INCENTIVIZING REASONING EFFORT IN LLM AGENTS WITHOUT GROUND-TRUTH FEEDBACK

Anonymous authors
Paper under double-blind review

ABSTRACT

Governing large AI swarms requires rewarding useful computation without verifying every answer. Multi-task peer prediction offers a route by paying agents from relationships among answers across tasks. To our knowledge, we provide the first empirical study of controllers choosing provider-supplied LLM reasoning settings under these payments. We model a game where utility is reward times peer score minus token cost and evaluate six models. A five-run online comparison finds no clear accuracy advantage over raw-agreement or flat payments. In a symmetric signal model, large rewards sustain maximum-effort equilibria in which agents systematically merge distinct answers.

1 INTRODUCTION

Governing large swarms of autonomous AI agents will require institutions that allocate resources, reward useful work and hold agents accountable without verifying every judgment. Multi-agent interactions create risks of conflict, collusion and information asymmetry (Hammond et al., 2025). Multi-task peer prediction (MTPP) offers a promising foundation: agents' reports across shared tasks determine incentives without requiring ground-truth labels at payment time (Shnayder et al., 2016; Kong, 2020). LLM peer elicitation and source-incentive mechanisms provide concrete precedents (Chen et al., 2025; Jiang et al., 2026). Our ambition is to turn this principle into an operational component of swarm governance, connecting incentives to the computation agents perform and the information they deliver.

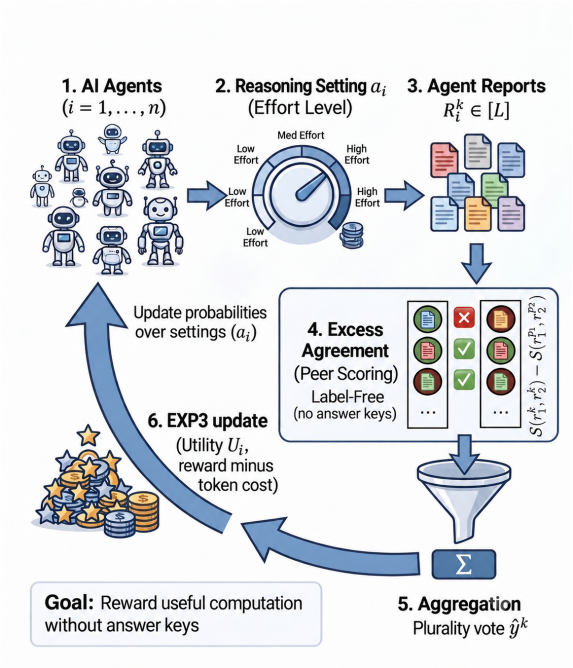
An agent's native reasoning setting changes both computation cost and the information available before it reports. Peer prediction already studies costly information acquisition and payment design (Dasgupta & Ghosh, 2013; Liu & Chen, 2016; 2017; Zhang & Schoenebeck, 2023; Xu et al., 2024). Our question is operational: **when do payments computed from peer reports induce useful reasoning choices in LLM APIs?** Answering it requires separating the agent's score–cost tradeoff, the controller that selects its action, and the answer delivered to the requester. Truthful reporting means revealing a signal; it does not mean that the signal is correct.

To our knowledge, this is the first empirical study of how multi-task peer-prediction payments shape agents' choices among metered, provider-native LLM reasoning actions. Using an established equality-minus-baseline score (Shnayder et al., 2016; Witt et al., 2026), we give a practical reward-calibration criterion: which reward values make a target setting's expected payment minus token cost at least as high as every alternative? We test whether selected settings retain their utility advantages on new questions, then evaluate delivered answer accuracy within a common inference allowance.

The equilibrium contribution quantifies a separate obstacle. Truthful payoff dominance and coarse-report equilibria are established ideas (Shnayder et al., 2016; Frongillo et al., 2026); neither implies that larger rewards select truth. In our symmetric channel model, every nonconstant signal partition becomes a strict joint reporting-and-effort equilibrium above an explicit reward threshold, even with all original report labels available. Truth can pay every agent more while no agent benefits from unilaterally restoring the lost distinctions. The result links costly computation to a reporting coordination failure, rather than supplying a new general truthfulness mechanism.

The static study issues two separate API requests per question and setting, using 20 settings in total across six models. We select settings with the highest estimated utility on 252 calibration questions, then test those unchanged selections on 182 separate confirmation questions, with peers held at

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.



1. **AI agents.** Pair each LLM with a controller; fix questions, settings and reward r before each run.
2. **Reasoning setting.** Select an API reasoning setting and obtain the model's answer.
3. **Agent reports.** Send the model's answer unchanged.
4. **Excess agreement.** Subtract cross-question matches from same-question peer matches.
5. **Aggregation.** Take a plurality vote. Use answer keys only for evaluation.
6. **EXP3 update.** Update setting probabilities using $r\widehat{q}_i - c_i$. Cost c_i counts input plus output kilotokens; the vote is not feedback.

Figure 1: Six-step peer-incentive loop. Each agent chooses a reasoning setting a_i, reports its answer, receives an excess-agreement score and updates its setting probabilities with EXP3. The requester combines reports by plurality vote. Coins denote token-equivalent payments; the arrows summarize the interaction cycle, not a dependency of learning on the aggregate answer.

maximum settings. Six models and seven reward levels give 42 combinations; 37 selections retain the highest estimated utility. For 18, uncertainty bounds also place the selected setting's utility above every alternative. These approximate bounds adjust jointly for the tested settings and rewards, with a further correction across models. The 182 questions are the uncertainty units; the 42 combinations are not independent replications (Section 4.1).

The online study fixes question assignments, rewards and learning parameters before collection; controllers then adapt their reasoning settings while sending answers unchanged. Five 105-question workloads, three models and five methods require 7,875 separate API calls. At a \$1 allowance per method/workload, excess agreement answers 83.8% of assigned questions correctly, versus 83.6% for raw agreement and 84.0% for flat payment. Against each adaptive comparator, only three of $5 \times 15 \times 3 = 225$ run/round/model choices differ, so this short experiment shows no clear accuracy advantage. Separate transformations of stored answers can preserve rewards while degrading accuracy; withholding tests expose a score-denominator vulnerability. Historical data and synthetic tests supply additional evidence.

We connect costly information-acquisition games to LLM APIs by treating reasoning settings and reporting policies as choices, metered token use as computation cost, and peer payments as incentives. We characterize which settings a reward makes optimal, compare delivered accuracy and resource use, and derive explicit costly-effort partition equilibria in a symmetric signal model. Our online learner updates reasoning settings only; the equilibrium analysis permits joint effort and reporting deviations. These results identify computation incentives and reporting failures that swarm governance must address. Section 6 sets out the incentive, verification and population-level evidence needed for large autonomous deployments.

2 Decisions, Payments and Evaluation

Who decides what. A requester assigns questions with $L \geq 2$ answer options to $n \geq 2$ agents. Each agent combines an LLM with a software controller. The controller selects a provider's reasoning setting $a_i \in \mathcal{A}_i$, such as an effort level or thinking-token budget; this finite menu defines its available

inference actions. The model produces an answer, and the agent sends a report R_i. Truthful reporting passes on that answer unchanged, even if it is wrong. The requester combines reports by plurality vote, breaking ties toward the first label. Figure 1 separates computation, reporting and payment.

The controller's objective. For a setting a_i, reporting policy F_i and fixed behavior b_{-i} of the other agents, let q_i be expected peer score and c_i expected computation cost. The controller maximizes

$$U_i(a_i, F_i; r, b_{-i}) = rq_i(a_i, F_i; b_{-i}) - c_i(a_i), \qquad r \geq 0. \tag{1}$$

The quantities have distinct roles: q_i measures agreement, rq_i is the incentive payment, and U_i subtracts the cost of producing an answer. In our controllers, c_i counts input plus output tokens in thousands (kilotokens), so payment and utility are token-equivalent quantities. Dollar inference charges are separate provider costs. We test software optimizing this objective, not an LLM's preference for money. Participation is assumed even at negative utility; payments are not clipped to prevent losses.

How peer score is calculated. Excess agreement rewards matching a peer on the same question and subtracts matches across different questions (Shnayder et al., 2016; Witt et al., 2026). The subtraction discounts matches due to frequent labels. For $m \geq 2$ assigned questions,

$$\widehat{q}_{ij} = \frac{1}{m}\sum_{t} e(R_i^t, R_j^t) - \frac{1}{m(m-1)}\sum_{t \neq v} e(R_i^t, R_j^v), \quad e(a,b) = \mathbf{1}\{a = b,\ a \in [L]\}. \tag{2}$$

Here $[L] = \{1, \ldots, L\}$ is the valid-label set; R_i^t is agent i's report on question t. Scores average over the other $n-1$ agents. Both denominators count all assignments, and missing reports never match. This fixed-assignment convention prevents one form of score inflation; Section 5.3 tests withholding explicitly. Shared failures can still give otherwise constant reports positive scores. Under independent, identically distributed complete reports, (2) estimates the covariance score in Section 5.

The requester's objective. Models answer ten-option MMLU-Pro questions (Wang et al., 2024). Each API record contains the chosen setting, answer label, token use and provider charge when available. *Delivered accuracy* is correct swarm answers divided by all assigned questions; invalid, missing and unserved answers count as incorrect. We compare this accuracy within the same dollar inference allowance. Answer keys are used only for evaluation, never for payments or setting choices. Agreement, utility and correctness must therefore be measured separately (Burrell & Schoenebeck, 2023; Kim et al., 2025).

Two complementary studies. The *online study* adapts settings after each round on 525 questions: five 105-question runs with three models and five methods. "Online" means adaptation during collection, not model training. The *static study* instead calls all 20 settings across six models, twice per question/setting. We select settings on 252 calibration questions and test them on 182 different confirmation questions, balanced across 14 subjects: $2 \times 20 \times (252 + 182) = 17{,}360$ calls. Its stored answers also support reporting interventions without new model calls. Neither study shares questions with the other. Appendices A and B give the protocols, separate 560-call pilot and reduction in confirmation questions made before testing. Historical archives and simulations are supplementary evidence, not additional independent LLM experiments (Appendices F–I).

3 Do peer payments improve accuracy within a budget?

We first test the requester's objective: correct answers within a dollar inference allowance. The controllers send model answers unchanged, so this comparison tests adaptation of computation alone. Each of five runs uses a different 105-question workload, split into 15 rounds of seven questions. GPT-5 nano, Qwen3.5 Flash and DeepSeek V4.1 Flash each choose one setting per round. We compare three adaptive payment rules: excess agreement, raw agreement (same-question matches without subtraction), and flat payment (no report-dependent reward). Each adaptive controller uses EXP3, which updates setting probabilities using only the chosen setting's payment minus token cost. Before collection we fix reward $r = 32$ and exploration $\gamma = 0.2$, so 20% of the sampling probability is uniform over settings. Fixed-minimum and fixed-maximum baselines always use each model's

Table 1: Five methods at a $1 inference allowance per 105-question workload. Accuracy is the percentage of all assigned questions answered correctly by plurality vote. Entries are five-run means; ranges are the lowest/highest run, not confidence intervals. All methods serve all questions here. Cost includes reservations for unknown charges and excludes agent payments.

Method	Accuracy (%)	Run range (%)	Cost ($)
Excess agreement	83.8	[81.0, 86.7]	0.303
Raw agreement	83.6	[81.9, 86.7]	0.300
Flat payment	84.0	[81.9, 85.7]	0.309
Fixed minimum	82.3	[77.1, 85.7]	0.085
Fixed maximum	84.2	[80.0, 87.6]	0.569

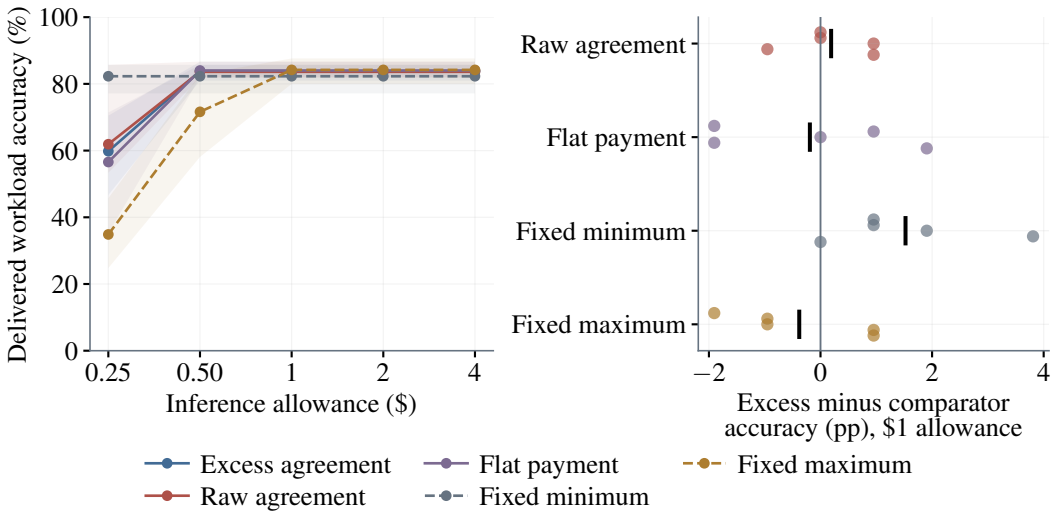


Figure 2: Accuracy within an inference allowance. Left: correct swarm answers divided by all 105 assigned questions, averaged over five runs; shaded bands span the observed lowest/highest run, not a confidence interval. Right: each dot is one run's accuracy difference between excess agreement and a comparator at $1, in percentage points; black marks are means. Unserved questions count as incorrect.

lowest or highest setting. Every method makes its own API calls on the same questions; outputs are not shared.

The primary endpoint is accuracy across all 105 assigned questions with a $1 inference allowance per method/run. We evaluate the collected call sequence at that allowance by keeping only the affordable prefix: before each call, its maximum reserved charge plus prior accounted cost must fit within the allowance. This decision uses neither that call's answer nor its realized cost. A question is served only when all three scheduled calls fit; unserved questions count as incorrect. Unknown charges retain their maximum reservation. This compares equal allowances, not equal spending or agent payments. All 7,875 planned calls were attempted once; failures remain in the data. Appendix A specifies the learner, accounting and independent audit.

At $1, every method serves all 105 questions (Table 1). Excess agreement achieves 83.81% accuracy at mean accounted cost $0.303, versus 83.62% at $0.300 for raw agreement and 84.00% at $0.309 for flat payment. Comparing methods on the same run gives excess-minus-raw differences averaging $+0.19$ percentage points (run range $[-0.95, +0.95]$), and excess-minus-flat differences averaging -0.19 ($[-1.90, +1.90]$). These five runs show no clear accuracy advantage and do not establish equivalence.

