intervention uses a shared online fleet reliability estimate $\bar{q}$ and liability $\lambda_i = (1 - \bar{q})b_i$. It adds a pairwise route penalty proportional to $(\lambda_i + \lambda_j)$ times shared non-goal vertices and opposing edges. Opposite-flow cost remains the primary criterion; the added penalty enters the secondary criterion with vertex flow and path length. With the new liability disabled, the tested implementation reproduces Guided25 in the parity checks. This diagnostic policy is separate from MCRP and does not establish an individual-reliability benefit.

Staged transfer evidence. The unchanged pooled policy was first tested on seeds 106–107, then on four additional seeds 108–111 after the first results were seen. All three maps use the low-failure profile, 1,000 robots and 3,000 physical ticks. Tasks completed at initialization are excluded. Table 27 retains every paired row; these data are not pooled with the frozen room-map evaluation. Mean task differences are -240.83 on room-8, $+3{,}827.33$ on the random map, and $-1{,}357.00$ on warehouse. Only 7/18 rows improve. Averaging log task ratios equally over maps within each seed and then over six seed clusters gives 0.0131, with a descriptive t_5 interval $[-0.1350, 0.1612]$; three of six cluster means are positive. There is no supported general transfer claim.

Two random-map trajectories account for the striking positive tail. On seed 107, Guided25 completes 19,322 tasks and the pooled policy 34,405; their final 500-tick counts are 394 and 5,289. On seed 110, totals are 19,926 and 33,331, with final-window counts 475 and 5,943. These are from-start policy comparisons, not rescue after an observed collapse. Moreover, the existing individual comparator already reaches 29,749 tasks on seed 107. Neither the full gain nor the outcome-selected pair uniquely credits the new route-load construction.

Assignment controls. At each visited state, a deterministic nonzero cyclic rotation of the current liability vector preserves its multiset and sum before assigning entries to agents. A uniform control instead preserves the sum alone. Each controller subsequently visits its own states, so the controls do not hold realized costs or liability histories identical across trajectories. On seed 107, aligned/rotated/uniform totals are 34,405/30,460/15,930; on seed 110 they are 33,331/34,764/34,229. Thus the particular original agent alignment is not necessary to avoid these two degradations. This does not eliminate the possible role of closure-derived magnitudes or other state variables. The controls also have costs: on stable random seeds 106/108, rotated losses are 2,465/7,621 tasks and uniform losses 4,113/31. General improvement is therefore unsupported.

Table 27: Complete staged development results for the unchanged pooled dependency-load intervention (two initial seeds, then four additional seeds). Values are completed tasks over 3,000 physical steps; Δ is pooled minus Guided25.

Map	Seed	Guided25	Pooled	Δ
room8	106	4,092	4,593	+501
room8	107	3,969	4,179	+210
room8	108	4,768	3,709	-1,059
room8	109	4,900	4,531	-369
room8	110	4,075	4,312	+237
room8	111	5,116	4,151	-965
random64	106	35,694	35,204	-490
random64	107	19,322	34,405	+15,083
random64	108	34,739	32,579	-2,160
random64	109	34,981	30,609	-4,372
random64	110	19,926	33,331	+13,405
random64	111	33,775	35,273	+1,498
warehouse	106	19,384	19,862	+478
warehouse	107	23,339	21,209	-2,130
warehouse	108	24,955	21,889	-3,066
warehouse	109	21,164	19,146	-2,018
warehouse	110	20,509	19,554	-955
warehouse	111	20,225	19,774	-451

Common-history timing interventions. Let z_t be the fraction of robots with no currently empty neighboring cell. A first pressure guard used a 500-tick calibration and an exponential moving average with coefficient 0.01 and drift margin 0.10. Activation at ticks 1,469/1,364 on seeds 107/110 yielded totals 20,837/21,333 and final windows 1,509/672. A development revision selected baseline $z \leq 0.25$, reset the moving average after calibration, and used margin 0.04. Activation at ticks 1,315/969 then yielded totals 19,187/20,561 and final windows 375/865. Before activation, histories match Guided25 exactly in audited positions, goals, proposals, faults and task consumption. The revised thresholds were selected after the 18 transfer contexts had been observed; these are not independent warning-system tests. Inactive guards preserve actions, but monitoring and passive learning still consume computation.

A finite intervention shares the first 500 baseline ticks, applies pooled liability on ticks 500–1499 when the calibration gate passes, and disables it thereafter. Seed 110 reaches 30,488 tasks, with 3,039 in the final window, after 1,500 ticks with no added penalty. Seed 107 reaches only 18,406, with 118 in the final window. Four other random contexts have task differences $+421, +708, -67, +1{,}320$; the tested room and warehouse contexts remain behaviorally identical to the baseline. An adaptive-duration variant stops at tick 1,250 unless a development-selected pressure rule extends protection. It extends seed 107 to the horizon but still gives only 20,124 total and 688 final-window tasks. Seed 110 stops earlier and reaches 28,983/2,292. Neither controller satisfies the joint development criteria.

Interpretation and provenance. Figure 7 supports persistent closed-loop path dependence and sensitivity to intervention timing. An "early basin" interpretation is consistent with these traces, but no dynamical basin, irreversibility, or unique mediator is established. Failing two specific late-start policies does not prove that all late recovery is impossible. All seeds 106–111 were exposed before these follow-ups. The matched-assignment and timing batches comprise 66 runs and 136,400 physical ticks, including retained short failed gates; summed outer wall time is 7,914.4 seconds, native CPU time 6,075.2 seconds, and maximum resident memory 405,504 KiB. Physical replay, learner, load, common-prefix, stopping-time, parity and source-identity checks passed. These checks establish execution validity, not broad scientific generality.

