UPON THE CERTIFIED REGULATION OF DELIBERATION IN MULTI-AGENT SYSTEMS POSSESSED OF UNSHARED EVIDENCE

Anonymous authors
Paper under double-blind review

ABSTRACT

Where the evidence pertinent to a decision lies dispersed among several agents, deliberation constitutes the sole instrument by which it may be consolidated; yet deliberation is customarily left unregulated. Fixed schedules, round-robin exchange and unconstrained debate afford no assurance whatsoever that a given conversational act is the one warranted by the circumstances, and assemblies of language models, in the manner of human committees, are disposed to rehearse what is already held in common and to acquiesce prematurely in a plausible error. We advance QUORUM (Quantified Uncertainty and Oversight of Regret for Unshared-evidence Multi-agent deliberation), an inference-time framework which regulates deliberation rather than merely countenancing it. At each round the framework reduces the proceedings to a compact state comprising uncertainty, dissent, redundancy, evidential gain and premature concurrence; thereupon it determines both the intervention warranted and the agent most fittingly charged with its execution. Each proposal must thereafter withstand a round-wise conformal examination, which circumscribes its one-step regret under a distribution-free, finite-sample assurance and substitutes a certified alternative wherever the proposal is adjudged inadmissible. Regulated deliberation surpasses its unregulated counterpart throughout twelve models and two benchmarks, and not infrequently surpasses the condition wherein every agent is furnished with the entirety of the evidence.

1 INTRODUCTION

Large language models (LLMs) are increasingly deployed as multi-agent systems in which several agents propose solutions, exchange arguments, critique one another, and jointly produce a decision (Du et al., 2023; Liang et al., 2024; Xiong et al., 2023; Wu et al., 2024; Li et al., 2023). Such systems have shown promise in mathematical reasoning, question answering, software development, planning, and tool-assisted problem solving. Their appeal stems from the idea that multiple agents can contribute complementary reasoning, identify errors overlooked by individual models, and produce more reliable collective decisions. However, recent studies question whether discussion alone consistently delivers these benefits. Multi-agent deliberation can duplicate the capabilities of a well-prompted single agent, amplify correlated errors, or converge prematurely on persuasive but incorrect conclusions (Wang et al., 2024; Liang et al., 2024). Simply increasing the number of agents or discussion rounds therefore does not guarantee better collective reasoning.

This limitation becomes especially consequential when task-relevant information is distributed across agents. Many collaborative decisions in medicine, intelligence analysis, and organizational planning involve participants who observe different pieces of evidence. No individual has access to the complete information required for a correct decision; the group succeeds only if its members communicate and integrate their complementary knowledge. This setting is captured by the *hidden-profile paradigm*, originally introduced in social psychology to study information pooling in group decision making (Stasser & Titus, 1985; Stasser, 1988). In a hidden profile, information shared by all group members supports a plausible but suboptimal alternative, whereas uniquely held information, when pooled across members, reveals the correct choice (Stasser & Titus, 1985; Lu et al., 2012; Schulz-Hardt & Mojzisch, 2012).

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.

Although discussion should theoretically recover distributed evidence, human groups commonly exhibit an information bias: information shared by several members is more likely to be mentioned, repeated, and reinforced than information known to only one member (Stasser & Titus, 1985; Stasser et al., 1989; Gigone & Hastie, 1993). As a result, discussion often reinforces initial preferences instead of correcting them. A meta-analysis of hidden-profile studies found that groups discussed substantially more common than unique information and were significantly less likely to identify correct solution under hidden- than full-information conditions (Lu et al., 2012). Effective hidden-profile reasoning therefore requires more than allowing participants to speak: groups must surface unshared evidence, consider dissenting perspectives, revise premature beliefs, and determine when consensus is supported to justify termination (Lu et al., 2012; Schulz-Hardt & Mojzisch, 2012).