The controllers scarcely choose different settings. Each adaptive comparison contains $5 \times 15 \times 3 = 225$ run/round/model decisions; excess agreement differs from each comparator in only three. The methods use the same random draws to select actions within a run, and their probability updates

remain too similar to separate most choices over 15 rounds. They still obtain separate model outputs. The small accuracy differences therefore provide little evidence that changing the payment rule changes reasoning behavior. Longer adaptation could produce different policies; this experiment does not test that outcome.

Fixed minimum reaches 82.29% at \$0.085, versus 84.19% at \$0.569 for fixed maximum. Excess agreement gains 1.52 percentage points over minimum at 3.58 times its inference cost; we do not test other uses of those savings. At the secondary \$0.25 allowance, minimum serves every question; excess serves 70.7% and answers 59.8% of all assigned questions correctly. Accuracy among served questions alone would hide the unserved workload (Figure 2).

The separate static study also finds no clear accuracy gain from maximum or reward-selected settings over minimum settings; Appendix B reports the token costs and paired uncertainty intervals. These results motivate two separate checks: whether payments make particular settings utility-maximizing (Section 4), and whether rewarded agents preserve their answers when reporting (Section 5). The latter is not tested by controllers that always pass answers through unchanged.

4 Which reasoning settings can payments incentivize?

The online comparison tests choices reached by a learner in 15 rounds. We now ask which choices would maximize utility if scores and costs were known. Costly information acquisition and reward selection are established questions (Dasgupta & Ghosh, 2013; Liu & Chen, 2016; 2017; Zhang & Schoenebeck, 2023; Xu et al., 2024; Kong, 2022); here the choices are measured API configurations.

A reward *implements* a setting when that setting maximizes payment minus cost among the available settings. Fix the reporting policy and other agents' behavior, and write q_a, c_a for each setting's expected score and cost. Its utility $rq_a - c_a$ is a line in the reward r; the highest line identifies the best choice. Intermediate settings must be included.

Proposition 1 (Finite-action implementation)**.** *A target a^* maximizes utility (is a best response) at reward $r \geq 0$ if and only if*

$$r\Delta q_b \geq \Delta c_b \quad \textit{for every } b \neq a^*, \qquad \Delta q_b = q_{a^*} - q_b, \quad \Delta c_b = c_{a^*} - c_b. \tag{3}$$

The rewards satisfying all comparisons form an interval, a single point or the empty set. If a competitor has the same score at lower cost, no reward implements the target. The target is uniquely optimal when every inequality in (3) is strict. Appendix E gives the interval endpoints and proof.

For three settings with $(q, c) = (0, 0), (1, 1), (1.5, 3)$, the highest setting matches or beats the lowest at $r \geq 2$, but the middle setting only at $r \geq 4$. Comparing just the endpoints therefore understates the required reward. A more costly action with no score advantage cannot be implemented by increasing r. The characterization also applies to finite libraries of joint inference/reporting actions, with guarantees limited to that library.

Testing estimated utility differences. Scores and costs must be estimated from model outputs. If two settings have nearly equal scores, dividing by their estimated score gap can give an unstable reward threshold. We instead check whether a chosen setting's payment-minus-cost advantage over every alternative remains positive after accounting for uncertainty. A utility-gap interval above zero supports the choice; one below zero contradicts it; one spanning zero leaves it unresolved. Appendix E gives sufficient inequalities. We test these utility comparisons on held-out questions below.

4.1 Do utility-maximizing choices hold on new questions?

For each model, all peers stay at maximum settings. On calibration questions we select the setting maximizing $r\widehat{q} - \widehat{c}^{\,\mathrm{upper}}$ at rewards 0, 2, 8, 32, 128, 512 and 2048. For missing token usage, we assign bounds from zero to the input allowance plus output cap; $\widehat{c}^{\,\mathrm{upper}}$ uses the upper bound. We keep each selection fixed and compare it with every alternative on confirmation questions. The test concerns utility, not answer accuracy.

We report two checks for each of the $6 \times 7 = 42$ model/reward combinations. A *point match* means that the selected setting still has the highest estimated utility. A *supported choice* additionally requires

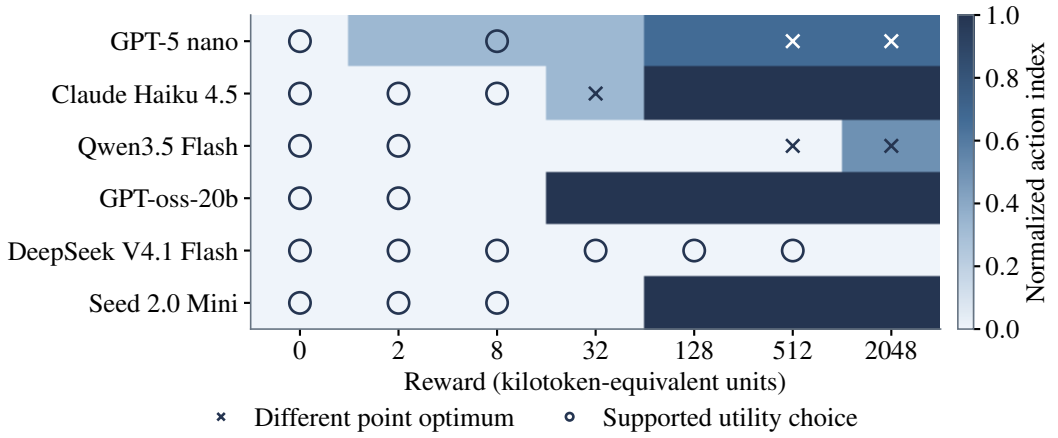


Figure 3: Testing settings selected on calibration questions. Each cell is one model/reward combination. Color locates the selected setting in that model's ordered menu: 0 is lowest, 1 highest, not a shared computation scale. A cross means another setting has higher estimated utility on confirmation questions. A circle means uncertainty bounds put the selected setting's utility above every alternative. Utility is payment minus token cost.

a strictly positive lower uncertainty bound on its utility advantage over every alternative. We resample questions 2,000 times within subject, keeping all models, settings and both calls for a question together. The resulting approximate simultaneous bands adjust for all compared rewards/settings and unknown-cost endpoints within each model, with error level $0.05/6$ across models. Appendix B gives the calculation; these are not exact finite-sample guarantees.

Selections match the highest confirmation utility estimate in 37/42 combinations, but only 18/42 meet the uncertainty criterion (Figure 3). Six supported choices are at zero reward; 12/36 are at positive rewards. The other 24 choices lack this support, including five with a different point optimum. These counts share questions and are not 42 independent replications. Table 6 reports every estimated utility shortfall and upper bound. Uncertainty is estimated from 182 question units, not 7,280 independent confirmation calls; the two calls per setting are analyzed separately and averaged, never pooled into a larger vote.

Effects differ across models. GPT-5 nano's lowest-to-highest accuracy rises from 43.7% to 71.7%, with 16.26 times the token cost; Claude's estimates are 78.8% and 78.3%. DeepSeek's minimum setting is selected at every reward and supported at six of seven. Nano selects low at rewards 2–32 and medium at 128–2048, never high. These are complete API configurations, not a universal measure of reasoning depth (Figure 7). Confirmation includes 7,130 valid outputs among 7,280 calls and eight unavailable token meters. Invalid answers count as wrong; missing meters widen utility bounds.

5 Can rewarded computation still produce misleading reports?

Utility-maximizing computation is only one requirement: agents must also preserve the information they obtain. The online controllers did not alter reports. We therefore examine reporting separately, first in a signal model and then by changing stored LLM answers. The theorem does not explain the nearly identical setting choices in Section 3; it identifies a different obstacle that can remain even when computation is maximal.

5.1 High rewards can sustain systematic answer merging

Let $Z_i \in [L]$ be agent i's answer before reporting. A reporting policy $F_i(z, b)$ gives the probability of sending label b after obtaining z; each row sums to one. Truthful reporting is $F_i = I$. For fixed settings and independent, identically distributed tasks, define the signal covariance matrix $\Delta_{ij}(z, w) = P(Z_i = z, Z_j = w) - P(Z_i = z)P(Z_j = w)$. After applying the reporting policies,

the score is

$$q_{ij}(F_i, F_j) = \mathrm{tr}(F_i^\top \Delta_{ij} F_j), \qquad q_i = \frac{1}{n-1} \sum_{j \neq i} q_{ij}. \tag{4}$$

Here tr sums a matrix's diagonal, and q_i averages over all other agents.

To isolate reporting incentives, replace the discrete API menu by an idealized continuous effort $s \in [0, 1]$. All agents share the same accuracy and cost functions. True labels are equally likely; agents' signals are independent given the true label. At effort s, an agent sees the correct label with probability $p(s)$ and each wrong label with probability $(1 - p(s))/(L - 1)$. Let $u(s) = p(s) - 1/L$ be accuracy above chance, where $u(0) = 0$, $u(1) > 0$, and $p'(1) > 0$. Assume differentiable p, c, nondecreasing concave p, strictly increasing c with $c(0) = 0$, and either strictly convex c or strictly concave p with convex c. Direct calculation gives

$$\Delta_{ij} = u(s_i)u(s_j)H, \qquad H = \frac{LI - \mathbf{1}\mathbf{1}^\top}{(L-1)^2}, \qquad \mathrm{tr}\, H = \kappa = \frac{L}{L-1}. \tag{5}$$

Here I is the identity matrix and $\mathbf{1}$ the all-ones vector. When both agents' effort is positive, the covariance diagonal is positive and the off-diagonals negative. This sign structure is an assumption of this model, not a property established for every empirical model pair.

Proposition 2 (Truthful maximum-effort equilibrium)**.** *Under the preceding assumptions, if $r\kappa u(1)p'(1) \geq c'(1)$, then truthful reporting at maximum effort is the unique best response to peers using that same strategy. The all-truthful, maximum-effort profile is therefore a strict symmetric equilibrium.*

The reward condition makes the last increment of computation worth its marginal cost. A strict equilibrium means that any one agent loses utility by changing its computation or reporting while the others keep their strategies. The proposition establishes such an equilibrium, not that agents will reach it. Existing correlated-agreement guarantees rank truthful joint payoffs highest without selecting a unique equilibrium (Shnayder et al., 2016). Coarse-report equilibria are also known when fewer reports than signals are available (Frongillo et al., 2026). Here every original label remains available: we quantify how continuous costly effort and arbitrary shared partitions interact. Partition the signal alphabet into $k \geq 2$ nonempty cells with sizes $m_1, \ldots, m_k$. Let G report one distinct label per cell, merging signals within the cell: for example, report A for either A or B, and C for C.

Theorem 3 (Strict informative partition equilibria)**.** *Under the assumptions above, define*

$$K_G = \mathrm{tr}(G^\top H G) = \frac{L^2 - \sum_{j=1}^{k} m_j^2}{(L-1)^2} > 0. \tag{6}$$

If $rK_G u(1)p'(1) > c'(1)$, then $(G, 1)$ is the unique joint best response to peers playing $(G, 1)$. Every such nonconstant partition supports a strict maximum-effort equilibrium at a sufficiently large reward.

Appendix C proves both results by optimizing reports and then effort.

Reward thresholds and payoff dominance. Write $r_G^* = c'(1)/(K_G u(1)p'(1))$ and $r_I^* = c'(1)/(\kappa u(1)p'(1))$ for the coarsened and truthful effort thresholds. Their ratio is

$$\frac{r_G^*}{r_I^*} = \frac{L(L-1)}{L^2 - \sum_j m_j^2} \in [1, L/2]. \tag{7}$$

The largest ratio is attained by cell sizes $(L - 1, 1)$. Consequently a reward above $(L/2)r_I^*$ sustains every nonconstant partition. At maximum effort, truth gives each agent $ru(1)^2(\kappa - K_G)$ more utility than a genuinely lossy partition, yet both profiles remain strict against unilateral joint deviations. Higher truthful payoff therefore does not select truthful play. The contribution is an explicit costly-effort characterization of this coordination failure, rather than the discovery that peer prediction can have undesirable equilibria.

For example, with three labels, $p(s) = 1/3 + 0.3s$ and $c(s) = 0.01s + 0.5s^2$, reward 12 sustains both truthful and two-cell maximum-effort equilibria (Appendix C). The lossy convention is not a shared permutation; excluding permutations does not eliminate it.

Equilibrium existence does not establish which convention learning will reach. Appendix D gives a local exact-gradient result and counterexamples to global convergence and joint-regret claims, complementing Feng et al. (2022). These results do not describe the noisy controller in Section 3.

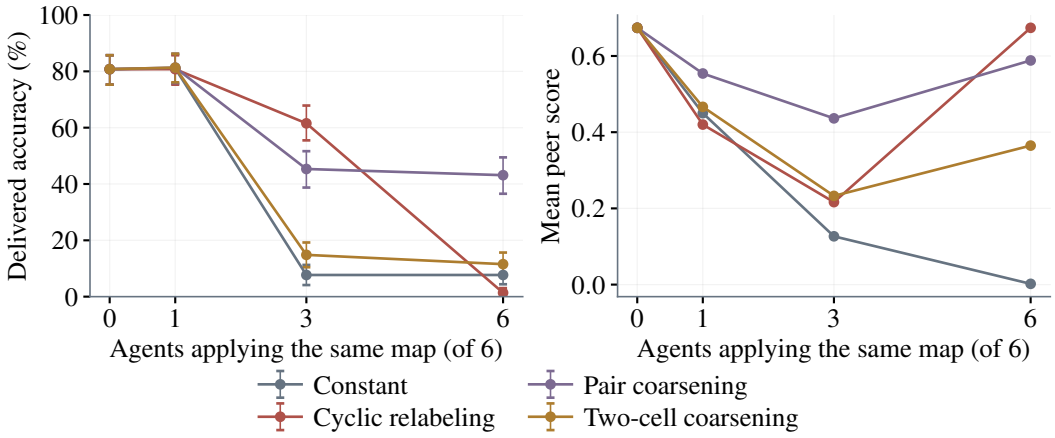


Figure 4: Changing stored answer labels without new model calls. The horizontal axis counts agents applying a shared reporting rule; zero is unchanged reporting. Rotating every agent's labels preserves peer score but changes which answer option is selected. Accuracy bars are exploratory unadjusted 95% question-bootstrap intervals; peer scores are point estimates. Lines connect only the tested group sizes.

5.2 What happens when model answers are changed?

We transform the stored confirmation answers obtained at maximum settings, without querying the models again. One, three or all six agents apply the same rule: always report A; rotate labels A→B through J→A; merge pairs A/B→A, C/D→C, and so on; or merge A–E→A and F–J→F. Coalition membership follows a fixed roster order. Missing reports remain missing and inference costs stay unchanged. We measure the resulting plurality accuracy and mean peer score. These imposed transformations test scoring vulnerabilities; the models did not choose to collude.

When all six rotate labels, plurality accuracy falls from 80.77% to 1.37%, while mean score stays exactly 0.673934: applying the same permutation preserves every equality comparison. Pair merging gives 43.13% accuracy and score 0.588307; merging into two categories gives 11.54% and 0.364818. Constant reports retain score 0.001983 because some agents fail on the same questions. Information loss thus extends beyond relabeling, as in Theorem 3, although these models need not satisfy its symmetric-signal assumptions.