L.1 Structural burden and intervention identities

Proposition 4 (Vacancy terminals and chain burden)**.** *Fix n robots on a free-cell graph, and let z be the fraction having no currently empty neighbor. Consider an injective adjacent-cell proposal*

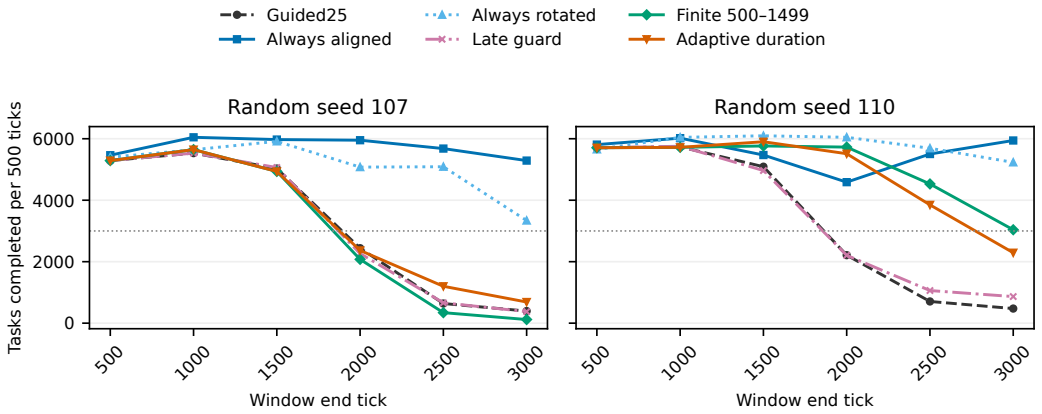


Figure 7: Timing interventions on the two outcome-selected random-map cases. Curves show consecutive 500-tick task counts. Aligned and rotated from-start policies avoid the observed late degradation. The tested pressure guard does not recover throughput; a finite intervention leaves persistent benefit only on the second case. The 3,000-task line is a development acceptance threshold, not a theoretical transition. The cases do not estimate degradation prevalence or unseen-seed performance.

with no active robot targeting a stationary robot's cell. Suppose M_{ch} *active robots occupy* $K > 0$ *directed-chain components. If* b_i *is the reverse closure size including* i*, then*

$$K \le n(1-z), \qquad \sum_{i \text{ in chains}} b_i \ge \frac{M_{\mathrm{ch}}^2}{2n(1-z)} + \frac{M_{\mathrm{ch}}}{2}.$$

Proof. Each chain has a distinct terminal robot proposing into an empty neighbor, so $K \le n(1-z)$. A chain of length ℓ has reverse closure sizes $1, \dots, \ell$. Hence the sum is $\sum_k \ell_k(\ell_k+1)/2 \ge M_{\mathrm{ch}}^2/(2K) + M_{\mathrm{ch}}/2$ by Cauchy–Schwarz; the terminal bound gives the result. A cycle of length ℓ contributes ℓ^2 under the same closure definition and does not reduce the chain bound. The condition $K > 0$ ensures a positive denominator. □

Multiplying by $1 - \bar{q}$ gives the pooled chain-liability bound. This is a combinatorial burden statement; overlapping dependency failures prevent its interpretation as an exact count of canceled moves or a throughput guarantee.

For completeness, the timing comparisons use two elementary pathwise facts. With shared initial state and paired exogenous randomness, a guard that is a stopping time of visible history and returns the baseline proposal before activation has the same realized pre-activation history. Induct on proposals, executed moves and goal consumption. After a finite intervention ends at H, if both policies apply the same decision rule and have equal complete states at H, their futures also agree by induction. The complete state includes positions, goals, planner memory, and random/tape indices. Thus post-removal differences reflect the state history produced by the intervention, including possible memory or random-index effects. The argument does not identify physical congestion as the sole persistent state variable.

M PLANNING-BUDGET AND ROUTE-SELECTION CONTROLS

These are development experiments on exposed seeds 106–111. The two random-map degradation cases, seeds 107 and 110, were chosen for mechanism analysis after their Guided25 outcomes were known. No policy in this section passed a cross-map development gate, and the unopened seeds 120–125 were not used. Counts exclude initial goals and cover $0 < t \le 3000$ physical ticks.

Table 28: Completed tasks in exposed random-map development cases. Portfolio arms compute three eight-iteration candidates at each tick. Guided8 computes only one. All controls use the same actuator/task contract; task streams can diverge after policy trajectories separate.

Controller	Seed 107	Seed 110
Guided25	19,322	19,926
Mixed scored portfolio	26,843	27,114
Q-off scored portfolio	26,847	26,248
Mixed round robin	17,813	18,643
Q-off round robin	17,893	18,436
Q-off fixed first candidate	29,180	34,743

Table 29: Guided8 minus Guided25 completed tasks over all exposed development seeds. Every listed value uses paired initial states and actuator seeds.

Map	106	107	108	109	110	111	Mean
Room-8	+698	−83	−1,126	−309	+162	−374	−172
Random64	−6,913	+9,858	−2,024	−3,301	+14,817	+257	+2,115.7
Warehouse	−337	−420	−10,459	−3,932	−3,625	−1,021	−3,299

Scored portfolios and matched candidate controls. Three route candidates were computed in isolated processes, each with eight route-LNS iterations and its own randomness. This is 24 route-improvement iterations in total versus Guided25's 25, but also three PIBT proposals and process communication; actual CPU is higher. A current-visible lexicographic score used proposed goal hits, guide-state coherence, vacancy terminals, dependency concentration, static goal-distance progress and path cost. It used no future task, fault or physical outcome. The mixed portfolio assigned online pooled reliability load to two candidates; the q-off portfolio assigned none. Round-robin and fixed-first-candidate policies kept the candidate-generation budget and removed the score. Table 28 reports all tested arms in the two exposed cases.

The mixed and q-off scored arms differ by only $-4/+866$ tasks, below the predeclared mean reliability-distinction threshold of 2,000. This rejects a necessity claim for reliability loading in the observed rescue. The all-q-off fixed candidate exceeds the all-q-off scored portfolio by 2,333/8,495 tasks. Thus the tested per-tick score does not justify switching route-generation processes. On these 6,000 scored ticks, proposed immediate goal hits decide 2,736 choices, guide-state coherence 3,258, and vacancy or dependency fields only six; these counts describe the implemented rule rather than identifying a causal mediator.

Single-planner budget control. The fixed first candidate is a q-off planner with eight rather than 25 route-LNS iterations. A separate single-process implementation matched its positions, goals, proposals, actuator outcomes and executed moves for all 120 ticks of a gated comparison. We therefore reuse its two longer exposed random-case trajectories as Guided8 outcomes. Table 29 retains all development differences versus Guided25, including losses.

The positive counts are 2/6 room, 3/6 random and 0/6 warehouse. Equal-map mean log ratios clustered by seed average -0.0369, with approximate t_5 interval $[-0.204, 0.130]$. The two random gains are substantial, but the random median difference is -883.5 tasks and warehouse seed 108 loses 10,459. Guided8 therefore failed the frozen nonnegative-map-mean, random 4/6-positive and severe-loss development gates. Its mean native CPU times are 59.9/60.2/109.2 seconds on random/room/warehouse, versus 107.5/134.3/222.5 for Guided25. Those are observed costs, not deadlines. The multi-candidate arms cost roughly 181–192 native CPU seconds per 3,000-tick random trajectory, above Guided25.