Recent work shows that LLM-based groups reproduce many of these collective reasoning failures. HiddenBench (Li et al., 2025) formalizes the hidden-profile paradigm for multi-agent LLM evaluation using 65 tasks. Each agent receives shared task information together with an asymmetric private clue, and the agents must deliberate before making a collective decision. Across 15 models from four families, HiddenBench finds persistent failures to integrate distributed evidence: agents may collectively have sufficient information yet reach the wrong answer, while model scale or reasoning strength does not reliably predict successful pooling.(Li et al., 2025). HiddenBench, therefore, exposes a gap not directly addressed by the conventional reasoning benchmarks. The bottleneck is not only whether an LLM can reason from the evidence it observes, but whether a group can control communication so that the evidence is elicited, routed, and incorporated before consensus forms.

Existing multi-agent LLM methods provide only partial solutions. Debate frameworks improve reasoning through critique and competing arguments (Du et al., 2023; Liang et al., 2024; Xiong et al., 2023); role-based frameworks coordinate specialized agents (Li et al., 2023; Wu et al., 2024; Hong et al., 2024); and coordination methods rely on predefined interaction topologies, voting, judges, or aggregation mechanisms (Chan et al., 2023; Chen et al., 2024; Zhu et al., 2025). These approaches work well when agents share most of the available information. Hidden-profile tasks, however, require adaptive communication control: the system must decide whether to elicit missing evidence, clarify disagreement, challenge unsupported conclusions, route the discussion, or terminate deliberation. Fixed round-robin protocols (Li et al., 2025) cannot adapt to the evolving discussion state, while unconstrained LLM orchestrators may make unreliable control decisions.

Moreover, existing debate and orchestration methods provide no statistical guarantee that the communication action selected at given state is appropriate. Such guarantees matter because communication decisions shape the information available to the group: selecting wrong speaker, repeating shared evidence, or voting too early can reinforce incorrect consensus, while unnecessary discussion increases cost and risks distraction or context degradation. Communication control should therefore be treated as a sequential decision problem rather than fixed prompting protocol. At each round, controllers use the agents' evolving beliefs and public transcript to select an action that improves the collective decision state while avoiding unnecessary communication. Our contributions are:

- **Closed-loop communication control.** We pose hidden-profile reasoning as a sequential communication-control problem and introduce QUORUM — a framework considering evolving discussion state to select, conformally certify, and route communication interventions, enabling closed-loop coordination among LLM agents.
- **Conformal reliability.** We propose a round-wise conformal calibration framework that conformally screens communication actions using one-step regret, providing finite-sample, regret guarantees and replacing inadmissible proposals with certified alternatives.
- **Comprehensive evaluation.** We evaluate QUORUM across several open- and closed-weight language models on HiddenBench (Li et al., 2025) and an LLM-generated benchmark based on GroupTravelBench task design (Cheng et al., 2026). We compare hidden-information, round-robin, and full-information baselines, and analyze communication efficiency, calibration, speaker routing, generalization, and heterogeneous agents.

2 Problem Formulation

We consider hidden-profile multi-agent decision tasks in which the information needed to identify the correct answer is distributed across multiple agents. A task is a tuple $(x, \mathcal{Y}, I_{\text{shared}}, \{I_i\}_{i=1}^{N})$,