Accuracy damage does not establish an equilibrium. We also ask whether one agent can profit by changing its reports while the others keep each convention. On calibration questions, we learn its highest-scoring answer-to-report rule, allowing withholding. For unchanged, rotated, pair-merged and two-category peer reports, all six fitted rules reproduce the incumbent reports on confirmation questions, giving zero realized score gain. This tests only rules using the original answer label. Identical outputs do not prove population stability or exclude changes to computation, prompts or richer private information.

5.3 An additional vulnerability: withholding answers

A report can also be withheld. After computation is paid for, truthful reporting is uniquely optimal only if the original answer contributes more to expected score than every replacement and withholding (Appendix E). A withholding rule sees the model's answer and whether token use is in the lower, middle or upper third of its calibration distribution. We fit separate rules for two score conventions: *fixed-assignment* scoring counts all assignments, as in (2); *conditional-on-valid* scoring averages only comparisons where both reports are valid. For the first, withhold answer/token categories whose estimated score contribution is negative (Appendix E); for the second, greedily remove categories that most increase calibration score. No rule sees current peer answers or answer keys. Computation has already occurred, so withholding saves no inference cost; a score gain Δq increases utility by $r\Delta q$.

All six conditional-score rules have positive estimated gains (Figure 8), but only Seed 2.0 Mini has an interval entirely above zero: 0.105, interval $[0.015, 0.200]$. Its *coverage*, the fraction of assigned questions receiving a valid report, falls from 99.7% to 4.4%; correct answers per assigned question fall from 77.5% to 3.6%. This exploratory interval is not adjusted for testing multiple models. All fixed-assignment rules retain every answer and gain zero score. The contrast identifies a denominator vulnerability, not a general guarantee against withholding. Sparse retained answers make the other conditional-score intervals wide.

6 Toward governing large autonomous swarms

Deploying multi-task peer prediction in large AI swarms requires payments that guide resource decisions, reports that preserve useful information, and answers that justify the total computation and verification cost. Four priorities follow.

Connect payments to resource control. Rewards should affect agents' compute allowances or their operators' returns. This requires recorded task assignments, metered actions and enforceable payments. Contracts must specify when agents may refuse tasks or leave, and how losses are bounded. The next systems test should give controllers a shared resource budget and let them allocate computation across tasks, rather than only select an API setting for a fixed workload.

Preserve the meaning of reports. Our partition result motivates testing mechanisms that reward information without sustaining lossy reporting conventions. Determinant mutual information (DMI) provides an alternative with dominant-truthfulness guarantees under its model assumptions (Kong, 2020). A complementary direction is to combine peer payments with randomly checked tasks, verifiable outcomes or human audits. Such hybrids should specify which external labels enter payments, measure the cost of obtaining them, and compare against direct spot checking (Xu et al., 2024). All assigned tasks, including withheld and missing reports, must remain visible in the evaluation.

Scale assignments and interactions. Large populations need enough overlapping tasks to estimate peer scores without assigning every task to every agent. Assignment and scoring designs should be tested as agents join, leave or change models. Correlated LLM errors (Kim et al., 2025) and multi-agent coordination and collusion risks (Hammond et al., 2025) motivate tests with shared model families, communicating groups and agents controlling multiple identities. Controllers should be allowed to change computation, reporting, prompts and tools together; these choices determine what information a mechanism actually purchases.

Test the complete incentive loop. We propose comparing excess agreement, DMI, peer payments with random audits, and direct verification at a common inference-and-verification budget. Controllers would choose both reasoning settings and reporting rules. Fix reward levels, audit rates, task splits and stopping rules before evaluation; measure action differences, correct answers per assigned workload, token use, verification cost, withheld reports and gains from coordinated deviations. Our five 15-round runs showed no clear accuracy advantage, with only 3/225 choices differing against each adaptive comparator. Longer runs and varied workloads should test when payment rules produce distinct resource allocations and whether those allocations improve delivered answers.

Conclusion. We take a step toward governing large swarms of autonomous AI agents by characterizing which reasoning settings peer payments can support, measuring their accuracy and cost, and showing that maximum effort can coexist with information-losing reports in a symmetric signal model. The next milestone is a joint computation-and-reporting system that delivers more correct work at a common total resource budget.

AI USE STATEMENT

Generative AI assisted framing, mathematics, experiments, code, analysis, literature search, writing and figures. Exact enumerations, archive reproduction, independent experiment audits and document inspection support these contributions. Author review of proofs, citations and interpretations remains required.

REPRODUCIBILITY STATEMENT

Appendices A–E give current protocols, proofs and estimators; Appendices F–I give historical analyses and simulations. Sanitized artifacts retain task hashes, configurations, failures, bootstrap draws and records for 17,360 static, 7,875 online and 560 pilot calls. Independent audits reproduce scores, decisions, costs and accuracy.

ETHICS STATEMENT

Peer agreement cannot establish correctness or resolve harmful shared beliefs. The experiments expose incentives to withhold information and preserve misleading agreement; they establish neither identity security, arbitrary-coalition resistance nor safe autonomy.

REFERENCES

Noah Burrell and Grant Schoenebeck. Measurement integrity in peer prediction: A peer assessment case study. In *Proceedings of the 24th ACM Conference on Economics and Computation*, 2023. URL https://arxiv.org/abs/2108.05521.

Baiting Chen, Tong Zhu, Jiale Han, Lexin Li, Gang Li, and Xiaowu Dai. Incentivizing truthful language models via peer elicitation games. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-1747. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/4b9eb7f2789a20be981f3fac91429b21-Abstract-Conference.html.

Anirban Dasgupta and Arpita Ghosh. Crowdsourced judgement elicitation with endogenous proficiency. In *Proceedings of the 22nd International Conference on World Wide Web*, pp. 319–330, 2013. URL https://arxiv.org/abs/1303.0799.

Shi Feng, Fang-Yi Yu, and Yiling Chen. Peer prediction for learning agents. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/6e469fbdc43ade121170f61096f4458b-Abstract-Conference.html.

Rafael Frongillo, Ian Kash, and Mary Monroe. Peer prediction with more signals than reports. In *Proceedings of the ACM Conference on Economics and Computation*, 2026. URL https://arxiv.org/abs/2503.16280v2. Extended version: arXiv:2503.16280v2.

Lewis Hammond, Alan Chan, Jesse Clifton, Jason Hoelscher-Obermaier, Akbir Khan, Euan McLean, Chandler Smith, Wolfram Barfuss, Jakob Foerster, Tomáš Gavenčiak, The Anh Han, Edward Hughes, Vojtěch Kovařík, Jan Kulveit, Joel Z. Leibo, Caspar Oesterheld, Christian Schroeder de Witt, Nisarg Shah, Michael Wellman, Paolo Bova, Theodor Cimpeanu, Carson Ezell, Quentin Feuillade-Montixi, Matija Franklin, Esben Kran, Igor Krawczuk, Max Lamparth, Niklas Lauffer, Alexander Meinke, Sumeet Motwani, Anka Reuel, Vincent Conitzer, Michael Dennis, Iason Gabriel, Adam Gleave, Gillian Hadfield, Nika Haghtalab, Atoosa Kasirzadeh, Sébastien Krier, Kate Larson, Joel Lehman, David C. Parkes, Georgios Piliouras, and Iyad Rahwan. Multi-agent risks from advanced AI. Technical Report 1, Cooperative AI Foundation, 2025. URL https://arxiv.org/abs/2502.14143.

Yanchen Jiang, Zhe Feng, and Aranyak Mehta. Incentive-aligned multi-source LLM summaries. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2509.25184.

Elliot Kim, Avi Garg, Kenny Peng, and Nikhil Garg. Correlated errors in large language models. *arXiv preprint arXiv:2506.07962*, 2025. URL https://arxiv.org/abs/2506.07962.

Yuqing Kong. Dominantly truthful multi-task peer prediction with a constant number of tasks. In *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms*, pp. 2398–2411, 2020. doi: 10.1137/1.9781611975994.147. URL https://arxiv.org/abs/1911.00272.

Yuqing Kong. More dominantly truthful multi-task peer prediction with a finite number of tasks. In *13th Innovations in Theoretical Computer Science Conference (ITCS 2022)*, volume 215 of *Leibniz International Proceedings in Informatics*, pp. 95:1–95:20, 2022. doi: 10.4230/LIPIcs.ITCS.2022.95. URL https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.ITCS.2022.95. Extended arXiv version 2103.02214v2 is titled Optimizing Multi-task Peer Prediction.

Yang Liu and Yiling Chen. Learning to incentivize: Eliciting effort via output agreement. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, pp. 3782–3788, 2016. URL https://www.ijcai.org/Proceedings/16/Papers/532.pdf.

Yang Liu and Yiling Chen. Sequential peer prediction: Learning to elicit effort using posted prices. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017. URL https://arxiv.org/abs/1611.09219.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level Google-proof Q&A benchmark. *arXiv preprint arXiv:2311.12022*, 2023. URL https://arxiv.org/abs/2311.12022.

Victor Shnayder, Arpit Agarwal, Rafael Frongillo, and David C. Parkes. Informed truthfulness in multi-task peer prediction. In *Proceedings of the 2016 ACM Conference on Economics and Computation*, pp. 179–196, 2016. doi: 10.1145/2940716.2940790. URL https://arxiv.org/abs/1603.03151.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://arxiv.org/abs/2406.01574.

Leon Witt, Togrul Abbasli, Kentaroh Toyoda, Wojciech Samek, and Lucy Klinger. Knowledge-free correlated agreement for incentivizing federated learning, 2026. URL https://arxiv.org/abs/2605.04747.

Shengwei Xu, Yichi Zhang, Paul Resnick, and Grant Schoenebeck. Spot check equivalence: An interpretable metric for information elicitation mechanisms. In *Proceedings of the ACM Web Conference 2024*, 2024. URL https://arxiv.org/abs/2402.13567.

Yichi Zhang and Grant Schoenebeck. High-effort crowds: Limited liability via tournaments. In *Proceedings of the ACM Web Conference 2023*, pp. 3467–3477, 2023. doi: 10.1145/3543507.3583334. URL https://yichiz97.github.io/Papers/High_Effort_Crowds.pdf.

A ONLINE ACCURACY–COST COMPARISON: PROTOCOL AND FULL RESULTS

The protocol, fixed before API collection, uses the first 525 questions of the separate online pool, partitioned into five disjoint 105-question workloads. Each workload has 15 rounds of seven questions and three models: GPT-5 nano, Qwen3.5 Flash and DeepSeek V4.1 Flash. Methods answer the same questions through separate API requests: excess agreement, raw agreement, flat payment, fixed minimum and fixed maximum. Five runs, 15 rounds, seven questions, three models and five methods give $5 \times 15 \times 7 \times 3 \times 5 = 7{,}875$ calls. The three-model scope was chosen to meet the spending limit before observing online results. These calls are separate from the six-model static study.

Online decisions and information. Each model's controller selects one setting per seven-question round. Excess agreement uses the fixed-assignment estimator in (2); raw agreement uses its same-task term alone. Flat payment contributes zero action-dependent score, so its learner observes cost differences only. All three adaptive methods use reward coefficient 32 and EXP3 exploration $\gamma = 0.2$, with uniform initial weights. For log weights w_a and K actions, the sampling distribution is

$$P(a) = (1-\gamma)\frac{\exp(w_a)}{\sum_b \exp(w_b)} + \frac{\gamma}{K}.$$

For run $j = 0, \ldots, 4$, seed $42 + j$ supplies uniform random numbers on $[0, 1]$ that select actions from these probabilities. The three adaptive methods share the draw for each model and round. Fixed policies always select the first or last native setting. Each new round starts only after the previous round's responses finish; request order within a round is shuffled independently of answers. No current or future evaluator labels enter these choices.

For the selected setting, $U = 32\widehat{q} - \widehat{c}^{\,\mathrm{upper}}$ uses the round's peer score and mean token cost. For each question, add its prompt's UTF-8 byte length, 1,024 framing tokens and the largest output cap in that model's menu. Averaging these sums across the seven questions and dividing by 1,000 gives the cost bound $C_{\max}$. The normalized loss is $\ell = (32 - U)/(64 + C_{\max}) \in [0, 1]$. Update only the selected log weight by $-\gamma\ell/(KP(a))$, then subtract the maximum log weight for numerical stability. EXP3's exponential-weight update is entropic mirror descent on an importance-weighted bandit loss, with separate exploration. PEG applies mirror descent to response and judgment policies (Chen et al., 2025); here only reasoning-setting probabilities change. Appendix D analyzes exact-gradient reporting dynamics.

Inference allowance and experimental overhead. The primary allowance is \$1 per 105-question workload; secondary allowances are \$0.25, \$0.50, \$2 and \$4. Evaluate prefixes of actual collected trajectories in predeclared question/model order. Before a call, add its known maximum cost reservation to the charges already accounted for. Stop if this would exceed the allowance, without using that call's realized answer or cost. After an accepted call, use 1.1 times the larger of the recorded provider cost and the conservative tariff estimate; retain the entire reservation when usage is unknown. These accounting dollars need not equal provider invoices. Charges for calls on a partially completed question still count. A delivered task requires all three calls; its prediction is plurality over valid reports, with smallest-label tie breaking and abstention if all reports are invalid. Every incomplete or unserved task counts incorrect on the full workload.

The same allowance does not force identical expenditure: reservations can leave unused funds, and some policies finish all tasks earlier. Full trajectories are collected to evaluate larger allowances; calls beyond a smaller allowance's stopping point are additional experimental costs and remain in the study ledger. The allowance rule was specified before collection and applied afterward to each recorded call sequence. We did not train a separate controller for each allowance. Payment transfers are not inference costs and are not matched by fixing the nominal reward coefficient.

Replication and scope. The five task-block runs, each with its own seed, are the replication units. We report every within-run excess-minus-comparator accuracy difference at the primary allowance, alongside its mean and observed range. The 15 rounds and individual calls are not additional independent runs. Fifteen rounds cannot establish convergence or rule out later benefits; one reward scale, benchmark and three-model population cannot establish general mechanism superiority. A DMI online arm is absent because the signed ten-label determinant-product estimator requires at least 20 tasks, exceeding a seven-task round. Online controllers send model answers unchanged. Only

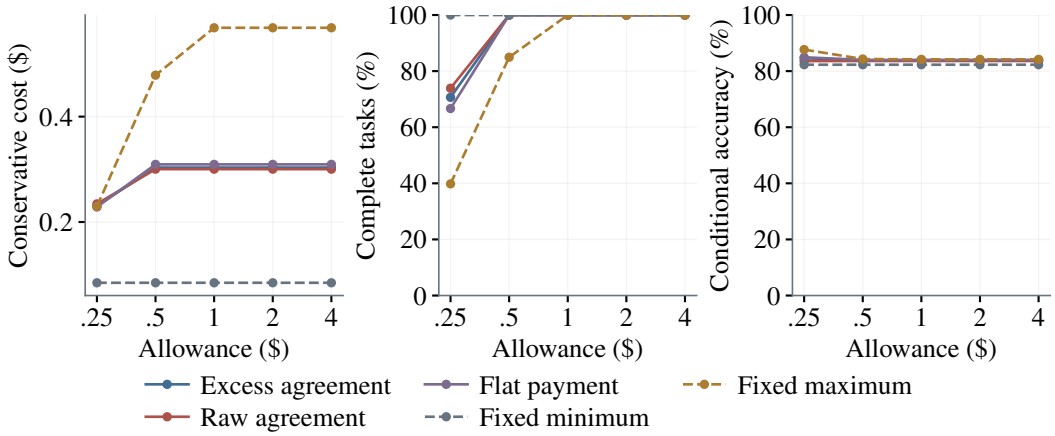


Figure 5: Online inference cost, completed-task fraction and accuracy conditional on complete tasks. Curves are five-run means. At $1 every method completes the workload and spends less than its allowance. At smaller allowances, conditional accuracy excludes unserved questions from its denominator; Figure 2 retains them as incorrect.