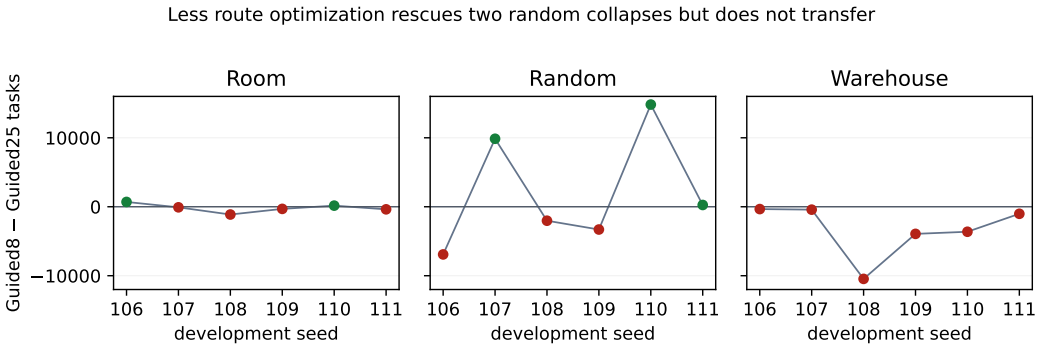


Figure 8: The full exposed-seed planning-budget boundary. Guided8 avoids the two known random degradations but loses on three other random seeds and every warehouse seed. It is a development counterexample to monotone closed-loop return in search depth, not a general lower-budget recommendation.

Interpretation and audit. The planner improves an internal route surrogate, whereas completed tasks are generated by a stochastic closed-loop process. Increasing route-search iterations need not improve that return: the eight-iteration policy wins on two paired cases and loses sharply elsewhere. Changing the iteration count also changes selected guides, random consumption, initial actions, subsequent physical states and task-consumption times; none is isolated as the unique mechanism. Likewise, fixed versus scored candidate selection is a whole-controller comparison, not proof that all adaptive selection fails.

The three rounds of portfolio, fixed-candidate and budget comparisons launched 36 runs. Thirty-two were accepted, two 120-tick gates were retained as rejected after a process-lifecycle error, and two start-up failures had zero physical ticks. Accepted trajectories cover 78,720 physical ticks; including rejected trajectories gives 78,960. The accepted runs passed proposal and physical replay, task accounting, q-off and liability checks, scorer recomputation, source identity and shared-input checks. The combined compact audit is preserved with SHA-256 `682651a5af307442d4aabb83996950a9c15900eea69135d65aab99feb372b1e6`. The frozen final same-budget learned-reliability result is separate and unchanged.

N Reproducibility and Audit Contract

The final Guided-PIBT and learned-method executable has SHA-256 `08682e34b0df43cd8e1131e61143dc92d9bc88bbc4e8ca6eeac323b225e8f053`. The unchanged author HR executable has SHA-256 `e1727676ec1f3e99965dcb7da37d3ad9eb6b4c700d4546de00916a428e4c42fa`. The accepted compact audit has SHA-256 `baf1b9a150e2f5e0aeb72ebe4420359de107d980e2906513239a5730399fe7bb`. It records PASS, 180 runs, 540,000 steps, paired initial/fault streams, and zero frozen-input changes.

The audit checks hashes and process exits, replays every cascading move and learner update, spot-checks the comparator, verifies paired states and fault tapes, and recomputes task-window, cohort, calibration and cost summaries. The launcher ran the matrix once with one audit. Four earlier development trajectories needed post-run metadata reconstruction after a scratch-cap interruption; no final trajectory used that repair.

Open-source MD-PIBT (Jiang et al., 2026b) supplies the task host via a one-step action oracle, not a compared planner. Planner, gateway and simulator shared a pinned x86 Linux CPU (24, 29, 35 or 36). Retained builds used `g++ -O3 -DNDEBUG -std=c++17`; Guided/learned additionally used `-DGUIDANCE_LNS=25`. Both executable hashes match the records, and their ELF comments identify GCC 13.3.0 (Ubuntu package 13.3.0-6ubuntu2~24.04). Post-run inspection found that the same host boot predates the matrix, CPU 24 was Intel Xeon 6767P, kernel Linux 6.9.12-060912-generic, and the Ubuntu 24.04.2 OS-release file predates the run. These reconstructed facts do not attest historical user-space package versions.

O FULL POSITION FEEDBACK: LIKELIHOOD, IDENTIFIABILITY, AND CONTROL TRANSFER

The censored Beta learner in the confirmed controller discards cyclic observations to retain independent own-actuation labels. This is sufficient for the room-scope task result, but it is not the full statistical experiment. The following statements concern the position-only observation model, not a claim that more accurate reliability must improve lifelong task throughput.

Proposition 5 (Conditional position likelihood)**.** *Condition on a legal joint proposal and independent actuator successes with probabilities $q_i \in (0,1)$. Its moving-agent dependency graph decomposes into vacancy-terminated paths and directed cycles. On a vacancy path ordered $i_1, \dots, i_L$ from the vacant endpoint, the executed movers form a prefix of length K, with*

$$\Pr(K = k) = \Big(\prod_{j=1}^{k} q_{i_j}\Big) \begin{cases} 1 - q_{i_{k+1}}, & 0 \le k < L, \\ 1, & k = L. \end{cases}$$

Every member of a cycle C moves together with probability $Q_C = \prod_{i \in C} q_i$. Conditional component likelihoods multiply.

Proof. Unique proposed targets, including waiters' own cells, give each mover at most one predecessor and one successor; no mover targets a waiter. Repeated cancellation therefore leaves a disjoint union of vacancy paths and cycles. On a vacancy path, i_1 moves iff its actuator succeeds; inductively i_j moves iff all first j actuators succeed. The first failed actuator stops the remaining upstream movers, yielding the prefix law. In a cycle, no member can vacate unless all members do, so its observed all-move bit is the conjunction of independent successes. □

Proposition 6 (Structural identifiability for a repeated exogenous design)**.** *Let E contain the robots appearing in vacancy paths with positive design probability, and let $C_1, \dots, C_m$ be the distinct cycle-member sets with positive design probability. Form a matrix A with rows $e_i^\top$ for $i \in E$ and $\mathbf{1}_{C_j}^\top$ for $j = 1, \dots, m$. On $(0,1)^n$, the individual vector q is identifiable from the position-observation distribution iff $\operatorname{rank}(A) = n$. A single observation of cycle C contributes Fisher information*

$$I_C(\log q) = \frac{Q_C}{1 - Q_C}\, \mathbf{1}_C \mathbf{1}_C^\top,$$

which has rank one.