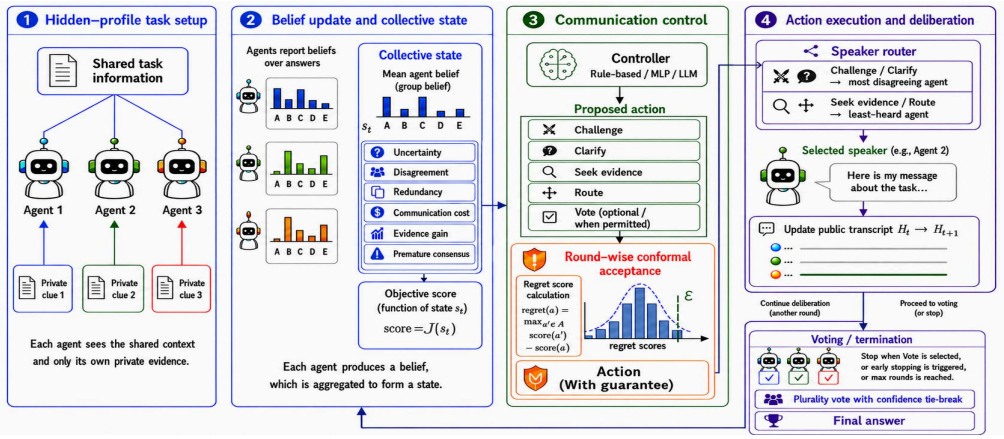

Figure 1: Proposed framework *QUORUM*: (1) Each agent receives shared task information and a private clue. (2) Agents report beliefs conditioned on the public transcript, which QUORUM aggregates with transcript-level signals into the collective state s_t. (3) A controller proposes a communication action, and round-wise conformal acceptance retains or replaces it using calibrated one-step regret. (4) An action-conditional router selects the speaker, whose message updates the transcript and beliefs. The loop repeats until termination, followed by voting with confidence-based tie-breaking.

where x is task description, $\mathcal{Y} = \{y_1, \ldots, y_K\}$ is a finite answer set, I_{shared} is public to all agents, and I_i is visible only to agent i. Agent i initially observes $(x, \mathcal{Y}, I_{\text{shared}}, I_i)$, but not I_j for $j \neq i$.

The agents communicate through a shared public transcript $\mathcal{H}_t = (m_1, \ldots, m_t)$, with $\mathcal{H}_0 = \varnothing$. Private evidence becomes available to the group only when an agent communicates it on the public transcript. Goal is to produce a collective answer $\widehat{y} \in \mathcal{Y}$ while integrating the relevant private evidence with limited communication. QUORUM treats this process as closed-loop communication control: the controller chooses what kind of action intervention is needed, and a separate action-conditional router chooses which agent should carry it out. Neither controller nor router answers the task itself or generates evidence on an agent's behalf. The final answer is produced from the quorum among the LLM agents from their beliefs formed from the information on the public transcript.

3 QUORUM: Conformally Calibrated Communication for Multi-Agent Systems

3.1 Method Overview

Figure 1 summarizes the QUORUM loop. Let $Y \in \mathcal{Y}$ denote the latent correct answer. An ideal controller would choose the intervention that maximizes the expected quality of the next collective belief while penalizing communication:

$$a_t^\star = \arg\max_{a \in \mathcal{A}} \mathbb{E}[U(b_{t+1}) - \lambda C(\mathcal{H}_{t+1}) \mid s_t, a], \tag{1}$$

where b_{t+1} is the collective posterior induced after the next message, $U(b) = -H(b)$ is negative posterior entropy, and $C(\mathcal{H})$ is communication cost. Exact optimization of Equation 1 is intractable because each action can induce many possible language-model responses and future discussion trajectories. QUORUM, therefore, uses an observable collective state and a tractable one-step objective to propose an intervention, round-wise conformal acceptance to screen that proposal, and an action-conditional router to select its speaker. The resulting message updates the public transcript and agent beliefs, closing the loop. The following subsections formalize state construction, action proposal, conformal acceptance, speaker routing, and termination.

Stage 1: Belief elicitation and state construction

After observing $\mathcal{H}_t$, each agent reports $p_i^{(t)} = \left[p_i^{(t)}(y_1), \ldots, p_i^{(t)}(y_K)\right]$, $\sum_{k=1}^K p_i^{(t)}(y_k) = 1$. This is the agent's reported belief conditional on $(x, I_{\text{shared}}, I_i, \mathcal{H}_t)$; it is not assumed to be a

calibrated Bayesian posterior. The aggregate belief is the arithmetic mean $\bar{p}^{(t)} = \frac{1}{N}\sum_{i=1}^{N} p_i^{(t)}$. QUORUM represents the current discussion by $s_t = [H_t, D_t, R_t, G_t, C_t, P_t]$, whose components are defined as:

Group uncertainty (H_t). We use the normalized entropy of the aggregate belief: $H_t = -\frac{\sum_{k=1}^{K} \bar{p}^{(t)}(y_k) \log \bar{p}^{(t)}(y_k)}{\log K}$. Thus, larger values of H_t indicate a less decisive group belief.

