FROM PEER REPORTS TO REVISABLE INTENT EVIDENCE: GROUNDED PEER EVIDENCE REVISION FOR EMBODIED COLLABORATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Grounded Peer Evidence Revision (GPER) raises the rate of completed goal units in a Cooperative Embodied Language Agent (CoELA) pipeline by 3.3 and 2.9 points. The gains hold on two suites derived from the PARTNR and TDW-MAT simulators, with 300 and 200 episodes. Each suite carries ambiguity that a peer can resolve, a regime in which a supported report can still leave competing intentions open. The control receives the same qualified facts with no intent update (QF), and both combined 95% intervals exclude zero. On a diagnostic suite of 600 snapshots with matched inputs, GPER makes the fewest harmful revisions among the tested conditions. GPER is an add-on to CoELA. GPER separates whether observations support a claim from how far that claim, assumed true, shifts competing explanations of teammate behavior. Authorization never conditions on active intent preferences, and compatibility never observes support strength or source reputation. An adapter multiplies record support, freshness and a conflict flag into a support strength in $[0, 1]$. The support strength scales a compatibility score from an evaluator that never sees strength or source. A ledger indexed by events stores each contribution as an active event group and keeps that contribution revisable. We will make the code, prompt templates, diagnostic snapshots and logs for every call available upon acceptance.

1 INTRODUCTION

Suppose two agents retrieve tools and a peer reports that a cabinet is empty. Under partial observability the receiver must distinguish support for that claim from its bearing on the teammate's intention. Driess et al. (2023) plans against this setting. Zhang et al. (2024) runs a cooperative pipeline in it. Mandi et al. (2024) adds plan negotiation. An exterior view cannot establish an empty interior. An interior observation can go stale. A correct report may not exclude a retrieval intention. We call this *ambiguity that a peer can resolve*.

Agents powered by an LLM often pass peer messages straight to a planner (Zitkovich et al., 2023). Cooperative pipelines do the same (Zhang et al., 2024). A supported claim may nonetheless leave competing intent hypotheses unchanged. Support can be checked without modeling the teammate. Compatibility asks how the claim, assumed true, distinguishes candidate goals. Physical transitions invalidate earlier contributions, so a claim must be insertable, attenuable and withdrawable, not a permanent addition to the state.

We therefore propose Grounded Peer Evidence Revision (GPER), an interface keeping the two judgments separate in computation and combining them in one weighted update. Authorization determines whether a contribution is supported. Compatibility determines whether the supported claim shifts the plausibility of candidate intentions. Support strength scales the centered compatibility score. The revised state is kept in a ledger indexed by events. Two exclusions hold. Authorization never conditions on active intent preferences. Compatibility never observes authorization strength or source reputation. The interface operates on transmitted records rather than internals specific to a model. The interface also fits a range of cooperative pipelines, and we demonstrate it as an add-on to CoELA.

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.

One line acts on message content (Zhang et al., 2024). Another negotiates plans through a validator (Mandi et al., 2024). A separate line maintains a revisable teammate model (Cross et al., 2025). Credibility weights act on a source rather than a record (Ebrahimi et al., 2025). GPER revises neither a plan nor a teammate model but a single record's contribution. The question is when such a revision is licensed.

We test a screened suite derived from PARTNR (Chang et al., 2025) with 300 episodes. A second suite derived from TDW-MAT (Zhang et al., 2024) carries 200 episodes. We also test three adapted transfer probes and 24 official TDW-MAT episodes, and we report a separate contrast on the unscreened generator over 700 episodes. On the screened suites, GPER improves the rate of completed goal units over QF by 3.3 and 2.9 points at 2.6k to 2.7k extra tokens per episode. On the diagnostic suite with matched inputs it makes the fewest harmful revisions among the conditions compared. A control matched in call count recovers 1.0 and 0.8 points. The unscreened effects are 0.8 and 0.6 points with intervals including zero. On the official split, transport rate is 70.83%, below PCE's 87.50% (Seo et al., 2026). Conditions appear in Table 16. Code, prompts, diagnostic snapshots and logs for every call will be made available upon acceptance.

2 Related Work

Embodied multi-agent cooperation. LLM planners decompose long-term embodied goals into executable action sequences under partial observability (Driess et al., 2023). CoELA combines observation, memory, communication, planning and execution (Zhang et al., 2024). RoCo negotiates plans with a validator for motion planning, so it validates plans rather than the reports behind them (Mandi et al., 2024). LLaMAR plans, acts, corrects and verifies in a closed loop (Nayak et al., 2024). A-ToM aligns recursive teammate reasoning to the partner's order in Theory of Mind (Mu et al., 2026). COMBO learns a compositional world model with tree search guided by vision and language (Zhang et al., 2025a). All of these act on message content or plans rather than on the support behind a report. The support behind a report is the intermediate step we target.

Hypothesis modeling and message credibility. Work on peer reports divides by what it revises, either a teammate model or a source weight. Hypothetical Minds generates, evaluates and refines hypotheses about other agents' strategies stated in natural language, and it scores them by predictive success (Cross et al., 2025). Prompting with belief states for teammates in cooperative text games has been studied (Li et al., 2023). Neither line separates observational grounding from conditional bearing on intent. A second line learns a credibility weight for each agent from earlier contributions (Ebrahimi et al., 2025) and revises reliability adaptively (Antuley et al., 2026). Source weighting acts on a source rather than a record, so a generally reliable but here ungrounded agent goes undetected. We instead revise a record's contribution. Our "Source trust" control instantiates that alternative (Table 33).

Belief updating and reasoning about others. Our update revises a preference distribution by a factor scaled with the strength. That shape goes back to virtual evidence and to discounted evidence combination, for which we include a control (Pearl, 1988; Shafer, 1976). We borrow the shape and deploy it as an operation at the record level inside a pipeline, not as a calculus of belief revision. The remaining relatives in that tradition are alternatives we do not use (Chan & Darwiche, 2005; Bissiri et al., 2016; Howard, 1966; Raiffa & Schlaifer, 1961; Alchourrón et al., 1985; de Kleer, 1986). Jeffrey conditioning and fusion of subjective logic are two more (Jeffrey, 1965; Jøsang, 2016), the latter relevant to Appendix H.3.

A further group treats the teammate's state explicitly. CoBel-World maintains a symbolic belief state over environment and mental states from peer messages on TDW-MAT and C-WAH (Wang et al., 2025), and it is our closest published comparator. A related strand separates environmental state from another agent's information and objectives (Gmytrasiewicz & Doshi, 2005). Theory of Mind infers latent beliefs, goals and intentions from behavior (Baker et al., 2017). Unlike these joint models, GPER asks a narrower question. When does a received observation justify revising hypotheses about a teammate's intent, and how far should it move them. Appendix B states what we claim and do not.

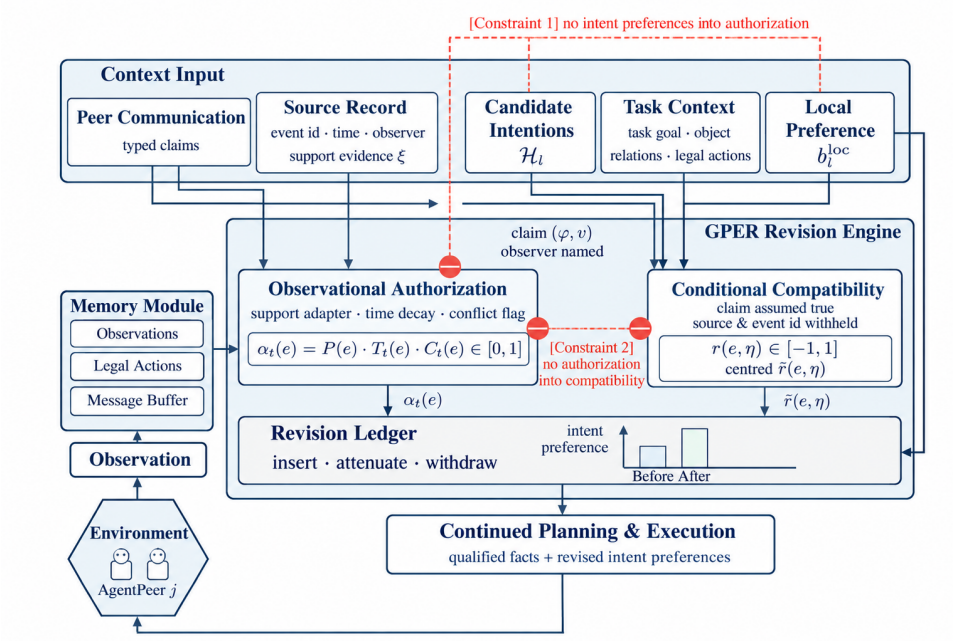


Figure 1: GPER revision engine (unified interface for revising evidence). Authorization licenses a record and produces a support strength in $[0, 1]$. Compatibility assigns a centered direction across candidate intentions. The ledger combines both with the local preference b_t^{loc}. Dashed red paths, each ending in a glyph for exclusion, are the structural exclusions of Constraints 1 and 2. Notes: Appendix A.

3 Problem Definition

A receiver in a cooperative embodied setting must sometimes commit to an irreversible action whose payoff depends on a teammate's hidden intention, and communication is costly. The receiver keeps a short list of candidate intentions for the teammate and a set of legal actions. Some legal actions are commitments. Once taken, a commitment permanently removes goal states the teammate could still have reached. Peer observations arrive only as transmitted records, so hidden goals, untransmitted memory and state that is global to the simulator never enter the runtime inputs and serve only as offline labels.

The case we study is *ambiguity that a peer can resolve*. At such a step two of the plausible intentions for the teammate, those in the plausible set $\mathcal{H}_t^{\text{pl}}$ defined by the plausibility threshold $p_{\min} = 0.15$ of Table 11, imply different commitments. The receiver cannot settle the choice by looking further on its own within the step horizon, while a single request to a peer does settle it. We use this criterion to build the test suites and never compute it at run time. Screening instead uses a reference quantity instantiated with simulator ground truth, which we call the certificate.

4 Method

GPER adopts the modular design common in embodied agent cooperation. The module set of Zhang et al. (2024) is Observation, Memory, Planning, Communication and Execution. Mandi et al. (2024) uses a comparable decomposition. The redesign turns the Communication Module into an Observational Authorization, Conditional Compatibility and Ledger pipeline. A supported record moves the intent state only in the direction it discriminates, and only while the observation behind it holds. Prompting details are in Appendix L, and the state is defined in Section 4.1.

GPER treats a received record as a contribution licensed by available support before it can move the intent state, and directed by conditional compatibility once licensed. A strongly supported claim therefore cannot buy a more favorable direction, the compatibility evaluator never seeing the support strength. A favorably scored claim cannot borrow credibility from its source, the score computed

without it. Figure 1 shows the two exclusions enforcing this. Authorization never conditions on active intent preferences (Constraint 1). Compatibility never observes authorization strength or source reputation (Constraint 2). A whitelist at the field level asserts both in every prompt constructor.

One received record passes through the pipeline in a fixed order. The adapter checks the observation behind it. A record that fails is dropped and nothing is written. A surviving record is assigned a strength from its support, its age and the conflict flag. Compatibility then scores the claim against each candidate intention, without seeing that strength or the source identity. The ledger stores the contribution under its event identifier, and a later invalidating observation withdraws it.

4.1 Intent state and the reference distribution

The planner produces candidate intentions, a local support value $s_t(\eta)$ for each, and a preferred action per hypothesis. These give the reference distribution b_t^{loc}, a softmax over the local support scores with temperature $\tau_b = 0.5$, mixed with a uniform component of weight $\epsilon_b = 10^{-3}$ so that no candidate is ruled out. Both constants are those of Table 11. The reference distribution represents local preferences rather than calibrated posteriors. Local support excludes premises derived from a peer and values revised on peer evidence, and the planner call never receives the preference distribution, so peer evidence enters the ledger exactly once.

The receiver queries a peer only when its state is genuinely split. Communication fires when the normalized intent entropy of the current preference distribution exceeds $u_0 = 0.7$ and the plausible intentions disagree about the action. A received claim is the unit of accounting for the ledger and the conflict check. The compatibility evaluator receives a strict subset of its fields, and Appendix F.1 reports the firing statistics.

4.2 Observational authorization: licensing a contribution

Authorization decides whether a contribution may be used at all, before any question of direction. Three factors enter. The first is whether the record is available and supports the claim. The second is how much it has decayed since acquisition. The third is whether a supported conflict overrides it.

Physical support. Physical support is a three-way conjunction, evaluated in code rather than by a model. The observation is available, its source has the required capability, and it supports the claim. Deterministic adapters check entity identifiers and verify that the record satisfies the relation's prerequisites exactly (Table 13). An absence claim therefore requires positive evidence of complete coverage, since failing to detect is not observing absence.

Temporal applicability. Applicability is one at acquisition and decays exponentially with the record's age in simulator steps, at the per-predicate time constant κ_ϕ of Table 11. It resets to zero once an invalidating transition fires (Table 13). Records dated beyond the current step are discarded.

Conflict flag. The conflict flag takes three values and comes from a separate call that checks conflicts. The flag is zero when a supported conflict overrides the claim, a midpoint value of $c_{\mathrm{amb}} = 0.5$ for unclear priority, and one when none qualifies. The flag is the only factor the model generates, the other two being computed in code.

Support strength. Write e for a received claim, t_e for its acquisition step and t for the current decision step, so that $t - t_e$ is the record's age in simulator steps. The three factors are physical support $P(e)$, temporal applicability $T_t(e)$ and the conflict flag $C_t(e)$. The three factors combine into one scalar, shared by every condition retaining the adapter.

$$\alpha_t(e) = P(e)T_t(e)C_t(e) \in [0, 1]. \tag{1}$$

The remaining conditions replace this product as listed in Table 16. The multiplicative form is a design choice rather than a derived likelihood ratio. Here P is a hard admissibility predicate, T a freshness decay and C a conflict flag that takes three values. Appendix N.3 reports how well the resulting score tracks an empirical estimate of the same quantity.

Authorization never sees current intent preferences or records derived from the claim being checked. A claim failing the support check receives $\alpha_t(e) = 0$, and the system records neither the claim nor its negation, since failing to verify the claim is not evidence for its negation.

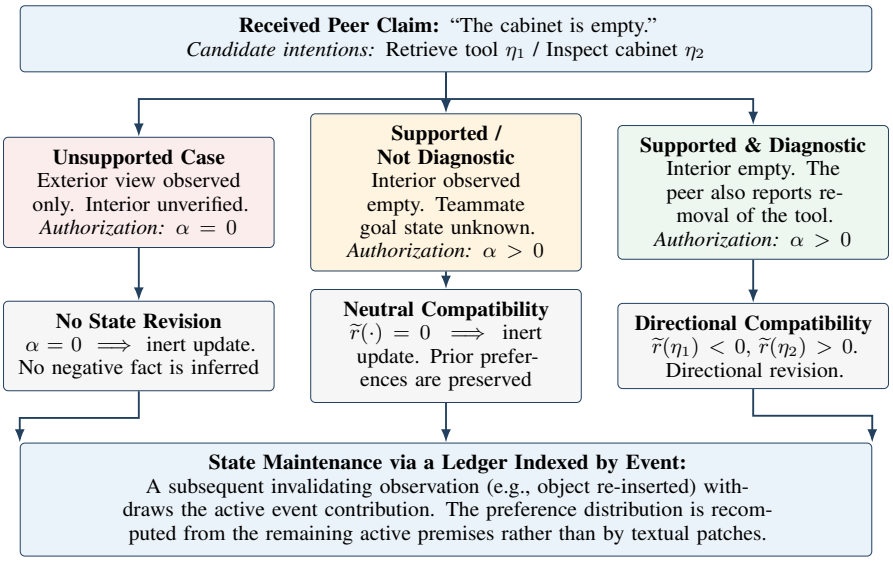


Figure 2: A taxonomy of evidence revision under identical message surface form. Notes: Appendix A.

4.3 Conditional compatibility: assigning a direction

A separate evaluator scores each candidate intention. The score $r(e, \eta) \in [-1, 1]$ is negative against candidate η and positive in its favor. The evaluator reads the claim, the task goal and object relations, the legal actions, and the candidate intentions. The evaluator assumes the claim correct and is blind to support strength and source.

Isolation is enforced by field exclusion. The compatibility call receives the claim with the observer named in it, the goal and legal actions, and the candidate intentions, but no support, source or record field (Table 51). The event identifier is replaced by a random opaque reference mapped internally to the real one, so neither source nor acquisition step is recoverable from the rendered text.

The claim *cabinet is empty* lowers the plausibility of a retrieval intention only if the teammate is known to have observed the cabinet. The claim also names that observer, so the link is carried by the claim rather than the source field. Absent the link the claim is neutral across hypotheses, the centered score is zero, and the update is inert. Figure 2 shows this case and its two neighbours under an identical surface form. The centered score, written $\widetilde{r}(e, \eta)$, subtracts from each candidate's compatibility the average over all candidates. Its sign gives the direction of the revision, and it vanishes whenever a claim is equally compatible with every candidate. Appendix H.4 probes the boundary in both directions.

Compatibility supplies a direction but not a warrant. Whether a direction that held at one step still holds after the next transition is a question the evaluator is never asked.

4.4 A ledger indexed by event: keeping contributions revisable

A ledger indexed by event $\mathcal{M}_t$, the set of active event groups, keeps the active contributions. Each contribution multiplies the current preference by a factor that grows with its strength and its centered compatibility, scaled by a single revision coefficient β, and the result is renormalized. Every deployed decision step begins from b_t^{loc} of Section 4.1, so the update of a single record and the ledger share the same prior.

Contributions depending on one physical event form a group. A group's strength $\alpha_{t,g}$ is its weakest member's, and its direction $\widetilde{r}(g, \eta)$ is the average over its members. Contributions from distinct events add in the exponent, giving the combined preference distribution over the active events.

$$b_t(\eta) \propto b_t^{\mathrm{loc}}(\eta) \exp\left[\beta \sum_{g \in \mathcal{M}_t} \alpha_{t,g} \widetilde{r}(g, \eta)\right]. \tag{2}$$

A forwarded copy keeps the original event identifier, replacing the original contribution rather than adding a second. Invalidated events are removed immediately, and every other event once its strength falls below 0.01. Exact deduplication by event ID does not catch two peers reporting the same fact under different identifiers, and Appendix H.3 reports the resulting overconfidence.

A single contribution cannot move the preference state arbitrarily far. Because every candidate keeps strictly positive mass, the relative preference between any two candidates shifts by at most twice the revision coefficient multiplied by that record's own strength. A single report therefore cannot overturn a strong preference.

Appendix K reports the derivation and evaluates the sum over the active set against the realized ledger. The guarantee covers one contribution only, and it does not bound the realized policy.

5 EXPERIMENT

Benchmarks. One controlled ambiguity set comes from the PARTNR simulator (Chang et al., 2025). A second comes from TDW-MAT (Zhang et al., 2024). The sets carry 300 and 200 episodes in 30 and 20 scene clusters, and each cluster concentrates decisions of the kind in Section 3.

We also report three adapted transfer probes. One comes from MAP-THOR (Nayak et al., 2024). A second comes from TEACh (Padmakumar et al., 2022). A third comes from VLABench (Zhang et al., 2025b). For the 700 unscreened generated episodes we report the primary contrast only. We report the 24 official **TDW-MAT Benchmark** episodes under the standard transport-rate metric (Zhang et al., 2024) (Appendix E.1).

Screening selects on the phenomenon the method targets. The certificate flags only 2.2% of decision steps on the official TDW-MAT episodes, and 7.3% on the PARTNR validation split (Appendix F.9). The flagged steps are too sparse to estimate an effect there. Without screening the primary contrast falls to under a point (Table 21). Both intervals include zero, so the effect holds inside the targeted regime rather than on the natural distribution.

Derived episodes are nine times shorter than official ones (Table 20). MAP-THOR, TEACh and VLABench are adapted transfer probes, not native benchmarks for peer collaboration. Two of the three lack teammate intent that is relevant to the task, and they serve as negative controls (Appendix C).

Metrics. All methods share initial states and teammate controllers within a set. *Score* is the percentage of completed goal units. A goal unit is an assigned subgoal milestone in episodes derived from PARTNR, or an item delivered to the target drop-off container in ones derived from TDW-MAT. *Risk count* counts episodes with an incorrect commitment. An incorrect commitment is an irreversible action that causes permanent task failure or makes the true teammate goal unreachable. *Timeout count* counts incomplete episodes reaching the step horizon.

Risk is not a second success measure but a failure mode the score aggregates away. Conditions can trade one for the other, as the diagnostic suite shows. We therefore read risk against the score, not against other methods.

A *scene cluster* is ten instances sharing one floorplan and room topology. The instances vary object spawn poses and the teammate controller, a fixed family of six scripted controllers spanning cooperative, self-interested and erratic behavior. The internal goal state of a cluster defines the intention treated as ground truth, used only for offline labels.

A *decision step* is one planning cycle. A *simulator step* is one primitive action execution. The horizon and the freshness decay both use the scale of simulator steps. The diagnostic instances of Table 2 carry two to three injected claim bundles each. The token cost of a single instance is therefore not comparable to the closed-loop average.

Baselines. We compare GPER with the cooperative pipelines of Section 2 and with controls that each isolate one design choice, all defined in Table 16. Two controls carry the paired contrasts. Qualified Facts Only (QF) supplies the same authorized peer reports without an intent update. A variant matched in call count takes the same compatibility call on the same inputs but performs no belief update.

Table 1: Closed-loop comparison on the two controlled sets. Score is in percent. Best score and lowest risk are in bold. The † symbol marks ports run by the authors. Paired differences against QF are reported only for the focal comparisons; dashes mark the remaining methods. A positive Δ Score and a negative Δ Risk indicate improvement relative to QF. Notes: Appendix A.

Environment	Method	Score	Risk	Timeout	Tokens (k/episode)	Δ Score vs. QF (paired)	Δ Risk vs. QF (paired)
PARTNR-derived	LLaMAR†	58.9	71/300	34/300	22.6	–	–
	CoELA†	59.4	68/300	32/300	19.9	–	–
	RoCo†	60.7	61/300	31/300	24.3	–	–
	Source trust	61.5	56/300	28/300	21.4	–	–
	A-ToM†	62.6	59/300	27/300	26.7	–	–
	PCE†	63.1	57/300	29/300	23.0	–	–
	Four-check	64.5	47/300	27/300	25.6	–	–
	COMBO† (structured-state)	65.4	50/300	24/300	23.9	–	–
	Frozen Intent State	66.4	43/300	25/300	27.3	–	–
	Qualified Facts Only (QF)	67.1	40/300	24/300	26.8	Reference	Reference
	CoBel-World†	67.7	36/300	26/300	32.4	$+0.6, [-0.6, 1.8]$	$-1.3, [-4.5, 1.9]$
	QF (call-matched)	68.1	34/300	25/300	28.3	$+1.0, [0.1, 1.9]$	$-2.0, [-4.8, 0.8]$
	GPER	**70.4**	**30/300**	26/300	29.4	$+3.3, [1.6, 5.0]$	$-3.3, [-6.5, -0.2]$
TDW-MAT-derived	LLaMAR†	56.4	50/200	25/200	22.9	–	–
	CoELA†	57.5	46/200	24/200	20.2	–	–
	Source trust	59.8	39/200	23/200	22.0	–	–
	RoCo†	60.9	38/200	21/200	24.6	–	–
	PCE†	62.1	37/200	22/200	23.2	–	–
	A-ToM†	63.2	35/200	20/200	27.1	–	–
	Four-check	63.4	32/200	20/200	25.9	–	–
	COMBO† (structured-state)	64.4	30/200	19/200	24.1	–	–
	Frozen Intent State	66.2	28/200	18/200	27.6	–	–
	Qualified Facts Only (QF)	66.8	26/200	17/200	27.0	Reference	Reference
	CoBel-World†	67.3	24/200	20/200	31.8	$+0.5, [-1.1, 2.1]$	$-1.0, [-4.0, 2.0]$
	QF (call-matched)	67.6	23/200	18/200	28.5	$+0.8, [-0.3, 1.9]$	$-1.5, [-4.4, 1.4]$
	GPER	**69.7**	**21/200**	18/200	29.7	$+2.9, [0.9, 4.9]$	$-2.5, [-6.3, 0.9]$

The remaining controls substitute the adapter with a weight derived from source accuracy. Other controls delegate the support judgment to a single LLM call. One control applies the Shafer transform to the same support product. The last control ablates the intent update and the text of the qualified facts.