Proof. Every downstream success prefix has positive probability, so ratios of adjacent prefix-survival probabilities identify each q_i for $i \in E$. Each observed cycle law identifies only Q_{C_j}. In coordinates $z_i = \log q_i$, these are exactly the linear equations $e_i^\top z$ and $\mathbf{1}_{C_j}^\top z$. Full rank is sufficient. If A is rank deficient, a nonzero null direction produces two nearby interior parameter vectors with identical path parameters and cycle products, proving nonidentifiability. The Bernoulli cycle score is proportional to $\mathbf{1}_C$; taking its expected outer product gives the displayed Fisher information. □

The rank result concerns repeated availability of an *exogenous* design; it does not say that a finite trace uniquely determines continuous q, nor does it give adaptive-policy consistency without exposure assumptions.

Proposition 7 (Identifiability of release-score comparisons)**.** *Under the repeated exogenous design of Proposition 6, consider two prospective partial-release scores $S_P(q) = \prod_{i \in P} q_i$ and $S_R(q) = \prod_{i \in R} q_i$. Their ratio is identifiable from the position-feedback distribution if and only if $\mathbf{1}_P - \mathbf{1}_R$ lies in the row span of A. For a finite collection of such comparisons, all ratios are identifiable if and only if every corresponding incidence-vector difference lies in that span. Full identification of q is sufficient but can be unnecessary.*

Proof. In log coordinates, $\log(S_P/S_R) = (\mathbf{1}_P - \mathbf{1}_R)^\top z$. The observation distribution identifies exactly Az on the interior parameter space. A contrast in the row span is a known linear combination of these observations. Otherwise a null vector h of A has a nonzero inner product with the contrast. Perturbing any interior z by a small multiple of h preserves every observed path and cycle law but changes the score ratio. Apply this argument to each member of the finite collection. □

The statement concerns score *ratios*, not every ordering: nested chains have a known ordering for $q_i \in (0,1)$ even if the ratio is not identified. For incomparable candidates, ambiguity can change the preferred action. Consider four robots circulating around a legal 2×2 grid square embedded in a larger grid with adjacent vacancies. If this cycle is their only prior exposure, $(q_1, q_2, q_3, q_4) = (0.9, 0.5, 0.8, 0.8)$ and $(0.5, 0.9, 0.8, 0.8)$ both move the cycle with probability 0.288 but reverse prospective single-robot release-score rankings q_1 versus q_2. Exposing robots 1 and 2 on vacancy paths identifies that comparison while q_3 and q_4 can remain unresolved. This establishes an information requirement for that local comparison, not a guarantee that choosing the higher score maximizes lifelong tasks. The row-space criterion is the log-product specialization of decision-relevant observability in linear partial monitoring (Kirschner et al., 2023); its motion-path and cycle feedback structure is specific to this setting.

Proposition 8 (Identifiability of exact immediate task values)**.** *Retain the repeated exogenous design and interior parameter domain of Proposition 6. Fix a decision state with no already satisfied current goal and a legal complete plan P. Let $b_D(P)$ count its current-goal movers whose motion requires precisely the actuator successes in dependency set D, taking zero for absent sets, and let $a_D = \mathbf{1}_D$. The numerical conditional expected current-goal count*

$$r_P(z) = \sum_D b_D(P) e^{a_D^\top z}, \qquad z = \log q,$$

is globally structurally identifiable from position feedback if and only if every a_D with $b_D(P) > 0$ belongs to $\mathrm{row}(A)$. For two fixed plans P, R in that state, their numerical score difference $r_P - r_R$ is identifiable if and only if every a_D with $b_D(P) - b_D(R) \neq 0$ belongs to $\mathrm{row}(A)$; common unidentifiable terms may cancel. This criterion concerns the numerical values, not merely the sign of a comparison or lifelong task value.

Proof. Any incidence vector in $\mathrm{row}(A)$ has a known log product from the feedback distribution, proving sufficiency. For necessity, collect identical dependency sets and write the plan value or difference as $f(z) = \sum_D c_D e^{a_D^\top z}$, with nonzero coefficients c_D. If f is constant on every observation-equivalence fiber $z + \ker(A)$ over an open interior domain, then for every $u \in \ker(A)$,

$$0 = \frac{d}{dt} f(z + tu)\big|_{t=0} = \sum_D c_D (a_D^\top u) e^{a_D^\top z}$$

on an open set of z. Distinct exponential monomials are linearly independent there, so $a_D^\top u = 0$ for each active term and every null direction u. Equivalently $a_D \in \mathrm{row}(A)$. □

Corollary 1 (Information required by a finite plan menu)**.** *Under Proposition 8, fix a nonempty finite menu $\mathcal{P}$ of legal complete plans at one state. Let*

$$\mathcal{V} = \mathrm{span}\{\mathbf{1}_D : b_D(P) \neq b_D(R) \text{ for some } P, R \in \mathcal{P}\}.$$

All numerical pairwise immediate task-value differences are identifiable if and only if $\mathcal{V} \subseteq \mathrm{row}(A)$.

Proof. For each pair, Proposition 8 requires precisely the incidence rows with nonzero coefficient difference. Taking the union over pairs and then its linear span gives the stated condition in both directions. □

The menu determines the information required: rows belonging only to terms common to every plan impose no constraint. This characterizes numerical differences; identifying merely their signs can require still less information.

The distinction between individual and decision-value identification can be strict. Observing only a four-robot cycle identifies its product, so the expected immediate goal count of a prospective goal-reaching rotation is identified although no member reliability is. A separate legal 5-robot example has two candidate plan values that each depend on an unobserved single-robot q, but their difference cancels that term and equals one observed fifth-robot q. Exact actuator-mask enumeration and the two indistinguishable reliability vectors are given in the companion evidence. Another enumerated legal example has two mutually exclusive four-robot rotations sharing one robot: their observed cycle products suffice to rank their immediate goal values ($567/1250 > 216/625$) even though all seven individual reliabilities remain underidentified. The inference result does not turn an identified myopic score into a long-horizon safe selector.