Inter-agent disagreement (D_t). Aggregate confidence can conceal incompatible individual beliefs. We measure disagreement by $D_t = \frac{1}{N}\sum_{i=1}^{N} \mathrm{JSD}\left(p_i^{(t)} \parallel \bar{p}^{(t)}\right)$, where JSD is the Jensen–Shannon divergence (Lin, 2002). A large D_t indicates that some agents interpret the available evidence differently from the group.

Message redundancy (R_t) and evidence gain (G_t). Let $R_t = \phi_R(m_t, \mathcal{H}_{t-1})$ and $G_t = \phi_G(m_t, \mathcal{H}_{t-1}, \mathcal{Y})$, where the fixed estimators $\phi_R, \phi_G \in [0, 1]$ are applied identically across controllers. Redundancy estimator ϕ_R combines lexical overlap with an LLM assessment of whether m_t repeats previously public information. Evidence-gain estimator ϕ_G assesses if m_t contributes previously unshared, relevant fact supporting or eliminating candidate answers. Higher R_t indicates more repetition, while higher G_t indicates a more informative contribution. We set $R_0 = G_0 = 0$.

Communication cost (C_t). We use normalized transcript length: $C_t = \frac{\mathrm{Words}(\mathcal{H}_t)}{1000}$.

Premature consensus (P_t). Let $p_t^\star = \max_{y \in \mathcal{Y}} \bar{p}^{(t)}(y)$. We distinguish supported agreement from early anchoring as $P_t = p_t^\star(1 - G_t)\,\mathbf{1}[p_t^\star > 0.70,\ G_t < 0.30,\ t < K + 2]$. The score is nonzero when the group becomes highly confident early in the discussion despite weak recent evidence.

Surrogate discussion loss. The state variables define $J(s_t) = \alpha_H H_t + \alpha_D D_t + \alpha_R R_t - \alpha_G G_t + \alpha_C C_t + \alpha_P P_t$. Lower values are preferred. The coefficients are fixed design hyperparameters, selected before evaluation and held constant across controller variants and test tasks. The objective is a local control signal, not a substitute for answer correctness. To summarize, Stage 1: the current transcript and private agent contexts produce the observable state s_t and its scalar discussion loss $J(s_t)$. Stage 2 uses these to propose the next intervention.

Stage 2: Communication-Action Proposal

Communication-action space. QUORUM uses four communication interventions $\mathcal{A}_{\mathrm{comm}} = \{\textsc{Challenge}, \textsc{Clarify}, \textsc{SeekEvidence}, \textsc{Route}\}$. Controllers that can terminate adaptively additionally include VOTE. All controller variants use same action definitions and message templates.

- CHALLENGE asks the speaker to examine the current leading answer, identify unsupported assumptions, and provide potentially contradictory evidence.
- CLARIFY asks an agent with a divergent belief to explain the private evidence/interpretation for that disagreement.
- SEEKEVIDENCE requests one previously unshared, decision-relevant fact and an explanation of how it supports or eliminates candidate answers.
- ROUTE is the evidence-sharing intervention. It asks an under-participating agent for concise discriminative facts.
- VOTE, when available, terminates deliberation and invokes the final voting rule.

Controller variants. Every controller maps s_t from Stage 1 to a proposal action $a_{t,0} = \pi_m(s_t)$. Routing and messaging procedures remain fixed, so controller comparisons isolate action proposal.

Rule-based controller. The interpretable controller applies fixed thresholds to the state variables. It selects CHALLENGE when confidence is high but recent evidence is weak, CLARIFY when disagreement is high, SEEKEVIDENCE when communication is redundant or uninformative, and ROUTE otherwise. It permits voting only after a minimum number of rounds, when the leading belief is sufficiently strong, disagreement is low, and premature consensus is absent.