Ports run by the authors, marked †, lack verified fidelity to the native benchmark. Those ports therefore support an ordering within our harness only (Appendix E). Table 4 reports the re-runs matched in call count.

LLMs and statistical protocol. The reference model is `gpt-4o-mini-2024-07-18`. Additional conditions use `Qwen/Qwen2.5-72B-Instruct` and `Qwen/Qwen2.5-7B-Instruct` (Appendix G.3). Token counts are comparable only within a model, since prompts and parameters are fixed across conditions.

The primary confirmatory contrast is GPER versus QF over five inference seeds on the full episode sets, paired by scene cluster. Each interval combines a component from a bootstrap over clusters with a component from repeated inference in quadrature (Appendix G). With 20 to 30 clusters a percentile interval can fail to cover the true value. An interval whose half-width is comparable to the effect it brackets is inconclusive. All diagnostic comparisons are paired, using McNemar tests or Wald intervals built from discordant counts whose components and values for each seed are in Appendix G.

5.1 Comparative results

In Table 1, GPER has the highest score and lowest risk count on both sets. Against QF, score rises by 3.3 and 2.9 points, with intervals excluding zero. Risk reduction resolves only on episodes derived from PARTNR. Two environments do not support estimating variance between environments.

The gains stem from keeping a report's grounding separate from its bearing on intent. Four-check, the strongest internal control that replaces the adapter with a single LLM verifier call, sits several points below GPER in both sets. Frozen Intent State retains the qualified facts but never revises the intent state, and its call budget matches that of QF. CoBel-World, the closest published system, sits marginally above QF with intervals including zero, and it spends more tokens per episode than GPER.

Table 2: Revision controls with matched inputs on 600 snapshots. Lower is better. Best in each column in bold. Notes: Appendix A.

Method	Errors/600	CE/420	HR/380	Balanced	Tokens (k)
Unqualified update	205	244	103	0.426	8.1
Fixed strength ($\alpha = 0.5$)	181	225	76	0.368	8.6
Source trust	172	207	80	0.352	9.2
CoELA (ported evaluator)	165	211	73	0.347	11.2
Thresholded verifier	158	215	52	0.324	10.5
Soft verifier	143	240	58	0.362	11.8
PCE (ported assumption tree)	138	234	56	0.352	12.8
Four-check	133	**175**	54	**0.279**	13.4
Frozen Intent State	130	250	47	0.359	13.2
Dempster–Shafer discounting	128	255	46	0.364	13.0
Qualified Facts Only	125	257	43	0.363	12.7
Revised State Only	119	264	40	0.367	12.1
CoBel-World (ported)	116	268	35	0.365	13.6
GPER	**102**	245	**29**	0.330	14.3

An interface that keeps the two judgments separate costs more per step than piping message content to the planner. The QF control matched in call count takes the same compatibility call on the same inputs and recovers about a third of the gain. The residual difference is the revision and the ledger. Raising Four-check to our budget moves it up in both environments, but the residual exceeds what the match accounts for. The gain over QF costs about the same in extra tokens as the jump from QF to the call-matched control, and it brings two additional timeouts.

5.2 Mechanism analysis and revision controls

Snapshots are fixed input states with a designated reference intention. An error is the event the Risk metric counts on an episode. Errors are recorded offline, since snapshots are not executed.

With the input fixed, differences cannot come from where the agent went or what it observed. Differences can only come from what the condition does with the same record.

The suite holds two overlapping populations. Instances of correction error, written CE, start incorrect, and they carry supported opposing evidence that a condition revising well should repair. Instances of harmful revision, written HR, start correct, and they carry unsupported opposing reports that an overeager condition should break. The two columns measure opposite failure modes, and the error total decomposes exactly as in Appendix F.3.

One caveat is structural. The labels come from the same harness that supplies support metadata to the adapter. The contrast therefore partly tests agreement with that rule. The adapter's advantage over the LLM verifier reverses at 35% noise in coverage (Appendix F.4).

Explicit revision improves on the error total over the same qualified facts. Revised State Only keeps the revised state and ledger but withholds the text of the qualified facts. Revised State Only recovers part of that gain. GPER commits the fewest harmful revisions, and paired tests on that column favor it against both QF and Four-check with intervals excluding zero (Table 31).

The HR instances start correct. Any error they acquire is therefore one the condition introduced. The HR column gives the cleanest attribution in the suite.

The error total and the correction column rank conditions differently. CE mixes two subpopulations of different difficulty. A leading error total is therefore not a uniform advantage across both failure modes.

Ranking on the error total favors GPER more than a balanced criterion over the CE rate and the HR rate. On the balanced criterion GPER ranks third of fourteen, behind Four-check and the thresholded verifier. Four-check is the only condition whose margin there has a paired interval excluding zero (Table 23).

The Dempster–Shafer control, which uses no LLM, lands within a few errors of QF. At the conservative dose that transform implies, the discounted support product alone does not improve over the qualified facts. Without a variant matched in dose this is not a test of fusion at the deployed dose.

CoBel-World is the strongest external result here, but its balanced rate leaves the two unseparated at this sample size. Isolation of prompt fields is varied only in the experiment on the organization of the

Table 3: Component interventions on common instances with fixed inputs. Paired Δ intervals vs GPER. The GPER row is the reference row, and the b and c columns follow the convention of Appendix F.8.

Condition	Errors/600 [95% CI]	Paired vs GPER Δ [95% CI]	b	c	CE/420	HR/380	Balanced
GPER	**102** [84, 120]				245	**29**	0.330
No physical support	140 [120, 160]	+38 [17, 59]	76	38	225	63	0.351
No temporal applicability	128 [108, 148]	+26 [8, 44]	55	29	233	50	0.343
No independent conflict check	123 [104, 142]	+21 [4, 38]	48	27	239	46	0.345
No invalidation or withdrawal	127 [107, 147]	+25 [7, 43]	55	30	237	48	0.345
No event deduplication	114 [95, 133]	+12 [−2, 26]	31	19	259	41	0.362
No compatibility refresh	120 [101, 139]	+18 [2, 34]	42	24	246	42	0.348
Binarized authorization	145 [124, 166]	+43 [20, 66]	90	47	208	58	**0.324**

evaluator (Appendix N.1). That experiment is a point estimate whose paired interval includes zero. Further controls appear in Appendices H and N.

5.3 Component ablations

On 600 paired snapshots with fixed inputs (Table 3), every intervention increases total errors. Both paired and marginal intervals are shown.

The two most damaging interventions remove the two ends of the authorization factor. Binarizing α and dropping the adapter's physical support predicate are statistically indistinguishable. Losing graded support strength therefore costs about as much as losing the support check itself.

Removing temporal applicability, withdrawal, conflict checking or compatibility refresh increases errors with paired intervals excluding zero. Removing event deduplication does not establish a paired difference. The value of event deduplication remains undemonstrated.

The two largest degradations of the error total produce the lowest correction counts, the same trade-off seen in Table 2.

On the balanced column every intervention except binarized authorization is worse than the full method. Binarized authorization is marginally better instead. Coarsening support strength buys fewer harmful revisions at the cost of many more unpaired ones. The design with two columns was meant to expose this trade-off.

6 Conclusion

GPER separates observational grounding from intent compatibility in a weighted, revisable event ledger. On screened suites it raises the rate of completed goal units by 3.3 and 2.9 points at 2.6k to 2.7k extra tokens per episode. Combined 95% CIs are $[1.6, 5.0]$ and $[0.9, 4.9]$ over five seeds. A control matched in call count recovers 1.0 and 0.8 points. All component interventions increase diagnostic total errors, and baselines matched in call count remain behind GPER.

The scope is limited. Isolation of prompt fields is unresolved. Unscreened effects are $+0.8$ and $+0.6$ points with intervals including zero. On 24 official episodes, transport rate rises from 66.67% to 70.83% (interval including zero), below PCE's 87.50%. Four-check wins the balanced diagnostic criterion, where GPER ranks third of fourteen. As an add-on to CoELA, GPER's gains on the controlled sets apply to the regime where a peer can resolve the ambiguity. Future work should test native benchmarks, replace the certificate instantiated from the simulator, and address double counting of correlated evidence (Appendix H.3). Limitations appear in Appendix M.

AI Use Statement

We used AI to assist with language polishing and editing, including grammar, clarity, and conciseness. We did not use AI tools for generating research ideas, designing methods, conducting experiments, analyzing data, or writing the core technical content. All AI-assisted text was reviewed, verified, and edited by the authors. The authors take full responsibility for the final content.

REFERENCES

Carlos E. Alchourrón, Peter Gärdenfors, and David Makinson. On the logic of theory change: Partial meet contraction and revision functions. *The Journal of Symbolic Logic*, 50(2):510–530, 1985. doi: 10.2307/2274239.

Usama Antuley, Shahbaz Siddiqui, Sufian Hameed, Waqas Arif, and Syed Attique Shah. SORA-ATMAS: Adaptive trust management and multi-LLM aligned governance for future smart cities. *Knowledge-Based Systems*, 337:115403, 2026. doi: 10.1016/j.knosys.2026.115403.

Chris L. Baker, Julian Jara-Ettinger, Rebecca Saxe, and Joshua B. Tenenbaum. Rational quantitative attribution of beliefs, desires and percepts in human mentalizing. *Nature Human Behaviour*, 1(4): 0064, 2017. doi: 10.1038/s41562-017-0064.

Pier Giovanni Bissiri, Chris C. Holmes, and Stephen G. Walker. A general framework for updating belief distributions. *Journal of the Royal Statistical Society: Series B*, 78(5):1103–1130, 2016. doi: 10.1111/rssb.12158.

Hei Chan and Adnan Darwiche. On the revision of probabilistic beliefs using uncertain evidence. *Artificial Intelligence*, 163(1):67–90, 2005. doi: 10.1016/j.artint.2004.09.005.

Matthew Chang, Gunjan Chhablani, Alexander Clegg, Mikael Dallaire Cote, Ruta Desai, Michal Hlavac, Vladimir Karashchuk, Jacob Krantz, Roozbeh Mottaghi, Priyam Parashar, Siddharth Patki, Ishita Prasad, Xavier Puig, Akshara Rai, Ram Ramrakhya, Daniel Tran, Joanne Truong, John Turner, Eric Undersander, and Tsung-Yen Yang. PARTNR: A benchmark for planning and reasoning in embodied multi-agent tasks. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/a3cf318fbeec1126da21e9185ae9908c-Abstract-Conference.html.

Logan Cross, Violet Xiang, Agam Bhatia, Daniel L. K. Yamins, and Nick Haber. Hypothetical minds: Scaffolding theory of mind for multi-agent tasks with large language models. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/12f483f624b378f9f3058d8ecd3c7ff5-Abstract-Conference.html.

Johan de Kleer. An assumption-based TMS. *Artificial Intelligence*, 28(2):127–162, 1986. doi: 10.1016/0004-3702(86)90080-9.

Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Pete Florence. PaLM-E: An embodied multimodal language model. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 8469–8488, 2023. URL https://proceedings.mlr.press/v202/driess23a.html.

Sana Ebrahimi, Mohsen Dehghankar, and Abolfazl Asudeh. An adversary-resistant multi-agent LLM system via credibility scoring. In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pp. 1676–1693, 2025. doi: 10.18653/v1/2025.ijcnlp-long.90.

Piotr J. Gmytrasiewicz and Prashant Doshi. A framework for sequential planning in multi-agent settings. *Journal of Artificial Intelligence Research*, 24:49–79, 2005. doi: 10.1613/jair.1579.

Ronald A. Howard. Information value theory. *IEEE Transactions on Systems Science and Cybernetics*, 2(1):22–26, 1966. doi: 10.1109/TSSC.1966.300074.

Richard C. Jeffrey. *The Logic of Decision*. McGraw-Hill, New York, 1965.

Audun Jøsang. *Subjective Logic: A Formalism for Reasoning Under Uncertainty*. Artificial Intelligence: Foundations, Theory, and Algorithms. Springer, 2016. doi: 10.1007/978-3-319-42337-1.

Huao Li, Yu Chong, Simon Stepputtis, Joseph Campbell, Dana Hughes, Charles Lewis, and Katia Sycara. Theory of mind for multi-agent collaboration via large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 180–192, 2023. doi: 10.18653/v1/2023.emnlp-main.13.

Jie Liu, Pan Zhou, Yingjun Du, Ah-Hwee Tan, Cees G. M. Snoek, Jan-Jakob Sonke, and Efstratios Gavves. CaPo: Cooperative plan optimization for efficient embodied multi-agent cooperation. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/b07091c16719ad3990e3d1ccee6641f1-Abstract-Conference.html.

Zhao Mandi, Shreeya Jain, and Shuran Song. RoCo: Dialectic multi-robot collaboration with large language models. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 286–299, 2024. doi: 10.1109/ICRA57147.2024.10610855.

Chunjiang Mu, Ya Zeng, Qiaosheng Zhang, Kun Shao, Chen Chu, Hao Guo, Danyang Jia, Zhen Wang, and Shuyue Hu. Adaptive theory of mind for LLM-based multi-agent coordination. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 29608–29616, 2026. doi: 10.1609/aaai.v40i35.40204.

Siddharth Nayak, Adelmo Morrison Orozco, Marina Ten Have, Vittal Thirumalai, Jackson Zhang, Darren Chen, Aditya Kapoor, Eric Robinson, Karthik Gopalakrishnan, James Harrison, Brian Ichter, Anuj Mahajan, and Hamsa Balakrishnan. Long-horizon planning for multi-agent robots in partially observable environments. In *Advances in Neural Information Processing Systems*, volume 37, pp. 67929–67967, 2024. doi: 10.52202/079017-2169.

Aishwarya Padmakumar, Jesse Thomason, Ayush Shrivastava, Patrick Lange, Anjali Narayan-Chen, Spandana Gella, Robinson Piramuthu, Gokhan Tur, and Dilek Hakkani-Tur. TEACh: Task-driven embodied agents that chat. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 2017–2025, 2022. doi: 10.1609/aaai.v36i2.20097.

Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988. ISBN 9781558604797.

Howard Raiffa and Robert Schlaifer. *Applied Statistical Decision Theory*. Division of Research, Graduate School of Business Administration, Harvard University, Boston, 1961.

SeungWon Seo, SeongRae Noh, Junhyeok Lee, SooBin Lim, Won Hee Lee, and HyeongYeop Kang. REVECA: Adaptive planning and trajectory-based validation in cooperative language agents using information relevance and relative proximity. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 23295–23303, 2025. doi: 10.1609/aaai.v39i22.34496.

SeungWon Seo, SooBin Lim, SeongRae Noh, Haneul Kim, and HyeongYeop Kang. From assumptions to actions: Turning LLM reasoning into uncertainty-aware planning for embodied agents. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=GODFBZhFcX. Planner-Composer-Evaluator (PCE).

Glenn Shafer. *A Mathematical Theory of Evidence*. Princeton University Press, 1976. ISBN 9780691081755.

Zhimin Wang, Duo Wu, Shaokang He, Jinghe Wang, Linjia Kang, Jing Yu, Kai Zhu, Jiawei Li, and Zhi Wang. Collaborative belief reasoning with LLMs for efficient multi-agent collaboration. *arXiv preprint arXiv:2509.21981v3*, 2025. URL https://arxiv.org/abs/2509.21981v3.

Hongxin Zhang, Weihua Du, Jiaming Shan, Qinhong Zhou, Yilun Du, Joshua B. Tenenbaum, Tianmin Shu, and Chuang Gan. Building cooperative embodied agents modularly with large language models. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/54b8b4e0b4ba4aad112e84f32e3b5dbb-Abstract-Conference.html.

Hongxin Zhang, Zeyuan Wang, Qiushi Lyu, Zheyuan Zhang, Sunli Chen, Tianmin Shu, Behzad Dariush, Kwonjoon Lee, Yilun Du, and Chuang Gan. COMBO: Compositional world models for embodied multi-agent cooperation. In *International Conference on Learning Representations*,

2025a. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/7d03c6bf9f07acb4038eea96c63db52d-Abstract-Conference.html.

Shiduo Zhang, Zhe Xu, Peiju Liu, Xiaopeng Yu, Yuan Li, Qinghui Gao, Zhaoye Fei, Zhangyue Yin, Zuxuan Wu, Yu-Gang Jiang, and Xipeng Qiu. VLABench: A large-scale benchmark for language-conditioned robotics manipulation with long-horizon reasoning tasks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 11142–11152, 2025b. doi: 10.1109/ICCV51701.2025.01037.

Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, Quan Vuong, Vincent Vanhoucke, Huong Tran, Radu Soricut, Anikait Singh, Jaspiar Singh, Pierre Sermanet, Pannag R. Sanketi, Grecia Salazar, Michael S. Ryoo, Krista Reymann, Kanishka Rao, Karl Pertsch, Igor Mordatch, Henryk Michalewski, Yao Lu, Sergey Levine, Lisa Lee, Tsang-Wei Edward Lee, Isabel Leal, Yuheng Kuang, Dmitry Kalashnikov, Ryan Julian, Nikhil J. Joshi, Alex Irpan, Brian Ichter, Jasmine Hsu, Alexander Herzog, Karol Hausman, Keerthana Gopalakrishnan, Chuyuan Fu, Pete Florence, Chelsea Finn, Kumar Avinava Dubey, Danny Driess, Tianli Ding, Krzysztof Marcin Choromanski, Xi Chen, Yevgen Chebotar, Justice Carbajal, Noah Brown, Anthony Brohan, Montserrat Gonzalez Arenas, and Kehang Han. RT-2: Vision-language-action models transfer web knowledge to robotic control. In *Proceedings of the 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 2165–2183, 2023. URL https://proceedings.mlr.press/v229/zitkovich23a.html.

Lizheng Zu, Lin Lin, Song Fu, Na Zhao, and Pan Zhou. Collaborative tree search for enhancing embodied multi-agent collaboration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 29513–29522, 2025. doi: 10.1109/CVPR52734.2025.02748.

A TABLE AND FIGURE NOTES

This appendix gathers the notes for the tables and figures, the extended related work, and a statement of the scope. The sections that follow cover the adapted transfer probes and the implementation details. The appendix also defines each baseline and control, and it records how the diagnostic suite was built. The paired statistical analysis and the mechanism controls come next. Later sections give the prompt contracts, together with their verbatim templates, and the update properties. The evaluation protocol and the limitations close the appendix. The tables hold the reported values. The prose gives the definition of each value and states the scope of the claim it supports.

Table 1 (closed-loop comparison). Score and Tokens are five-seed means (101, 203, 307, 409, 503) on the full episode sets, and Tokens are generated tokens per episode. Risk and Timeout are episode counts for the primary seed (101), so their differences are descriptive rather than inferential. Δ intervals combine paired 95% cluster-bootstrap percentile intervals (10,000 resamples) as in Section 5. Where available, they also include the paired repeated-inference component of Table 35. Bold marks the best score and the lowest risk count. Token and timeout columns are not bolded, because their preferred direction conflicts with the score objective. An empty cell indicates no reported value, including Δ columns of rows with no paired contrast against QF. Rows marked † are author-run ports whose native-benchmark fidelity is not verified. Those ports support an ordering within our harness only (Appendix E).

Table 4: Call-matched baselines on the two controlled sets, reference model.

Environment	Method	Score (matched)	Score (unmatched)	Change	Δ vs matched
PARTNR-derived	Four-check	65.2	64.5	$+0.7$	-5.2
	CoBel-World†	67.7	67.7	0.0	-2.7
TDW-MAT-derived	Four-check	64.1	63.4	$+0.7$	-5.6
	CoBel-World†	67.3	67.3	0.0	-2.4

Each comparator issuing fewer calls than GPER is re-run with its call budget raised to GPER's 4.1 calls per decision. The re-run adds the same compatibility-score call that the QF (call-matched) control adds, so the comparison is not confounded with inference budget. CoBel-World already issues 4.4 calls per decision, so its row is unmatched. Its gap to GPER is if anything understated. Δ is the score difference to GPER on the same episodes, and it is a point estimate. No paired interval is reported for these re-runs. Matching calls reduces the gap but does not close it.

Table 2 (matched-input revision controls). CE counts correction errors out of 420, and HR counts harmful revisions out of 380. Lower is better in both columns. *Thresholded verifier* binarizes the continuous verifier score at 0.5. *Dempster–Shafer discounting* applies the Shafer reliability transform to the same record-support product. The Balanced column is the unweighted mean $\frac{1}{2}(\mathrm{CE}/420 + \mathrm{HR}/380)$, the primary diagnostic endpoint. Errors/600 weights harmful revisions 380 against 220, which is why that endpoint differs. The columns do not agree on one winner, and GPER has the lowest Errors and HR. The Balanced column places GPER third, behind Four-check and the thresholded verifier. Four-check has the lowest CE, while Source trust and the PCE port tie at 0.352.

Table 3 (component interventions). *Binarized authorization* replaces α with $\mathbb{I}[\alpha \geq 0.5]$. Paired Δ intervals are paired-difference Wald intervals $\Delta \pm 1.96\sqrt{b+c}$ against GPER over the shared 600 instances. Here $b + c$ is the discordant count, with the same b/c convention as Table 29. Bold marks the best Errors, the lowest HR and the lowest Balanced. As in Table 2, no condition wins all columns.

Figure 1 (interface). Solid arrows denote information flow, and dashed red arrows denote the structural exclusions enforcing information isolation. Authorization determines $\alpha_t(e)$ without conditioning on active intent preferences (Constraint 1). Compatibility scores $r(e, \eta)$ are computed without observing authorization strength or source reputation (Constraint 2). Constraint 1 is interpreted at the conflict-check call, and Constraint 2 is interpreted at the compatibility call. A field-level whitelist enforces both, and every prompt constructor applies it at run time (Appendix L). The ledger keeps contributions revisable until a commitment action is selected.

Figure 2 (revision taxonomy). The same sentence can be unsupported, supported but nondiagnostic, or supported and intent-discriminating. Factual qualification controls contribution weight, and conditional compatibility determines revision direction. A single contribution is bounded by $2\beta\alpha_t(e)$

(Appendix K). A single contribution can therefore reverse only hypothesis pairs whose prior log-odds fall below that bound (Table 19). Invalidating observations withdraw the contribution, and the preference distribution is recomputed from the remaining active premises.

Replacement rules for Eq. 1. Unqualified updating uses $\alpha = 1$, and fixed strength uses $\alpha = 0.5$. Source trust uses a Laplace-smoothed source-accuracy weight and skips the adapter, which isolates source reliability from record-level grounding. Dempster–Shafer discounting applies the Shafer reliability transform with $\alpha = P(e)T_t(e)$, which isolates the combination rule rather than the support check. Four-check delegates the support judgement to one LLM call, as do the thresholded and soft verifiers. Four-check asks for the three support predicates of Eq. 1 plus the conflict flag $C_t(e)$, and the harness multiplies them. The thresholded and soft verifiers ask for a single continuous score in $[0, 1]$, and they differ only in how the harness uses it. In all these conditions $r(e, \cdot)$ is returned by the continuation call. Their per-decision call count is therefore that of QF (Table 8). "CoELA (ported evaluator)" is the evaluator the CoELA port uses instead of the adapter. That evaluator is one call deciding whether the message reaches the planner, with no support predicates and no conflict flag. Appendix N.3 reports how well $\beta\alpha\tilde{r}$ tracks an empirically estimated log-likelihood ratio. The exponent explains under 40% of that ratio's variance, so Eq. 2 is a heuristic scoring rule.