Proposition 9 (Finite-sample decision-value confidence)**.** *Fix horizon T, $q_i \in [\epsilon, 1-\epsilon]$, and a preselected finite collection H of feedback rows a_r. A row may be an individually exposed vacancy-path actuator bit, a vacancy-path prefix all-move bit, or a directed cycle all-move bit; its Bernoulli mean is $p_r = e^{a_r^\top \log q}$ and $p_r \geq \ell_r = \epsilon^{\|a_r\|_1}$. Proposals are selected before the current fault draw, possibly using past feedback. For each row with $K_r \geq 1$ observations by any step at most T, clip its sample mean $\bar{Y}_r$ to $\widetilde{p}_r \in [\ell_r, 1]$ and set*

$$h_r = \sqrt{\frac{\log(2|H|T/\delta)}{2K_r}}, \qquad L_r = \max\{\ell_r, \bar{Y}_r - h_r\}.$$

Let $H_+ = \{r \in H : K_r \geq 1\}$ at that step. For a fixed plan value or signed two-plan contrast $f = \sum_D c_D e^{a_D^\top \log q}$, suppose every active dependency row has a representation $a_D = \sum_{r \in H_+} \lambda_{Dr} a_r$. Estimate it by

$$\widehat{f} = \sum_D c_D \exp\left(\min\left\{0, \sum_{r \in H_+} \lambda_{Dr} \log \widetilde{p}_r\right\}\right).$$

With probability at least $1-\delta$, simultaneously at every such step,

$$|\widehat{f} - f| \leq B_f := \sum_D |c_D| \sum_{r \in H_+} |\lambda_{Dr}| \frac{h_r}{L_r}.$$

Consequently $|\widehat{f}| > B_f$ certifies the sign of a plan contrast at that step. A needed row with $K_r = 0$ gives no certificate.

Proof. For each row, the observed success sequence can be coupled to iid Bernoulli(p_r) draws. Prefix and cycle events are selected before their fault draws. A vacancy-path robot's individual bit is observed only when its downstream members succeeded, an event independent of that robot's current bit. Hoeffding's inequality over every latent row/count prefix, followed by a union bound over $|H|T$ pairs, gives $|\bar{Y}_r - p_r| \leq h_r$ simultaneously at every realized $K_r \geq 1$. Different rows need not have independent observations. Clipping cannot increase the error. Both p_r and $\widetilde{p}_r$ are at least L_r on the confidence event, so $\log$ is $1/L_r$-Lipschitz between them. Because the true $a_D^\top \log q \leq 0$, projection of its estimated log product onto $(-\infty, 0]$ cannot increase the error; $\exp$ is one-Lipschitz there. Apply the triangle inequality to each term and then to the signed sum. □

The path-prefix indicator is a directly observed product even though adding its row does not enlarge the span generated by exposed singleton rows. It can improve the finite-sample bound when estimating a particular plan value. Since the confidence event holds for every row simultaneously, the representation may be selected after observing counts: for each active a_D, minimizing $\sum_{r \in H_+} |\lambda_{Dr}| h_r / L_r$ subject to $\sum_{r \in H_+} \lambda_{Dr} a_r = a_D$ is a linear program. It uses redundant direct-prefix observations when they tighten the certificate. There is a sharper direct-observation case: if each active a_D is itself an observed row $a_{r(D)}$, estimate its product by $\bar{Y}_{r(D)}$ rather than a log-row reconstruction. The same simultaneous Hoeffding event gives $|\widehat{f} - f| \leq \sum_D |c_D| h_{r(D)}$, with no factor $1/L_r$. Thus repeated position observations of a prospective goal-reaching cycle, or the relevant vacancy prefix, can certify its immediate task value without identifying any member reliability separately. For the five-robot example above, the unidentified terms in two plan values cancel and their contrast is q_5; a certificate needs only fifth-robot singleton exposure. The radius can be loose for rare long-cycle products, although its observed lower bound L_r can far exceed the worst-case ℓ_r. This proposition certifies an immediate score under its exposure counts, not a better lifelong policy.

Proposition 10 (Adaptive direct-product certification)**.** *Fix a fleet of n robots and at most T physical steps, with independent stationary $q_i \in [0,1]$. Every proposed mover i directly supplies the Bernoulli product bit $\mathbf{1}\{i \text{ moves}\}$ for its dependency set D_i; count an identical set at most once per step. No list of prospective dependency sets need be fixed in advance. For every nonempty $D \subseteq [n]$ with $K_D \geq 1$ prior observations, set*

$$h_D^* = \sqrt{\frac{\log(2nT\binom{n}{|D|}/\delta)}{2K_D}}.$$

With probability at least $1-\delta$, simultaneously for every discovered set and every step, $|\bar{Y}_D - \prod_{i\in D} q_i| \le h^*_D$. Consequently, even for a current state and two plans selected using past feedback, a fixed-plan immediate contrast $f = \sum_D c_D \prod_{i\in D} q_i$ whose active sets all have $K_D \ge 1$ obeys $|\sum_D c_D \bar{Y}_D - f| \le \sum_D |c_D| h^*_D$; a strictly larger estimated margin certifies its sign. This is no long-run task guarantee.

Proof. For each D, its exposure is selected before the current fault draw, and its deduplicated bits at exposure times are iid Bernoulli($\prod_{i\in D} q_i$). Allocate to D the prior weight $\pi_D = [n\binom{n}{|D|}]^{-1}$, for which $\sum_{D\neq\varnothing} \pi_D = 1$. Hoeffding's bound with failure probability $\delta\pi_D/T$ for each row/count pair, followed by a union bound, gives the simultaneous event. The plan and coefficients may be chosen after past feedback because this event holds for all rows and counts. Apply the triangle inequality. □

The penalty for a k-robot dependency is $\log\binom{n}{k}$, rather than uniformly paying order n for the entire set universe. This removes the preselected-row premise of Proposition 9 for direct-product scores, but low exposure still gives a wide radius. One can avoid charging for the full subset universe by ordering each newly proposed distinct dependency set before that tick's fault draw and assigning it the next discovery slot. At most nT slots can be filled. Conditioning on the pre-fault discovery history and union bounding over these slots and at most T counts gives the alternative simultaneous radius

$$h^{\mathrm{slot}}_D = \sqrt{\frac{\log(2nT^2/\delta)}{2K_D}}.$$

It is smaller than h^*_D for $|D| = k$ whenever $\binom{n}{k} > T$, but still requires repeat observations of the exact same set. The two alternatives each have failure probability at most δ; to take their minimum on one joint event, allocate $\delta/2$ to each. This is a sequential union-budget specialization, not a long-run control result.