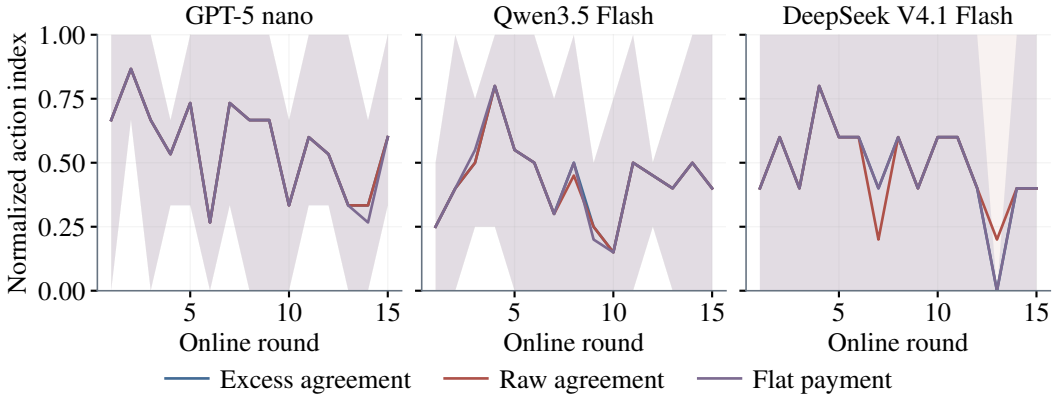


Figure 6: Adaptive setting choices: each menu index is scaled from 0 (lowest) to 1 (highest). Lines are five-run means; bands span the observed lowest/highest run, not confidence intervals. Shared random draws and similar probability updates produce extensive overlap. Methods still make separate API calls when settings coincide. Fixed minimum and maximum would stay at zero and one.

the separate stored-answer experiments alter reporting, so neither design tests joint adaptation of reporting and effort.

Completed collection and independent reconstruction. The run completed on September 26, 2026 with one record for each of 7,875 planned calls and all 25 method/run sequences. There are 7,706 valid answers, 158 invalid answers and 11 unknown-completion failures (ten DeepSeek and one Qwen); no paid call was retried. Every assignment stays in the denominator, and unknown charges retain their full reservations. The complete online stage adds $7.832 in conservatively accounted API costs. Including every earlier pilot and static call, the total is $33.138 under the unchanged $50 study cap; these study costs are separate from the per-workload comparison.

Collection revision `f2a09b02060f` and verification revision `fcbe57ed7220` are recorded in the sanitized provenance. A separate program verifies the fixed protocol and API controls without making model calls. Before accessing answer keys, it reconstructs all 1,125 setting decisions (5 runs $\times$15 rounds $\times$3 models $\times$5 methods), scores, costs and EXP3 updates. It checks round timing and reproduces all 150 allowance evaluations (25 method/run sequences $\times$6 conditions, including full

Table 2: Within-run accuracy differences at the $1 allowance: excess agreement minus comparator, in percentage points. Seeds 42–46 use disjoint question sets. Each column compares methods on the same questions. These five observations describe variation across runs; they do not establish superiority.

Comparator	Run 1	Run 2	Run 3	Run 4	Run 5	Mean
Raw agreement	+0.00	+0.00	+0.95	-0.95	+0.95	+0.19
Flat payment	-1.90	+0.95	+0.00	-1.90	+1.90	-0.19
Fixed minimum	+0.95	+0.95	+1.90	+3.81	+0.00	+1.52
Fixed maximum	-1.90	-0.95	-0.95	+0.95	+0.95	-0.38

trajectories) and every primary accuracy difference. All checks pass. It also records the selected-action differences: excess versus raw agreement has run counts $(0, 2, 1, 0, 0)$ out of 45 decisions per run; excess versus flat payment has $(0, 0, 0, 2, 1)$. Each comparison differs on only 3/225 decisions. This is observed short-horizon similarity, not a convergence guarantee or an equivalence test.

Table 3: All specified allowances and complete call sequences ("Full"). Accuracy counts all 105 questions; "Complete" is the percentage with all three calls attempted within the allowance; conditional accuracy uses only those questions. Entries are five-run means, with observed accuracy ranges rather than confidence intervals. Cost includes reservations for unknown charges.

Method	Allowance ($)	Accuracy (%)	Run range (%)	Complete (%)	Conditional accuracy (%)	Cost ($)
Excess agreement	0.25	59.81	[46.67, 70.48]	70.7	84.6	0.231
Excess agreement	0.5	83.81	[80.95, 86.67]	100.0	83.8	0.303
Excess agreement	1	83.81	[80.95, 86.67]	100.0	83.8	0.303
Excess agreement	2	83.81	[80.95, 86.67]	100.0	83.8	0.303
Excess agreement	4	83.81	[80.95, 86.67]	100.0	83.8	0.303
Excess agreement	Full	83.81	[80.95, 86.67]	100.0	83.8	0.303
Raw agreement	0.25	61.90	[53.33, 85.71]	73.9	83.6	0.234
Raw agreement	0.5	83.62	[81.90, 86.67]	100.0	83.6	0.300
Raw agreement	1	83.62	[81.90, 86.67]	100.0	83.6	0.300
Raw agreement	2	83.62	[81.90, 86.67]	100.0	83.6	0.300
Raw agreement	4	83.62	[81.90, 86.67]	100.0	83.6	0.300
Raw agreement	Full	83.62	[81.90, 86.67]	100.0	83.6	0.300
Flat payment	0.25	56.57	[35.24, 71.43]	66.7	84.9	0.229
Flat payment	0.5	84.00	[81.90, 85.71]	100.0	84.0	0.309
Flat payment	1	84.00	[81.90, 85.71]	100.0	84.0	0.309
Flat payment	2	84.00	[81.90, 85.71]	100.0	84.0	0.309
Flat payment	4	84.00	[81.90, 85.71]	100.0	84.0	0.309
Flat payment	Full	84.00	[81.90, 85.71]	100.0	84.0	0.309
Fixed minimum	0.25	82.29	[77.14, 85.71]	100.0	82.3	0.085
Fixed minimum	0.5	82.29	[77.14, 85.71]	100.0	82.3	0.085
Fixed minimum	1	82.29	[77.14, 85.71]	100.0	82.3	0.085
Fixed minimum	2	82.29	[77.14, 85.71]	100.0	82.3	0.085
Fixed minimum	4	82.29	[77.14, 85.71]	100.0	82.3	0.085
Fixed minimum	Full	82.29	[77.14, 85.71]	100.0	82.3	0.085
Fixed maximum	0.25	34.86	[24.76, 45.71]	39.8	87.7	0.230
Fixed maximum	0.5	71.62	[58.10, 82.86]	85.0	84.3	0.479
Fixed maximum	1	84.19	[80.00, 87.62]	100.0	84.2	0.569
Fixed maximum	2	84.19	[80.00, 87.62]	100.0	84.2	0.569
Fixed maximum	4	84.19	[80.00, 87.62]	100.0	84.2	0.569
Fixed maximum	Full	84.19	[80.00, 87.62]	100.0	84.2	0.569

B STATIC API COLLECTION, HELD-OUT EVALUATION AND REPORTING INTERVENTIONS

Task pools and decisions fixed before testing. Collection on September 25, 2026 uses the cached MMLU-Pro test revision `b189ec765aa7`, restricted to ten-option questions. We exclude all 294 historical MMLU-Pro IDs and matching canonical question hashes; before hashing, text is normalized by collapsing whitespace and ignoring letter case. Question text is deduplicated even if options differ. Seed 42 shuffles within subject and samples round-robin across 14 subjects. Each question uses the same deterministic answer-option order for every model, setting and repeated call. The engineering pilot (14 questions), calibration (252), confirmation (182) and reserved online pools are disjoint. Each calibration subject supplies 18 questions; each confirmation subject supplies 13. Reported means therefore weight the 14 subjects equally, rather than in proportion to their frequency in the benchmark.

Before any confirmation call, we reduced the planned confirmation pool from 364 to its first 182 balanced questions to meet the corrected spending limit. All 252 calibration questions were retained. The removed questions were not reused in the online study. This decision used no answer labels, estimated utilities or confirmation results. The label-free calibration policy was saved at 06:37 UTC, before the first confirmation request. The selections were fixed before this held-out test, but were not externally preregistered. Collection revision is `3c6dabaa1e89`; the independent offline audit uses `824517f1bc5f`.

Native configurations and parsing. The requested models and setting menus follow. Each output cap limits the request's total output-token allowance, separately from any thinking budget.

- `gpt-5-nano-2025-08-07`: minimal, low, medium and high reasoning effort; output cap 16,384 tokens.
- `claude-haiku-4-5-20251001`: thinking off or budgets 1,024, 1,536 and 2,048 tokens; output cap 6,144.
- `qwen/qwen3.5-flash-02-23`, Alibaba provider: reasoning off or budgets 512, 1,024, 2,048 and 4,096; output cap 8,192.
- `openai/gpt-oss-20b`, Darkbloom FP8 provider: low, medium and high reasoning effort; output cap 16,384.
- `deepseek-flash`, routing to DeepSeek V4.1 Flash during this collection: thinking off/on; output cap 16,384. This version differs from the historical V4 Flash.
- `bytedance-seed/seed-2.0-mini`, Seed FP8 provider: reasoning off/on; output cap 16,384.

OpenRouter requests are restricted to the named provider, with fallback routing disabled and support for the requested parameters required. Returned model identifiers are recorded. OpenAI uses provider-default temperature. Claude uses 0.7 with thinking off and the required provider default with thinking on. OpenRouter and DeepSeek request 0.7, subject to provider semantics. These complete configurations are the actions; switching modes does not isolate reasoning depth from every other API behavior. All settings remain in the analysis, including cases where a higher setting uses fewer tokens or has lower accuracy.

The prompt asks the model to end with an answer-option letter. A parser accepts an unambiguous final label, repeated identical final labels, or a bare letter, stripping Markdown bold delimiters; conflicting labels and inferred guesses from prose remain invalid. The broader engineering pilot triggered these parser changes before calibration. Pilot records preserve their original parser outcomes and are not pooled with confirmation. Each question/model/setting/repetition combination receives one paid attempt; failed calls are retained, never selectively retried. Across the pilot and static collection the 17,920 records contain 17,559 valid answers, 335 invalid answers, 15 HTTP errors and 11 failures with unknown completion status. These failures and invalid answers count as incorrect rather than being dropped.

Generations, scores and costs. A generation is one API attempt for a given question and setting; failed attempts remain in the data. We make two separate requests for every question/setting; neither

Table 4: Six-model test on 182 held-out questions, with two API calls per setting/question. Accuracy counts missing answers as incorrect. Token ratio is maximum-setting cost divided by minimum-setting cost, using upper costs for missing meters. Across seven rewards, "Point match" counts selected settings that still maximize estimated utility; "Band support" counts those whose lower utility-advantage bounds exceed zero against every alternative.

	Accuracy (%)		Token	Point	Band
Model	Min.	Max.	ratio	match	support
GPT-5 nano	43.7	71.7	16.26	5/7	2/7
Claude Haiku 4.5	78.8	78.3	1.89	6/7	3/7
Qwen3.5 Flash	80.5	83.0	2.48	5/7	2/7
GPT-oss-20b	58.5	66.8	2.06	7/7	2/7
DeepSeek V4.1 Flash	82.1	81.0	5.93	7/7	6/7
Seed 2.0 Mini	74.2	77.5	1.99	7/7	3/7

output is a copy of an archived answer. The first output is scored against peers' first outputs at maximum settings, and the second against their second outputs; scores, costs and accuracies are then averaged. Plurality combines the six agents within one generation only, with deterministic smallest-label tie breaking and abstention if no valid vote exists. It never votes over the two generations. Table 5 reports all settings and their separate generation accuracies. The second call measures variation between repeated outputs on the same question; it does not add an independent question or provider deployment.

Costs are total input plus output kilotokens per assignment. For missing token usage, we assign a lower bound of zero and an upper bound equal to prompt UTF-8 bytes plus 1,024 framing tokens plus the output cap, divided by 1,000. This assumes the requested output cap and input allowance hold. Calibration uses this upper cost, hence lower utility, for the affected call. Confirmation propagates both cost endpoints into the utility differences. Table 4's ratios are ratios of these upper-cost summaries, not bounds on a population cost ratio. We do not report a complete mean provider charge when any contributing charge is unknown. The study ledger retains maximum reserved charges for uncertain calls and includes every pilot and failure. Its price-based accounting estimates have not been reconciled against provider invoices.

Confirmation bands. Each of 2,000 seed-42 bootstrap draws resamples 13 questions with replacement within each of the 14 confirmation subjects. Each sampled question includes all models, settings and both repeated calls. For each reward and ordered action pair, form the lower and upper utility-gap endpoints and their bootstrap draws. Subtract each original endpoint estimate from its resampled estimates and divide by their bootstrap standard deviation. For each draw, take the maximum absolute standardized deviation over every contrast and endpoint within the model. Its empirical $1 - 0.05/6$ quantile supplies a common critical value. Expand the lower endpoint downward and the upper endpoint upward by the corresponding standard errors times this critical value. Zero-variance endpoints receive zero expansion; that convention is not evidence of zero population variance. Bands are approximate, including error from estimating an extreme quantile with 2,000 draws, and can fail in small or degenerate samples.

A calibration-selected setting is supported only if every lower utility gap against another action is strictly positive. Point matching instead compares conservative upper-cost utilities with deterministic tie breaking. Point shortfall is the best alternative's conservative utility minus that of the selected setting, truncated below at zero. The reported shortfall upper bound maximizes the negatives of its lower utility-gap bounds and zero. These measure different things: zero estimated shortfall does not mean that uncertainty excludes better alternatives. Table 6 retains every reward condition, including cases where the selected setting has positive shortfall or lacks support.