Combined intervals. Task variability uses 10,000 bootstrap resamples of paired scene clusters (seed 20260920) with empirical 95% percentile intervals. Each half-width divided by 1.96 is $\mathrm{SE}_{\mathrm{cluster}}$. The repeated-inference component is $\mathrm{SE}_{\mathrm{inference}} = \mathrm{Paired}\,\Delta\,\mathrm{SD}/\sqrt{R}$ with $R = 5$. The two combine as $\mathrm{SE}_{\mathrm{total}} = \sqrt{\mathrm{SE}^2_{\mathrm{cluster}} + \mathrm{SE}^2_{\mathrm{inference}}}$. The seed fixes the LLM sampling stream, the spawn-seed list and the teammate-controller draw jointly. Per-seed differences are therefore paired and strongly correlated across rows (-0.90 to $+0.98$, Appendix G.2). That correlation makes the quadrature combination conservative, and we do not report a two-level (cluster $\times$ seed) bootstrap. Risk intervals use the same paired cluster bootstrap, but the risk difference carries no inference-variability component. Table 36 gives both components for every contrast with a reported combined interval.

B Extended Related Work and Scope of the Claim

What we claim and what we do not. We claim no new combination rule. Eq. 2 is a virtual-evidence update (Pearl, 1988), and Eq. 1 is a product of three hand-set factors with no claimed probabilistic reading. The ledger's operations are expansion, revision and contraction, built on an assumption structure (Alchourrón et al., 1985; de Kleer, 1986). Two claims are empirical and narrow. First, the two quantities are elicited by separate prompts, and the prompt constructor asserts the field exclusion (Appendix L). The experiment that varies isolation itself is reported in Appendix N.1, and it is not resolved at our sample size. Second, the packaging and correlated-evidence stresses exercise the ledger's insert, attenuate and withdraw semantics (Appendix H.2, H.3). The ledger's deduplication clause is not shown to help on the diagnostic suite (Table 3). As a negative result, the isolation step is not resolved at the sample size we ran. An extra call per authorized claim recovers 1.0 of the 3.3 and 0.8 of the 2.9 closed-loop points (Section 5.1).

Controls we did not run. Two controls would test the remaining claim. The first is a Jeffrey-conditioning update with the same α and the same r (Jeffrey, 1965). The second is a subjective-logic discounting-plus-dependent-fusion rule (Jøsang, 2016). Neither is claimed as a result anywhere in this paper. Appendix M records the same gap from the limitations side. A subjective-logic fusion rule is the natural remedy for the double counting quantified in Appendix H.3, so its absence is both a missing control and a missing fix.

Prior work we do not compare against method-by-method. We do not compare against LLM hypothesis-refinement pipelines of the Hypothetical Minds type (Cross et al., 2025). That pipeline is the closest published analogue of an explicit intent state, and Table 16 lists its port as not evaluated. We also do not compare against per-agent credibility estimation (Ebrahimi et al., 2025; Antuley et al., 2026), and the Source trust control instantiates that idea inside our harness (Table 33). None of these separates observational grounding from conditional bearing on intent, the decomposition this paper studies.

Table 5: Adapted transfer interaction probes. Scores are adapter-mediated and are not comparable with published results on the corresponding environment.

Environment	Method	Score	Risk	Timeout	Tokens (k)	Δ vs QF [95% CI]	Δ vs QF (cm)
MAP-THOR	Qualified Facts Only	65.9	36/250	21/250	27.1	Reference	
	Frozen Intent State	65.3	38/250	22/250	27.8	$-0.6\,[-2.6, 1.4]$	
	CoBel-World (ported)	66.4	34/250	22/250	30.4	$+0.5\,[-0.7, 1.7]$	
	LLaMAR (ported)	63.7	41/250	24/250	22.6	$-2.2\,[-4.6, 0.2]$	
	GPER	**68.0**	**33/250**	22/250	29.8	$+2.1\,[0.5, 3.7]$	$+1.4$
TEACh	Qualified Facts Only	65.4	34/240	23/240	26.5	Reference	
	Frozen Intent State	65.2	36/240	24/240	27.2	$-0.2\,[-2.1, 1.7]$	
	CoBel-World (ported)	65.8	33/240	24/240	29.7	$+0.4\,[-1.5, 2.3]$	
	QF (call-matched)	66.3	33/240	23/240	28.0	$+0.9\,[-0.2, 2.0]$	Reference
	GPER	**66.5**	**32/240**	23/240	29.1	$+1.1\,[-0.5, 2.7]$	$+0.2\,[-1.5, 1.9]$
VLABench	Qualified Facts Only	65.1	39/260	25/260	27.3	Reference	
	Frozen Intent State	64.8	41/260	26/260	27.9	$-0.3\,[-2.4, 1.8]$	
	CoBel-World (ported)	66.0	37/260	26/260	29.5	$+0.9\,[-0.7, 2.5]$	
	QF (call-matched)	67.4	36/260	25/260	28.2	$+2.3\,[0.6, 4.0]$	Reference
	GPER	**67.6**	**35/260**	25/260	29.6	$+2.5\,[0.9, 4.1]$	$+0.2\,[-1.4, 1.8]$

C Adapted Transfer Interaction Probes

This appendix documents the three adapted transfer probes and the rationale for each baseline coverage. MAP-THOR, TEACh and VLABench are not native symmetric peer-collaboration benchmarks, and they appear only as probes of GPER under adapted interaction protocols. All numbers are adapter-mediated, so they are comparable only with the internal controls in the same table. TEACh and VLABench are negative controls, because the injected peer's goal does not enter the receiver's task. No revision of a teammate intention can therefore affect the outcome, and the third probe is the only informative one. In VLABench the robot is the sole actor, so the intent-revision component is inert by construction. Only the qualified-fact channel can act in that probe. TEACh is an offline dialogue dataset whose Commander has oracle access, so every claim passes the support check and the authorization stage never rejects. The per-episode Risk and Timeout counts of Table 6 are defined by the interaction adapter, and they do not correspond to native TEACh episodes. For both probes Risk counts an irreversible action that makes the scripted peer's goal index unreachable. Risk here is a property of the adapter rather than of a cooperative task, since that goal index does not affect the receiver. The negative-control prediction is specific and falsifiable. On TEACh and VLABench, GPER should match a call-matched QF control, and any QF-relative gain should disappear once the extra call is granted to the control.

C.1 Probe construction and baseline coverage

Peer reports are injected through the interaction adapter of Table 6. For TEACh and VLABench the injected peer is a scripted controller of the same six-controller family used in the controlled sets. The reference intention labels used to construct the adapter's success criterion come from that controller's internal goal index. MAP-THOR is the only probe with a native external comparator, because LLaMAR is native to that environment (Table 15) and is reported there. For TEACh and VLABench any external peer-collaboration baseline would require changing the benchmark's fundamental interaction topology. We therefore report only the internal controls plus the CoBel-World port, and that port accepts adapter-mediated typed messages without changing the topology.

Bold marks the highest score and the lowest risk. Δ is the paired difference against QF on the same episodes for every row, including Frozen Intent State. The MAP-THOR call-matched QF reference that the last column uses (*cm* = call-matched) scores 66.6 and is not tabulated as a row. The MAP-THOR block reports the native comparator LLaMAR instead. The two probes with no teammate intention that can affect the receiver's task show a QF-relative gain that disappears against the call-matched control. The negative-control design predicts that pattern, and the MAP-THOR probe retains most of its gain. On those two probes the call-matched control's advantage over QF reflects the annotation channel of Table 16, not intent revision. An empty cell indicates that no paired contrast against the call-matched QF control is defined for that row.

The MAP-THOR probe includes an LLaMAR port, because LLaMAR is native to that environment and its planner directly processes message content. That practice is the one the paper contrasts with GPER. The LLaMAR port scores 63.7, below both internal controls. It is the one probe whose

Table 6: Adaptation protocol for non-native transfer probes.

Environment	Native formulation	Peer-report injection	Adaptation boundary
MAP-THOR	Multi-agent planning on AI2-THOR	Navigation agent receives typed peer state claims through the shared channel, and the peer is a scripted controller with an internal goal index	We evaluate 250 adapted episodes, against MAP-THOR's 45 official tasks. Results are therefore not comparable with published MAP-THOR numbers
TEACh	Asymmetric oracle Commander–Follower dialogue	Commander utterance mapped to a peer claim with the Follower as receiver; the Commander's oracle access means every claim passes the support check	Asymmetric dialogue, so intent revision cannot be exercised. Used as a negative control
VLABench	Single-robot language-conditioned manipulation	Peer state claims injected as messages with the robot as the sole actor	Single-robot manipulation, so intent revision cannot be exercised. Used as a negative control

comparator action space differs, so its interval is computed on the adapted action space and is not directly comparable with the others. The TEACh and VLABench rows behave as the design predicts. Against QF they appear to gain $+1.1$ and $+2.5$. Against the call-matched control the gains fall to $+0.2$ on both, with intervals spanning zero. Because no teammate intention that affects the receiver is present in either probe, we read those rows as evidence about the qualified-fact channel. The additional call, not intent revision, produces the QF-relative movement. The VLABench row is the sharpest case, because its QF-relative interval excludes zero ($+2.5$ $[0.9, 4.1]$) while the same comparison against the call-matched control does not. A probe with no teammate cannot support an intent-revision claim. MAP-THOR does contain a scripted teammate with an internal goal index determined by the task, and it retains $+1.4$ against the call-matched reference. Its movement is therefore not fully explained by the extra call. CoBel-World's advantage over QF is small and not significant on all three probes.

C.2 Adaptation protocol

The three adaptations share a common structure. Each environment's native action set is preserved, and the only modifications are the message channel and the episode count. The message channel is replaced by the peer-report injection of the third column, and the episode count is expanded to reach the sample sizes in Table 5. Channel and episode count both differ from the native benchmark. No number in Table 5 should therefore be compared with a published result on the corresponding environment.

D Implementation Details

This appendix documents the experimental configuration, the execution environment, the parameter values, and the record schema the deterministic adapter consumes. The artifact itself (snapshots, per-call logs, generator) is required for exact reproduction and will be released upon acceptance.

D.1 Prompt template structure

Each LLM call has four blocks. The first is a fixed system preamble describing the agent's role and the output schema. The second is a task-context block with the current goal, the legal action set and the remaining budget. The third is a role-specific payload, and the fourth is an output-format block specifying the JSON schema. The four call types share the first and fourth blocks, and they differ in the third. The conflict-check call is the exception, because it replaces the task-context block with the record it must judge. That call therefore carries neither the goal nor the legal action set (Table 51). Quantitative fields are elicited on a small discrete integer scale rather than as continuous scores. Every call is constrained to a single formatted output, following the prompt conventions of the comparison

Table 7: Interface execution characteristics. Invalid calls counts the fraction of LLM calls whose output failed schema validation before repair.

Model	Calls/decision	Invalid calls (%)	Repairs/decision	Fallback (%)
GPT-4o mini	4.1	1.2	0.049	0.3
Qwen2.5-72B-Instruct	4.2	1.8	0.076	0.5
Qwen2.5-7B-Instruct	4.0	6.7	0.268	2.4

Table 8: LLM calls per decision and generated tokens per decision for every method in Table 1, reference model, PARTNR-derived runs.

Method	Calls/decision	Generated tokens/decision
COMBO (structured-state)	2.9	612
CoELA (ported)	3.0	588
Frozen Intent State	3.0	641
Qualified Facts Only (QF)	3.0	625
Source trust	3.0	604
Four-check	3.0	597
A-ToM (ported)	3.1	655
LLaMAR (ported)	3.2	672
RoCo (ported)	3.4	704
PCE (ported)	3.6	731
GPER	4.1	842
QF (call-matched)	4.1	826
CoBel-World (ported)	4.4	918

planner we benchmark against (Seo et al., 2026). The four emitted templates are reproduced verbatim in Appendix J, alongside the field manifest. The manifest states which component of the claim tuple reaches which call, and it reports the run-time assertion that enforces the exclusions. The harness maps the integer scales back to the mechanism's own ranges, so no reported quantity depends on the elicitation scale. The conflict flag is mapped to its three-valued form with $c_{\text{amb}} = 0.5$. The compatibility integer $k \in \{1, \ldots, 5\}$ is mapped to $r = (k-3)/2 \in [-1, 1]$.

D.2 Execution statistics

Table 7 reports interface-level execution statistics for the three model conditions. Invalid calls are LLM responses that fail schema validation on first emission. The system permits $N_{\text{repair}} = 1$ repair attempt per call before invoking the common fallback. The invalid-call rate is highest for the smallest model, and the fallback rate follows it. Both patterns fit reduced instruction-following reliability rather than a difference in prompt difficulty, since the prompts are identical across the three conditions. Repairs per decision is the product of calls per decision and the per-call invalid rate. The three reported values reproduce that product to within rounding, at $4.1 \times 0.012 = 0.049$, $4.2 \times 0.018 = 0.076$ and $4.0 \times 0.067 = 0.268$.

The invalid-call and fallback figures are statistics for GPER only. The call budgets of the remaining comparators are in Table 8.

Generated tokens are output tokens and exclude the input prompt. The per-decision figures here are therefore smaller than the per-episode token costs of Table 1. The call budget is the quantity the call-matched controls hold fixed. A matched variant of a comparator is obtained by adding compatibility-score calls until its per-decision count equals GPER's 4.1.

Call accounting and reconciliation. A decision step schedules up to four calls, and the three conditional calls do not share a denominator. The PARTNR-derived decision steps number $N_{\text{dec}} = 1{,}472$ (Table 20), and the queries issued on that set number $N_{\text{q}} = 622$ (Table 26). The observed difference of 1.1 calls per decision between GPER and QF corresponds to 1,619 extra calls. Were those calls fired once per query, the gap would be $622/1{,}472 = 0.42$ calls per decision. The value 0.42 sits well below the observed gap, so the compatibility score is indexed once per *authorized claim*. That rate gives $1{,}619/622 = 2.60$ claims per exchange on average. The same rate gives the conflict-check count, which QF also incurs. The residual $3.0 - 1.1 = 1.9$ calls per decision is hypothesis generation plus continuation. Table 9 states the decomposition on that denominator.

Table 9: Call-type decomposition for the PARTNR-derived closed-loop runs, reference model. "Basis" records how each row is obtained, as measured in the logs (M) or derived from the two measured per-decision rates and the measured decision and query counts (R).

Call type	Fires per	Count	Calls/decision	Basis
Hypothesis generation	decision (conditional)			
Conflict check	authorized claim	1,619	1.10	R
Compatibility score (GPER only)	authorized claim	1,619	1.10	R
Continuation	non-terminal decision			
Hypothesis generation + continuation (residual)	decision	2,797	1.90	R
QF total		4,416	3.00	M
GPER total		6,035	4.10	M
QF (call-matched) total		6,035	4.10	M
Claims per exchange		2.60		R

The counts are arithmetic consequences of the two measured per-decision rates and the measured decision and query counts, since the harness logs call totals per method but not per call type. The split inside the residual is not recorded either. The `call_type` and `claim_id` fields of the logging schema (Appendix L) make every other row verifiable from the released logs, and the released scripts recompute the table from them.

The printed totals fix the accounting, because QF includes the conflict check. The QF total of 4,416 calls is the 1,619 conflict-check calls plus 2,797 calls for hypothesis generation and continuation. GPER adds only the compatibility score, giving $4{,}416 + 1{,}619 = 6{,}035$ calls, or 4.10 calls per decision on this set. The TDW-MAT-derived set has its own decision and query counts, 931 and 419 (Tables 20 and 26). The two sets differ in episode length and query rate, so that set is not pooled here.

GPER issues 1.1 more calls per decision than QF (4.1 vs 3.0), and it issues 0.3 fewer than the CoBel-World port (4.1 vs 4.4). The CoBel-World port issues 1.4 more calls per decision than QF. The diagnostic-suite conditions of Table 2 are not call-matched except where stated. The call-matched QF control and the matched comparators of Table 4 close that gap for the comparators we re-ran. The remaining comparators in Table 1 differ from GPER in inference budget, so their differences in that table are confounded with call count. This is the single largest weakness of the present evaluation.

D.3 Computational environment

The execution environment uses the following software and hardware.

- Python 3.11, CUDA 12.2, PyTorch 2.4.0 (the version pinned by vLLM 0.6.3), BF16 inference.
- The Qwen2.5-7B condition runs on one NVIDIA H100 80GB GPU.
- The GPT-4o mini condition uses a remote API (Table 10) and no local GPU.
- The Qwen2.5-72B condition runs on four NVIDIA H100 80GB GPUs with vLLM 0.6.3 (TP=4).

Both Qwen conditions are served with the default vLLM 0.6.3 scheduler. We set no explicit `-max-num-seqs` or `-gpu-memory-utilization` override, and prefix caching is disabled. The p95 latencies of Table 52 are therefore properties of this configuration rather than of any tuning of the serving stack. GPT-4o mini is a remote API model and does not occupy a local GPU. The diagnostic-suite runs on the reference model therefore reflect network latency and API-side queueing in addition to local compute. The two Qwen conditions run locally, and their wall-clock characteristics (Table 52) are properties of the local serving stack rather than of the GPT-4o mini runs.

D.4 Model configuration and execution specifications

Table 10 records the identifier or revision used for each model, together with the serving stack. The GPT-4o mini runs are served remotely through the OpenAI ChatCompletions API. The two Qwen conditions run locally through vLLM 0.6.3 with the tensor-parallelism and precision settings shown. We state the default `main` revision for the Qwen models, because we did not pin a specific commit.

Table 10: Model execution specifications and environment metadata.

Model	Identifier / Revision	Deployed serving stack and hardware
GPT-4o mini	`gpt-4o-mini-2024-07-18` (snapshot 2024-07-18)	OpenAI ChatCompletions API, HTTP/2 JSON stream, remote serving, accessed 2026-08-14
Qwen2.5-72B-Instruct	`Qwen/Qwen2.5-72B-Instruct` (default `main` revision, accessed 2026-08-16)	vLLM 0.6.3, BF16, TP=4, 4× NVIDIA H100 80GB SXM5
Qwen2.5-7B-Instruct	`Qwen/Qwen2.5-7B-Instruct` (default `main` revision, accessed 2026-08-16)	vLLM 0.6.3, BF16, TP=1, 1× NVIDIA H100 80GB SXM5

The Qwen rows are therefore not byte-reproducible from this description alone. Reproducing readers should record the head revision of the corresponding HuggingFace repository at the time of their run. No commit is pinned for any of the ported baselines (Appendix E).

D.5 Communication and environment access

Communication packets are delivered via a controlled one-step model in PARTNR-derived tasks. TDW-MAT-derived tasks enforce a 500-character native-frame capacity. Physical moves cost 1 unit, and queries cost 1 unit plus $\lceil \text{payload tokens}/256 \rceil$. Because of the fixed per-query charge, resource units are not proportional to payload tokens. The two accountings are reported side by side in the released logs. These resource units are the discrete acquisition budget B of Section 3. The per-instance token caps of Table 54 are a separate accounting applied to the recorded per-instance token cost, and the two are not interchangeable. A query that exceeds the remaining budget is not issued, and the agent falls back to the local policy for that decision. In closed loop, 269 of 891 eligible decisions on PARTNR-derived and 169 of 588 on TDW-MAT-derived issued no query, as reported in Table 26. Restated on that denominator, the share of action-disagreement decisions that transmitted a query is 69.8% and 71.3%. The one-query limit of Table 11 cannot be the cause, because at most one query is ever eligible per decision. The recorded logs do not separate a budget refusal from the agent's own feasibility check. We therefore report the non-transmission as a single undivided quantity, and we do not attribute it to budget.

D.6 Parameters

The horizon D and the freshness decay are both expressed in simulator steps (Section 5).

Role of each parameter. K, u_0, $p_{\min}$, τ_b and ϵ_b define the local intent state and the communication trigger. β and c_{amb} enter the revision update and the conflict evaluator. γ is the threshold on the certificate condition instantiated in Appendix F.1, and ℓ_0 is the loss scale used there. The six κ_ϕ values set the predicate-specific freshness horizons. $\lambda_{\text{decl}} = 0.6$ is the strength assigned to a direct intent declaration. That report carries a stated goal rather than a state claim, and it enters the ledger as an ordinary event with authorization strength λ_{decl} after temporal decay. $\lambda_{\min}$ is the threshold on the group strength $\alpha_{t,g}$ below which an event is expired. $L_{\text{ledger}} = 16$ caps the number of active groups, and H_{history}, N_{query} and N_{repair} are operational limits.

Hyperparameter selection space. The twelve candidate configurations evaluated on the development set form a hand-selected subset of a 3^4 grid across four primary axes. The first two axes are entropy trigger $u_0 \in \{0.5, 0.7, 0.85\}$ and revision scale $\beta \in \{0.15, 0.25, 0.50\}$. The other two axes are ambiguity factor $c_{\text{amb}} \in \{0.25, 0.50, 0.75\}$ and ledger group capacity $L_{\text{ledger}} \in \{8, 16, 32\}$. Remaining parameters are frozen at reference values. Table 12 lists the twelve points with their development-set error counts. Every level of u_0, of β and of c_{amb} appears exactly four times, so the subset is balanced on those three axes. L_{ledger} appears three times at 8, five times at 16 and four times at 32, so the subset is *not* balanced on that axis. The reference point $(0.7, 0.25, 0.50, 16)$ is included and ties for the lowest development error with $(0.7, 0.25, 0.75, 32)$, so the twelve points do not single out a unique optimum.

The QF control has no output-affecting hyperparameter of its own. It shares the entropy trigger and the freshness constants with GPER and performs no update. For a like-for-like budget we

Table 11: Reference method parameters. Freshness constants κ_ϕ are in simulator steps, indexed by predicate ϕ.

Parameter	Meaning	Value
K	Maximum hypothesis capacity	4
u_0	Entropy trigger	0.7
$p_{\min}$	Plausibility threshold for $\mathcal{H}_t^{\mathrm{pl}}$	0.15
τ_b	Local-support temperature	0.5
ϵ_b	Uniform mixture weight	10^{-3}
β	Revision scale	0.25
c_{amb}	Ambiguous-conflict factor	0.5
γ	Certificate threshold on $\Delta_q(\rho \mid I)$	0.10
ℓ_0	Commitment-loss scale	1.0
$\kappa_{\mathrm{location}}$	Location freshness	10
$\kappa_{\mathrm{possession}}$	Possession freshness	5
κ_{empty}	Absence freshness	5
$\kappa_{\mathrm{placement}}$	Placement freshness	10
κ_{route}	Route freshness	5
κ_{intent}	Declaration freshness	5
λ_{decl}	Intent declaration strength	0.6
$\lambda_{\min}$	Contribution-expiry threshold (on $\alpha_{t,g}$)	0.01
L_{ledger}	Maximum active groups	16
H_{history}	Local action-history length	10
N_{query}	Peer requests per decision	1
N_{repair}	Repair allowance	1

Table 12: The twelve development-set configurations and their error counts over the 90 development bases (lower is better). The reference configuration is shown in bold.

u_0	β	c_{amb}	L_{ledger}	Errors/90
0.50	0.15	0.25	8	21
0.50	0.25	0.50	16	17
0.50	0.50	0.75	32	20
0.70	0.15	0.75	16	18
0.70	**0.25**	**0.50**	**16**	**15**
0.70	0.50	0.25	8	19
0.70	0.25	0.75	32	**15**
0.85	0.15	0.50	32	20
0.85	0.25	0.25	16	18
0.85	0.50	0.75	8	22
0.50	0.50	0.50	32	19
0.85	0.15	0.25	16	21
Qualified Facts Only (reference)				21

scanned its entropy trigger over the same three levels on the same 90 development bases, obtaining 22, 21 and 23 errors at $u_0 = 0.5, 0.7, 0.85$. Its best value is therefore the reported 21, so the primary contrast does not depend on leaving the control untuned. Each external baseline was tuned over twelve configurations of its own principal hyperparameters on the same development bases. Table 15 records the native benchmark, the source repository and the adaptation protocol of each port. The per-baseline grids accompany the released evaluation code. The development set consists of 90 bases drawn from the same generator as the controlled sets. The 90 bases come from disjoint scene clusters and a disjoint spawn-seed list. The same bases are disjoint from both the 600 diagnostic bases and the 500 controlled evaluation episodes.