Proposition 11 (Consistency under adaptive persistent exposure)**.** *Fix a finite fleet and grid, and let $q^\star$ lie in the interior of $\mathcal{Q} = [\epsilon, 1-\epsilon]^n$ for $\epsilon > 0$. Before each actuator draw, a planner may select any legal joint proposal using the entire past and fresh planner randomness independent of that draw. A proposal type d records both current joint positions and the legal joint proposal. Suppose the empirical frequency of each type d converges almost surely to w_d, and the exposure matrix formed from proposal types with $w_d > 0$ has full column rank. Then any maximizer over $\mathcal{Q}$ of the conditional position-feedback log likelihood converges almost surely to $q^\star$.*

Proof. There are finitely many legal proposal types and feedback outcomes. For a fixed candidate $q \in \mathcal{Q}$, subtract the conditional expectation of each log-likelihood ratio against $q^\star$. The resulting bounded martingale differences obey the strong law. Compactness and a common Lipschitz bound on the log likelihood over $\mathcal{Q}$ extend this convergence uniformly in q by a finite-net argument. The normalized conditional log-likelihood ratio therefore converges uniformly to $-\sum_d w_d \,\mathrm{KL}(P^d_{q^\star} \| P^d_q)$. A zero divergence for every positive-frequency type gives exactly the individual and cycle-product equations of Proposition 6; full rank makes $q^\star$ the unique maximizer. Uniform convergence and compactness imply the stated argmax consistency. □

This is a sufficient persistent-excitation result. It does not assert that Guided-PIBT or the tested controllers produce full-rank positive-frequency exposure, and it does not quantify finite-horizon estimator error.

Proposition 12 (Adaptive consistency of identifiable comparisons)**.** *Retain the compactness, predictability and limiting-frequency assumptions of Proposition 11, but do not require full column rank. Let A_w be the exposure matrix of proposal types with positive limiting frequency. For any fixed chain-score contrast $v = \mathbf{1}_P - \mathbf{1}_R$ in the row span of A_w almost surely, every conditional-likelihood maximizer $\widehat{q}_T$ obeys $v^\top \log \widehat{q}_T \to v^\top \log q^\star$ almost surely. Consequently a strictly unequal pair of these scores is eventually ranked correctly, even if the individual reliability vector is not identifiable.*

Proof. The uniform likelihood limit in the preceding proof still holds. Its maximizer set is the compact equivalence class $\{q \in \mathcal{Q} : A_w \log q = A_w \log q^\star\}$. Uniform convergence implies that the distance of any likelihood maximizer to this class tends to zero. A row-span contrast has the same value at every member of the class, so continuity gives the stated limit. A strict score ratio has log ratio bounded away from zero and therefore an eventually fixed correct sign. □

This is a statistical guarantee for persistent, informative exposure and fixed prospective comparisons, not a guarantee that the planner chooses such exposures or that a locally better-ranked action improves lifelong throughput.

The same compact-equivalence-class argument applies to Proposition 8: under the adaptive assumptions above, if the active dependency incidence vectors for a fixed prospective plan (or for its numerical contrast with another fixed plan) lie in $\mathrm{row}(A_w)$ almost surely, every conditional-likelihood maximizer consistently estimates that immediate value (or contrast). A strictly nonzero consistently estimated contrast is eventually ranked correctly. This does not require individual-q consistency and still says nothing about the value of altering the subsequent policy.

In a postoutcome structural check of one retained mean-q warehouse trajectory per exposed start, every one of the 1,800 robots had appeared in a vacancy-terminated proposal path by zero-indexed tick 8/3/4/8/9/3 for seeds 106–111. Thus the corresponding finite-prefix exposure matrix already contains all singleton rows; the warehouse task contrast is not explained by missing *structural* rank in these six prefixes. A handful of appearances is not accurate estimation, and this one-tape check does not verify positive-frequency exposure in the consistency propositions.

A postoutcome read-only check of all twelve exposed PS1 shield trajectories found that only 12/13,625 room and 1,933/66,573 warehouse multi-robot current-goal products had at least 20 prior observations of the same robot set. Each of the six per-setting traces was below 10% on this descriptive exposure measure. This does not evaluate confidence coverage, independent population transfer or control benefit; complete counts and the preserved pre-readout plan are in the compact supplement. At $\delta = .05$ and $T = 3000$, even omitting the nonnegative row-size penalty in Proposition 10, a single-product radius $h_D^* \le .1$ needs at least 931 prior observations for $n = 1000$ or 960 for $n = 1800$. No room multi-robot term and only nine warehouse terms in this panel had even 100; hence at most nine could meet that radius target. This necessary-count calculation does not test confidence coverage or certify a multi-term plan contrast. The discovery-slot alternative requires at least 1,331/1,360 observations for the same .1 radius in the room/warehouse fleet, so it does not change this panel-level limitation.

Cascade-bandit prefix feedback (Kveton et al., 2015) and network tomography's incidence-based latent inference (Chen et al., 2007) are relevant precedents. The contribution here is the legal joint-motion observation decomposition and its interface with a persistent task planner.

Using cycle observations. For independent incoming $\mathrm{Beta}(a_i, b_i)$ marginals, let $M_{i,k} = \mathbb{E}[q_i^k]$, $\mu_i = M_{i,1}$ and $Q = \prod_{i \in C} \mu_i$. A successful cycle exactly increments each member's success count. For a failed cycle,

$$\mathbb{E}[q_i^k \mid \text{cycle fails}] = \frac{M_{i,k} - M_{i,k+1} \prod_{j \in C \setminus \{i\}} \mu_j}{1 - Q}, \qquad k = 1, 2. \tag{7}$$

In particular, its posterior mean decreases by $\mathrm{Var}(q_i)Q/[\mu_i(1-Q)]$, rather than assigning a full failure to every member. We match the resulting two marginal moments to Beta laws after each failed cycle, an $O(|C|)$ assumed-density filter. Equation (7) is exact for one cycle under the independent incoming approximation; repeated projection discards induced correlations and is not an exact full posterior.

Proposition 13 (Convex exact-likelihood MAP benchmark)**.** *Let $z_i = \log q_i < 0$, let s_i, f_i count individually identified successes and failures, and include each successful cycle once in s_i for each member. Let n_C count failed cycles with member set C. With independent* $\mathrm{Beta}(2,2)$ *priors on q_i, the log posterior evaluated at $q_i = e^{z_i}$ is, up to a constant,*

$$L(z) = \sum_i \left[(s_i + 1)z_i + (f_i + 1)\log(1 - e^{z_i})\right] + \sum_C n_C \log\left(1 - e^{\sum_{i \in C} z_i}\right).$$

It is strictly concave on $z_i < 0$ and has a unique global maximizer. Without cycle observations, its maximizer equals the censored $\mathrm{Beta}(1,1)$ *posterior mean. This is a MAP in q, not the mode of the posterior density after a change of variables to z.*

Proof. For $u < 0$, $\log(1 - e^u)$ has second derivative $-e^u/(1-e^u)^2 < 0$. The negative Hessian of L is

$$\mathrm{diag}\left(\frac{(f_i + 1)q_i}{(1 - q_i)^2}\right) + \sum_C \frac{n_C Q_C}{(1 - Q_C)^2} \mathbf{1}_C \mathbf{1}_C^\top,$$

which is positive definite. The prior makes L tend to minus infinity at both parameter boundaries, so the optimum exists in the interior. Differentiating the no-cycle objective gives $q_i = (s_i + 1)/(s_i + f_i + 2)$. □

An offline benchmark optimized this exact objective from compact sufficient statistics on the same six exposed V2-L1 traces. At tick 200, true-q cross entropy was below the censored learner in all six by 0.0000087–0.000398, but above the cycle ADF in all six by 0.0000049–0.0000410. All 36 snapshot fits passed a projected-gradient check; fit time after count extraction was 10–22 ms at tick 200. The prewritten five-of-six advance prediction failed. Since this exact benchmark returns a posterior mode whereas ADF approximates marginal means, the comparison does not isolate ADF projection error or validate an online controller. Source, six complete checkpoint records and the disclosed numerical amendment are retained in the V2 study artifacts.