Transforming stored answer labels. Number the answer options A–J as 0–9. The reporting rules are constant $g(z) = 0$, cyclic $g(z) = (z + 1) \bmod 10$, pairs $g(z) = 2\lfloor z/2 \rfloor$, and two cells $g(z) = 0$ for $z < 5$ and 5 otherwise. The last representative differs from the historical two-cell map; both are fixed before evaluation. Coalitions contain the first 1, 3 or 6 models in the order nano, Claude, Qwen, GPT-oss, DeepSeek, Seed, without optimizing membership for damage. Unchanged reports

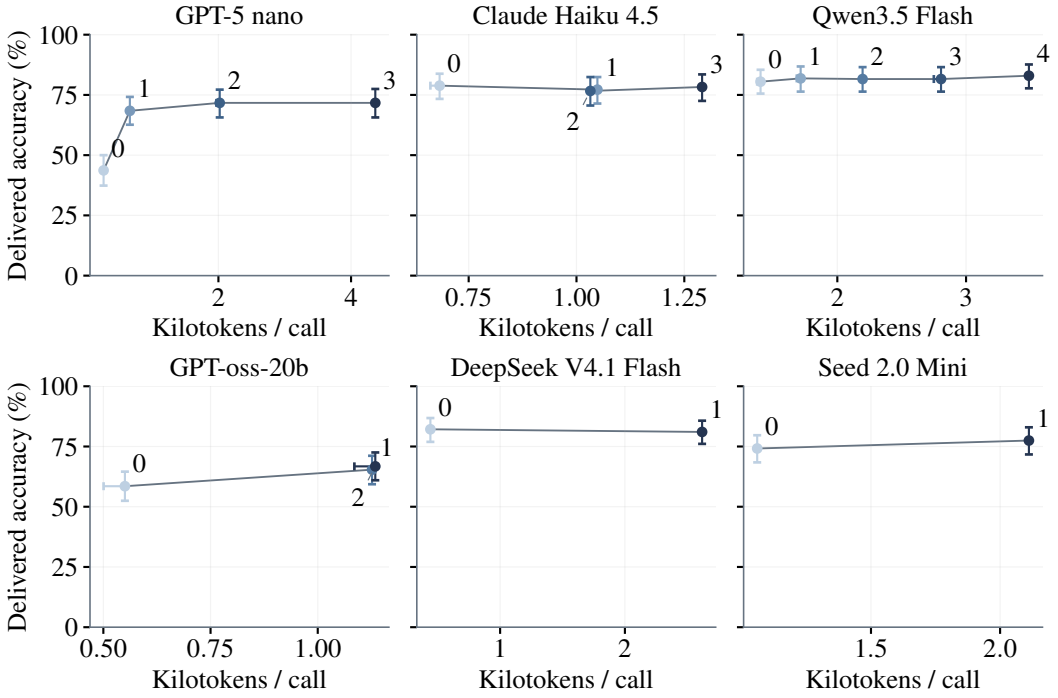


Figure 7: Accuracy and token cost for every API setting on 182 held-out questions. Numbers identify settings in each provider's ordered menu. Vertical bars are unadjusted 95% intervals obtained by resampling whole questions; horizontal bars bound costs of calls with missing meters. These cost bounds are not sampling intervals. Lines connect settings; larger menu indices need not improve accuracy or increase measured tokens.

are the baseline for every comparison. Original missing reports stay missing. Score invariance under a common permutation is exact even with these missingness conventions; plurality accuracy can depend on its tie rule.

To test one agent's deviation, we pool both calibration repetitions to estimate the score contribution of each possible answer-to-report change. Withholding has coefficient zero; peers retain the specified reporting rule. The fitted reporting rules are applied to the original confirmation answers. Choosing the highest coefficient for each answer maximizes calibration score among all rules using only that answer, including randomized rules. This optimization does not change inference, prompts or rules using richer information. Against unchanged, cyclic, pair-merged or two-cell peer reports, the fitted rules reproduce the incumbent confirmation reports. Under constant peers, fitted unilateral score gains range from -0.00192 to $+0.00272$ and all six unadjusted intervals include zero. We therefore do not infer a strict empirical equilibrium from these tests.

For withholding, each private state combines an answer and a calibration token-count group (lower, middle or upper third) into one of 30 states. Both calibration repetitions are used to fit the token-group thresholds and reporting rules; the fixed-score rule retains zero-coefficient states, while the conditional-score greedy rule stops when no deletion improves score by more than 10^{-12}. Invalid or unavailable states stay unchanged. During confirmation, a rule sees only its own model's answer and token count, not current peer answers. Each score convention trains its own rule; the comparison changes the rule as well as the score. Changing an already computed answer leaves inference cost unchanged; the local cost of applying the rule is not metered.

All reporting accuracy and score-gain intervals are unadjusted percentile intervals from 2,000 within-subject resamples of complete question records, keeping fitted policies fixed. They describe variation across questions in the collected outputs, without accounting for refitting rules, multiple testing or later changes to the providers. They are distinct from the simultaneous utility bands. Table 7 reports

Table 5: All 20 API settings tested on 182 held-out questions. Score averages excess agreement over two calls per question against maximum-setting peers. Cost bounds reflect missing input/output token meters, in thousands of tokens per call. "Valid" is the percentage of calls returning a usable label. "Acc. G1/G2" give percentage accuracy separately for the first and second calls; missing answers count as wrong. Equal rounded cost endpoints may hide a small interval.

Model	Action	Score	Cost interval	Valid	Acc. G1	Acc. G2
GPT-5 nano	minimal	0.371	0.269–0.269	98.4	41.2	46.2
GPT-5 nano	low	0.634	0.665–0.665	98.9	67.0	69.8
GPT-5 nano	medium	0.667	1.970–2.019	98.6	72.0	71.4
GPT-5 nano	high	0.673	4.366–4.366	95.6	73.1	70.3
Claude Haiku 4.5	off	0.678	0.661–0.683	98.9	77.5	80.2
Claude Haiku 4.5	think1024	0.671	1.049–1.049	99.5	76.9	77.5
Claude Haiku 4.5	think1536	0.672	1.032–1.032	99.5	76.4	76.9
Claude Haiku 4.5	think2048	0.692	1.291–1.291	99.5	76.4	80.2
Qwen3.5 Flash	off	0.695	1.378–1.404	97.8	78.6	82.4
Qwen3.5 Flash	think512	0.698	1.686–1.715	97.8	79.7	84.1
Qwen3.5 Flash	think1024	0.693	2.196–2.196	97.5	81.9	81.3
Qwen3.5 Flash	think2048	0.690	2.750–2.806	97.8	83.0	80.2
Qwen3.5 Flash	think4096	0.698	3.488–3.488	98.1	81.3	84.6
GPT-oss-20b	low	0.553	0.501–0.550	95.3	56.6	60.4
GPT-oss-20b	medium	0.614	1.127–1.127	94.8	64.8	65.9
GPT-oss-20b	high	0.621	1.085–1.134	96.4	69.2	64.3
DeepSeek V4.1 Flash	off	0.703	0.441–0.441	100.0	81.9	82.4
DeepSeek V4.1 Flash	on	0.677	2.619–2.619	95.3	81.9	80.2
Seed 2.0 Mini	off	0.667	1.059–1.059	99.5	74.2	74.2
Seed 2.0 Mini	on	0.682	2.112–2.112	99.7	76.9	78.0

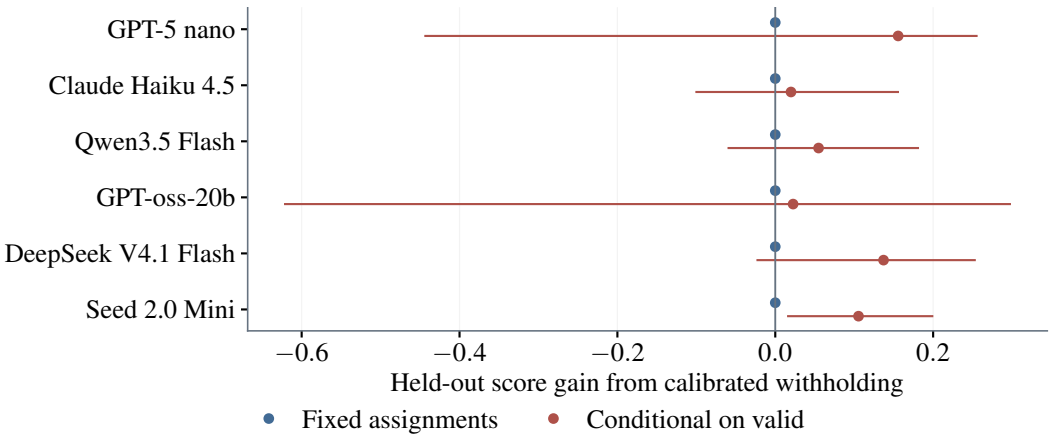


Figure 8: Change in peer score when a rule withholds selected model answers, tested on held-out questions. Rules were learned separately for each score convention on calibration questions. Bars are exploratory unadjusted 95% intervals from 2,000 resamples of whole questions, keeping the rules fixed. Fixed-assignment rules keep every available answer and gain zero score.

every coalition and conditional-withholding condition; the six fixed-score rules all retain every state and produce exactly zero gain.

Static comparison of accuracy and token use. On the static study's 182 confirmation questions, plurality at all-minimum settings is 79.95% accurate at a total of 4.406 kilotokens across the six agents per question; all-maximum gives 80.77% at 15.009 kilotokens. The estimated gain is $+0.82$ percentage points, with an exploratory paired bootstrap interval $[-1.65, +3.30]$, added after the initial readout. It permits a loss or gain and does not establish a cost-effective benefit. None of the seven reward-selected combinations has an accuracy-gain interval entirely above zero versus minimum.

Table 6: All 42 model/reward combinations. Utility is payment minus token cost. "Point shortfall" is the estimated utility lost by keeping the calibration-selected setting instead of the best confirmation setting, clipped at zero. "Upper bound" accounts for simultaneous sampling uncertainty and missing token meters. "Supported" means every lower utility-advantage bound is positive. Values are kilotoken-equivalent units, not dollars.

Model	Reward	Selected action	Point shortfall	Upper bound	Supported
GPT-5 nano	0	minimal	0.000	0.000	Yes
GPT-5 nano	2	low	0.000	0.060	No
GPT-5 nano	8	low	0.000	0.000	Yes
GPT-5 nano	32	low	0.000	1.388	No
GPT-5 nano	128	medium	0.000	3.904	No
GPT-5 nano	512	medium	0.885	22.297	No
GPT-5 nano	2048	medium	10.579	95.919	No
Claude Haiku 4.5	0	off	0.000	0.000	Yes
Claude Haiku 4.5	2	off	0.000	0.000	Yes
Claude Haiku 4.5	8	off	0.000	0.000	Yes
Claude Haiku 4.5	32	think1024	0.571	1.745	No
Claude Haiku 4.5	128	think2048	0.000	3.318	No
Claude Haiku 4.5	512	think2048	0.000	11.406	No
Claude Haiku 4.5	2048	think2048	0.000	43.762	No
Qwen3.5 Flash	0	off	0.000	0.000	Yes
Qwen3.5 Flash	2	off	0.000	0.000	Yes
Qwen3.5 Flash	8	off	0.000	0.058	No
Qwen3.5 Flash	32	off	0.000	0.638	No
Qwen3.5 Flash	128	off	0.000	4.192	No
Qwen3.5 Flash	512	off	0.871	18.872	No
Qwen3.5 Flash	2048	think1024	9.925	88.623	No
GPT-oss-20b	0	low	0.000	0.000	Yes
GPT-oss-20b	2	low	0.000	0.000	Yes
GPT-oss-20b	8	low	0.000	0.630	No
GPT-oss-20b	32	high	0.000	1.560	No
GPT-oss-20b	128	high	0.000	6.050	No
GPT-oss-20b	512	high	0.000	24.058	No
GPT-oss-20b	2048	high	0.000	96.103	No
DeepSeek V4.1 Flash	0	off	0.000	0.000	Yes
DeepSeek V4.1 Flash	2	off	0.000	0.000	Yes
DeepSeek V4.1 Flash	8	off	0.000	0.000	Yes
DeepSeek V4.1 Flash	32	off	0.000	0.000	Yes
DeepSeek V4.1 Flash	128	off	0.000	0.000	Yes
DeepSeek V4.1 Flash	512	off	0.000	0.000	Yes
DeepSeek V4.1 Flash	2048	off	0.000	0.427	No
Seed 2.0 Mini	0	off	0.000	0.000	Yes
Seed 2.0 Mini	2	off	0.000	0.000	Yes
Seed 2.0 Mini	8	off	0.000	0.000	Yes
Seed 2.0 Mini	32	off	0.000	0.682	No
Seed 2.0 Mini	128	on	0.000	4.181	No
Seed 2.0 Mini	512	on	0.000	13.596	No
Seed 2.0 Mini	2048	on	0.000	51.283	No

Table 7: Transformations of stored model answers. Top: number of agents using each reporting rule, swarm accuracy and mean peer score. Bottom: withholding rules for scores conditioned on valid reports; score gain compares with retaining every available answer. Accuracy and coverage are percentages of assigned questions; withheld answers count as wrong. Intervals are exploratory unadjusted 95% intervals. Fixed-assignment rules withhold nothing.

Map	Agents	Accuracy	95% interval	Mean score
Identity	0	80.77	[75.27, 85.71]	0.673934
Constant	1	81.04	[75.55, 85.99]	0.449935
Constant	3	7.69	[4.12, 11.26]	0.126731
Constant	6	7.69	[4.40, 11.54]	0.001983
Cyclic	1	80.77	[75.27, 85.71]	0.420015
Cyclic	3	61.54	[55.49, 67.86]	0.216470
Cyclic	6	1.37	[0.00, 3.02]	0.673934
Pair merge	1	81.32	[75.82, 86.26]	0.554040
Pair merge	3	45.33	[38.74, 51.65]	0.436316
Pair merge	6	43.13	[36.54, 49.45]	0.588307
Two cells	1	81.32	[76.10, 86.26]	0.466463
Two cells	3	14.84	[10.44, 19.23]	0.232802
Two cells	6	11.54	[7.42, 15.66]	0.364818

Model	Score gain	95% interval	Coverage	Accuracy
GPT-5 nano	0.156	[-0.445, 0.256]	1.6	1.4
Claude Haiku 4.5	0.020	[-0.101, 0.157]	4.7	3.6
Qwen3.5 Flash	0.055	[-0.060, 0.182]	3.8	3.8
GPT-oss-20b	0.023	[-0.622, 0.299]	1.9	1.4
DeepSeek V4.1 Flash	0.137	[-0.024, 0.254]	2.5	2.2
Seed 2.0 Mini	0.105	[0.015, 0.200]	4.4	3.6

Table 8: Accuracy differences on the static response bank, relative to all-minimum settings, in percentage points. "Calibrated reward" uses the settings selected on calibration questions at that reward. Intervals resample questions with all their model outputs and are unadjusted 95% intervals. This exploratory comparison was added after the initial readout and does not match inference budgets.