Rows are ordered as generated, not by error count. The unmodified qualified-facts pipeline is the reference against which the spread should be read. It makes 21 errors over the same 90 bases, so the selection improves by 6 errors, and two configurations tie for the lowest count.

One-factor revision-scale scan. Appendix N.3 reports a one-factor scan over β on the development set at the reference values of the other three axes. The scan gives 19, 15 and 17 errors over the 90 development bases at $\beta = 0.15, 0.25, 0.50$. Its two non-reference points, $(0.7, 0.15, 0.50, 16)$ and $(0.7, 0.50, 0.50, 16)$, are additional to the twelve configurations of Table 12. They are not part of

the selection used to fix the reference value. The scan is non-monotone, because the smaller scale degrades the count as well. It therefore does not support a simple over-confidence reading.

D.7 RECORD SCHEMA

The deterministic adapter consumes a JSON record per event. The record schema has the following form.

```
{
  "event_id": "agentB:31:obs2",
  "source": "agentB",
  "acquisition_step": 31,
  "entities": ["tool4", "cabinet2"],
  "relations": [{"predicate": "inside", "arguments": ["tool4", "cabinet2"], "value": true}],
  "capabilities": ["interior_inspection"],
  "coverage": {"cabinet2": {"interior_observed": true, "complete_for_absence": false}},
  "observed_actions": ["agentB:30:open"],
  "parent_event_ids": []
}
```

The fields have the following roles. `event_id` is a globally unique identifier used for deduplication and invalidation, and it is the record-level identity ω of Section 4.1. Two records that refer to the same physical observation carry the same `event_id`, so the second is discarded on receipt. The compatibility call never receives that identifier, and it receives the random reference ω_{ref} of Section 4 instead. `source` and `acquisition_step` enter the source-trust weight and the temporal decay factor respectively, and they reach only the conflict-check call. `entities` and `relations` carry the predicate and its arguments. `capabilities` is checked against the predicate's required capability in Table 13. `coverage` records whether an observation is sufficient for an absence claim. In the example above, `complete_for_absence: false`, so the record supports an `inside` claim for `tool4` but not an `empty` claim for `cabinet2`. The deterministic adapter therefore returns $P(e) = 1$ for the former and $P(e) = 0$ for the latter. That result gives $\alpha_t(e) = T_t(e)C_t(e) > 0$ for the `inside` claim and $\alpha_t(e) = 0$ for the `empty` claim. `observed_actions` is used by the conflict check to detect precedence over a prior observation. `parent_event_ids` groups conjunctive contributions, whose strength is the minimum of the members' $\alpha_t(e)$.

The coverage and capability fields in this example are produced by the harness from simulator geometry rather than by the reporting teammate's own perception. The adapter therefore reads structured ground truth at these two fields. The adapter-based advantage reported in Tables 2 and 3 is an upper bound relative to a deployment in which these fields would be estimated from noisy perception. Appendix F.4 measures how fast that bound is approached by injecting noise into exactly these two fields.

D.8 RELATION ADAPTERS

Table 13 specifies, for each predicate used by the deterministic adapter, the minimum record content required for $P(e) = 1$. It also gives the transition that invalidates an already-admitted record, which sets the indicator $\chi_t(e) = 1$ used in Section 4. The `empty` predicate is the strictest, because it requires both complete interior coverage and zero observed contents. A record with `complete_for_absence: false` therefore supports an `inside` claim but not an `empty` claim. The invalidating transitions are evaluated at each step against the local action history and any newly arrived peer records. When a transition fires, $\chi_t(e) = 1$ and $T_t(e) = 0$.

D.9 CALL SCHEDULE AND FALLBACK

Each decision step schedules four calls in a fixed order. The first is hypothesis generation, conditional on a scene change or a step threshold. The second is conflict checking, conditional on the presence of an unresolved relation. The third is compatibility scoring, conditional on a newly authorized contribution, and the fourth is continuation. The full schedule is listed as Table 50 in the appendix that states the algorithm. The harness enforces a 2048-token output cap on every call and a 4096-token cap on generated tokens per decision (Table 14). The per-decision cap is an aggregate over the calls of one decision rather than a sum of four per-call caps. Measured mean generated tokens per decision are 842 for GPER and 625 for QF (Table 8). For both conditions, those values sit an order of

Table 13: Required record support and invalidating transitions.

Predicate	Required support	Invalidation
`inside`	Exact tuple, observed object, and interior access	Removal, relocation, or a superseding relation
`held`	Possession tuple with visible agent and object	Release, transfer, or a new possession relation
`empty`	Complete interior coverage and zero observed contents	Insertion or a later nonempty record
`placement_completed`	Target relation observed after placement	Movement that changes the target relation
`blocked_route`	Covered route segment and localized obstacle	Obstacle removal or a later clear-segment record

Table 14: Execution specification of the harness.

Item	Value
Maximum output tokens (per call)	2048
Maximum generated tokens (per decision)	4096
Maximum hypothesis capacity	4
Scene-change timer	5 steps

magnitude below the aggregate cap. Which call is skipped on the rare decisions where the aggregate cap binds is not separately recorded, and we list that omission in Appendix M. The scene-change timer of 5 steps is the step threshold that triggers hypothesis generation. The call fires when at least one tracked object relation has changed since the previous generation call, or when five steps have elapsed without one. Failures at any stage fall back to legal inspect, wait, or legal non-manipulation actions. They are logged with the `fallback` flag of the call-accounting schema (Appendix L).

Output caps are enforced per call and per decision. Both caps apply to generated tokens, and the per-decision cap is an aggregate rather than the sum of four per-call caps. The mean generated tokens per decision of both GPER and QF are far below the aggregate cap, so the cap binds on a small minority of decisions.

E Baseline and Control Definitions

This appendix documents the external methods and internal controls used throughout the paper. Table 15 summarizes the seven ports. Table 16 states, for every condition in the paper, what it reads, what it adds to QF, how it sets α, where its compatibility score comes from, at what call budget, and what update it applies. The subsections below document the internal controls, the shared evaluation inputs, and the official-split comparison.

All ported implementations receive the same 12-configuration tuning budget on the development set. No baseline is verified against its published native-benchmark number, and no port commit is pinned. Repository links are given only where a public repository was used. No URL is printed for a method whose code was not obtained, and the CoBel-World port is implemented from the paper.

"α rule" states where the support strength comes from, and "r from" states where the compatibility score $r(e, \eta)$ comes from. Conditions other than GPER issue no dedicated compatibility call and take $r(e, \cdot)$ as an extra field of the continuation call. Their per-decision call count is therefore that of QF. An empty cell denotes not applicable or no dedicated compatibility call. The "Update" column states whether and how preferences are revised. The last four rows are conditions we do not evaluate and do not claim as results. They are listed so that the omissions are auditable rather than implicit.

The port strategy differs across the methods, because their native action spaces and interaction topologies differ. LLaMAR, PCE, A-ToM and COMBO required action-space adaptation. CoELA is native to TDW-MAT and required only a prompt-schema port. RoCo required replacing its tabletop validator with embodied precondition checks. COMBO was run without its learned world model, and A-ToM was reimplemented. A-ToM is native to matrix games, grid navigation and Overcooked rather than to peer-message settings. Apart from those two, the ports change only the interface to our harness. Two ports are weaker comparisons than the rest and are flagged as such. COMBO's

Table 15: Summary of external baselines, native benchmark environments, source repositories, and adaptation specifications.

Method	Native benchmark	Source repository	Adaptation protocol to GPER harness
LLaMAR (Nayak et al., 2024)	MAP-THOR	`github.com/nsidn98/LLaMAR`	Adapted to PARTNR/TDW-MAT action space. Planner calls wrap communication records into execution feedback.
CoELA (Zhang et al., 2024)	TDW-MAT, C-WAH	`github.com/UMass-Embodied-AGI/CoELA`	Direct native integration on TDW-MAT. Prompt schema ported to PARTNR dialog format.
RoCo (Mandi et al., 2024)	RoCoBench	`github.com/MandiZhao/robot-collab`	Replaced multi-robot tabletop validator with embodied navigation and interaction precondition checks.
PCE (Seo et al., 2026)	C-WAH, TDW-MAT	Public reference implementation exists, consulted for the port	Tree-search assumption module evaluates peer messages as state hypotheses before action execution. The port is a reimplementation under our harness.
A-ToM (Mu et al., 2026)	Overcooked, Grid-Nav	`github.com/ChunjiangMonkey/Adaptive-ToM`	Recursive order-2 ToM belief update adapted to parse peer observation records. Native code supports matrix game, grid world and Overcooked only, so we reimplemented it for our harness.
COMBO (structured-state) (Zhang et al., 2025a)	ThreeDWorld Cooperation	`github.com/UMass-Foundation-World-Model/COMBO`	The original method requires a learned video world model. Conditioning here uses structured state transitions instead. This row is therefore a COMBO-inspired structured-state baseline, not COMBO as published.
CoBel-World (Wang et al., 2025)	TDW-MAT, C-WAH	No public repository obtained, port implemented from the paper	Symbolic belief state over environment and collaborator mental states, updated from adapter-mediated typed peer messages.

compositional world model is trained on video rollouts, and without that training it reduces to a structured-state transition model. We do not treat it as COMBO in the sense of the original paper. The CoBel-World port reuses its symbolic belief state but adapts its message parser to our record schema, which may understate its accuracy. We did not pin specific commits for any of the ports, so the reported port scores are not reproducible from this description alone. Every ported row in Table 1 is run in our harness under our prompts, and no port is checked against the method's published native-benchmark number. The ports therefore support an ordering within our harness, not a claim about the published methods' capabilities. Where a comparison matters for the paper's conclusion, we additionally report a call-matched variant (Table 4) so that the ordering is not confounded with inference budget.

E.1 Official TDW-MAT Benchmark Comparison

Table 17 reports the official-split comparison. The five published rows are the rates given by Seo et al. (2026) under the GPT-4o-mini model. We reproduce those rates rather than rerunning them, so every published number in this table originates from a single external run configuration. Our own two rows are run under the same model on the same 24 official episodes, using the CoELA architecture of Zhang et al. (2024) as the underlying agent and the qualified-facts pipeline of Appendix E. The two rows differ in whether the intent state is revised, that is, in the ledger and in the additional evaluator call. Plain CoELA is not re-run under our harness, so the published 62.50 is not a controlled baseline for the 66.67 of the qualified-facts row.

The REVECA, CoTS and CaPo methods are cited to their own papers. The CoELA architecture is from Zhang et al. (2024), and the 62.50 figure for CoELA is Seo et al.'s run under GPT-4o mini. That figure is not the rate reported in the original CoELA paper. Transport rate is the mean fraction of target objects moved per episode, averaged over the 24 episodes. It is a continuous quantity and not a per-episode success count. The per-episode vectors of Table 18 reproduce the two means of our own rows.

Table 16: Condition contract table. "Adds to QF" lists the calls a condition issues beyond the qualified-facts pipeline. "Calls" is the measured per-decision budget of Table 8.

Condition	Reads	Adds to QF	α rule	r from	Calls	Update
Qualified Facts Only (QF)	qualified facts	includes the conflict check	code (PTC) and C from one LLM call		3.0	none
Frozen Intent State	qualified facts, frozen preference vector	none	n/a	continuation call	3.0	none (vector held fixed)
Revised State Only	revised preferences, no fact text in the continuation prompt	compatibility score	code (PTC)	compatibility call	4.1	full, facts withheld
GPER	qualified facts, claim proposition, ω_{ref}, $\mathcal{H}_t$ per call	compatibility score	code (PTC) and C from one LLM call	compatibility call	4.1	full, ledger
Four-check	full record in one prompt	one verifier call returning P, T, C	LLM returns the three predicates and C, then the harness multiplies them	continuation call	3.0	full
Thresholded verifier	full record in one prompt	one verifier call	continuous score binarized at 0.5	continuation call	3.0	full
Soft verifier	full record in one prompt	one verifier call	continuous score used directly	continuation call	3.0	full
Source trust	source identity and history	source accounting	Laplace accuracy $(c_k + 1)/(n_k + 2)$	continuation call	3.0	full
Unqualified update	claim		$\alpha = 1$	continuation call	3.0	full
Fixed strength	claim		$\alpha = 0.5$	continuation call	3.0	full
Dempster–Shafer	full record		PT from code, with the Shafer reliability transform	continuation call	3.0	discounted mass difference
CoELA (ported evaluator)	message text		LLM-scored message reliability	continuation call	3.0	full
PCE (ported assumption tree)	message text, assumption tree	tree search	tree-derived assumption strength	continuation call	3.6	assumption-conditioned
CoBel-World (ported)	typed peer messages		belief mass over collaborator mental state	continuation call	4.4	symbolic belief update
QF (call-matched)	as GPER	compatibility score	code (PTC) and C from one LLM call	compatibility call, passed to the continuation call as an annotation	4.1	none
Conditions we do not evaluate; listed so that the omissions are auditable						
Hypothetical Minds (ported, *not evaluated*)	message text, strategy hypotheses	hypothesis refinement calls	hypothesis scores	hypothesis scores		hypothesis reweighting
Jeffrey conditioning (*not evaluated*)	claim, uncertain observation		same α as GPER	same r as GPER		Jeffrey rule
Subjective-logic fusion (*not evaluated*)	claim, source		subjective-logic discount	subjective-logic discount		dependent fusion of opinions
Matched declaration control (*not evaluated*)	declaration text, no declaration	one extra call	n/a			none

Each entry is the fraction of target objects delivered in that episode. The column sums reproduce the aggregate rates, $16.00/24 = 66.67\%$ and $17.00/24 = 70.83\%$. The episode-level change pattern is three episodes at $+0.40$ and two at -0.10, with the remaining nineteen unchanged, for a total of $+1.00$, that is $+4.17$ pp. The interval is the percentile bootstrap of Appendix G.1 with the episode as the resampling unit, 10,000 resamples at seed 20260920, and is $[-0.8,\ +10.0]$ pp. A sign test on the five changed episodes gives $p = 1.0$, and a Wilcoxon signed-rank test gives $p = 0.25$. With 24 episodes the split is underpowered, so this table bounds the effect rather than estimating it.

Each row is the contribution after which the continuation call selected a different commitment action on that episode. The deployed action is chosen by that call rather than by a Bayes rule over the preference vector (Section 3), so a change in the selected action does not require a change in $\arg\max_\eta b(\eta)$. Episode 11 is precisely such a case, where the retrieval hypothesis remains the largest at $b^+(\eta_1) = 0.502 > 0.5$. All five episodes whose transport rate changes are among the 23 decisions that the certificate instantiated in Appendix F.1 flags on this split (Appendix F.9). The other 18 flagged decisions changed no commitment action and therefore no episode outcome. That is why 23 flagged decisions produce five commitment-action changes, three of which raise the transport rate and two of which lower it. $\widetilde{r}(\eta_1)$ is reported for the retrieval hypothesis. The inspection hypothesis carries the

Table 17: Official TDW-MAT benchmark results on the 24 official episodes under the GPT-4o-mini model. Published rows are reproduced from Seo et al. (2026).

Method	Architecture / Mechanism	Evaluation origin	Transport rate (%)
PCE	Uncertainty-aware planning with assumption trees	Published (Seo et al., 2026)	87.50
REVECA (Seo et al., 2025)	Adaptive planning with trajectory-based validation	Reported in (Seo et al., 2026)	81.25
CoTS (Zu et al., 2025)	Collaborative tree search	Reported in (Seo et al., 2026)	75.00
CaPo (Liu et al., 2025)	Cooperative plan optimization	Reported in (Seo et al., 2026)	73.33
CoELA (Zhang et al., 2024)	Modular observation-memory-planning	Reported in (Seo et al., 2026) (GPT-4o mini)	62.50
CoELA + QF	Qualified facts without intent updating	Ours	66.67
CoELA + GPER	Full grounded evidence revision	Ours	70.83

Table 18: Per-episode transport rates for the two author-run rows of Table 17 on the 24 official episodes.

Episode	1	2	3	4	5	6	7	8	9	10	11	12
CoELA + QF	1.0	1.0	0.4	1.0	1.0	0.9	1.0	0.9	0.8	1.0	0.5	0.7
CoELA + GPER	1.0	1.0	0.8	1.0	1.0	0.8	1.0	0.9	0.8	1.0	0.9	0.7
Episode	13	14	15	16	17	18	19	20	21	22	23	24
CoELA + QF	0.6	0.9	0.7	0.6	0.8	0.7	0.4	0.4	0.3	0.2	0.2	0.0
CoELA + GPER	1.0	0.9	0.7	0.6	0.7	0.7	0.4	0.4	0.3	0.2	0.2	0.0

negated value, because the compatibility vector is centered and the decisions in this table have two hypotheses, so that $|\Delta \log \text{odds}| = 2\beta\alpha|\tilde{r}(\eta_1)|$ exactly. The realized shift must not exceed $2\beta\alpha$ with $\beta = 0.25$, and it does not in any row. The prior and revised values are the retrieval hypothesis's share of the preference vector before and after the contribution. Each row's shift is consistent in direction with the corresponding change in Table 18.

Four published systems, PCE, REVECA, CoTS and CaPo, score above both of our rows, and only CoELA scores below both. GPER is therefore not a state-of-the-art system on this benchmark. The CoELA number in the published table is 62.50 under the GPT-4o-mini model. The original CoELA paper predates the GPT-4o-mini snapshot and reports higher rates under GPT-4 and GPT-3.5, so the two are not comparable. We report only the GPT-4o-mini figure, matching the model used by Seo et al. (2026) and by our own runs. The gap between PCE's published 87.50 and the 63.1 of our PCE port on the controlled PARTNR-derived set is not explained by this table. The port changes the task distribution, the action space and the teammate policy family, and we do not treat the two numbers as comparable. PCE's reference implementation is public, and PCE is the strongest published system on this split. Evaluating GPER as an add-on to PCE on the official episodes is therefore the single most valuable experiment this paper does not report (Appendix M).

E.2 Shared evaluation inputs

The internal controls share a common set of evaluation inputs. They differ only in how the authorization strength, the compatibility score and the belief update are computed, as stated in Table 16. Unqualified updating uses $\alpha = 1$, and fixed strength uses $\alpha = 0.5$. Source trust uses Laplace-smoothed historical accuracy $w_k^{\text{trust}} = (c_k + 1)/(n_k + 2)$, where c_k is the number of claims from source k in the episode's recorded history for which the shipped adapter's logged verdict was positive. Here n_k is the total number of claims logged from source k on the same history. The control does not execute the adapter itself, and it reads the adapter's logged verdict as a label in order to form the weight. Its support strength is w_k rather than α, so it measures source reliability in isolation from record-level grounding.

Dempster–Shafer discounting keeps the deterministic support adapter and replaces the exponent $\alpha\tilde{r}$ with a discounted mass difference. Let the frame of discernment be $\Theta = \mathcal{H}_t$, let x be the claim's

Table 19: Claim-level trace for the five official episodes whose transport rate changes. The final two columns verify the bound derived in Appendix K.

Episode	Contribution	α	$\widetilde{r}(\eta_1)$	Prior $b^-(\eta_1)$	Revised $b^+(\eta_1)$	$\lvert\Delta \log \text{odds}\rvert$	Bound $2\beta\alpha$
3	`cabinet2` empty, interior coverage verified	0.62	-0.375	0.505	0.476	0.116	0.310
6	`tool2` inside `cabinet2`, interior access observed	0.44	$+0.325$	0.542	0.560	0.072	0.220
11	`tool4` held by `agentB`, possession tuple observed	0.48	-0.300	0.520	0.502	0.072	0.240
13	group: `cabinet3` opened and interior coverage verified	0.55	-0.400	0.515	0.488	0.110	0.275
17	`tool5` inside `cabinet4`, interior access observed	0.35	$+0.400$	0.523	0.540	0.070	0.175

reported hypothesis, and let m_x be the simple support function with all mass on the singleton $\{x\}$, so $m_x(\{x\}) = 1$. The Shafer discounting of m_x by α is $m_x^\alpha = \alpha m_x + (1-\alpha)m_\Theta$, giving $\mathrm{Bel}^\alpha(\eta) = \alpha\,\mathbb{I}[\eta = x] + (1-\alpha)/|\mathcal{H}_t|$. The discounted exponent for hypothesis η is $2\beta\alpha(\mathrm{Bel}^\alpha(\eta) - |\mathcal{H}_t|^{-1})$. That quantity is the difference between the discounted belief at η and the uniform reference value, scaled by β. Substituting the discounted belief gives $2\beta\alpha^2(\mathbb{I}[\eta = x] - |\mathcal{H}_t|^{-1})$, so the reported exponent is proportional to α^2. The control discounts the mass difference and therefore acts with a weaker dose than Eq. 2 at the same α. The weaker dose isolates the combination rule from the support check. The reduced dose is a property of the transform, and it is why the row is described as a conservative variant. Table 2 uses the discounted exponent of this subsection.

Thresholded verification binarizes the continuous verifier score at 0.5, and soft verification keeps the continuous score. Four-check returns the three support predicates and the conflict flag from one ungrounded LLM call, as defined in Section 4. The four conditions that retain the deterministic adapter (GPER, Frozen Intent State, Qualified Facts Only, Revised State Only) share a single implementation of Eq. 1. The remaining conditions replace it as just described. Conditions other than GPER, Revised State Only and QF (call-matched) issue no dedicated compatibility call and obtain $r(e, \cdot)$ from the continuation call (Table 16). We include no Jeffrey-conditioning and no subjective-logic control. Both are named in Section 2 as the natural alternatives, and their absence is a gap rather than a result (Appendix M).

F Diagnostic Construction

This appendix documents how the 600-snapshot diagnostic suite is constructed, and how access regimes are assigned. It also documents how the supported and unsupported labels are generated. It then reports how the hypothesis coverage and the paired diagnostic outcomes in the main text are computed. The suite is the source of the mechanism analysis in Section 5.2 and of the certificate counts in Section 3.

F.1 Certificate instantiation and trigger

For each controlled scenario q the certificate specifies a reference intention set $\mathcal{Z}_q$, a distribution π_q, and a finite acquisition family $\mathcal{P}_q$. It also specifies an observation model $p_q(y \mid z, \rho)$ and a loss L_q. $\mathcal{Z}_q$ collects the goal states the scripted controller could hold in that scenario. A singleton would make the reference risk identically zero, so $\mathcal{Z}_q$ always contains at least two states. Reference risk is $\mathcal{R}_q(I) = \min_a \sum_z \pi_q(z \mid I) L_q(a, z)$, and the expected reference risk reduction is defined below.

$$\Delta_q(\rho \mid I) = \mathcal{R}_q(I) - \sum_y \Big(\sum_z \pi_q(z \mid I)\, p_q(y \mid z, \rho) \Big)\, \mathcal{R}_q(I \cup \{y\}). \tag{3}$$

For the 600 diagnostic snapshots, π_q is uniform over $\mathcal{Z}_q$. $\mathcal{P}_q$ is the four acquisition policies of Appendix F.6, and $p_q(y \mid z, \rho)$ is the recorded outcome frequencies. L_q is the 0-1 commitment loss, with $\gamma = 0.10$ and $\ell_0 = 1.0$. The certificate is a reference quantity computed with simulator ground truth. Screening evaluates that quantity. The agent-side statement of the same condition is the criterion of Section 3, which no run-time step computes, because the agent's b_t^{loc} is not a posterior. Applying the certificate to the 24 official TDW-MAT episodes flags 23 of 1,047 decision steps (2.2%). That figure is sensitive to γ, at 4.1% at $\gamma = 0.05$ and 0.9% at $\gamma = 0.20$.