Exposed development readout. The six already-observed low-failure E219 q-off trajectories were replayed read-only, with exactly the same positions and feedback for the censored learner and the cycle filter. The latter reduced first-200-step true-q cross entropy and fragile-robot mean absolute error in all six rows; the absolute cross-entropy reductions ranged from 0.000082 to 0.000833. These are modest estimation gains, not task gains. The online filter was then inserted as the *only* change in the original 25-iteration Guided backbone, with matched old individual controls. Its own-policy true-q cross entropy again improved in all six contexts, but tasks did not: the three map-by-seed paired differences were room $+923/-232$, random $-454/+3996$, and warehouse $-1382/+6640$. Room108 violated the predeclared -200 task stop rule, so this controller was not promoted to an unopened-seed confirmation. The large positive seed108 differences partly reflect old-policy collapses; the new policy remains below the vacancy backbone on random108 and warehouse108. All nine new trajectories passed physical, task and online-estimate replay. Full row/window and cost data, protocols and original traces are retained in the study artifacts.

This contrast makes the inferential levels explicit: more cycle feedback improved predictive estimation in these development data; a fixed plug-in planner did not turn that gain into a reliable task benefit. It does not identify a unique traffic mediator or establish a population effect from six exposed contexts.

Cycle-centered preference falsification. A separate predeclared control test favored reliability of a currently closed dependency cycle inside equal-guidance decisions. Against the same Guided25 backbone on the three development maps at seeds 106/108, its task differences were room $+247/-342$, random $-170/-742$, and warehouse $-131/-8079$. Both room rows failed the required positive gate and the warehouse loss crossed the prespecified harm gate; the method was not extended or confirmed. A comparison can score a two-robot swap candidate that the final legality check rejects, so strict score-comparison counts are not counts of legal cycle opportunities. The full engineering correction, row results and physical audits are retained with the V2-C1 study. This falsifies that particular direct cycle preference, not the value of cycle observations for estimating q.

P PAIRED FAULT-SUFFIX CONTINUATION DIAGNOSTIC

This is a development mechanism test on the six already-exposed E220 single-pulse contexts, not a newly confirmed controller. For each context, the q-off and one-pulse arms reproduce the archived native prefix through the step before the pulse. Starting at that step, both use the same independent per-robot fault tape for each of eight suffix seeds $61000007 + 1009s + 7919r$, where $s \in \{106, 108\}$ and $r = 0, \ldots, 7$. The pulse changes one joint proposal; both arms then use the unchanged q-off planning rule for the rest of the 3,000-step horizon. The chronological task-generator RNG is identically initialized in all runs. It is not a fixed future goal list per robot after the policies diverge.

Four 120-step engineering gates checked archived-prefix parity, switch-tick fault-mask regeneration, the original pulse proposal, exactly one selector activation and task/physical replay before the full matrix. All 96 full runs completed once and passed an independent 3,000-step prefix, fault, physical, selector and task audit. All 48 pairs matched physical reliability vectors, fragile identities, input hashes and the fault-suffix contract. The native planner was unmodified and received neither true physical reliabilities nor future fault masks. The full raw remains remote; the compact per-run audits, prewritten protocol, runner/auditor sources and analysis JSON accompany the research artifacts.

The old path loses tasks in all six contexts despite five positive exact *next-step* goal values. Four of those five new sample means are positive; all six t_7 intervals cross zero. These continuations leave every conditional expected long-run sign unresolved while exposing strong pathwise sensitivity. Results condition on selected prefixes and a fixed chronological task stream; suffixes are not new map seeds or a validated controller.

Table 30: One changed action followed by the same closed-loop rule for 3,000 steps. Task differences are changed minus original. The eight-suffix intervals are descriptive within each already-exposed state, not map-level confidence intervals.

Map	Seed	One-step expected goal Δ	Old path Δ	Eight-suffix task mean [95% t_7]
room	106	$+0.095$	-57	$+269.6\ [-226.1, 765.4]$
room	108	$+0.117$	-374	$+312.9\ [-171.3, 797.0]$
random	106	$+0.150$	-930	$+624.9\ [-3051.1, 4300.8]$
random	108	$+0.177$	-3354	$+287.4\ [-3498.6, 4073.3]$
warehouse	106	-0.012	-6826	$+1111.1\ [-2362.0, 4584.3]$
warehouse	108	$+0.335$	-577	$-879.8\ [-4908.9, 3149.4]$

Table 31: All eight paired task differences, one changed action minus q-off, ordered by fault suffix replicate. The original E220 continuation was a different single fault tape and is shown in Table 30.

Map	Seed	Δ completed tasks for suffix replicates 0–7
room	106	$+625, +568, +723, +241, +616, -663, +705, -658$
room	108	$+620, +672, +960, +876, -147, +314, -103, -689$
random	106	$+334, +364, -5480, +5447, +5306, -4653, +5152, -1471$
random	108	$-1375, +365, -419, +4954, +523, -6325, -3488, +8064$
warehouse	106	$-3656, +318, +3451, +2389, +6697, +5760, -1779, -4291$
warehouse	108	$-5859, -5172, +8621, -3882, -224, +3225, -1226, -2521$

Table 32: Paired continuation summaries. Window entries are the means of task differences in successive 500-step windows; move differences sum over the full 3,000 steps. All per-pair windows and moves are in the compact analysis JSON.