Policy	Difference	95% interval
Fixed maximum	+0.82	[-1.65, +3.30]
Calibrated reward 0	+0.00	[+0.00, +0.00]
Calibrated reward 2	-0.82	[-2.20, +0.55]
Calibrated reward 8	-0.82	[-2.20, +0.55]
Calibrated reward 32	+0.00	[-2.20, +2.20]
Calibrated reward 128	+0.82	[-1.10, +2.75]
Calibrated reward 512	+0.82	[-1.10, +2.75]
Calibrated reward 2048	+0.55	[-1.65, +2.75]

These comparisons reuse stored outputs; they do not measure adaptive learning or equal-budget deployment.

Static aggregate quality and independent checks. Fixed-minimum, fixed-maximum and reward-selected policies each choose a setting per model. For each question and repetition, we combine the six stored outputs at those settings. Policy fitting assumes maximum-setting peers, whereas plurality combines the selected actions; these static aggregate comparisons do not assert a joint equilibrium. Table 8 gives the exploratory paired accuracy contrasts against fixed minimum. They were added after the initial static readout and reuse the original question-level bootstrap draws, without correction for multiple comparisons. The zero interval for reward zero arises because it selects the same responses as fixed minimum, not because its population accuracy is known.

A separate audit program verifies that each planned call has one record, the question pools do not overlap, the selected settings were saved before confirmation, and those choices can be reproduced without answer keys. It verifies the optimized score by explicitly enumerating ordered question pairs,

checks subject counts and costs in every stored bootstrap draw, checks scores in selected draws, and reproduces individual and swarm accuracy. All checks pass and consume no additional API calls. Sanitized manifests and tabular outputs accompany the source; original local records preserve raw provenance.

C SIGNAL MODEL AND EXPLICIT DEVIATION CALCULATION

Let $C_s(a, y) = P(Z = a \mid Y = y, s)$ be the probability of observing signal a when the true label is y and effort is s. Uniform latent labels imply uniform signal marginals. Conditional independence gives

$$P(Z_i = a, Z_j = b) = \frac{1}{L} \sum_y C_{s_i}(a, y) C_{s_j}(b, y).$$

Subtracting $1/L^2$ and writing $p_i = p(s_i) = 1/L + u_i$ yields diagonal covariance $u_i u_j/(L-1)$ and off-diagonal covariance $-u_i u_j/(L-1)^2$, proving (5). Independent randomized reporting gives the reported covariance $F_i^\top \Delta_{ij} F_j$. Its trace equals same-task agreement minus independent-task agreement. A peer average preserves this calculation.

Truthful maximum effort.

Proof. At each positive effort, each signal's highest-scoring report against truthful peers is the same label, so identity uniquely maximizes the score. At effort zero it remains a maximizer. Every joint deviation is therefore bounded above by $V(s) = r\kappa u(1)u(s) - c(s)$, a strictly concave function. Its endpoint derivative is nonnegative, so its unique maximum is $s = 1$. At that effort any nonidentity reporting policy strictly reduces score. □

Strict partition equilibria.

Proof. For a signal in a cell of size m, the coefficient in HG for reporting that cell's representative label is $(L-m)/(L-1)^2 > 0$. The coefficient for another used cell of size m' is $-m'/(L-1)^2 < 0$; unused labels have coefficient zero. Thus G is the unique row-wise maximizer at every informative effort. Summing the own-cell coefficients gives (6). As in Proposition 2, joint deviations are bounded above by the strictly concave function $rK_G u(1)u(s) - c(s)$, whose endpoint derivative is positive. The effort maximum is $s = 1$, where G is the unique optimal reporting matrix. □

Threshold inflation and payoff comparison. For a nonconstant partition, $L \le \sum_j m_j^2 \le (L-1)^2 + 1$. The minimum uses singleton cells; merging cells increases the sum of squares, and among two nonempty cells it is maximal at sizes $1, L-1$. Substituting these extremes into κ/K_G proves (7). The truthful and partition profiles incur the same full-effort cost, so their per-agent payoff difference is exactly $ru(1)^2(\kappa - K_G)$. A genuine coarsening has $\sum_j m_j^2 > L$, making that difference strictly positive. Changing together to truthful reports benefits every agent; changing alone from either equilibrium reduces the deviator's utility.

The fixed-effort reporting argument also holds for any covariance matrix with zero row sums and strictly negative off-diagonal entries. For a signal z in a proper cell C, the own-cell coefficient is $\sum_{w \in C} \Delta(z, w) = -\sum_{w \notin C} \Delta(z, w) > 0$; another cell contains only negative off-diagonal terms, and an unused report has coefficient zero. The scalar symmetric channel model supplies the additional explicit effort threshold. No heterogeneous effort equilibrium follows from the reporting calculation alone.

Explicit three-label example. Set $p(s) = 1/3 + 0.3s$, $c(s) = 0.01s + 0.5s^2$, and $r = 12$. The map $g = (1, 1, 3)$ reports 1 for signals 1 or 2 and reports 3 for signal 3; its reporting matrix G has $K_G = 1$. Against maximum-effort peers using g, any report matrix F satisfies

$$U(F, s) = 1.08s \operatorname{tr}(F^\top HG) - 0.01s - 0.5s^2 \le 1.07s - 0.5s^2. \tag{8}$$

The unique maximizer is $(G, 1)$, with utility 0.57. The best distinct deterministic map, after optimizing its continuous effort, earns 0.32; linearity in F extends the proof to randomized reporting. Truthful

maximum effort is also strict, since its threshold is approximately $7.48 < 12$. The coarsening is informative and is not a shared permutation. Excluding shared label permutations therefore does not remove it.

For this three-label construction, the exact matrices are

$$H = \begin{pmatrix} 1/2 & -1/4 & -1/4 \\ -1/4 & 1/2 & -1/4 \\ -1/4 & -1/4 & 1/2 \end{pmatrix}, \qquad HG = \begin{pmatrix} 1/4 & 0 & -1/4 \\ 1/4 & 0 & -1/4 \\ -1/2 & 0 & 1/2 \end{pmatrix}.$$

The unique maximum in each row is the column prescribed by $g = (1, 1, 3)$. For a deterministic deviation f, set $K_f = \sum_a (HG)_{a,f(a)}$. Its optimal effort is

$$s_f^* = \min\{1, \max\{0, 1.08K_f - 0.01\}\}, \qquad U_f^* = (1.08K_f - 0.01)s_f^* - \tfrac{1}{2}(s_f^*)^2.$$

Enumerating the 27 maps with exact rational arithmetic gives a unique maximum $U_g^* = 57/100$; the largest value of any other deterministic map is $8/25$. The row-maximization proof, rather than a finite grid, handles every stochastic reporting matrix. At $s > 0$ the unique row maxima make G the unique maximizing matrix. At $s = 0$ all matrices give utility zero, strictly below $57/100$. A lottery over efforts and reporting matrices is an average of pure realized utilities and cannot improve on the unique maximizer.

The threshold for truthful reporting at maximum effort is

$$\frac{c'(1)}{\kappa u(1) p'(1)} = \frac{1.01}{1.5(0.3)^2} \approx 7.48148,$$

and the corresponding threshold for merged reports is $1.01/(0.3)^2 \approx 11.22222$. Reward 12 exceeds both. The partitions with sizes $(1, 1, 1)$ and $(2, 1)$ have coefficients 1.5 and 1, respectively. Truthful reporting can deliver greater joint payoff and still coexist with a strictly stable coarsened convention. This statement does not assert historical priority for the existence of undesirable equilibria in peer prediction.

Information-acquisition scope. Effort selects a channel from the prescribed symmetric family before reporting. Agents cannot directly purchase only the coarsened signal at a lower cost. Adding such acquisition strategies could alter both the maximum-effort conclusion and partition stability; the theorem makes no claim about that enlarged action space.

Observable semantics. These equilibria are conventions over signal reports, not claims that agents know the true label. A permutation renames labels without discarding distinctions; coarsening merges labels and discards distinctions. Excluding permutations therefore does not exclude coarsening. An external check, such as an audited answer key, adds information. Any claim that it eliminates these equilibria must specify how often checks occur and how much they change payments.

D LOCAL DYNAMICS AND THE LIMITS OF COORDINATE LEARNING

Learning and interpretation. An equilibrium can attract nearby behavior without attracting all initial strategies. With exact score gradients and strictly positive incentive margins, agents starting sufficiently close to truthful maximum effort converge to it under the proposition below. The same conclusion need not hold from other starting points: uniform random reports can leave computation at its minimum forever. Updating reporting and effort separately also does not guarantee low regret for their joint choice, meaning performance close to the best fixed reporting-and-effort pair. The examples below show both failures. These local guarantees do not describe the noisy online learner in Section 3.

Proposition 4 (Local deterministic attraction)**.** *Suppose expected utilities have continuous gradients near the profile in which every agent reports truthfully at maximum effort. At that corner, every truthful report column has strictly greater gradient than every false column in its row, and every own-effort derivative is strictly positive. Agents use exact-gradient multiplicative updates for reporting and projected gradient ascent for effort, with positive steps whose cumulative sums diverge. There is a feasible neighborhood of the corner from which all reporting policies converge to identity and all effort coordinates reach their upper endpoint in finite time.*

Proof. Continuity and finitely many agents and labels give uniform lower bounds $\gamma_F > 0$ on truthful-versus-false reporting-gradient gaps and $\gamma_s > 0$ on effort gradients in a neighborhood. Choose $\rho, \delta > 0$ small enough that the set

$$s_i \in [1-\delta, 1], \qquad \frac{F_i(a,b)}{F_i(a,a)} \le \rho \quad (b \neq a)$$

lies in that neighborhood. In particular, the denominator is positive. If $g_{i,t}$ is the reporting gradient, a multiplicative update gives

$$\frac{F_{i,t+1}(a,b)}{F_{i,t+1}(a,a)} = \frac{F_{i,t}(a,b)}{F_{i,t}(a,a)} e^{-\alpha_{F,t}(g_{i,t}(a,a)-g_{i,t}(a,b))} \le \frac{F_{i,t}(a,b)}{F_{i,t}(a,a)} e^{-\alpha_{F,t}\gamma_F}.$$

Every false-to-true ratio decreases. Projected effort ascent increases each coordinate by at least $\alpha_{s,t}\gamma_s$ before clipping at 1. Thus simultaneous updates cannot leave this set. Divergent cumulative reporting steps make the ratios vanish; divergent cumulative effort steps make all efforts reach 1 after finitely many iterations. Strict margins keep them there. □

The strict version of Proposition 2's endpoint inequality supplies the effort margin. Equality can still yield a strict equilibrium but does not supply this proof's strictly positive endpoint gradient. No stochastic conclusion follows from this deterministic argument: noisy or biased finite differences may reverse a gradient sign and leave the neighborhood.

Failure of the joint-regret inference. Let J have every entry $1/L$, so it reports uniformly regardless of the signal. Against truthful maximum-effort peers in the example, the family $F(x) = (1-x)J + xI$, $x \in [0,1]$, has utility $1.62sx - 0.01s - 0.5s^2$. Its Hessian is

$$\begin{pmatrix} 0 & 1.62 \\ 1.62 & -1 \end{pmatrix},$$

whose determinant is negative, so one curvature direction is positive and one negative. Separate linearity in x and concavity in s do not imply joint concavity. In a comparator decomposition,

$$U(F^\circ, s^\circ) - U(F_t, s_t) = [U(F^\circ, s_t) - U(F_t, s_t)] + [U(F^\circ, s^\circ) - U(F^\circ, s_t)],$$

the second term involves effort utility at F°, while the actual effort update uses F_t. A coordinate regret bound at F_t cannot simply bound that term.

If one agent starts with uniform reporting and zero effort, $(J, 0)$, the reporting gradient vanishes because the signal covariance is zero; the effort gradient is $-c'(0) = -0.01$, and projection leaves effort at zero. The agent stays there. The fixed comparator $(I, 1)$ earns $12(1.5)(0.3)^2 - 0.51 = 1.11$ each round, giving cumulative utility loss $1.11T$ over T rounds relative to that fixed strategy (external regret). This construction uses fixed truthful peers. If the protocol instead requires all agents to start uniformly, $HF_j = 0$ for every peer forever, so all scores vanish and only cost reduction remains. Minimizing cost in this state does not teach agents to report informative signals.

E ESTIMATORS AND UNCERTAINTY

Reporting incentives with missing answers. At a fixed inference action, allow $R_i \in [L] \cup \{\perp\}$, where $\perp$ means no report. An unavailable answer must be missing; an available answer may be deliberately withheld. Missing reports never count as matching a label. With fixed peers, define

$$M_i(z,b) = \frac{1}{n-1} \sum_{j \neq i} \mathrm{Cov}(\mathbf{1}\{Z_i = z\}, \mathbf{1}\{R_j = b\}), \qquad M_i(z, \perp) = 0. \tag{9}$$

The coefficient $M_i(z,b)$ measures how much reporting b after seeing z contributes to excess agreement; withholding contributes zero. For a policy depending only on Z_i, score is $\sum_{z,b} F_i(z,b) M_i(z,b)$. At $r > 0$ and report-independent cost, sending the original answer is uniquely optimal for every signal with positive probability if and only if $M_i(z,z) > \max\{0, \max_{b \neq z} M_i(z,b)\}$ for each such z. In words, each original answer must score more than every replacement and withholding. Each row can be optimized separately; no uniform prior or conditional independence is required. A positive diagonal and negative off-diagonal for every truthful peer pair are sufficient but stronger conditions. This fixed-action statement does not cover policies using richer private metadata; Section 5.3 tests one such class. Labels enter evaluation only, never scoring or policy fitting.

Finite-action implementation interval. For the score and cost gaps in Proposition 1, each inequality restricts the permitted reward. The target's implementing set is the intersection of these half-lines with $[0, \infty)$, hence is an interval, a singleton, or empty. A zero score gap with positive cost gap makes the set empty. Otherwise its endpoints are

$$\underline{r} = \max\left(0, \max_{\Delta q_b > 0} \frac{\Delta c_b}{\Delta q_b}\right), \qquad \overline{r} = \min_{\Delta q_b < 0} \frac{\Delta c_b}{\Delta q_b}, \tag{10}$$

with empty lower maximum $-\infty$ and empty upper minimum $+\infty$; feasibility requires $\underline{r} \le \overline{r}$. The target is uniquely optimal precisely when all inequalities in (3) are strict.

Proof. Subtract each competitor's utility from the target's utility in (1). Dividing by a positive score gap supplies a lower bound; dividing by a negative gap reverses the inequality and supplies an upper bound. Equal-score actions are ordered by cost. Intersecting these constraints proves the result, including ties. □

Utility bounds instead of threshold ratios. When two settings have nearly equal scores, dividing by their estimated score difference can give unstable or unbounded reward thresholds. We instead compare their payment-minus-cost differences directly on the same questions and evaluate (3) at specified rewards. On an event with simultaneous bounds $\Delta q_b \ge \ell_{q,b}$ and $\Delta c_b \le u_{c,b}$, the inequalities

$$r\ell_{q,b} - u_{c,b} \ge 0 \quad \forall b \tag{11}$$

are sufficient to support the target choice: even its lower estimated score advantage pays for its upper estimated extra cost. Failure leaves the choice unresolved; it need not be impossible. Direct simultaneous bands for $r\Delta q_b - \Delta c_b$ can exploit score–cost dependence. Calibration questions select candidate settings; separate held-out questions test those choices. If peers also change behavior, scores change, so this fixed-peer test does not establish a population equilibrium.