Learned controller. Exhaustive training rollouts evaluate every admissible action from each encountered state using Stage-3 counterfactual reward and record the vector $\left[r_t(s_t, a_1), \ldots, r_t(s_t, a_{|\mathcal{A}_{\mathrm{comm}}|})\right]$. A multilayer perceptron $f_\theta : \mathbb{R}^6 \to \mathbb{R}^{|\mathcal{A}_{\mathrm{comm}}|}$ with one

12-unit ReLU hidden layer is trained to predict this vector. At inference time, it proposes $\pi_{\text{MLP}}(s_t) = \arg\max_{a \in \mathcal{A}_{\text{comm}}} f_\theta(s_t)_a$.

LLM-based controller. The LLM controller receives task, candidate answers, public transcript, state vector, agent beliefs, and speaker counts, and selects an action from the admissible action space. A second variant includes VOTE, enabling adaptive termination. Speaker selection remains separate. Regardless of controller type, Stage 2 outputs only proposed action $a_{t,0}$. Stage 3 evaluates the proposal against other admissible actions and determines action a_t that may actually be executed.

Stage 3: Counterfactual evaluation and conformal acceptance

Conformal prediction (Vovk et al., 2005) provides distribution-free, finite-sample marginal coverage for exchangeable non-conformity scores on calibration datapoints $(Q_1, \ldots, Q_n)$ and the test input Q_{test}. Specifically, if $Q_1, \ldots, Q_n, Q_{\text{test}}$ are exchangeable and are produced by the same fixed scoring procedure, then, for $k = \lceil (n+1)(1-\alpha) \rceil$, and ϵ as the k-th smallest calibration score, we have the following probabilistic guarantees: $\mathbb{P}(Q_{\text{test}} \leq \epsilon) \geq 1 - \alpha$.

Counterfactual action outcomes. For a nonterminal action a applied at state s_t, the fixed Stage-4 router selects a speaker, the speaker generates a message, all agents update their beliefs, and QUORUM obtains a candidate next state s^a_{t+1}. All candidate actions are evaluated independently from the same frozen pre-action state. For each candidate action, QUORUM caches the selected speaker, generated message, resulting belief updates, and candidate next state. Only the cached branch corresponding to the accepted action is committed to the public trajectory. The one-step improvement is $r_t(s_t, a) = J(s_t) - J(s^a_{t+1})$. We propose one-step regret as non-conformity score: $\mathcal{R}_t(s_t, a) = \max_{a' \in \mathcal{A}^m_t(s_t)} r_t(s_t, a') - r_t(s_t, a) = J(s^a_{t+1}) - \min_{a' \in \mathcal{A}^m_t(s_t)} J(s^{a'}_{t+1})$, where $\mathcal{A}^m_t(s_t)$ is the set of actions admissible for controller m at round t. Hence $\mathcal{R}_t(s_t, a) \geq 0$, and at least one admissible action has regret zero. Terminal actions, when available, are evaluated using the same fixed terminal-state scoring rule in calibration and testing.

Offline round-wise calibration. For a fixed controller method m, define the non-conformity score of its proposal at round t as

$$Q^m_t(s) = \mathcal{R}_t(s, \pi_m(s)) \tag{2}$$

Calibration uses complete trajectories generated by the fixed controller, routing policy, model, prompts, and score-generating procedure. Because some trajectories terminate early, only calibration trajectories that reach round t contribute to the round-t multiset . Let $n_t = |\mathcal{Q}^{m,(t)}_{\text{cal}}|$ and $k_t = \lceil (n_t + 1)(1-\alpha) \rceil$. The threshold ϵ^m_t is k_t-th smallest calibration score.

Online conformal acceptance. At test time, non-conformity score (regret) for controller's proposed action is computed, and if it's within the conformal threshold ϵ^m_t, then it's passed to router for stage 4. Else, near-optimal conformal action set $\Gamma^m_t(s)$ is generated by executing all possible actions:

$$\Gamma^m_t(s) = \{a \in \mathcal{A}^m_t(s) : \mathcal{R}_t(s, a) \leq \epsilon^m_t\} . \tag{3}$$

Here, QUORUM applies a fixed fallback Select, which returns a minimum-regret member of $\Gamma^m_t(s)$, with ties resolved by a fixed action ordering: $\text{Select}(\Gamma^m_t(s)) \in \arg\min_{a \in \Gamma^m_t(s)} \mathcal{R}_t(s, a)$. The set is nonempty because a minimum-regret action has regret zero and $\epsilon^m_t \geq 0$. Constructing either the calibration score in (2) or the test-time set in (3) requires evaluating every admissible action from the same state. These evaluations are counterfactual branches: they are used to calculate regret, and only the accepted branch updates the actual transcript. Stage 3 therefore maps proposal $a_{t,0}$ to an accepted action a_t with its cached counterfactual branch. The real transcript is not modified during candidate evaluation; Stage 4 commits the cached branch associated with the accepted action.