Map	Seed	Mean move Δ	Mean task Δ by 500-step window
room	106	$+25{,}123.6$	$+5.5, +28.6, +73.2, +49.1, +27.4, +85.8$
room	108	$+26{,}305.5$	$+21.2, +4.0, +56.6, +78.0, +73.8, +79.2$
random	106	$+11{,}985.5$	$+24.4, +62.5, -1.6, -7.0, +275.9, +270.8$
random	108	$+28{,}775.0$	$-29.4, -58.0, -167.1, -270.9, +231.2, +581.5$
warehouse	106	$+104{,}567.0$	$-56.5, +288.2, +142.9, +155.2, +376.2, +205.0$
warehouse	108	$-32{,}032.4$	$-51.9, -304.6, -447.1, +25.4, -189.6, +88.1$

Q POST-OUTCOME MOTION ACCOUNTING FOR THE EXACT-INPUT TESTS

Scope and provenance. This analysis uses all completed room/1,000 and warehouse/1,800 exact-input tests from Section 5.2: six starts, twelve matched fault tapes per start, two arms, and 3,000 physical ticks per trajectory. The 144 pairs are the original tests, with no new trajectories or selected endpoints. Task and aggregate realized-motion results were already known. An initial every-tenth-tick readout was specified before reading these component statistics; the subsequent all-tick readout, initial-state comparison, six-window analysis, and proposal-composition analysis are explicitly post-outcome. We report the complete all-tick result.

An accounting identity. At each tick let A count robots whose proposed target differs from their current cell, F count proposed movers whose own fault bit is active, and R count robots whose realized cell changes. Define $C = A - F - R$: these are proposed movers with successful own actuators that stay because a required dependency fails. A mover with both an own fault and a failed dependency is assigned to F, so the categories are disjoint. Planned waits are outside A. Thus $R = A - F - C$ is an exact partition under the execution contract, not an additional causal model. All-tick realized totals match the existing independent replay audit. The analysis uses hidden fault bits offline to describe execution; these bits are not inputs to the learned policy.

Pairing and intervals. Arms have identical initial positions and goals, matched physical faults at every tick, and robot-indexed goal sequences. Their subsequent states, active goals, and planning histories may diverge. We first average the twelve paired differences within each start, then average the six start means equally. Descriptive intervals are $\bar{d} \pm t_{5,.975} s_d/\sqrt{6}$. For Table 33, counts and interval endpoints are divided by $3{,}000n/100$; all entries are percentage points of robots per physical tick. The interval does not account for post-outcome selection of explanatory metrics.

Table 33: Complete motion decomposition with descriptive 95% start-level intervals. Savings columns negate execution losses, so the first three columns sum to realized motion.

Setting	Quantity	Difference	95% interval
Room / 1,000	Proposals ΔA	-0.248	$[-0.511, +0.015]$
	Direct-failure savings $-\Delta F$	$+0.262$	$[+0.241, +0.283]$
	Cascade savings $-\Delta C$	$+1.648$	$[+1.491, +1.805]$
	Realized ΔR	$+1.662$	$[+1.307, +2.017]$
Warehouse / 1,800	Proposals ΔA	-4.713	$[-6.735, -2.691]$
	Direct-failure savings $-\Delta F$	$+0.262$	$[+0.182, +0.342]$
	Cascade savings $-\Delta C$	-0.152	$[-0.309, +0.004]$
	Realized ΔR	-4.603	$[-6.647, -2.559]$

Room loses $7{,}451.08$ proposed moves per trajectory but avoids $7{,}868.03$ direct failures and $49{,}436.57$ propagated blocks, for $49{,}853.51$ extra realized moves. Its cascade count decreases in six of six start means and 70 of 72 tapes. Warehouse loses $254{,}507.89$ proposed moves, avoids $14{,}169.26$ direct failures, and incurs $8{,}231.71$ more propagated blocks, for $248{,}570.33$ fewer realized moves. Its proposal count decreases in six of six starts and 53 of 72 tapes. The warehouse cascade-change interval includes zero. Degree-two-cell occupancy changes are -0.015 points $[-0.085, 0.054]$ in room and $+1.744$ $[-0.266, 3.754]$ in warehouse; these readouts do not establish a consistent narrow-cell congestion increase.

When the deficit appears. At the common initial state, each arm produces an identical proposal across its twelve fault tapes, leaving six distinct planning states per setting. In warehouse, exact identity changes a mean of 197.17 individual actions, but the proposed-mover-count differences over the six starts are $-1, +2, -5, -5, -4, -3$, a mean of -2.67. The full-trajectory difference is -84.84 movers per tick. Room's initial mean is -0.83, with action Hamming distance 174.67. Initial-state comparisons therefore locate the large warehouse activity deficit after the initially matched state; they do not identify which later state or planner-history feature produces it.

Warehouse's first-window proposal difference has interval $[-10.68, 7.58]$ movers per tick and realized-motion difference $[-6.26, 14.95]$. Its later proposal deficit grows across windows, while room cascade

Table 34: All six fixed 500-step windows. Motion entries are paired differences in movers per tick; task entries are paired completed-task counts per window from the original endpoint audit.

Readout	0–499	500–999	1000–1499	1500–1999	2000–2499	2500–2999
Warehouse ΔA	-1.55	-36.65	-88.72	-107.57	-126.44	-148.09
Warehouse ΔR	$+4.34$	-30.78	-90.00	-111.40	-131.69	-137.61
Room ΔC	-11.65	-13.50	-14.87	-19.68	-19.75	-19.44
Room ΔR	$+10.71$	$+12.97$	$+13.26$	$+21.09$	$+22.67$	$+19.01$
Warehouse tasks	$+11.9$	-163.6	-499.3	-625.7	-669.7	-641.8
Room tasks	$+58.7$	$+44.0$	$+44.0$	$+59.0$	$+75.8$	$+66.1$

savings persist throughout. This temporal description links the reported totals to their development along the trajectories; it does not identify causal mediation or imply that maximizing proposed moves would improve task throughput.

Which robots are proposed to move? An offline composition check sums true failure probabilities over proposed movers. Room's fragile-robot fraction among proposed moves changes from 20.644% to 18.756%, with paired change -1.888 points $[-1.989, -1.787]$; warehouse changes from 20.490% to 20.115%, or -0.375 points $[-0.444, -0.307]$. Both directions hold in every start. Mean expected direct-failure rates per proposal change from 4.526% to 4.157% in room and 4.496% to 4.422% in warehouse. These are means of trajectory-level rates. Expected direct-failure count differences, $-7{,}842.76$ and $-14{,}129.52$, are close to observed differences $-7{,}868.03$ and $-14{,}169.26$. The comparison describes the policy-dependent set of attempted moves, rather than attributing the effect to an immediate local preference at matched later states.

The companion motion-audit supplement includes all paired compact rows, analysis programs, readout protocols, and checksums. Original raw trajectories remain in the existing experiment archive. This addition changes no frozen task result or prospective-test status.