Fixed-assignment score. For $m \ge 2$ assigned tasks, let $N_i(a) = \sum_t \mathbf{1}\{R_i^t = a\}$ for valid categories and $B_{ij} = \sum_t e(R_i^t, R_j^t)$. The cross-task sum in (2) is $\sum_a N_i(a)N_j(a) - B_{ij}$, so the score can be computed without listing all $m(m-1)$ ordered question pairs. When complete per-question records are independent and identically distributed, the same-task and cross-task terms estimate the joint probability and the product of marginals on the same assigned-task population. With missing responses, neither denominator is reduced to count only valid reports. Missing–missing is not treated as categorical agreement. The archived score instead conditions on valid reports; changing that denominator changes incentives as well as estimates.

Signed finite-task DMI. Correlated agreement distinguishes truthful payoff dominance from equilibrium selection (Shnayder et al., 2016; Witt et al., 2026). Determinant mutual information (DMI) supplies a complementary multi-task approach (Kong, 2020); its finite-task estimator is a signed product of determinants on disjoint blocks.

Split assigned tasks randomly into disjoint blocks of sizes $m_1, m_2 \ge L$, and let $N_b(a, c)$ count tasks on which agents i, j report a, c in block $b \in \{1, 2\}$. With $(m)_L = m(m-1)\cdots(m-L+1)$, use

$$\widehat{D}_{ij} = \frac{\det N_1 \det N_2}{(m_1)_L (m_2)_L}. \tag{12}$$

For independent, identically distributed complete reports with joint probability matrix P, expanding a count determinant selects L distinct observations: repeated use of one observation cannot contribute to distinct rows and columns. Thus $\mathbb{E}[\det N_b] = (m_b)_L \det P$. Independence of blocks yields $\mathbb{E}[\widehat{D}_{ij}] = (\det P)^2$. The estimator can be negative in finite samples; taking its absolute value or clipping negative transfers changes it. It is defined for binary reports too, with at least four tasks. Applying the same relabeling to one agent's reports in both blocks changes both determinant signs and leaves their product invariant.

The implementation's exhaustive binary check sums over all $4^4 = 256$ sequences of four iid paired observations under $P = \left(\begin{smallmatrix} 0.4 & 0.1 \\ 0.2 & 0.3 \end{smallmatrix}\right)$. It reproduces $(\det P)^2 = 0.01$, verifies that negative realizations exist, and checks unilateral relabeling invariance. Applying zero categorical indicators to missing reports is a computational extension; the complete-report truthfulness theorem is not asserted for endogenous abstention. The new code validates this baseline but does not include an adaptive LLM experiment using DMI.

F HISTORICAL PROTOCOL AND PROVENANCE

The historical response ZIP contains 187 files and directory entries and has SHA-256

`16bcdda3a3cf9b4f724cd5b282eeaaf016fc`
`19ebdd6ace46feb30dd7d280cb87`

The archived source revision is `7768451eaab6`. The offline reanalysis records its code revision through the run supervisor, seed 42, the exact 75/75 item-ID split, all 20 action rows, nine rewards, 2,000 bootstrap draws per model and five-fold aggregation assignments. It uses Python 3.11 with locked dependencies. This reanalysis makes no model API calls. Missing high-setting records in the historical aggregation bank are represented as missing reports on the archived assignment pool, not silently dropped; known task labels from other records supply the evaluator label and are checked for consistency. Action-cost analysis instead requires an explicit record for each assigned task/action and rejects missing cost records.

Historical replay bootstrap procedure. For each model, let $\widehat{U}_{r,a}$ be its estimated validation utility at reward r and action a. Each replicate draws 75 task positions with replacement and carries along every peer response, focal action response and action-specific cost on the selected task. It recomputes the full score and mean cost. Let $\widehat{d}_{r,a,b} = \widehat{U}_{r,a} - \widehat{U}_{r,b}$ and $\widehat{d}^{(h)}$ be the replicate differences. Let $\widehat{\sigma}_{r,a,b}$ be their bootstrap standard deviation. We use the empirical 95th percentile z of

$$\max_{r,a,b:\widehat{\sigma}_{r,a,b}>10^{-12}} \frac{|\widehat{d}^{(h)}_{r,a,b} - \widehat{d}_{r,a,b}|}{\widehat{\sigma}_{r,a,b}}$$

and report $\widehat{d} \pm z\widehat{\sigma}$. Zero-variance contrasts receive zero radius. These are approximate bootstrap bands, simultaneous within a model rather than across all six models. Repeated observations, a degenerate empirical distribution or small samples can undermine coverage. We make no exact 95% finite-sample certification claim. In particular, zero observed variance cannot prove zero population variance.

An action is not ruled out unless another action has a strictly positive lower bound on its utility advantage. A calibration-selected action is supported by the approximate band only when all its lower gaps against alternatives are nonnegative. This historical convention differs from the strict-positive criterion used for the September study. All 2,000 replicates are retained, including negative score gaps, and both costs and scores vary. This avoids the historical endpoint bootstrap's conditioning on positive score differences and fixed point-estimate costs.

Historical controller. The historical six-model controller uses seed 42 and 24 rounds of 12 questions for each reward. A base round uses the current setting; the next probe round tests an adjacent setting on the same questions, holding peer reports fixed for scoring. After pair k, the controller updates with step $0.15/\sqrt{k}$, exponential gradient smoothing 0.7 and clipping to $[-2, 2]$. It skips updates if either round has fewer than 80% valid reports. Terminal effort averages the chosen setting index, scaled from 0 to 1, over second-half base rounds. There are 144 unique task IDs across the paired sequence; sharing tasks across arms creates dependence and does not multiply the independent sample size. Per-call failures and metering are part of the experiment's resource record. USD values are computed from recorded token usage and configured prices; they are not independently reconciled provider invoices.

Offline live-label recovery. The local MMLU-Pro test cache has revision `b189ec765aa7`. Its native option order agrees with all 4,242 archived calibration label records across 150 unique IDs. The archived loader uses that native order for MMLU-Pro. We join by question ID, reject duplicate or unmatched IDs, validate every report vector against its assignment length and alphabet, and verify paired task order across agents. Live and calibration questions have disjoint text hashes. The evaluation module runs after the historical decisions and never feeds labels to scoring or learning. Its manifest records the Arrow-file hash and question hashes; per-item and per-round outputs make the aggregation reproducible. Terminal metrics use the last six base rounds, totaling 72 questions; cost for each reward condition also includes earlier rounds and all probes. No confidence interval over independent runs is available from this one archived seed.

Cross-fitted aggregation. Five deterministic folds are formed using seed-42 hashes of the legacy task IDs. For each held-out fold, each agent's mean excess-agreement score on the other four folds becomes its voting weight after negative values are set to zero. Those weights combine valid votes on the held-out fold. If all agent weights are zero, equal weights are used. If a task receives no positive vote mass, the aggregate abstains. Ties with positive vote mass choose the smallest label. Labels never enter weight fitting and abstentions count as incorrect in delivered accuracy. Truncating aggregation weights is separate from clipping payments; the incentive analysis uses the untruncated score.

The September calibration/confirmation study uses separate model calls and the different protocol in Appendix B. The preceding archive analyses are historical; their task counts and intervals must not be pooled with that study.

G Reporting and sensitivity experiments

The historical reporting and score-sensitivity study uses the same unmodified archive and a separately recorded specification, `feedback-study-2026-09-25-v1`. Its run revision is `020ddf865ae4`. It makes zero API calls and uses seed 42 and 2,000 paired bootstrap draws. The supplied CSV files include every model, reporting map, withholding state, calibration token-bin boundary, evaluation condition and interval; a manifest records the exact calibration/validation IDs, archive hash, code hash and dependency versions. The split was fixed before evaluating these answer transformations, but after the archive had already been inspected for other analyses.

Response banks and missingness. The ten models, in the fixed order used for coalition selection, are Claude Haiku 4.5, DeepSeek V4 Flash, GLM-4.7 Flash, GPT-5 nano, GPT-oss-20b, HY3 Preview, Mistral Small 2603, Nemotron Nano 30B, Qwen3.5 Flash and Seed 2.0 Mini. Each agent contributes its archived maximum-setting responses on the common assignment pool. Missing records and failed parses are represented by -1, which is never a categorical match, including against another -1. Missing token records are not replaced by zero cost. Reporting experiments do not estimate counterfactual inference costs: all transformations are post-inference, preserving the original calls and modeled costs. Extra local wrapper computation is not metered.

Factorial score sensitivity. For the fixed denominator, same-task and cross-task denominators count assignments; for the conditional denominator, they count valid comparisons. The full estimator uses all ordered pairs $t \neq v$ for its penalty. In the conditional full variant, that denominator is $N_i N_j - N_{ij}$, where N_i, N_j count valid reports and N_{ij} counts jointly valid same-task pairs. The split estimator partitions tasks into three disjoint blocks: one for the bonus and one per penalty marginal, using the exact cross-product of the last two blocks. We average 25 partitions with seeds $42, \ldots, 66$, using the same partitions for every action. Empty valid sets contribute zero. The four variants have the same expectation for independent, identically distributed complete reports, but need not agree in realized samples or under nonresponse. The conditional full variant is a declared sensitivity estimator, not a verbatim implementation of the archived code. All threshold calculations use the full action menu with fixed maximum-setting peers and the same per-assignment costs.

Maps and coalition selection. In zero-based category notation the prespecified maps are identity, constant $g(z) = 0$, cyclic $g(z) = (z + 1) \bmod L$, pair merging $g(z) = 2\lfloor z/2 \rfloor$, and two cells $g(z) = 0$ if $z < L/2$ and $L - 1$ otherwise. For each nonidentity map we transform the first 1, 5 or 10 models in the fixed order above. Coalition identity is not optimized for damage. GPQA has four labels and MMLU-Pro ten, so pair merging leaves two and five used categories. Missingness is preserved in every condition. Plurality uses equal valid votes, deterministic smallest-label tie breaking, and abstains when no report is valid. Consequently the exact accuracy under a relabeling can depend on this tie convention even though score invariance holds identically.

For each full-population profile, one agent learns a potentially different answer-to-report rule from its original signal, with peers held to the specified profile. Training coefficients use the fixed-assignment sample version of (9); each row chooses the largest coefficient among all categorical reports and abstention, whose coefficient is zero. Equal coefficients choose the first categorical index before abstention. The fitted map is then applied to original held-out responses and compared with the incumbent transformed response, using the same peer/task vectors and costs. This is a best response

Table 9: Historical transformations of stored answers. "Coalition" counts agents using the specified map; accuracy is plurality accuracy and mean score averages agents. Unadjusted 95% bootstrap intervals resample held-out questions, with all model outputs kept together; they do not describe new independent model calls.

Benchmark	Map	Coalition	Accuracy (%)	Interval	Mean score
GPQA	identity	0	84.0	[74.7, 92.0]	0.411
GPQA	constant	1	78.7	[69.3, 86.7]	0.332
GPQA	constant	5	20.0	[12.0, 29.3]	0.083
GPQA	constant	10	18.7	[10.7, 28.0]	0.012
GPQA	permutation	1	81.3	[72.0, 89.3]	0.310
GPQA	permutation	5	36.0	[25.3, 48.0]	0.101
GPQA	permutation	10	4.0	[0.0, 9.3]	0.411
GPQA	pair merge	1	82.7	[73.3, 90.7]	0.352
GPQA	pair merge	5	37.3	[26.7, 49.3]	0.209
GPQA	pair merge	10	33.3	[22.7, 44.0]	0.275
GPQA	two cells	1	81.3	[72.0, 89.3]	0.354
GPQA	two cells	5	48.0	[36.0, 60.0]	0.217
GPQA	two cells	10	41.3	[30.7, 53.3]	0.275
MMLU-Pro	identity	0	90.0	[80.0, 98.0]	0.680
MMLU-Pro	constant	1	90.0	[80.0, 98.0]	0.541
MMLU-Pro	constant	5	12.0	[4.0, 22.0]	0.150
MMLU-Pro	constant	10	10.0	[2.0, 20.0]	0.010
MMLU-Pro	permutation	1	90.0	[80.0, 98.0]	0.528
MMLU-Pro	permutation	5	62.0	[48.0, 76.0]	0.259
MMLU-Pro	permutation	10	2.0	[0.0, 6.0]	0.680
MMLU-Pro	pair merge	1	90.0	[80.0, 98.0]	0.593
MMLU-Pro	pair merge	5	44.0	[30.0, 58.0]	0.435
MMLU-Pro	pair merge	10	36.0	[22.0, 50.0]	0.606
MMLU-Pro	two cells	1	90.0	[80.0, 98.0]	0.562
MMLU-Pro	two cells	5	62.0	[48.0, 74.0]	0.285
MMLU-Pro	two cells	10	32.0	[20.0, 44.0]	0.382

to the empirical training distribution among all randomized rules that use only the original answer label: linearity makes a deterministic row maximum sufficient. It is not a best response over inference effort, token-dependent reporting, task-dependent prompts, or unobserved private information. Zero held-out gain can arise because both policies produce exactly the same reports; a degenerate bootstrap interval in that case is not evidence that all population alternatives have been excluded.

Withholding rules. A private state combines the answer z with its token-count group $h \in \{0, 1, 2\}$: lower, middle or upper third. The state index is $z + Lh \in \{0, \ldots, 3L - 1\}$. Group boundaries use finite token counts of valid calibration responses only. The fixed-score policy withholds exactly those states with negative empirical own-label covariance coefficient, retaining zero-coefficient states. The conditional-score policy begins by retaining every state, evaluates each one-state deletion on calibration reports, selects the largest improvement (breaking ties toward the largest state index), and repeats until no improvement exceeds 10^{-12}. It is a greedy heuristic, not a globally optimal conditional-score policy. At evaluation, policies receive only their own answer and token count. Invalid or unavailable state records remain unchanged. Different score definitions induce different fitted policies; the comparison changes both the score definition and the fitted withholding rule; it does not compare score definitions while holding that rule fixed.

Uncertainty and endpoints. For each benchmark the bootstrap jointly resamples all held-out task positions, recomputes focal scores before and after a deviation, and reports the 2.5th and 97.5th percentiles of paired gains. Fitted rules and calibration bins remain fixed. These unadjusted exploratory intervals do not account for policy-training uncertainty, model-generation variability, archive selection or multiple comparisons. The intervals do not establish a discovery after correction for multiple tests. The coalition table reports intervals for accuracy levels, not a test of the difference between any two conditions. An agent's cost is unchanged when it alters a stored report, so its utility

gain is reward times its score gain. This differs from the coalition's mean score and the swarm's accuracy.