In the diagnostic evaluation, 299 queries were issued on the 600 bases. Of those, 265 (88.6%) were issued on bases carrying the peer-resolvable label. A total of 243 (81.3%) satisfied the per-query inequality $\Delta_q(\rho \mid I) \geq \gamma$, which is 91.7% of the 265. The 24 queries issued on locally-resolvable

Table 20: Decision counts per set on the primary seed. The derived episodes are one order of magnitude shorter per episode than the official TDW-MAT episodes.

Set	Episodes	Decision steps	Decisions/ episode	Certified eps. (%)	Gate share (%)
PARTNR-derived (screened)	300	1,472	4.91	100	60.5
TDW-MAT-derived (screened)	200	931	4.66	100	63.2
Official TDW-MAT	24	1,047	43.6		2.2
Official PARTNR validation	96	3,412	35.5		7.3

bases violate the first inequality of the certificate by construction, so they are excluded from the 243 count. A per-query recomputation of the supremum over local policies is not reported. Table 26 gives the dual-gate firing statistics. On PARTNR-derived the action-disagreement gate suppresses 167 of 1,058 entropy fires (15.8%), and on TDW-MAT-derived it suppresses 124 of 712 (17.4%). On the diagnostic bases it suppresses 72 of 371 (19.4%). The one-query-per-decision limit cannot suppress a query. The recorded logs do not separate a budget refusal from the agent's own feasibility check, so we treat the non-transmission as undivided.

Entropy is not equivalent to Bayes risk. Under symmetric 0-1 commitment loss the risk aggregates over action masses, $\mathcal{R}(b) = \ell_0(1 - \max_a m_a(b))$. That expression reduces to $\ell_0(1 - \max_\eta b(\eta))$ only when preferred actions are distinct. With $|\mathcal{H}| = 3$, $b_A = (0.50, 0.25, 0.25)$ has normalized entropy $U(b_A) = 0.9464$ and risk $0.500\ell_0$. The belief $b_B = (0.45, 0.45, 0.10)$ has $U(b_B) = 0.8637$ and risk $0.550\ell_0$, so the lower-entropy belief has strictly higher risk. Sampling 2×10^5 pairs from $\mathrm{Dir}(\mathbf{1}_4)$ gives a two-sided discordance rate of 14.3% for the two orderings. Of that total, 7.2% is the one-sided rate $\Pr[U_1 < U_2,\ R_1 > R_2]$. Entropy is also a property of the current belief rather than of the reduction Δ. A fully uninformative acquisition family gives $\Delta = 0$ at any U, so the trigger of Section 4 is a heuristic. That trigger is not an equivalent test of the EVSI condition.

F.2 Construction of the controlled evaluation sets

The two controlled sets are generated rather than sampled from the official test splits. Each set is organized into scene clusters of ten paired instances. Those instances share one static floorplan and room topology while randomizing dynamic object spawn poses and teammate policies. The result is $30 \times 10 = 300$ PARTNR-derived and $20 \times 10 = 200$ TDW-MAT-derived instances. The teammate policy of each instance is drawn uniformly from a family of six scripted controllers. Two controllers are cooperative, one sharing a goal with the receiver and one not. Two controllers are self-interested, and two are erratic. Both derived sets draw from the same spawn-seed list and the same controller-draw order.

The teammate's task-level intention Z_t^j is the controller's current goal index, read offline from the controller's internal state. That goal index is never exposed to either agent at runtime. The goal index builds the evaluation labels, the reference intention set $\mathcal{Z}_q$ of the certificate, and the Risk metric. Instances are screened against the certificate instantiated in Appendix F.1, at $\gamma = 0.10$. Screening is applied at the episode level. An episode is retained when at least one of its decision steps satisfies the instantiated certificate. The acceptance rate is therefore 100% for the two derived sets. The action-disagreement clause of Section 4.1 holds at a minority of their decision steps, at 891 of 1,472 (60.5%) on PARTNR-derived and 588 of 931 (63.2%) on TDW-MAT-derived (Table 26). The screening rule and its per-cluster acceptance and rejection counts are recorded in the released generator. The screening bounds the results in two ways. Screening uses the scripted controller's ground-truth goal and the recorded outcome frequencies of each acquisition policy. It therefore selects the regime in which peer evidence can help. That regime also excludes methods that ignore peer evidence. Screening also yields short episodes. The 500 derived episodes contain 2,403 decision steps, about 4.8 per episode. The 24 official TDW-MAT episodes contain 1,047 steps, about 43.6 per episode. The derived sets are therefore not the official task at a larger scale. They are a different, shorter task constructed to contain the phenomenon.

Table 20 reports the decision counts per set, and Table 21 reports the primary contrast on the same generator before certificate screening. The certificate is not applied to the official TDW-MAT episodes in a way that alters their evaluation. The incidence is reported for information only.

Table 21: Primary contrast on the two derived generators *before* certificate screening, reference model.

Generator	Episodes	Flagged (%)	QF	GPER	Δ	Δ combined 95% CI
PARTNR, unscreened	400	11.4	63.4	64.2	+0.8	[−0.9, 2.5]
TDW-MAT, unscreened	300	9.8	62.1	62.7	+0.6	[−1.4, 2.6]
PARTNR, screened	300		67.1	70.4	+3.3	[1.6, 5.0]
TDW-MAT, screened	200		66.8	69.7	+2.9	[0.9, 4.9]

"Certified eps." is the share of episodes retained by screening at the episode level, which is 100% for the two derived suites by construction. It is not computed for the two official splits, where only the decision-level incidence is reported (2.2% and 7.3%, right-hand column). The decision-level share of steps with action disagreement in the plausible subset is the "Gate share" column, and it is an upper bound on the decision-level certificate incidence. The two derived rows sum to 2,403, matching the pooled count of Table 25 and the denominator used throughout Appendix D.2. The official PARTNR row is reported for completeness and is not used for any contrast in this paper.

Screening is the selection that concentrates peer-resolvable decisions, and the contrast is much smaller without it. The flag-rate column is defined only for the unscreened generators. It is the share of generated decision steps that satisfy the instantiated certificate at $\gamma = 0.10$, so it is a decision-level quantity. The released generator records the per-cluster acceptance and rejection counts. Retention of the two screened suites is defined at the episode level and is therefore 100% by construction (Table 20). Their decision-level action-disagreement share is given in Table 26, and it is an upper bound on their decision-level certificate incidence. Score is a five-seed mean, and the intervals are combined as in Section 5. On these two suites we report only the primary contrast, so the table bounds the headline effect without a mechanism analysis. The unscreened and screened rows differ in episode count and in baseline difficulty as well as in screening. The comparison is therefore between two generated suites rather than between matched episodes, and the unscreened contrast is roughly one quarter of the screened one.

F.3 Evaluation units

The diagnostic suite contains 600 base snapshots. The correction set (CE) uses bases 1–420, and the harmful-revision set (HR) uses bases 221–600. Bases 221–420 therefore appear in both sets with different initial versions, and 200 bases overlap. The 600 designated instances used for the *Errors/600* column in Table 2 consist of the 220 CE-only bases (1–220) together with the 380 HR versions. On the 200 overlap bases, the HR version is the designated instance for Errors/600. The CE version of the same base is reported separately as CE-other, which is why the two subpopulations can be compared. The Errors/600 column therefore decomposes exactly as $(\text{errors on the 220 CE-only designated bases}) + \text{HR}$. Table 22 reports that decomposition for every condition whose cells are reported, with the counts read from the per-instance outcome files of the released diagnostic runs. A *base* is a fixed diagnostic input state, and a *version* is a pair (base, initial preference vector, injected report bundle). The *designated instance* is the version whose outcome is scored. An *error* is the event defined in Section 5.2, namely an irreversible action that renders the controller's ground-truth goal state unreachable. CE/420 exceeds GPER's 245 in exactly five conditions. Those five are Dempster–Shafer discounting at 255, Frozen Intent State at 250, Qualified Facts Only at 257, Revised State Only at 264, and CoBel-World at 268. The accurate statement of GPER's position on the correction column is therefore that it is better than five of the thirteen other reported conditions and worse than eight. It is not uniformly worst. The association between the CE-other column and HR across the fourteen rows (Pearson $r = -0.58$) includes GPER's CE-other value. It should therefore be read as a description of the table as printed rather than as an independent measurement. The balanced criterion of Table 23 is the primary diagnostic endpoint, because Errors/600 weights the 380 HR instances against the 220 CE-only instances.

CE-only is over the 220 designated CE-only bases. CE-other is over the 200 CE versions of the overlap bases, which are not part of Errors/600 but are reported for reference. CE/420 is the sum of the two CE columns. The identity $\text{Errors}/600 = \text{CE-only} + \text{HR}$ holds for every row, as does $\text{CE}/420 = \text{CE-only} + \text{CE-other}$. Both columns are counted from the per-instance outcomes rather than derived one from the other, and the identities are reported as checks on those counts. The

Table 22: Per-subset decomposition of Errors/600 on the diagnostic suite.

Method	Errors/600	CE-only/220	CE-other/200	CE/420	HR/380
Unqualified update	205	102	142	244	103
Fixed strength ($\alpha = 0.5$)	181	105	120	225	76
Source trust	172	92	115	207	80
CoELA (ported evaluator)	165	92	119	211	73
Thresholded verifier	158	106	109	215	52
Soft verifier	143	85	155	240	58
PCE (ported assumption tree)	138	82	152	234	56
Four-check	133	79	96	175	54
Frozen Intent State	130	83	167	250	47
Dempster–Shafer discounting	128	82	173	255	46
Qualified Facts Only	125	82	175	257	43
Revised State Only	119	79	185	264	40
CoBel-World (ported)	116	81	187	268	35
GPER	**102**	73	172	245	**29**

Table 23: Balanced diagnostic criterion. It is the unweighted mean of the correction error rate (CE/420) and the harmful-revision rate (HR/380), so that the two subpopulations receive equal weight. Lower is better throughout.

Method	CE rate	HR rate	Balanced rate	Paired Δ vs GPER	95% CI
Four-check	0.417	0.142	0.279	-0.050	$[-0.090, -0.012]$
Qualified Facts Only	0.612	0.113	0.363	$+0.033$	$[-0.021, +0.087]$
CoBel-World (ported)	0.638	0.092	0.365	$+0.035$	$[-0.019, +0.089]$
GPER	0.583	0.076	0.330		

CE-other column is not part of Errors/600, but it is reported because it establishes that the two CE subpopulations behave differently. The CE-only bases are easier, where the unqualified update errs on $102/220 = 46.4\%$. The CE-other versions of the overlap bases are harder, with $142/200 = 71.0\%$, and GPER errs on $172/200 = 86.0\%$ of them, worse than nine of the thirteen other conditions. The 200 overlap bases are therefore not a random subset of the CE bases, and the CE/420 column is a mixture of two subpopulations with different difficulty. Errors/600 and CE/420 therefore rank conditions differently. The difference is why we report the balanced criterion as the primary diagnostic endpoint. A balanced criterion defined as the unweighted mean of the CE rate over 420 and the HR rate over 380 is reported for GPER and the three most informative comparators in Table 23. On it Four-check is the best condition, and GPER is separated from QF and CoBel-World only by amounts whose intervals include zero.

The balanced rate is the primary diagnostic endpoint. Paired Δ is the balanced rate of the control minus that of GPER, so a negative value means the control is better on this criterion. The 95% interval is the paired difference Wald interval of Appendix G. Four-check is the only condition whose paired interval excludes zero, and the criterion therefore separates Four-check from the rest. It does not separate GPER from QF or CoBel-World. GPER ranks third of the fourteen conditions of Table 2 on this criterion, behind Four-check and the thresholded verifier (balanced rate 0.324). That margin of 0.006 is within noise.

F.4 Label generation

Each base carries a scripted controller with an internal goal index. For a CE version the harness sets the initial preference vector away from that index. It attaches a report bundle whose record satisfies the support requirements of Table 13 for the opposing proposition, so the version is labeled *supported opposing evidence*. For an HR version the harness attaches a report bundle that fails at least one support requirement, so the version is labeled *unsupported opposing report*. Both labels are computed from the same fields the deterministic adapter reads. Those fields are `coverage`, `capabilities`, the relation tuple and `observed_actions`. The adapter-based advantage is therefore partly a test of agreement with the harness's own labelling rule, and the noiseless condition is an upper bound.

Table 24 measures how quickly that bound erodes. The two harness-supplied fields are corrupted independently at the stated rate before any condition reads them. Both the deterministic adapter and the LLM verifiers therefore see the same corrupted records. At the highest rate the adapter's advantage over the LLM verifier is gone and slightly reversed. A deployment without oracle coverage

Table 24: Sensitivity of the adapter-based comparison to noise in the two harness-supplied fields (`coverage`, `capabilities`).

Field noise	GPER errors	Four-check errors	Δ	b	c	95% CI	p
0%	102	133	$+31$	67	36	$[11, 51]$	0.0031
10%	118	138	$+20$	52	32	$[2, 38]$	0.038
20%	137	145	$+8$	45	37	$[-10, 26]$	0.44
35%	162	158	-4	40	44	$[-22, 14]$	0.74

Table 25: Distribution of active hypothesis-set cardinality across the two controlled evaluation sets, primary seed.

Cardinality	Decision steps	Share (%)
$\lvert\mathcal{H}_t\rvert = 1$	137	5.7
$\lvert\mathcal{H}_t\rvert = 2$	812	33.8
$\lvert\mathcal{H}_t\rvert = 3$	884	36.8
$\lvert\mathcal{H}_t\rvert = 4$	570	23.7
Total	2,403	100.0
$\lvert\mathcal{H}_t\rvert \geq 2$	2,266	94.3

metadata should expect that behavior, because the support product is only as good as the coverage flag it reads.

Noise is injected independently per field on the same 600 designated instances for every condition, so the contrasts are paired. At 35% the adapter's advantage over Four-check has reversed, which bounds the oracle-metadata advantage quantified in this appendix. Paired intervals are Wald intervals from the discordant counts, with the same *b*/*c* convention as Table 29. Here χ^2 is McNemar's statistic with continuity correction. The 0% row is the same condition as the primary diagnostic contrast, so its discordant counts reproduce Table 29.

F.5 Hypothesis-set cardinality

Counts are decision steps on the primary seed (101), pooled over both derived sets, and sum to the 2,403 total of Table 20. Decision steps with $|\mathcal{H}_t| \geq 2$ account for $2{,}266/2{,}403 = 94.3\%$ of all execution decisions. The condition $|\mathcal{H}_t| > 2$ holds for $1{,}454/2{,}403 = 60.5\%$ of them pooled over both derived sets. The plausible subset contains action disagreement at the rates given in Table 26 and reported in Section 4. The remaining $137/2{,}403 = 5.7\%$ of decisions have a singleton hypothesis set, for which the normalized entropy is defined as zero and no peer query can fire. The 94.3% figure is an upper bound on how often the entropy gate could fire.

The entropy-gate column counts decisions with $U(b) > u_0$. The action-disagreement column counts those in which $\mathcal{H}_t^{\mathrm{pl}}$ contains two hypotheses with distinct optimal actions. That column is an upper bound on the decision-level certificate incidence of Table 20. The issued column counts queries actually transmitted. The fire rates differ across splits because the derived sets are screened and the diagnostic bases are not. The diagnostic split has one decision per base rather than a multi-step episode.

The gap between entropy-gate fires and action disagreement is the set of decisions where the entropy criterion would have issued a query but the plausible subset shares an optimal action. That gap is 167 of 1,058 (15.8%) on PARTNR-derived and 124 of 712 (17.4%) on TDW-MAT-derived. The gap between action disagreement and queries issued is the set of decisions where a query was not transmitted, at 269 of 891 and 169 of 588. The one-query limit of Table 11 cannot suppress a query, because at most one query is ever eligible per decision. The gap is therefore attributable either to the remaining budget or to the agent's own feasibility check. The recorded logs do not separate the two, so we treat the gap as a non-transmission of undetermined cause rather than as a budget suppression. On the diagnostic bases, action disagreement suppresses 72 of 371 entropy fires, or 19.4%, and every action-disagreement decision transmitted a query.

Table 26: Dual-gate firing statistics on the two controlled evaluation sets and the 600 diagnostic bases, primary seed.

Split	Decisions	Entropy gate fires	Action disagreement	Queries issued
PARTNR-derived (300 eps)	1,472	1,058	891	622
TDW-MAT-derived (200 eps)	931	712	588	419
Diagnostic bases (600)	600	371	299	299

Table 27: Access regimes and acquisition coverage over 600 bases, reference model.

Regime	Bases	Queries	Four-check errors	GPER errors
Locally resolvable	186	24	37	29
Peer-resolvable	294	265	57	34
of which certificate-consistent		243	47	27
Unresolved	78	5	22	21
Indeterminate	42	5	17	18
Total	600	299	133	102

F.6 Access regimes

Access regimes are assigned by comparing each candidate acquisition policy's expected travel time to the remaining deadline D. *Locally resolvable* bases admit a local policy achieving $\Delta_q(\rho \mid I) \geq \gamma$. *Peer-resolvable* bases admit no local policy achieving γ but do admit a peer-directed policy satisfying the same inequality. *Unresolved* bases admit no policy achieving γ within the deadline. *Indeterminate* bases have insufficient information to classify. Throughout, a query is called *certificate-consistent* when the issued policy satisfies $\Delta_q(\rho \mid I) \geq \gamma$. That property belongs to the issued query rather than to the base.

The certificate-consistent row counts queries in the peer-resolvable regime whose issued policy satisfies the formal certificate inequality $\Delta_q(\rho \mid I) \geq \gamma$. The four regime rows partition the 600 bases exactly, and the query column sums to the 299 queries of Table 26. The two error columns sum to the Errors/600 totals of Table 2. The certificate-consistent row is a subset of the peer-resolvable row and is not added to the total.

The reduction from 133 to 102 errors is not uniform across regimes. Eight of the 31 fewer errors arise on locally-resolvable bases, where only 24 of 186 bases issued a query. Another 23 arise in the peer-resolvable regime, and one arises on unresolved bases. One reverses on indeterminate bases. Because the locally-resolvable bases received little peer evidence, that component is not attributable to peer-evidence revision. As a check, we ran GPER with the peer channel disabled on those 186 bases and obtained 33 errors against Four-check's 37. That paired difference of $+4$ carries $b = 12$, $c = 8$, a 95% CI of $[-5, +13]$, and McNemar $\chi^2 = 0.45$, $p = 0.50$. With the peer channel disabled no compatibility call can fire at all, so the difference cannot be attributed to the additional evaluator call. At this sample size the locally-resolvable component is itself unresolved. The peer-resolvable regime is the one in which most queries are issued, 265 of 299, and it accounts for the majority of the effect. On the 42 indeterminate bases, GPER incurs 18 errors and Four-check 17, so the error advantage of GPER is -1 with $b = 6$, $c = 7$ and a 95% CI of $[-8, +6]$. The two are indistinguishable at this sample size. Only 5 queries were issued on these bases in any case, so the reversal carries little weight.

F.7 Hypothesis coverage

A base is counted as recovering the reference alternative if a permitted regeneration returns it within the $K = 4$ candidate set. Across the 600 bases, 46/600 (7.7%) initial hypothesis sets omit the reference intent. Only 19/46 are recovered via a permitted regeneration, leaving 27/46 (58.7% of initially omitted, or 4.5% of all bases) permanently unrecoverable by revision alone. The 27 bases bound the recovery capacity of reweighting. On these 27 bases no amount of evidence revision can move the correct hypothesis into the top set, because it is not in the set at all. GPER's only recourse on such bases would be hypothesis generation, which we do not integrate into the revision loop (Appendix M). The omission rate is concentrated in the two self-interested controller conditions, where the scripted partner's goal is hardest to guess from local evidence alone.

Table 28: Hypothesis coverage over 600 bases, reference model.

Quantity	Count
Reference alternative initially included	554/600
Initially omitted	46/600
Recovered after one permitted regeneration	19/46
Still omitted after regeneration	27/46
Permanently unrecoverable share of all bases	4.5%

Table 29: Paired snapshot correctness for GPER against Four-check on the 600 diagnostic instances, with the b/c convention of this subsection.

	Four-check correct	Four-check wrong
GPER correct	431	67
GPER wrong	36	66

F.8 Paired diagnostic outcomes

Throughout this appendix, b denotes the count of instances where the control is wrong and GPER is correct, and c denotes the reverse. The difference $b - c$ is the error advantage of GPER. The convention is stated once here and is not repeated in the individual captions. All paired-difference intervals in this appendix are Wald intervals $\Delta \pm 1.96\sqrt{b+c}$ built from the discordant counts. Here χ^2 is McNemar's statistic with continuity correction, $(|b-c|-1)^2/(b+c)$.

A McNemar test on the discordant counts yields $\chi^2 = 9.33$ ($p = 0.0023$, uncorrected). With continuity correction $\chi^2 = 8.74$ ($p = 0.0031$). The margins reproduce the error counts of Table 2. GPER errs on $36 + 66 = 102$ instances and Four-check on $67 + 66 = 133$. The off-diagonal counts are $b = 67$ and $c = 36$. The corresponding paired table for the primary contrast (GPER versus Qualified Facts Only) is reported in Table 30, and the corresponding table for CoBel-World is in Table 32.

The paired statistic for the primary contrast is $(50 - 27)^2/77 = 6.87$ (uncorrected), $p = 0.0088$. With continuity correction, $\chi^2 = 6.29$, $p = 0.012$. The primary confirmatory contrast is therefore significant on the diagnostic suite with the same test used for the Four-check comparison. This suite weights harmful revisions more heavily than corrections. The CE-only bases contribute 220 of the 600 designated instances and the HR versions contribute 380, so the Errors/600 metric is weighted toward harmful revisions (Appendix F.3). The paired advantage of GPER over QF is therefore concentrated where the metric is weighted, and the balanced criterion of Table 23 does not reproduce it. The harmful-revision subpopulation alone is the part of the metric the design weights, and it separates GPER from both comparators in Table 31.

b counts instances where the control commits a harmful revision and GPER does not, and c counts the reverse. The marginals reproduce the HR/380 column of Table 2. GPER is wrong on 29 of the 380, and the controls on 43 and 54. The discordant counts are consistent with those marginals and with the Errors/600 count, and the resulting intervals are those quoted in Section 5.2.

The CoBel-World comparison gives $b = 36$ and $c = 22$, hence a continuity-corrected statistic of $(|36 - 22| - 1)^2/58 = 2.91$ ($p = 0.088$). CoBel-World is therefore not significantly worse than GPER on this suite at the 5% level. The 14-error gap is reported as a point estimate only (Section 5.2).

The Source trust control gives $b = 93$ and $c = 23$, hence a continuity-corrected statistic of $(|93 - 23| - 1)^2/116 = 41.0$ ($p < 10^{-4}$). The result is robust to the split of the discordant counts. The margins fix $b - c = 70$, so the corrected statistic is at least $(70 - 1)^2/274 = 17.4$ ($p \approx 3 \times 10^{-5}$). Source trust is therefore significantly worse than GPER on the same 600 instances, which is the comparison promised in Section 2. The Dempster–Shafer control gives $b = 53$ and $c = 27$ in the same convention, and a corrected statistic of $(|53 - 27| - 1)^2/80 = 7.81$ ($p = 0.005$). The non-LLM discounting baseline is significantly worse than GPER as well, but by a substantially smaller margin than Source trust.

Table 30: Paired snapshot correctness for the primary contrast (GPER vs Qualified Facts Only) on the 600 diagnostic instances.