H CONTROLLED SYNTHETIC EXPERIMENTS

Interpretation of the diagnostics. In the historical estimates, covariance diagonals are positive and off-diagonals negative for 41/45 GPQA pairs and 2/45 MMLU-Pro pairs. Yet the score margin for original reporting against the average peer in (9) is strictly positive for all signal rows in 10/10 and 9/10 models on retrospective held-out tasks; the remaining model has one signal with zero margin. These small-sample diagnostics do not establish population assumptions.

Synthetic tests isolate what the empirical study cannot identify (Figure 9). Exact-gradient dynamics initialized near truth or coarsening approach different high-effort conventions; effort falls to zero under uniform reporting. Under asymmetric channels and nonuniform priors, a truthful-reporting margin can remain positive after the pairwise sign condition fails. Finally, shared latent error changes correctness from 75% to 12.5% while preserving the entire report distribution and expected score 0.260. Reports alone cannot distinguish these worlds. This illustrates an information limit, not the frequency of shared errors among LLMs.

Different attractors. Use the three-label channels and costs from Section 5. Agents with identical channels and costs start with $s = 0.85$ and $F = 0.97T + 0.03J$, for $T = I$, G or J, where $g = (1, 1, 3)$ and J is uniform. At each step, every agent updates simultaneously using the exact own-strategy gradients, holding peers fixed in differentiation:

$$\nabla_F U = 12(0.3s)^2 HF, \qquad \partial_s U = 12(0.3)(0.3s)\,\mathrm{tr}(F^\top HF) - 0.01 - s.$$

Multiplicative row updates and projected effort ascent use step $0.3/\sqrt{t}$ for 500 iterations. The final efforts are $1, 1, 0$, with utilities approximately $1.110, 0.570, 0$. These selected initializations illustrate local behavior, not the fraction of starting strategies that reach each equilibrium. The local proof applies near any strict deterministic corner with the stated gradient margins, including this coarsening.

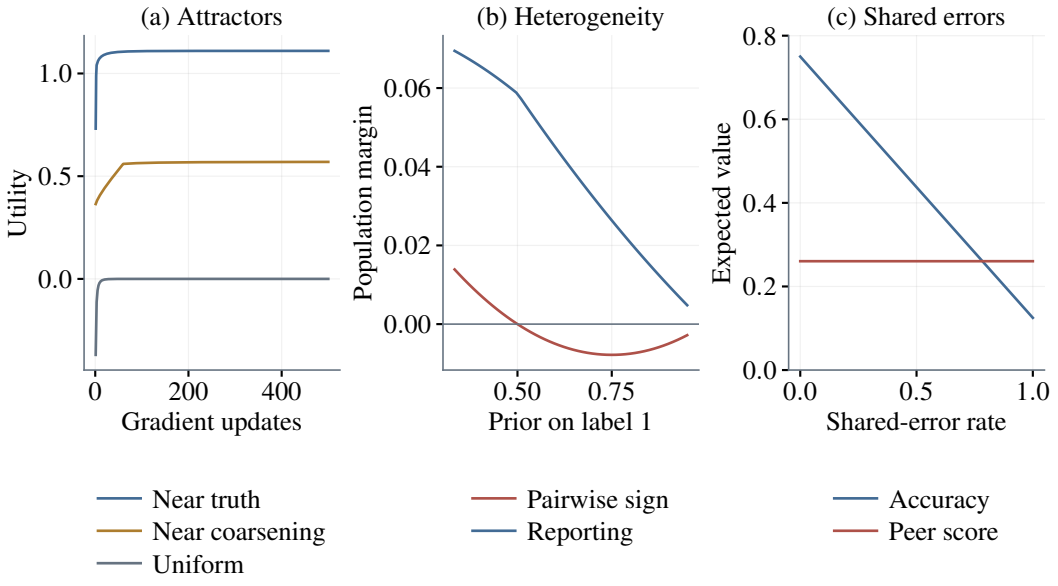


Figure 9: Controlled synthetic tests. (a) Exact-gradient dynamics approach distinct high-effort truthful and coarsened profiles. (b) Nonuniform priors and asymmetric channels can violate pairwise covariance signs while retaining a positive truthful-reporting margin. (c) Shared latent errors change correctness without changing reports or score. These are analytic or deterministic calculations, not additional LLM runs.

Heterogeneous channels and nonuniform priors. Use columns as latent labels and rows as observed signals:

$$C_1 = \begin{pmatrix} .75 & .125 & .125 \\ .125 & .75 & .125 \\ .125 & .125 & .75 \end{pmatrix}, \quad C_2 = \begin{pmatrix} .8 & .1 & .2 \\ .15 & .65 & .15 \\ .05 & .25 & .65 \end{pmatrix}, \quad \pi = (v, (1-v)/2, (1-v)/2).$$

For 50 equally spaced $v \in [1/3, .95]$, calculate $\Delta = C_1 \operatorname{diag}(\pi) C_2^\top - (C_1\pi)(C_2\pi)^\top$. The pairwise sign margin is the minimum of all diagonal entries and negated off-diagonal entries. The focal reporting margin is the minimum over rows of the diagonal minus the largest alternative, including zero for abstention. At $v = .95$ these margins are -0.0028125 and 0.0048046875, respectively. This example relaxes symmetry and uniformity for a fixed-action reporting question; it does not extend the partition equilibrium theorem to arbitrary channels.

Observation-equivalent common errors. Let the true answer Y be uniform on labels $\{0, 1, 2\}$ and let $B \sim \text{Bernoulli}(\rho)$ be independent. Set the shared latent belief $W = Y$ if $B = 0$ and $W = Y + 1 \pmod 3$ otherwise. Conditional on W, agents independently receive symmetric signals correct about W with probability $p = .75$. Then W is uniform for every ρ, so the entire joint report distribution, and every report-only payment statistic, is identical across these worlds. Nonetheless accuracy relative to Y is

$$(1-\rho)p + \rho\frac{1-p}{2}, \qquad q = \frac{3}{2}\left(p - \frac{1}{3}\right)^2 = \frac{25}{96}.$$

At $\rho = 0$ accuracy is .75; at $\rho = 1$ it is .125. The conditional-independence assumption relative to external truth Y is absent in the intermediate worlds. Thus the same reports and payments can accompany different rates of correct answers without claiming a new general impossibility theorem.

I Additional historical results

The ten-model maximum-setting archive covers 150 GPQA (Rein et al., 2023) and 99 MMLU-Pro items. The six-model action bank covers 150 MMLU-Pro items and 20 settings. The reanalysis uses 75 calibration and 75 validation questions for the action bank and GPQA, and 49 calibration and 50 validation questions for MMLU-Pro reporting. These splits were made after inspecting the archive; they do not create new model outputs.

All-action sensitivity and replay. Comparing only minimum and maximum settings with the original conditional score suggests reward 17.60 suffices for GPT-5 nano's maximum. Comparing every setting raises the requirement to 192.56 because the low setting is the strongest competitor. Claude's threshold rises from 6.92 to 13.75. Qwen's think2k yields both a higher score and lower cost than its maximum, although maximum outperforms minimum on score. DeepSeek's maximum scores below its minimum. Table 10 separates denominator choice from cross-task estimation using identical responses and costs. With full penalties, changing the denominator raises nano's threshold from 191.28 to 610.37 and makes GPT-oss's maximum unimplementable. With fixed denominators, changing split to full penalties moves nano's threshold from 500.53 to 610.37. These ratios are descriptive point estimates, not stable uncertainty summaries. Calibration and validation choices agree in 36 of 54 combinations: six models at nine rewards. The approximate simultaneous bands support 17 of 54 selected settings against every alternative (11 of 48 at positive rewards). These comparisons reuse questions and are not independent trials. The remaining 37 choices lack that support; a single estimated reward threshold does not convey this uncertainty.

Historical reporting and withholding. Full-population cyclic relabeling reduces GPQA/MMLU-Pro plurality accuracy from 84%/90% to 4%/2%, preserving scores 0.411/0.680. Pair merging gives 33.3%/36% accuracy and scores 0.275/0.606. The calibration-fitted answer-only maps against identity, cyclic, pair and two-cell profiles reproduce the incumbent held-out reports for every model, with zero realized gain. Constant reports retain small scores 0.012/0.010 through shared missingness. These historical outcomes are separate from the September reporting interventions.

Conditional withholding has positive point gains for 8/10 GPQA and 4/10 MMLU-Pro models. Three unadjusted gain intervals lie entirely above zero: GPQA nano ($+0.134$), GPQA Mistral ($+0.135$),

Table 10: Minimum reward making the maximum setting optimal against every alternative, with maximum-setting peers. The four columns vary the score denominator and cross-question calculation. "Split" averages 25 partitions into three blocks; "Full" uses all distinct-question pairs. "None" means no reward implements the target at these estimates. Units are kilotoken equivalents.

	Conditional on valid		Fixed assignments	
Model	Split	Full	Split	Full
Claude Haiku 4.5	13.75	17.85	14.03	18.28
DeepSeek V4 Flash	None	None	None	None
GPT-5 nano	192.56	191.28	500.53	610.37
GPT-oss-20b	61.16	58.43	None	None
Qwen3.5 Flash	None	None	None	None
Seed 2.0 Mini	39.56	44.00	33.19	37.00

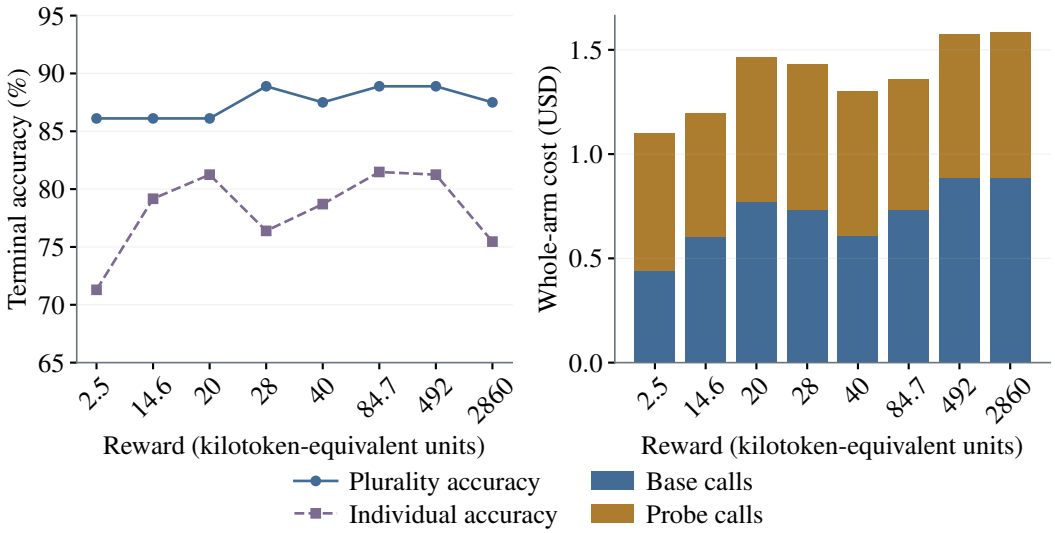


Figure 10: Historical live controller, evaluated offline. Accuracy uses the same 72 terminal base tasks; cost includes all 24 rounds and probes. One seed and unequal budgets do not identify a matched-cost benefit. These observations are separate from the September static study; Table 11 supplies every arm.

and MMLU-Pro DeepSeek (+0.117). Their delivered accuracies fall from 73.3%, 73.3%, 78% to 10.7%, 10.7%, 24%. No fixed-score gain interval lies entirely above zero, although one point gain is positive. These exploratory outcomes do not establish general resistance to withholding.

Historical live and aggregation results. The archived six-agent controller shows how settings changed under eight reward levels. Across eight reward arms, terminal normalized action index rises from 0.009 to 1.000, with intermediate reversals. It uses adjacent-action probes and smoothed, clipped finite differences, not the exact gradients of our local theorem. At the reward extremes, plurality answers 62 versus 63 of the same 72 terminal tasks correctly; whole-arm cost rises from \$1.097 to \$1.585. Probes account for 48.5% of the historical \$11.009 expenditure. These are one-seed observations, not independent replications. They are separate from the five-run comparison whose protocol was fixed before collection in Section 3.

Table 11 gives every historical reward arm. The highest-reward arm has 87.5% report validity and 75.5% per-agent delivered accuracy. Five-fold cross-fitted aggregation gives 81.3% GPQA accuracy versus 80.7% plurality, and 88.9% for both on MMLU-Pro (Table 12). Table 13 gives the complete archived native-action menu, score and cost estimates.

Table 11: Historical controller at eight reward levels. Terminal measures average the last six base rounds on the same 72 questions; effort is the setting index scaled from 0 (lowest) to 1 (highest). Reward is in kilotoken-equivalent units. Cost includes all 24 rounds and probes. The conditions share one seed and have unequal inference costs.

Reward	Terminal effort	Agent accuracy (%)	Plurality (%)	Total USD
2.51	0.009	71.3	86.1	1.097
14.59	0.449	79.2	86.1	1.197
20.00	0.442	81.2	86.1	1.464
28.00	0.542	76.4	88.9	1.430
40.00	0.535	78.7	87.5	1.301
84.74	0.782	81.5	88.9	1.360
492.31	0.812	81.2	88.9	1.574
2860.15	1.000	75.5	87.5	1.585

Table 12: Ten-model accuracy (%) on archived questions; N is the number of questions. Plurality gives each valid vote equal weight. Original weighted voting fits weights on the same answers it combines. Cross-fitted voting fits weights on four folds and evaluates on the fifth, repeated across all folds. Missing aggregate answers count as incorrect.

Benchmark	N	Plurality	Original weighted	New cross-fitted
GPQA	150	80.7	81.3	81.3
MMLU-Pro	99	88.9	88.9	88.9

Table 13: All 20 archived settings on 150 MMLU-Pro questions, with peers at maximum settings. Score uses all assignments and all distinct-question pairs. Cost is mean input plus output kilotokens per call, including failed calls with recorded usage. "Valid" is the percentage returning a usable label. Thinking-budget labels specify API controls, not realized token use.

Model	Native action	Score	Cost (ktok)	Valid (%)
Claude Haiku 4.5	think0	0.68899	0.508	99.3
	think1k	0.73956	0.867	100.0
	think1.5k	0.75981	0.863	100.0
	think2k	0.77070	1.062	100.0
DeepSeek V4 Flash	nothink	0.77014	0.356	99.3
	think	0.74974	1.832	100.0
GPT-5 nano	minimal	0.46241	0.269	99.3
	low	0.71964	0.682	100.0
	medium	0.71844	1.926	99.3
	high	0.72691	5.117	97.3
GPT-oss-20b	low	0.69349	0.625	99.3
	medium	0.74156	1.267	99.3
	high	0.73330	4.981	90.0
Qwen3.5 Flash	nothink	0.45180	0.646	77.3
	think512	0.74760	1.408	100.0
	think1k	0.75650	2.008	100.0
	think2k	0.77767	2.547	100.0
	think4k	0.76318	2.984	100.0
Seed 2.0 Mini	nothink	0.68884	0.645	98.0
	think	0.76277	3.380	100.0