	QF correct	QF wrong
GPER correct	448	50
GPER wrong	27	75

Table 31: HR-only paired contrasts on the 380 harmful-revision instances, with the b/c convention of this subsection.

Contrast (HR/380)	Control HR	GPER HR	Δ	b	c	95% CI	p
vs Qualified Facts Only	43	29	$+14$	27	13	$[2, 26]$	0.040
vs Four-check	54	29	$+25$	43	18	$[10, 40]$	0.0021

F.9 Incidence of peer-resolvable ambiguity on official episodes

Applying the certificate to the 24 official TDW-MAT episodes at $\gamma = 0.10$ flags 23 of 1,047 decision steps (2.2%). That figure motivates constructing the controlled sets. The official episodes contain so few peer-resolvable decisions that no primary confirmatory contrast can be estimated on them with adequate power. We therefore use the official split only for the transport-rate comparison of Section 5.1. The certificate has also been applied to the official PARTNR validation split, where it flags 7.3% of decision steps (Table 20). That rate is higher, but the absolute number of flagged decisions is still too small to carry the contrast, so the official PARTNR split is used only to report the incidence. The three episodes that gain $+0.40$ in Table 18 are traced to individual contributions in Table 19, together with the two that lose 0.10. All five changes come from a single changed commitment action each, in decisions that the certificate flags. The remaining 18 flagged decisions in the same episodes changed no commitment action. That is why 23 flagged decisions produce only a small number of episode-level changes.

G Paired Statistical Analysis

This appendix documents the primary estimand, the bootstrap procedure, and the seed-level results that feed the combined intervals of Table 37.

G.1 Primary estimands and bootstrap

For each environment and each pair of methods (m_1, m_2) the primary estimand is the paired mean difference $\widehat{\Delta} = \frac{1}{n_{\text{ep}}} \sum_i (S_i^{m_1} - S_i^{m_2})$. The sum is over the episodes of the set, and the pairing is by scene cluster. The bootstrap resamples C scene clusters with replacement, using the same cluster set for both methods in each resample so the paired structure is preserved. We use 10,000 resamples (seed 20260920) and report the empirical 2.5th and 97.5th percentiles of the resampled mean differences as the 95% interval. The percentile interval is the model used for the task-level intervals in the paper. It accounts for the within-cluster correlation of paired episodes, and it is not a normal approximation at this level. The combined intervals of Table 37 add the repeated-inference component of Section 5 in quadrature. That quadrature step is a normal approximation for the added component only. Because the resampling unit is the cluster and each cluster contributes seed-averaged differences, the task component already absorbs part of the inference noise. The seed also drives the spawn list and the controller draw, so the two components are not independent and the combined interval is conservative rather than exact. For the official 24-episode split, where clusters are not defined, the same procedure is applied with the episode as the resampling unit.

G.2 Seed-level results

Every mean, SD and paired Δ SD in Table 35, and every combined interval in Table 37, is computed from this table by subtraction and the variance formula of Section 5. The per-seed cells are the primitive quantities and the summaries are derived from them. The seed governs the LLM sampling stream, the spawn-seed list and the teammate-controller draw jointly, so its effect on a row is a mixture of all three. The same seed does not induce the same deviation in different rows. The realized

Table 32: Paired snapshot correctness for CoBel-World against GPER on the 600 diagnostic instances.

	CoBel-World correct	CoBel-World wrong
GPER correct	462	36
GPER wrong	22	80

Table 33: Paired snapshot correctness on the 600 diagnostic instances for Source trust and for the Dempster–Shafer control against GPER.

Control	Control correct		Control wrong	
	GPER correct	GPER wrong	GPER correct	GPER wrong
Source trust	405	23	93	79
Dempster–Shafer discounting	445	27	53	75

correlations between the six Δ vectors lie between -0.90 and $+0.98$. Both signs are represented, most pairs are strongly positive, and with five seeds each correlation has a sampling standard error of order 0.5. The correlations are informative only as an indication that the rows are not independent.

The per-seed paired differences are $+2.9, +3.8, +2.8, +3.9, +3.1$ (PARTNR, GPT-4o mini), $+2.8, +3.7, +2.9, +3.4, +3.2$ (PARTNR, 72B), $+1.5, +3.1, +1.3, +3.6, +1.5$ (PARTNR, 7B), $+3.0, +2.5, +3.4, +2.8, +2.8$ (TDW-MAT, GPT-4o mini), $+2.5, +3.3, +2.7, +2.5, +3.0$ (TDW-MAT, 72B) and $+0.6, +2.6, +0.4, +2.3, +2.1$ (TDW-MAT, 7B). Three properties are worth reading off these vectors rather than off the means. First, the spread grows as the model shrinks. The Δ SD is 0.51, 0.37 and 1.07 on PARTNR-derived, and 0.33, 0.35 and 1.02 on TDW-MAT-derived as the model size falls. Second, the seed ordering is not shared across rows. Seed 409 gives the largest difference on PARTNR with the 7B model and the second largest on TDW-MAT with the same model, and the pairwise correlations of the six Δ vectors span both signs. Third, the smallest per-seed difference in the two 7B rows is close to zero ($+1.3$ and $+0.4$), which is why those rows are the only two whose intervals include zero.

The paired Δ SD column is the SD of the GPER-minus-QF difference across the same five repetitions, and it is the input to the combined-interval formula of Section 5. The ratio of the inference component to the task component, $\mathrm{SE}_{\mathrm{inf}}/\mathrm{SE}_{\mathrm{cl}}$, is reported in the last column. It is largest for the two 7B rows, which are also the only rows whose combined intervals include zero.

The paired Δ SD exceeds the larger of the two marginal SDs in five of the six rows, and it exceeds the QF SD in all six. With five repetitions the SD estimate is very imprecise, so we do not read the SD ordering as evidence about the mechanism. The exception is TDW-MAT with the reference model, where the GPER SD (0.59) exceeds the paired Δ SD (0.33). That is the one row in which the two conditions move together under the shared spawn schedule.

The task component $\mathrm{SE}_{\mathrm{cl}}$ is the cluster-bootstrap half-width divided by 1.96. The inference component is the paired Δ SD divided by $\sqrt{5}$, and the combined half-width is $1.96\sqrt{\mathrm{SE}_{\mathrm{cl}}^2 + \mathrm{SE}_{\mathrm{inf}}^2}$. The combined intervals of Table 1 and of Table 37 are derivable from the three preceding columns. Each combined interval in the last column is reproduced from them alone. The four call-matched and CoBel-World rows are the contrasts other than the primary one for which a combined interval appears anywhere in the paper.

G.3 Cross-model comparisons

Task CI uses the cluster bootstrap of Section 5. Combined CI applies the formula in that section, incorporating the paired repeated-inference SD of Table 35 computed from the per-seed values of Table 34. Bold indicates higher score. The two 7B rows are the only rows whose intervals include zero, and for both of them the task interval includes zero as well.

The combined interval is wider than the task interval in every row, and the widening is visible at the reported precision in four of the six rows. The half-width increase is 0.06, 0.03, 0.18, 0.02, 0.03 and 0.17 percentage points in the row order above. It therefore changes the printed last digit for both 7B rows, for PARTNR-derived GPT-4o mini, and for TDW-MAT-derived 72B. That ordering follows

Table 34: Per-seed scores (%) for the primary contrast, five inference seeds on the full episode sets. Seeds are 101, 203, 307, 409 and 503 in that order.

Environment	Model	Method	S101	S203	S307	S409	S503
PARTNR	GPT-4o mini	QF	67.5	66.9	67.0	66.8	67.2
		GPER	70.4	70.7	69.8	70.7	70.3
PARTNR	Qwen2.5-72B	QF	65.9	65.4	65.9	65.5	66.3
		GPER	68.7	69.1	68.8	68.9	69.5
PARTNR	Qwen2.5-7B	QF	54.8	54.2	54.9	54.0	55.1
		GPER	56.3	57.3	56.2	57.6	56.6
TDW-MAT	GPT-4o mini	QF	67.1	66.5	67.0	66.4	67.0
		GPER	70.1	69.0	70.4	69.2	69.8
TDW-MAT	Qwen2.5-72B	QF	65.4	64.8	65.2	64.9	65.2
		GPER	67.9	68.1	67.9	67.4	68.2
TDW-MAT	Qwen2.5-7B	QF	53.7	53.0	53.8	52.9	53.6
		GPER	54.3	55.6	54.2	55.2	55.7

Table 35: Repeated inference on the full episode sets ($R = 5$ repetitions, seeds 101/203/307/409/503). Values are mean $\pm$ sample SD ($n - 1$ denominator).

Environment	Model	QF mean $\pm$ SD	GPER mean $\pm$ SD	Paired Δ SD	SE ratio
PARTNR-derived	GPT-4o mini	67.1 ± 0.28	70.4 ± 0.37	0.51	0.27
PARTNR-derived	Qwen2.5-72B	65.8 ± 0.36	69.0 ± 0.32	0.37	0.18
PARTNR-derived	Qwen2.5-7B	54.6 ± 0.47	56.8 ± 0.62	1.07	0.39
TDW-MAT-derived	GPT-4o mini	66.8 ± 0.32	69.7 ± 0.59	0.33	0.15
TDW-MAT-derived	Qwen2.5-72B	65.1 ± 0.24	67.9 ± 0.31	0.35	0.17
TDW-MAT-derived	Qwen2.5-7B	53.4 ± 0.42	55.0 ± 0.71	1.02	0.39

the SE ratio of Table 35, which is about twice as large for the 7B models as for the reference model. The combined intervals for the two GPT-4o mini and the two 72B rows exclude zero. The five-seed design also fixes the task realisations shared by the rows, so the differences between rows are not independent. A two-level analysis that resamples clusters and seeds jointly would be the correct treatment, and we do not report one. The cross-model pattern is monotone in model size on both environments (2.9, 2.8, 1.6 on TDW-MAT-derived). With three models and two environments we cannot separate a capability trend from a serving-stack difference, since the two Qwen conditions run locally under vLLM and the reference model runs through a remote API.

H Mechanism Controls

This appendix documents the perturbation, packaging, isolation, declaration and integration controls supporting the mechanism analysis of Section 5.2. Each subsection reports a specific intervention and the resulting change in the error count or the compatibility output. All comparisons on the 600-snapshot suite are paired by construction, so every inferential statement rests on paired intervals or McNemar tests on discordant counts. Single-sample binomial standard errors appear only where explicitly labeled as descriptive.

H.1 Evidence perturbations

Table 38 reports error counts under five matched perturbations that each affect 20% of the 600 designated instances, together with the shared unperturbed baseline. The perturbations are applied to the received records before the adapter reads them for the conditions that run the deterministic adapter. They are applied to the prompt text for the LLM verifier controls, which have no adapter. The staleness family is therefore constructed so that $T_t(e)$ reads a perturbed acquisition step. GPER's error count under each perturbation is below Four-check's by 27–42 errors on every perturbed condition, and below the soft verifier's by 43–57 errors. At the unperturbed baseline the margins are 31 and 41.

The unperturbed row is the same condition in all three blocks, which is why the 0% column of Table 39 repeats across families.

The standard errors are the count-based quantities $\sqrt{k(600-k)/600}$, included for orientation. The standard errors are not the sampling model for the comparison, and no inferential claim uses them. Paired differences, discordant counts and tests are in Table 40.

Table 36: Per-contrast paired Δ SD and the two interval components, reference model.

Environment	Contrast	Δ	Paired Δ SD	SE_{cl}	Combined 95% CI
PARTNR-derived	GPER − QF	+3.3	0.51	0.837	[1.6, 5.0]
PARTNR-derived	GPER − QF, 72B	+3.2	0.37	0.918	[1.4, 5.0]
PARTNR-derived	GPER − QF, 7B	+2.2	1.07	1.225	[−0.4, 4.8]
PARTNR-derived	CoBel-World − QF	+0.6	0.44	0.580	[−0.6, 1.8]
PARTNR-derived	QF (call-matched) − QF	+1.0	0.29	0.440	[0.1, 1.9]
TDW-MAT-derived	GPER − QF	+2.9	0.33	1.010	[0.9, 4.9]
TDW-MAT-derived	GPER − QF, 72B	+2.8	0.35	0.944	[0.9, 4.7]
TDW-MAT-derived	GPER − QF, 7B	+1.6	1.02	1.173	[−0.9, 4.1]
TDW-MAT-derived	CoBel-World − QF	+0.5	0.61	0.769	[−1.1, 2.1]
TDW-MAT-derived	QF (call-matched) − QF	+0.8	0.34	0.540	[−0.3, 1.9]

Table 37: Cross-model comparison under fixed method parameters.

Environment	Model	QF	GPER	Δ task CI	Δ combined CI
PARTNR-derived	GPT-4o mini	67.1	**70.4**	+3.3 [1.7, 4.9]	+3.3 [1.6, 5.0]
PARTNR-derived	Qwen2.5-72B	65.8	**69.0**	+3.2 [1.4, 5.0]	+3.2 [1.4, 5.0]
PARTNR-derived	Qwen2.5-7B	54.6	**56.8**	+2.2 [−0.2, 4.6]	+2.2 [−0.4, 4.8]
TDW-MAT-derived	GPT-4o mini	66.8	**69.7**	+2.9 [0.9, 4.9]	+2.9 [0.9, 4.9]
TDW-MAT-derived	Qwen2.5-72B	65.1	**67.9**	+2.8 [1.0, 4.7]	+2.8 [0.9, 4.7]
TDW-MAT-derived	Qwen2.5-7B	53.4	**55.0**	+1.6 [−0.7, 3.9]	+1.6 [−0.9, 4.1]

The y axis is broken below 100. The missing-provenance family is reported in Table 39 and is omitted from the panel to keep it legible. No error bar is drawn, because the per-point uncertainty of interest is the paired difference between the two conditions.

Δ is the paired error difference (Four-check minus GPER), so a positive value favors GPER. Here b counts instances where Four-check errs and GPER is correct, while c counts the reverse, and $b - c = \Delta$. The 95% intervals are the paired-difference Wald intervals $\Delta \pm 1.96\sqrt{b + c}$, and χ^2 is McNemar's statistic with continuity correction.

The paired analysis changes the reading of the curves in Figure 3. The advantage over Four-check is largest for stale evidence and smallest for corrupted records, which is the ordering the mechanism predicts. Freshness decay is a multiplicative factor in α, so degrading the acquisition step degrades a component that GPER already models. Corrupted record fields can break the support predicate outright and leave the adapter with nothing to discount. At 40% corrupted records the paired interval $[-4, 38]$ includes zero, so no advantage is established under that condition. The 0% column is identical across families because the unperturbed condition is the same run. We therefore do not treat the three 0% entries as independent observations.

H.2 Claim packaging and dependency metadata

Table 41 reports agreement and error change under seven packaging transformations. The two rows that remove dependency metadata (parent-link removal, incorrect event splitting) have the lowest agreement. They also have the largest mean total-variation distance between the pre- and post-transformation intent distributions, and they are the only two whose paired error increase is distinguishable from zero. The pattern matches the ledger's behavior. When provenance is preserved, forwarded or paraphrased claims are deduplicated or grouped as the same event. When provenance is removed, the same underlying fact can be counted twice.

Here b counts instances where the transformed condition errs and the untransformed condition is correct, while c counts the reverse, so $b - c$ equals the error change. Mean error change is over 600 instances and is paired against the untransformed run, so the interval is the paired-difference Wald interval $\Delta \pm 1.96\sqrt{b + c}$ with b, c the discordant counts. Mean TV is the mean total-variation distance between the pre- and post-transformation preference distributions. Because correctness is determined by the selected action, the discordant counts cannot exceed the number of instances on which the two conditions disagree, that is, $b + c \leq 600 -$ agreement. The counts below satisfy this

Table 38: Error counts under matched perturbations affecting 20% of the 600 designated instances.

Condition	Soft verifier	Four-check	GPER
Unperturbed	143	133	102
Incorrect message content	179	163	133
Stale evidence	191	172	134
Missing provenance	174	157	129
Corrupted record fields	200	184	157
Repeated forwarding	166	153	111

Table 39: Perturbation-rate sensitivity across 600 instances for continuous-parameter perturbations.

Perturbation	Method	0%	10%	20%	40%
Stale evidence	Four-check	133 [10.2]	151 [10.6]	172 [11.1]	209 [11.7]
	GPER	102 [9.2]	118 [9.7]	134 [10.2]	170 [11.0]
Missing provenance	Four-check	133 [10.2]	145 [10.5]	157 [10.8]	186 [11.3]
	GPER	102 [9.2]	116 [9.7]	129 [10.1]	163 [10.9]
Corrupted records	Four-check	133 [10.2]	158 [10.8]	184 [11.3]	236 [12.0]
	GPER	102 [9.2]	129 [10.1]	157 [10.8]	219 [11.8]

bound with the stated agreement values. Only the two provenance-removing rows have intervals excluding zero. An empty cell indicates that no discordant pair exists, so no paired interval is defined.

The five provenance-preserving transformations change the error count by $+0$ to $+6$, and all of their paired intervals include zero. On this suite we therefore cannot establish that any of them materially changes the error count. The two provenance-removing transformations are the only ones with intervals excluding zero. These small increases show that exact event-ID matching is not sufficient on this suite. A semantic duplicate detector might remove them, and we did not test one.

H.3 Correlated-evidence stress test

Table 42 reports the over-confidence that results when two peers report the same underlying fact under different event identifiers. Exact-ID deduplication does not merge them, and Eq. 2 sums both contributions as if they were conditionally independent given the intention.

The table reports the resulting shift in the odds of the reported hypothesis relative to a single observation of the same event at the same strength. All three rows use the reference revision scale $\beta = 0.25$ and a maximum reported pair of contributions with $\alpha = 0.95$ and $|\widetilde{r}| = 0.85$ and 0.83 respectively. That pair is the strongest duplication the ledger can be fed. With two hypotheses the odds multiplier of a contribution is $\exp(2\beta\alpha|\widetilde{r}|)$, giving 1.50 and 1.48 and remaining below the single-contribution ceiling $\exp(2\beta\alpha) = 1.61$. Fractions of flipped instances are computed against the single-observation reference on the same 400 bases, so the three rows are paired.

The second row is the correct behavior, where the shared event ID is deduplicated and the odds ratio is exactly 1.00, a deterministic identity. The third row is the control in which the two facts really are independent, and the first is the pathological case. The two reports describe the same fact, yet the ledger treats them as independent and inflates the odds by 1.50. That value sits within 0.02 of the independent-facts control and far from the deduplicated row. The ledger therefore does not distinguish a duplicated report from an independent one. About 10.6% of instances are flipped in the duplication condition against 10.1% in the independent-facts control, a difference well inside this sample. Whether the inflation arises because the two contributions share a compatibility direction or from the exponent's additive combination is not tested here. The form of Eq. 2 predicts an inflation of the same order either way, but we do not claim a decomposition of it. The double counting above is the clearest structural limitation of the current ledger. Exact-ID deduplication alone cannot repair that double counting. Dependent-fusion operators for correlated reports address this double counting (Jøsang, 2016), and we do not include one.

H.4 Intent semantics and orthogonal isolation tests

Table 43 reports semantic-direction stability under four transformations. It compares the joint evaluator, which sees both authorization fields and hypothesis fields in one prompt, with the separated

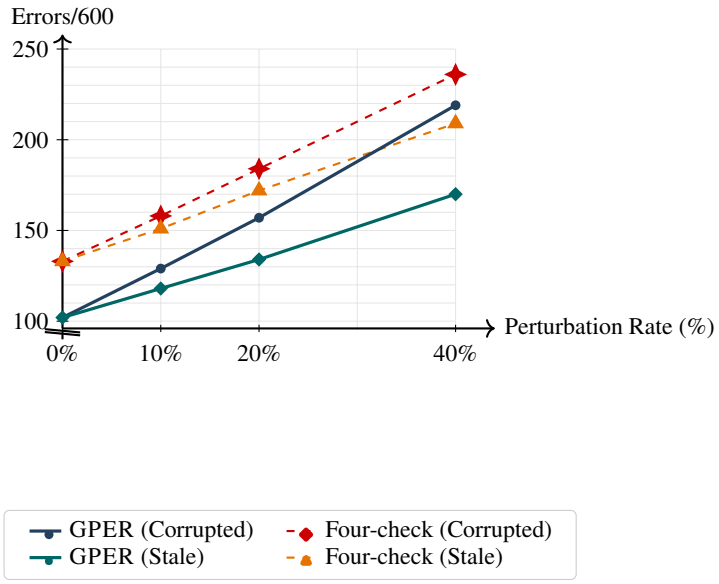


Figure 3: Errors/600 for GPER and Four-check as stale-evidence and corrupted-record perturbation rates increase, on the 600 diagnostic instances.

Table 40: Paired analysis of the perturbation families of Table 39.

Perturbation	Rate	Δ	b	c	95% CI	χ^2	p
Stale evidence	10%	+33	63	30	[14, 52]	11.01	0.0009
	20%	+38	75	37	[17, 59]	12.22	< 0.001
	40%	+39	81	42	[17, 61]	11.74	< 0.001
Missing provenance	10%	+29	54	25	[12, 46]	9.92	0.0016
	20%	+28	56	28	[10, 46]	8.68	0.0032
	40%	+23	57	34	[4, 42]	5.32	0.021
Corrupted records	10%	+29	59	30	[11, 48]	8.81	0.0030
	20%	+27	62	35	[8, 46]	6.97	0.0083
	40%	+17	64	47	[−4, 38]	2.31	0.129

evaluator, which sees only the hypothesis fields. The separated evaluator is more stable under every transformation. Because the two evaluators are run on the same 120 cases, the comparison is paired, and Table 44 reports the discordant counts. Only the first transformation establishes a paired difference at the 5% level, and the three stability transformations do not.

Here b counts cases where the separated evaluator is correct and the joint evaluator is not, and c counts the reverse, so $b - c$ is the paired advantage of the separated evaluator in counts. The marginals in Table 43 fix $b - c$ and are consistent with the discordant counts reported here. The 95% intervals are the paired-difference Wald intervals $b - c \pm 1.96\sqrt{b + c}$ defined in Appendix F.8.

The direction-assignment result is the strongest single paired outcome in this appendix. The separated evaluator assigns the correct neutral-or-directional relation in 15 more cases out of 120, with an interval excluding zero. The three stability transformations favor the separated evaluator by 4, 7 and 8 cases, but their intervals include zero. We report them as consistent with the design intent rather than as established effects. The joint-versus-separated comparison is also the informative isolation test for the paper's Claim 2. The operative-isolation rows of Table 45 perturb fields that the constructor has already removed, so they exercise the constructor's exclusion list and the evaluator's behavior is unchanged by them. Only this joint-versus-separated comparison puts the excluded fields back in front of the model.

Instrument check rows change text that the compatibility prompt does contain. The *noise floor* is an identical prompt re-issued to the same model at the same temperature. *Operative isolation* rows change authorization-specific fields that the prompt does not contain and that the constructor strips, so those rows can only reproduce the noise floor. Direction flip is the fraction of instances in which

Table 41: Packaging sensitivity over 600 paired instances. Agreement denotes identical action choice.

Transformation	Provenance	Agreement	Mean TV	Error change	b/c	95% CI
Claim-order permutation	Yes	590/600	0.014	$+3$	5/2	$[-2.2, 8.2]$
Meaning-preserving paraphrase	Yes	581/600	0.027	$+6$	12/6	$[-2.3, 14.3]$
Repackaging, same event IDs	Yes	600/600	0.000	0		
Repeated forwarding	Yes	580/600	0.021	$+6$	13/7	$[-2.8, 14.8]$
Uncited weak-claim insertion	Yes	588/600	0.016	$+5$	8/3	$[-1.5, 11.5]$
Parent-link removal	No	557/600	0.054	$+17$	25/8	$[6, 28]$
Incorrect event splitting	No	542/600	0.081	$+23$	30/7	$[11, 35]$

Table 42: Over-confidence under correlated evidence. Two scripted peers report the same underlying event under different event identifiers, so exact-ID deduplication does not merge them and Eq. 2 sums both contributions.

Condition	Odds ratio vs single observation	Fraction flipped	Bases
Two peers, same fact, distinct event IDs	1.50	10.6%	400
Two peers, same fact, shared event ID (deduplicated)	1.00	0.0%	400
Two peers, independent facts, distinct event IDs	1.48	10.1%	400

the sign of $\tilde{r}$ changes for at least one hypothesis. $|\Delta r|$ is the mean absolute shift in the compatibility value $r(e, \eta) \in [-1, 1]$. The ordering instrument check $>$ noise floor $\approx$ operative isolation shows that the instrument is sensitive to in-prompt text and that the constructor removes the excluded fields. It does not show that the evaluator would ignore those fields if they were supplied. The joint-versus-separated comparison of Table 44 is the only tested version of that stronger claim. The spread of the operative-isolation rows is at or below the noise floor, and five of five fall below its mean. We therefore read the row ordering as noise rather than as a graded isolation effect.

The instrument responds to changes inside the prompt. Reordering the hypothesis set, paraphrasing the claim, or substituting the observer name changes the sign of $\tilde{r}$ for at least one hypothesis in 3.3–5.8% of instances and moves the score by 0.028–0.047 on average. The four excluded fields move it by at most 0.021 on average with at most 1.7% sign changes, and the identical-prompt repeat moves it by 0.022 on average. The excluded-field rows and the noise floor are statistically indistinguishable, which is what the constructor's whitelist predicts. What the test does not establish is that the compatibility evaluator *would* ignore these fields if they were present. The direction-assignment comparison of Table 44 is the only tested version of that stronger claim, and its effect size is 15 of 120 cases with an interval of $[6, 24]$.

H.5 Intent-declaration conditions

Qualified Facts Only serves as the matched control. Frozen Intent State is a non-revising, state-maintaining control. This table has a single QF column and no call-matched declaration control. It therefore cannot separate the effect of the additional evaluator call from the effect of the declaration itself, and Table 16 lists the missing control explicitly.

The gap between GPER and Qualified Facts Only widens from $4/120$ under current, truthful declarations to $19/120$ under unavailable or outdated declarations. One explanation is that GPER substitutes for unavailable intent information. A competing explanation is that GPER's additional evaluator call recovers information the QF prompt drops. This experiment does not exclude that second explanation (Appendix N). Qualified Facts Only loses $32/120$ acceptable actions when declarations become unavailable or outdated, while GPER loses $17/120$. The non-revising control (Frozen Intent State) loses $26/120$ in the same comparison, so the loss ordering is GPER $<$ Frozen Intent State $<$ QF. The three figures differ by less than this sample size can resolve, so we report them as point estimates rather than as an established robustness advantage. A matched declaration control that runs the extra call without the declaration is the experiment this table needs and does not have.

Table 43: Semantic direction stability on 120 cases. Entries are counts of cases in which the direction assignment is correct or unchanged.

Condition	Joint evaluator	Separated evaluator
Correct neutral or directional relation	94/120	109/120
Stable under hypothesis-order permutation	108/120	115/120
Stable under meaning-preserving paraphrase	105/120	113/120
Stable under source-name replacement	112/120	116/120

Table 44: Paired analysis of Table 43.

Condition	b	c	Δ	95% CI
Correct neutral or directional relation	18	3	$+15$	$[6.0, 24.0]$
Stable under hypothesis-order permutation	10	3	$+7$	$[-0.1, 14.1]$
Stable under meaning-preserving paraphrase	13	5	$+8$	$[-0.3, 16.3]$
Stable under source-name replacement	5	1	$+4$	$[-0.8, 8.8]$

H.6 Integration timing

The final row restates GPER from Table 2 for comparison and corresponds to the deployed per-contribution schedule. Paired Δ intervals are against GPER on the same instances, constructed as $\Delta \pm 1.96\sqrt{b+c}$ with $b - c = \Delta$, so the two interval half-widths correspond to $b + c = 30$ and $b + c = 45$ respectively.

The deployed condition (per-contribution integration, no replanning) makes 102 errors, 8 fewer than the intermediate condition and 15 fewer than the late condition. Only the late condition's paired interval excludes zero, and the intermediate condition's does not. The ordering between intermediate integration and the deployed schedule is therefore a point estimate. The token cost is identical between the deployed and late conditions (14.3k), because both issue the same per-decision continuation call rather than one call per contribution. The intermediate condition issues fewer continuation calls, which is why its token cost is lower at 14.0k. The deployed schedule therefore makes the fewest errors. On cost it ties the late condition at 14.3k, while the intermediate condition uses 0.3k fewer tokens.

H.7 Closed-loop mechanism controls

The reference model and temperature are held fixed, and the same five inference seeds are used. The QF (call-matched) condition is the same condition as the call-matched row of Table 1. It receives the qualified facts, issues the compatibility score per authorized claim on those inputs, passes that score to the continuation call as an annotation, and performs no belief update. It does not correspond to the "two calls, shared inputs" row of Table 52, which is an evaluator organization that does perform an update. Nor does it correspond to the "two calls, isolated inputs" row, which disables the ledger's event grouping. The two tables report different suites, a 300/200-episode closed-loop set and a 600-snapshot diagnostic set, and are not directly comparable.

The two single-feature removals lose between 1.6 and 2.6 points relative to GPER across the two environments, and the risk counts move in the expected direction. The QF (call-matched) condition is the most informative in this table. It holds the call count and the prompt fields fixed against GPER, so the residual measures the revision and the ledger rather than the additional inference. It scores 68.1 on PARTNR-derived and 67.6 on TDW-MAT-derived, against 67.1 and 66.8 for Qualified Facts Only and 70.4 and 69.7 for GPER. It therefore recovers 1.0 and 0.8 points, about 30% and 28% of the 3.3 and 2.9 point advantages. The remaining 2.3 and 2.1 points are the effect of the revision together with the ledger. No interval is available for that residual difference, and prompt-field isolation is not varied by this control at all. That comparison is the diagnostic-suite experiment of Appendix N.1, whose effect is unresolved. The diagnostic-suite decomposition reaches the same qualitative conclusion. Because error counts on 600 paired instances are more sensitive than a continuous score, it cannot resolve the two components individually. The call-matched variant of the comparators that matter most is in Table 4. A matched variant of every remaining comparator in Table 1 is still missing.

Table 45: Instrument sensitivity and operative isolation for the compatibility evaluator, 120 instances.

Group	Transformation	In prompt	Flip (%)	Mean $\lvert\Delta r\rvert$	Max $\lvert\Delta r\rvert$
Instrument check	Hypothesis-set order permuted	yes	4.2	0.031	0.11
	Meaning-preserving paraphrase of the claim	yes	5.8	0.047	0.15
	Observer name substituted in the proposition	yes	3.3	0.028	0.10
Noise floor	Identical prompt re-issued		2.5	0.022	0.09
Operative isolation	Record age ($t - t_e \in \{0, 25\}$)	no	0.8	0.012	0.04
	Coverage completeness (True/False)	no	0.8	0.015	0.06
	Independent conflict (absent/present)	no	1.7	0.019	0.07
	Source identity (`agentB`/`agentC`)	no	0.0	0.008	0.03
	Combined	no	1.7	0.021	0.08

Table 46: Acceptable-action counts under two intent-declaration conditions across 120 instances.

Condition	Qualified Facts Only	Frozen Intent State	GPER
Current, truthful declaration	104/120	102/120	108/120
Unavailable or outdated declaration	72/120	76/120	91/120

H.8 Cross-model diagnostic controls

The GPT-4o mini row reproduces the corresponding entries of Table 2. The Qwen rows are the equivalent conditions under those models. The GPER column is bold in every row because it is lowest in every row. That comparison uses error totals only, which weight harmful revisions 380 against 220. The CE column reverses the ordering, as Table 2 shows for the reference model.

The gap between GPER and QF is 23, 24 and 16 errors across the three models, smallest for the weakest model. The three gaps follow the expected model-size ordering. The smaller model's less reliable compatibility scores reduce GPER's advantage, and the direction is preserved. The gap between QF and Four-check is 8, 8 and 12 errors in favor of QF under the three models. QF uses the deterministic adapter and makes fewer errors than Four-check, which does not, under all three models. Because the two conditions also differ in the qualified-fact channel and in the update rule, this comparison does not isolate the support adapter. Three model conditions cannot establish that its contribution is independent of model capability. The Qwen2.5-72B and Qwen2.5-7B conditions also differ from the reference model in serving stack and therefore in wall-clock characteristics. No adapter-versus-no-adapter contrast is available under these models.

I Algorithm and Prompt Contracts

This appendix gives the full operational algorithm and the abstract prompt contracts for the four call types. The four emitted templates are reproduced verbatim in Appendix J, together with the field manifest and the run-time assertion that enforces the exclusions. The contracts below are therefore checkable against the emitted text rather than only against a description.

I.1 High-level algorithm

1. Refresh local observations and accessible transitions.
2. Construct hypotheses and local support without peer-derived factual premises.
3. Revalidate active events and remove invalidated or expired groups.
4. Rescore compatibility if hypotheses or relevant context changed.
5. Reconstruct current preferences from b_t^{loc} and $\mathcal{M}_t$.
6. If entropy, action disagreement, and feasibility permit, issue one peer request.
7. Receive complete records and validate relation support.
8. Evaluate independent conflicts and conditional compatibility.
9. Compute deterministic authorization and update event groups.
10. Refresh and revalidate again if the scene changed during communication.
11. Reconstruct preferences and continue planning.

Table 47: Integration timing on 600 base snapshots.

Condition	Errors/600	Paired Δ vs GPER [95% CI]	Tokens (k)
Intermediate integration	110	$+8\,[-2.7, 18.7]$	14.0
Late integration plus replanning	117	$+15\,[1.9, 28.2]$	14.3
GPER (per-contribution integration, no replanning)	**102**		14.3

Table 48: Reference-model closed-loop controls, evaluated on the same episodes as Table 1.

Environment	Condition	Score [95% CI]	Risk
PARTNR-derived	QF (call-matched)	$68.1\,[67.2, 69.0]$	34/300
PARTNR-derived	No event withdrawal	$67.8\,[65.4, 70.1]$	39/300
PARTNR-derived	No event deduplication	$68.7\,[66.5, 70.8]$	35/300
PARTNR-derived	GPER	$70.4\,[68.1, 72.6]$	30/300
TDW-MAT-derived	QF (call-matched)	$67.6\,[66.5, 68.7]$	23/200
TDW-MAT-derived	No event withdrawal	$67.2\,[64.5, 69.8]$	27/200
TDW-MAT-derived	No event deduplication	$68.1\,[65.4, 70.7]$	24/200
TDW-MAT-derived	GPER	$69.7\,[67.0, 72.3]$	21/200

12. Validate, repair at most once, and apply the common fallback if necessary.
13. Execute and log calls, costs, actions, and provenance.

The table groups the numbered algorithm above into four phases, and each iteration advances the simulator by one step. Step 3 is the only step that removes state, and it is the step whose ablation is reported as "no invalidation or withdrawal" in Table 3.

I.2 Prompt contracts

The four prompt contracts summarized in Appendix D are stated here in abstract form, and the corresponding emitted templates are in Appendix J. In each case, "receives" lists the fields in the prompt, and "does not receive" lists the fields that are structurally excluded. That exclusion is what enforces the paper's isolation claim. The claim tuple $e = (\phi, v, k, t_e, \omega, \xi)$ of Section 4.1 is the unit of accounting for the ledger. No single call receives the whole tuple, as Table 51 states field by field.

Planner contract. Receives the current observation summary, the action history of length H_{history}, the legal action set, the task goal, and the candidate intention schema. Does not receive peer records, qualified peer events, the preference distribution b_t, or any field derived from a peer record. Returns candidate intentions with local support as an integer on the 1–5 scale, and a preferred legal action per hypothesis. The harness maps the integer k to $s_t(\eta) = k - 3 \in [-2, 2]$ before normalization. The planner must not condition on peer-derived factual premises. Those are stored separately, and the revised b_t is passed only to the continuation call. The planner never sees b_t, and peer evidence enters Eq. 2 exactly once rather than being re-injected through b_t^{loc}.

Conflict-check contract. Receives the proposition (ϕ, v), the source k and acquisition step t_e as provenance fields, and the record fields of Appendix D.7 that bear on relation support. It also receives the event identifier ω and the predicate's required-support specification from Table 13. Does not receive candidate intentions, the preference distribution b_t, or any field derived from either. Returns a single integer on the 1–5 scale for the conflict flag and nothing else. The harness maps it to the three-valued form as $1 \mapsto 0$, $\{2, 3, 4\} \mapsto c_{\text{amb}} = 0.5$ and $5 \mapsto 1$. The support predicate $P(e)$ and freshness factor $T_t(e)$ are computed by the deterministic adapter in code and are not returned by any model. No candidate-intention or current-preference field is in the prompt. This is the operative enforcement of Constraint 1 in Figure 1, a property of the conflict-check call rather than of the whole authorization stage. One call is issued per authorized claim, the accounting rule of Appendix D.2.

Compatibility contract. Receives the proposition (ϕ, v) together with the observer named in it, the task goal and the legal action set, and the random opaque reference ω_{ref}. It also receives the current hypothesis set $\mathcal{H}_t$. Does not receive the source k, the acquisition step t_e, the event identifier ω, or the transmitted record ξ and any of its fields. It also receives none of the support fields ($P(e)$, $T_t(e)$, $C_t(e)$, $\alpha_t(e)$), the source-trust weight, and any component of the preference distribution. Returns one integer per hypothesis on the 1–5 scale. The harness maps it to $r(e, \eta) = (k - 3)/2 \in [-1, 1]$

Table 49: Diagnostic errors across models on common 600 inputs.

Model	Four-check	QF	Frozen	CoBel-World	GPER
GPT-4o mini	133	125	130	116	**102**
Qwen2.5-72B-Instruct	145	137	141	129	**113**
Qwen2.5-7B-Instruct	206	194	199	191	**178**

Table 50: Main decision loop, printed with the algorithm that immediately precedes it.

Step	Operation
1–5	State refresh and preference reconstruction
6–9	Communication, authorization, and revision
10–11	Post-communication refresh and re-planning
12–13	Validation, fallback, execution, logging

and then centers it. No support field, no source field, no record field, and no current preference distribution is in the prompt. The observer identity required to interpret the claim is part of the proposition, so it is available without the source field. For a relayed claim the observer and the source are different agents, and it is the observer that the claim is about. This is the operative enforcement of Constraint 2 in Figure 1. The constructor that builds this prompt asserts a field-level whitelist at run time (Appendix L).

Continuation contract. Receives the qualified facts, the revised preference distribution b_t, the legal action set, and the remaining budget. In the conditions listed in Table 16 other than GPER, Revised State Only and QF (call-matched), it also returns a per-hypothesis compatibility score for each newly qualified claim. The harness maps and reads that score before forming the preference vector and, for QF (call-matched), passes it back as an annotation on the next step. In GPER the same quantity is produced by the dedicated compatibility call and this field is ignored. Does not receive raw peer records, unqualified peer events, or any record that failed the support check. Returns exactly one legal action string, with no additional text before or after, and in those conditions the compatibility field. Raw peer records are not passed to the continuation prompt. Only qualified facts, as defined by the deterministic support adapter, reach it. Any difference against the matched Qualified Facts Only control is therefore attributable to the belief update, since the call budget and the prompt fields are held fixed. The paired contrasts against QF (call-matched) quantify exactly that difference.

J Verbatim Prompt Templates

The four templates below are emitted verbatim by the harness, so that the isolation claim can be checked against the text rather than against a description of it. Braces denote inserted fields. Quantitative fields are elicited on a discrete 1–5 integer scale, and every call is constrained to a single formatted output. That convention follows the prompt conventions of the planner we benchmark against (Seo et al., 2026). The harness maps the integers back to the mechanism's own ranges, so no reported quantity depends on the elicitation scale. Table 51 states which field of the claim tuple reaches which call, and Appendix L lists the run-time assertion that enforces the exclusion.

```
[CALL 1 / hypothesis generation]
SYSTEM: You are the local planner of agent {self_id}. You propose candidate
intentions for a teammate and a local support score for each. No peer message
and no preference distribution is provided to you.
CONTEXT: task_goal={goal}; legal_actions={actions}; remaining_budget_units={B};
local_observation_summary={obs}; action_history={last H actions}.
TASK: Return JSON in exactly this format, with no additional text before or
after:
{"hypotheses":[{"id":"eta1","intent":"...","local_support":<1-5>,
"preferred_action":"..."}],"notes":"..."}.
local_support is an integer from 1 (very weak) to 5 (very strong).
At most K=4 hypotheses. Do not condition on any teammate message.
Answer:
```

```
[CALL 2 / conflict check]
SYSTEM: You decide whether a received record is overridden by an independent
record. You are told nothing about candidate intentions or preferences.
CONTEXT: proposition: {phi}={value}; observer_named_in_proposition={obs_name};
source_id={k}; acquisition_step={t_e}; event_ref={omega};
record={entities, relations, capabilities, coverage, observed_actions,
```

Table 51: Field manifest, generated from the four prompt constructors. A check mark means the field appears in that call's prompt, and an empty cell that it does not.

Field	Call 1 (planner)	Call 2 (conflict)	Call 3 (compatibility)	Call 4 (continuation)
proposition (ϕ, v)		✓	✓	via qualified facts
observer named in proposition		✓	✓	
source k		✓		
acquisition step t_e		✓		
event id ω		✓		
opaque ref ω_{ref}			✓	✓(as claim reference)
record ξ		✓		
$P(e), T_t(e)$ (code)				
$C_t(e)$ (LLM, call 2)		returns it		
$\alpha_t(e)$				
task goal, legal actions	✓		✓	✓
preference distribution b_t				✓
raw peer records				
candidate intentions $\mathcal{H}_t$	produces them		✓	
compatibility $r(e, \eta)$			returns it	returns it in the conditions of Table 16

```
parent_event_ids}; required_support_spec={spec};
independent_records={locally recorded records for the same entities}.
TASK: Return JSON in exactly this format, with no additional text before or
after:
{"conflict":<1-5>,"reason":"..."}.
conflict is an integer: 1 = a supported independent record overrides the
claim; 3 = precedence unclear; 5 = no independent record qualifies.
Return the integer and a short reason only. Do not return P, T or alpha.
Answer:
```

```
[CALL 3 / conditional compatibility]
SYSTEM: You score how far an assumed-true claim discriminates among candidate
intentions. Assume the claim is true. You receive no information about the
source of the claim, when it was acquired, or how well it is supported.
CONTEXT: claim: {observer} reports that {phi}={value};
event_ref={omega_ref} (opaque identifier, generated at random per claim: no
timestamp, no source, no relation to any other field of the record);
hypotheses: {eta_i: intent_i, preferred_action_i}_{i<=K};
task_semantics={goal}; legal_actions={legal_actions}.
TASK: Return JSON in exactly this format, with no additional text before or
after:
{"r":{"eta1":<1-5>,...},"reason":"..."}.
Each r value is an integer: 5 makes the intention much more plausible,
3 leaves it unchanged, 1 makes it much less plausible.
Fields not provided and not to be inferred: source identity, acquisition step,
record contents, support strength, conflict flag, source-trust weight,
current preference distribution.
Answer:
```

```
[CALL 4 / continuation]
SYSTEM: You select the next action. You receive facts that passed the support
check and a revised preference distribution over teammate intentions.
CONTEXT: task_goal={goal}; qualified_facts=[{phi}={v} (claim_ref {omega_ref}),
...]; preference_distribution={eta -> probability}; legal_actions={actions};
remaining_budget_units={B}; step={t}/{D}.
TASK: Return exactly one action string in this format:
[<verb>] <target>
Do not add any extra text before or after.
When the system message asks you to score newly qualified claims, also return
the optional "r" object on the 1-5 scale described in the system message;
otherwise omit it. Raw peer records and unqualified claims are not provided
and must not be assumed.
Answer:
```

P and T are computed in code and appear in no prompt. C is returned by call 2 only, as an integer on the 1–5 scale that the harness maps to its three-valued form. $\alpha_t(e) = P(e)T_t(t)C_t(e)$ is assembled in code and never returned by a model. ω is the ledger's event identifier and reaches only call 2. Call 3 receives the random reference ω_{ref}, which is not derivable from ω, the source or the acquisition step. Call 4 receives ω_{ref} only as the claim reference of an already qualified fact. An empty cell means the field does not appear in that call's prompt. The manifest is a complete statement of the isolation claim. It asserts that call 3 contains none of the authorization fields, and Appendix L asserts at run time that none of them appears in the rendered text.

K UPDATE PROPERTIES

This appendix derives the per-contribution bound on the log-odds shift, states the pairwise consequence of that bound, gives an explicit counterexample for the global version, and states the strengthened condition that version would require.

The per-contribution bound. If $\alpha_t(e) = 0$, all multiplicative factors equal one. If compatibility is constant across candidate intentions, centered compatibility is zero. Both cases preserve prior preferences. Centering cancels in the difference, so the update may be evaluated with either form of the compatibility score. For any two hypotheses, the normalizer cancels.

$$\log \frac{b^+(\eta)}{b^+(\eta')} = \log \frac{b^-(\eta)}{b^-(\eta')} + \beta\alpha_t(e)\,[r(e,\eta) - r(e,\eta')]\,.$$

Since $r(e,\cdot) \in [-1,1]$, the difference $r(e,\eta) - r(e,\eta')$ lies in $[-2,2]$. Multiplying by $\beta\alpha_t(e)$ therefore yields the upper bound $2\beta\alpha_t(e)$. The bound is a one-line consequence of the range of r, and we present it as such rather than as a substantive result.

Pairwise preference preservation. The same inequality read pairwise gives the following. Only its final step uses the assumptions. The argument $\log \frac{b^+(\eta_1)}{b^+(\eta_2)} \geq \log \frac{b^-(\eta_1)}{b^-(\eta_2)} - 2\beta\alpha_t(e)$ holds for any two hypotheses with $b^-(\eta_1) > b^-(\eta_2) > 0$. Neither the distinct-action hypothesis nor the symmetric-loss hypothesis enters the algebra. They explain why a preserved pairwise preference is of interest for commitment, not why it is preserved. The numerical identity for a pair with distinct actions follows because the two actions are then different, so a pairwise reversal is a change of preferred action within the pair.

Consider candidates η_1, η_2 with distinct preferred actions and $b^-(\eta_1) > b^-(\eta_2)$. This is without loss of generality, since the case $b^-(\eta_1) < b^-(\eta_2)$ follows by exchanging indices, and at equality the hypothesis fails, so there is nothing to prove. A pairwise preference reversal occurs iff $b^+(\eta_2) \geq b^+(\eta_1)$, equivalently $\log \frac{b^+(\eta_1)}{b^+(\eta_2)} \leq 0$. From the bound derived above, $\log \frac{b^+(\eta_1)}{b^+(\eta_2)} \geq \log \frac{b^-(\eta_1)}{b^-(\eta_2)} - 2\beta\alpha_t(e)$. If $\log \frac{b^-(\eta_1)}{b^-(\eta_2)} > 2\beta\alpha_t(e)$, then $\log \frac{b^+(\eta_1)}{b^+(\eta_2)} > 0$, guaranteeing that η_1 remains strictly preferred within the pair $\{\eta_1, \eta_2\}$. Sufficiently strong priors therefore prevent a single report from reversing the relative ranking of two hypotheses.

Counterexample for the global version. The argument above establishes only the pairwise preference. With $K = 4$, a third hypothesis η_3 can overtake the pair even when the pairwise ordering is preserved. Let $\beta = 0.25$ (Table 11), $\alpha_t(e) = 1$, and $\mathcal{H}_t = \{\eta_1, \eta_2, \eta_3\}$ with distinct preferred actions.

$$b^- = (0.45,\ 0.20,\ 0.35), \qquad \left|\log(b^-(\eta_1)/b^-(\eta_2))\right| = 0.8109 > 2\beta\alpha_t(e) = 0.5.$$

Take $r(e,\cdot) = (-1,-1,+1)$, so $\tilde{r} = (-\frac{2}{3}, -\frac{2}{3}, +\frac{4}{3})$. Then $b^+ = (0.3667,\ 0.1630,\ 0.4703)$. The pairwise ordering is preserved, with $\arg\max_{\{\eta_1,\eta_2\}}$ equal to η_1 before and after. The global $\arg\max$ shifts from η_1 to η_3, so the Bayes commitment action changes. The pairwise result is therefore correctly stated as pairwise and cannot be read as an action-invariance guarantee for the belief. A second counterexample shows that a hypothesis stated on hypotheses, rather than on action masses, fails even as a sufficient condition when actions are shared. With $b^- = (0.49, 0.26, 0.25)$, preferred actions (a, a', a'), $\beta = 0.25$, $\alpha = 1$ and $r = (+1, -1, -1)$, we get $b^+ = (0.613, 0.197, 0.190)$. The two largest action masses move from $(0.49, 0.51)$ to $(0.613, 0.387)$ and the Bayes action flips, although $\log(b^-(\eta_1)/b^-(\eta_2)) = 0.634 > 2\beta\alpha$. The two counterexamples show that an entropy condition alone cannot guard the commitment action. The paper's dual gate therefore adds an action-disagreement condition.

Scope of the pairwise result. For a global action-preservation guarantee the hypothesis must be stated on *action masses*, and a sufficient condition is given below.

$$\log \frac{m^-_{a^{(1)}}}{m^-_{a^{(2)}}} > 2\beta \sum_{g \in \mathcal{M}_t} \alpha_{t,g},$$

Here $m^-_{a^{(1)}}$ and $m^-_{a^{(2)}}$ are the two largest action masses under b^-, that is $m^-_a = \sum_{\eta:\, a^*(\eta)=a} b^-(\eta)$. Sufficiency follows because any two multipliers $w_\eta = \exp(\beta \sum_g \alpha_{t,g} \tilde{r}(g,\eta))$ differ by at most a factor $\exp(2\beta \sum_g \alpha_{t,g})$. So $m^+_{a^{(1)}}/m^+_a \geq \exp(-2\beta \sum_g \alpha_{t,g})\, m^-_{a^{(1)}}/m^-_a > 1$ for every other action a. The condition is sufficient and is not necessary. It is stated on action masses rather than on the two highest-mass hypotheses, because several hypotheses may share a preferred action. A condition on hypotheses alone is neither sufficient nor necessary. The sum on the right is the ledger quantity, not the single-contribution factor, so the guarantee weakens as active groups accumulate. The condition bounds a Bayes action computed from b_t. The deployed action is chosen by the continuation call (Section 3), so neither the pairwise result nor this condition constrains the realized policy directly.

Composition over the active ledger. The bound above applies to a single contribution. When the ledger is active, the relevant bound is $2\beta \sum_{g \in \mathcal{M}_t} \alpha_{t,g}$. With $L_{\text{ledger}} = 16$ and the empirical 95th-percentile active strength of 2.85, the realized bound is $2 \cdot 0.25 \cdot 2.85 = 1.43$. The theoretical bound of $2\beta L_{\text{ledger}} = 8$ corresponds to a fully saturated ledger in which every group has strength 1, which the empirical distribution does not approach. The five contributions traced in Table 19 realize log-odds shifts of 0.070–0.116, between 0.30 and 0.40 of the single-contribution bound.

L Evaluation Protocol, Artifacts, and Execution Schema

This appendix documents the release protocol, the per-call logging schema, the run-time isolation assertion, and the derived quantities reported in the tables.

Open-source repository. All experiment scripts, prompt templates, 600 base snapshot records and per-call trace records will be made available upon acceptance. The released package will contain five items. Item (i) is the four prompt templates of Appendix I.2, and item (ii) is the 600 base snapshot records with the derived CE and HR versions. Item (iii) is the per-call trace records with the schema below, and item (iv) is the scripts that regenerate every table and figure from those records. Item (v) is the controlled-set generator, including the screening thresholds, the per-cluster acceptance and rejection counts, and the spawn-seed list. The four templates are reproduced verbatim in Appendix J and are therefore checkable without the artifact.

Call accounting schema. Row granularity is per LLM call. The released logs follow the schema below.

```
run_id, episode_id, scene_id, decision_id, call_id,
method, model, model_revision, call_type, claim_id,
request_time, response_time,
input_tokens, output_tokens,
retry_flag, repair_flag, fallback_flag, success_flag
```

The `call_type` field records which of the four scheduled calls the row corresponds to, and the `claim_id` field records which authorized claim the row is indexed by. Together they make the decomposition of Table 9 verifiable and let the split inside the hypothesis-generation-plus-continuation residual be recomputed from the logs. The `input_tokens` and `output_tokens` fields support the per-instance token-cost histogram of Appendix N.2. The `repair_flag` field supports Table 7, and the `method` field supports the per-method call-matching analysis that separates the inference-budget effect from the revision effect (Appendix N). The `fallback_flag` field supports the fallback rates of Table 7 and the closed-loop query statistics of Table 26. The `success_flag` field allows the invalid-call rate of Table 7 to be recomputed from the raw logs. The `model_revision` field is the field that would make the Qwen rows of Table 10 byte-reproducible. It is present in the released schema but was not populated for the runs reported here.

Isolation assertion. Every prompt is assembled by a single constructor per call type. The compatibility constructor asserts a field-level whitelist at run time, so a field cannot reach the prompt by accident.

```
ALLOWED_COMPAT_FIELDS = {"observer", "phi", "value",
                         "omega_ref", "hypotheses",
                         "goal", "legal_actions"}
def build_compat_prompt(claim, omega_ref, hypotheses, goal, actions):
   fields = {"observer": claim.observer, "phi": claim.phi,
           "value": claim.value, "omega_ref": omega_ref,
           "hypotheses": hypotheses, "goal": goal,
           "legal_actions": actions}
```

```
  assert set(fields) == ALLOWED_COMPAT_FIELDS, \
     f"isolation violated: {sorted(set(fields) ^ ALLOWED_COMPAT_FIELDS)}"
  return render(COMPAT_TEMPLATE, **fields)
```

A substring test on the rendered text cannot be used here, because the template's own instruction block names the excluded fields in order to tell the model not to infer them. A substring assertion would therefore fail on every well-formed prompt. The whitelist above is the assertion that runs, and the released call logs record the rendered-prompt hash, so a reader can re-run the same check on the logs. The assertion is the reason the field manifest of Table 51 can be read as a claim about the emitted text and not only about the source code.

Derived quantities. Several reported quantities follow from identities among other reported cells rather than from a separate measurement. The four regime rows of Table 27 sum to 600 bases and 299 queries and to the Errors/600 totals of Table 2. The decision-count sum is $2{,}403 = 1{,}472 + 931$ in Table 20, and the query and gate sums are in Table 26. Every budgeted frontier entry of Table 54 is computed as $\mathbb{E}[\min(X, C)] = \mathbb{E}[X] - \mathbb{E}[(X - C)^+]$ from the stated bin values and counts. The additive identities of Table 22 are arithmetic consequences of the per-instance counts rather than a substitute for them. The McNemar statistics, Wald intervals and χ^2 values are deterministic functions of the integer discordant counts reported alongside them.

Reproduction of a single decision. A fixed-seed matched trace of a single controlled-set episode would make the mechanism auditable end to end, and we flag its absence in Appendix M. The closest available trace is the claim-level trace of the five official episodes in Table 19.

M Limitations, Ethics, and Licensing

Evaluation validity and limitations. The two controlled evaluation sets are curated from the corresponding environments and do not use the official test splits. They are screened with the certificate instantiated in Appendix F.1, which is computed from the scripted controller's ground-truth goal and recorded query outcomes. The reported effects therefore describe the peer-resolvable regime rather than the natural episode distribution. The size of that selection is measured rather than only declared. On the same generator without screening the contrast is $+0.8$ and $+0.6$ points with intervals including zero (Table 21), so the unscreened contrast is roughly one quarter of the screened one. The derived episodes are also about nine times shorter than the official TDW-MAT episodes (Table 20), so they are a different task rather than a larger sample of the same one. The certificate has been applied to the official PARTNR validation split for incidence reporting only.

The seven external-method rows in Table 1 are evaluated using ported implementations under a matched-input protocol. No port commit is pinned, no port is verified against its published native-benchmark number, and two ports are weaker comparisons than their names imply. The two weaker ports are COMBO without its world model and CoBel-World with an adapted message parser. PCE's reference implementation is public and was not run, so evaluating GPER as an add-on to PCE on the official split is the most informative missing experiment. Call matching has been added for the two comparators that matter most (Table 4), and it narrows but does not close the gaps. The remaining comparison is therefore no longer purely an inference-budget artefact for those two rows, but it still is for the others. Point estimates and intervals in Table 1 average over five inference seeds on the full episode sets, so inference variability is propagated for the primary contrast. The task and inference components are combined by a normal approximation in quadrature rather than by a two-level bootstrap. The seed design shares task realisations across rows, so the two components are not independent and the combined interval is conservative rather than exact.

The diagnostic suite is paired by construction and uses paired resampling or McNemar tests throughout. Its labels and its record-support metadata come from the same harness, so agreement with that harness is not independent evidence of grounding quality. The noise-injection condition of Table 24 bounds that advantage at about 20% field noise, beyond which it is no longer established. The diagnostic metric weights harmful revisions at 380/600, which is why the balanced criterion is the primary endpoint. GPER's advantage on the harmful-revision subpopulation is resolved (29 against 43 for QF and 54 for Four-check, with HR-only paired intervals $[2, 26]$ and $[10, 40]$ in Table 31). It is bought with correction errors, because its CE/420 of 245 is worse than eight of the thirteen other reported conditions. On the balanced criterion of Table 23 Four-check is 0.050 better than GPER with a paired interval excluding zero, while GPER, QF and CoBel-World are mutually

indistinguishable. The CE/HR trade-off is therefore the summary of GPER's diagnostic profile. It makes fewer harmful revisions and more correction errors, and the Errors/600 ranking should not be read as a recommendation over the balanced one.

The diagnostic-suite token costs are computed from the per-instance token-cost histogram recorded on the diagnostic runs (Appendix N.2), so cost-benefit statements resting on Table 54 describe that recorded distribution. The token-budget rows of that table are determined by the histogram. The wall-clock rows require a latency model the harness does not record, and no values are printed for them. The error counts under a token cap are not independently reproducible from the histogram alone, because the mapping from a capped instance to a specific outcome is not recorded. Those cells are therefore taken from the capped re-runs described in Appendix N.2.

Method limitations. GPER has several properties that bound its scope. Authorization depends on the source-local observation interface and does not solve image-only perception or authenticate arbitrarily compromised records. The interface supplies the record checks that the adapter consumes, so the adapter-based advantage reported here is an upper bound relative to a deployment in which coverage and capability fields come from noisy perception. Table 24 shows how fast that bound is approached. Compatibility can introduce unjustified evidence-to-intent links when behavioral context is unobserved. The isolation test of Appendix H.4 establishes that the compatibility evaluator is insensitive to fields that were removed from its prompt, not that its direction assignments are correct. The direction-assignment comparison is the only tested version of that claim and is quantified there. Entropy triggers can miss confident errors, and absent hypotheses cannot be recovered by reweighting alone (Appendix F.7), which bounds the method on 4.5% of bases. The correlated-evidence test of Appendix H.3 (Table 42) shows an odds inflation of 1.50 when two peers report the same fact under distinct event identifiers. That value is within 0.02 of the independent-facts control and therefore leaves the ledger unable to distinguish duplication from independence. Finally, the exponent $\beta\alpha\tilde{r}$ is a heuristic score rather than a calibrated log-likelihood ratio (Appendix N.3).

The absent controls are those listed in Appendix B. They are a Jeffrey-conditioning update (Jeffrey, 1965) and a subjective-logic discounting-plus-dependent-fusion rule (Jøsang, 2016), neither claimed as a result anywhere in this paper. They also include the Hypothetical Minds port, listed in Table 16 as not evaluated, and no matched declaration control for Table 46.

Use of language models in preparing this manuscript. The experimental pipeline executes LLM calls because the method under study is an LLM pipeline, and that use is part of the method. In addition, language models were used to draft and edit portions of the prose and to check the internal arithmetic consistency of the tables. We disclose this use here rather than in the acknowledgements, in accordance with the conference policy.

Open items. The following would still require additional runs or logs. The first is the per-instance call-type breakdown needed to split the hypothesis-generation-plus-continuation residual of Table 9. The second is the identity of the call that the harness skips when the per-decision token cap binds. The third is the latency model behind the deadline rows of Table 54, and the fourth is the matched declaration control behind Table 46. The fifth is the Jeffrey-conditioning and subjective-logic rows of Table 16, and the sixth is the Hypothetical Minds row of Table 16. The seventh is a fixed-seed matched trace of a single episode from claim through α, r and b to the selected action.

M.1 Ethics, Data, and Simulator Licensing

This work evaluates simulated agents without human subjects. Simulators and datasets are used under their open licenses, namely PARTNR (MIT), ThreeDWorld (BSD 2-Clause), AI2-THOR (Apache 2.0) and VLABench (MIT). We could not locate a license file for the TEACh release and treat its terms as unverified. No TEACh content is redistributed in the released artifact. No personally identifying information is collected or stored, and all interactions are between scripted agents in simulated environments. The controlled-set generator produces episodes from simulator primitives and contains no human data, so it can be released without a data-use agreement.

Table 52: Evaluator organization with a common 4096 generated-token allowance per decision (2048 per call, Table 14).

Condition	Errors	CE	HR	Tokens (k)	p95 (s)
One call, shared inputs	130	253	51	13.7	8.4
Two calls, shared inputs	122	266	43	14.2	10.1
Two calls, isolated inputs	110	273	34	14.0	9.8

Table 53: Paired contrasts for the evaluator-organization conditions of Table 52.

Contrast	Improvement	b	c	95% CI	χ^2	p
One call $\rightarrow$ two calls, shared	$+8$	34	26	$[-7, 23]$	0.82	0.366
Two shared $\rightarrow$ two isolated	$+12$	40	28	$[-4, 28]$	1.78	0.182
One call $\rightarrow$ two isolated	$+20$	46	26	$[3, 37]$	5.01	0.025

N Call-Budget and Calibration Controls

This appendix documents the analyses that separate the effect of the additional evaluator call from the effect of information isolation. It also reports the resource frontier under three token budgets and two wall-clock deadlines, and the calibration check on the update exponent. It is distinct from Appendix H, which holds the perturbation, packaging, declaration and integration controls.

N.1 Information isolation versus call decomposition

Table 52 decomposes the effect of the two-evaluator design on the 600 diagnostic instances. All three conditions are run on the same instances, so the contrasts are paired and Table 53 reports the discordant counts. Issuing a second LLM call with the *same* shared inputs reduces errors from 130 to 122. Adding prompt-field isolation on top reduces them further, to 110.

The Errors, CE, HR and Tokens columns are GPT-4o mini results on the 600 diagnostic instances. The p95 latency column is measured on a different configuration, the Qwen2.5-72B / vLLM 0.6.3 setup of Appendix D. That column is an execution characteristic of that stack rather than a property of the rows being compared, and it is not the latency of the GPT-4o mini diagnostic runs. Paired contrasts and intervals are in Table 53. The "two calls, isolated inputs" row is not the GPER condition. It uses the same two-call organization but disables the ledger's event grouping, so the 8-error gap against GPER (110 vs 102) is the contribution of event grouping. That gap is smaller than the 12-error gap of the "no event deduplication" row of Table 3, so exact-ID deduplication and event grouping are not interchangeable here.

The improvement column is the reduction in errors from the first condition to the second. Here b counts instances where the first condition errs and the second is correct, and c counts the reverse, so $b - c$ equals the improvement. The 95% intervals are $b - c \pm 1.96\sqrt{b + c}$, and χ^2 is McNemar's statistic with continuity correction.

The overall two-evaluator design reduces errors by 20 on the shared 600 instances, with a paired interval excluding zero. The decomposition of that 20 into its two steps is not individually resolved. The second-call step contributes 8 errors and the isolation step 12, and neither step's paired interval excludes zero at this sample size. The point estimates therefore suggest that roughly 40% of the total reduction is attributable to call decomposition rather than to information isolation. This split is a point estimate rather than an established decomposition. The direction of the second step is also the opposite of the CE/HR direction of the first, since the isolated condition incurs the most CE errors (273) and the fewest HR errors (34). The one-call condition incurs the fewest CE errors (253) and the most HR errors (51). Isolation therefore trades harmful revisions for correction errors, and HR falls from 51 to 34 while CE rises from 253 to 273. A two-step effect of $+12$ with a paired interval of $[-4, 28]$ would require roughly three to four times the instances to resolve at this effect size, which is why we report the direction and not the magnitude. The CE and HR columns of the three rows are also not consistent with a single movement of the CE/HR frontier. That inconsistency suggests that the isolation step and the second call are not interchangeable interventions even though their aggregate error reductions are comparable. The compatibility template and constructor exclude the source and acquisition fields at the field level (Appendix L), so this experiment is an organization-level contrast

Table 54: Fixed-input resource frontier computed from the per-instance token-cost histogram stated in the text.

Budget	Method	Errors/600	Exhaustions/600	Tokens (k)
10k total tokens	Four-check	402	530	9.90
	GPER	391	572	9.94
14k total tokens	Four-check	245	188	12.80
	GPER	231	283	13.15
18k total tokens	Four-check	135	13	13.25
	GPER	110	42	13.96
8 s and 12 s wall-clock deadlines are not tabulated; the harness does not record a latency model				

between shared and isolated evaluator inputs rather than a test of the constructor. It is therefore a different experiment from the constructor-level check of Table 45, which perturbs fields the prompt does not contain.

N.2 Resource frontier

Table 54 reports the resource frontier under three token budgets and two wall-clock deadlines. The token-budget rows apply a per-instance cap and fall back to the local policy for instances whose natural cost exceeds the cap. The deadline rows would apply a per-instance wall-clock limit and fall back for instances that do not complete within the limit. The two blocks therefore measure different things. The token-budget block measures robustness to a hard cost cap, and the deadline block measures robustness to a hard latency cap.

The costs come from a 16-bin histogram of the per-instance token cost recorded on the diagnostic runs. For GPER the bin values are $8.0, 9.0, 10.5, 11.5, 12.5, 13.5, 14.0, 14.5, 15.0, 16.0, 17.0, 19.0, 22.0, 26.0, 32.0, 45.0$ (k tokens) with counts $8, 20, 35, 46, 58, 70, 80, 82, 74, 52, 33, 18, 12, 7, 4, 1$. Those counts sum to 600 and give a mean of 14.31 with a right-skewed tail. For Four-check the bin values are $7.0, 8.5, 10.0, 11.0, 12.0, 13.0, 13.5, 14.0, 14.5, 15.0, 16.0, 18.0, 21.0, 25.0, 31.0, 43.0$ with counts $10, 22, 38, 48, 60, 72, 80, 82, 74, 52, 33, 16, 7, 4, 1, 1$. Those counts also sum to 600 and give a mean of 13.40. The exhaustion counts are the bins strictly above the cap. The budgeted costs are $\mathbb{E}[\min(X, C)]$ over the empirical distribution, using the identity $\mathbb{E}[\min(X, C)] = \mathbb{E}[X] - \mathbb{E}[(X - C)^+]$. Recomputation from the stated bin values and counts reproduces the six token rows exactly. The coupled exhaustion counts reproduce as 572, 283 and 42 for GPER and 530, 188 and 13 for Four-check at caps of 10k, 14k and 18k. The token-budget rows are fully determined by this histogram, and the error counts under a cap are not. Those error counts require the per-instance join between a capped instance and its outcome, which the harness does not record. Those error cells are therefore taken from capped re-runs rather than from per-instance measurements, and the token and exhaustion cells are the only fully histogram-determined entries in the table.

The deadline rows require a latency model that maps a per-instance token cost and the serving configuration to a completion time. We do not have that model in a form that fixes the reported values, so the deadline rows are not tabulated.

Token costs are read off that histogram. The error cells are taken from the capped re-runs described in the text, because the harness does not record the join between a capped instance and its outcome. For token-budget rows, Exhaustions/600 counts instances whose cost exceeds the stated cap and invokes the common fallback. For deadline rows, Exhaustions/600 would count instances that fail to complete within the wall-clock deadline and invoke the same fallback. Those two quantities answer different questions and are not directly comparable across blocks. C is the per-instance token cap of this appendix and is distinct from the acquisition budget B of Section 3, so the two are not interchangeable (Appendix D.5). The token-budget rows do not order the two methods the same way on the two axes. GPER commits fewer errors at every budget, while Four-check exhausts the cap less often.

At every cap the two methods are not separated by the frontier. GPER is better on the error axis and worse on the exhaustion axis at all three budgets. At the tightest budget (10k), GPER exhausts

on 572 instances against 530, and commits 391 errors against 402. We did not measure how the budget is divided between acquisition and continuation, so this should be read as consistent with an acquisition-heavy policy rather than as direct evidence for one. The error advantage is 11 errors at 10k, 14 at 14k and 25 at 18k, and the exhaustion gap is 42, 95 and 29 instances. The error advantage therefore grows with the cap, while the exhaustion gap does not. The two frontiers rank the methods differently depending on which axis is treated as the binding constraint, and we do not present a single ordering.

N.3 Calibration of the update exponent

Section 4 states that the exponent $\beta\alpha\tilde{r}$ of Eq. 2 is a design choice rather than a derived likelihood ratio. This subsection quantifies how far from a calibrated likelihood ratio it is. We assembled 240 held-out claim instances with recorded outcomes. For each instance the empirical log-likelihood ratio is estimated from the recorded outcomes of the scripted controller's goal index. For a hypothesis η, $p(\text{claim} \mid \eta)$ is the frequency of the observed claim among the recorded episodes in which the controller held goal index η, with a Jeffreys prior so that no cell is zero. The prior adds a half-count to each cell, and the ratio is formed for the hypothesis pair nominated by the base's reference alternative. For the same pair we record the model's exponent $\beta\alpha_t(e)\tilde{r}(e,\eta)$. Regressing the former on the latter gives a slope of 0.46 and an R^2 of 0.38, and the Spearman correlation is 0.62. The exponent is therefore moderately rank-correlated with the empirical log-likelihood ratio and explains under 40% of its variance. A slope below one would be consistent with over-confidence, but attenuation from noise in the regressor produces the same pattern. We therefore read the slope as a descriptive fit rather than as a calibration curve and do not estimate an over-confidence factor from it. The estimator above is coarse, because it binarizes each claim and uses a small number of episodes per goal index. It therefore bears on the sign and rank of the exponent rather than on its scale. A binned calibration curve with counts per bin would be required to sustain any claim about the exponent's magnitude. A substantial share of the exponent's variation is also not evidence-bearing, which is part of why the boundary stated in Appendix K is pairwise rather than global.

A one-factor scan over β on the 90 development bases at the reference values of the other three axes gives 19, 15 and 17 errors at $\beta = 0.15, 0.25, 0.50$. The spread is 4 errors against a development-set contrast of 6 errors, so β matters at a scale comparable to the effect being tuned. The reference value is the best of the three values scanned, and we cannot tell whether it is a unique minimum. The two non-reference points of this scan are not among the twelve configurations of Table 12, and the scan is reported on the development set rather than on the diagnostic suite. The scan is also non-monotone, because the smaller scale degrades the count as well as the larger one. The two endpoints therefore bracket the minimum rather than supporting a monotone over-confidence reading.