Stage 4: Action-conditioned routing and transcript update

After conformal acceptance chooses a_t, the deterministic router selects speaker $i_t = \rho(a_t, s_t)$. For disagreement-focused actions,

$$\begin{gathered} \rho(a_t, s_t) = \arg\max_i \text{JSD}\left(p^{(t)}_i \parallel \bar{p}^{(t)}\right), \\ a_t \in \{\text{CHALLENGE}, \text{CLARIFY}\}. \end{gathered} \tag{4}$$

The most disagreeing agent is the most likely to hold evidence or an interpretation not yet incorporated into the group belief. For evidence-acquisition actions,

$$\rho(a_t, s_t) = \arg\min_i n^{(t)}_i, a_t \in \{\text{SEEKEVIDENCE}, \text{ROUTE}\}, \tag{5}$$

where $n_i^{(t)}$ is the number of messages previously contributed by agent i. The least-heard agent is used as a proxy for the agent most likely to retain unshared evidence. Ties in Equations 4–5 are resolved by a fixed agent ordering.

The executed control is therefore $u_t = (a_t, i_t)$, where conformal communication control determines a_t and the router determines i_t. During the counterfactual evaluation in Stage 3, QUORUM caches, for each admissible action, the selected speaker, generated message, resulting belief updates, and corresponding next state. After a_t is accepted, Stage 4 retrieves the cached branch associated with that action and commits it to the public trajectory. In particular, the cached message m_{t+1} is appended to the public transcript and the cached agent beliefs $p_i^{(t+1)}$ are used to form s_{t+1}; neither the message nor the belief updates are resampled after acceptance. Thus, if no stopping condition is met, Stage 4 closes the feedback loop by returning the updated transcript to Stage 1.

Stage 5: Termination and Collective Prediction

Deliberation terminates when VOTE is selected by a controller that admits it, a fixed early-stopping condition is met, or the maximum number of rounds T is reached. Each agent then independently reports an answer $\widehat{y}_i \in \mathcal{Y}$ and confidence $c_i \in [0, 1]$. We define $N(y) = \sum_{i=1}^{N} \mathbf{1}[\widehat{y}_i = y]$, and $\mathcal{Y}_{\text{tie}} = \{y : N(y) = \max_{y' \in \mathcal{Y}} N(y')\}$. $\mathcal{Y}_{\text{tie}}$ contains all the plurality winners. The final prediction is $\widehat{y}_{\text{final}} = \arg\max_{y \in \mathcal{Y}_{\text{tie}}} \sum_{i:\widehat{y}_i = y} c_i$.

3.2 END-TO-END PROCEDURE AND THEORETICAL GUARANTEE

The five stages above specify one online control round:

$$\underbrace{(\mathcal{H}_t, \{I_i\}_{i=1}^{N})}_{\text{current information}} \xrightarrow{\text{Stage 1}} s_t \xrightarrow{\text{Stage 2}} a_{t,0} \xrightarrow{\text{Stage 3}} a_t \xrightarrow{\text{Stage 4}} \mathcal{H}_{t+1},$$

followed by either another round or the Stage-5 prediction. Calibration phase precedes online deliberation and supplies the round-specific thresholds used in Stage 3. Here, we formalize the statistical guarantee for the Stage-3 acceptance rule based on the theory of conformal prediction framework.

Proposition 1 specializes this construction to the round-specific one-step regret scores of a fixed QUORUM controller and gives the sharper finite-sample result for almost surely distinct scores.

Proposition 1 (Round-Wise Marginal Regret Coverage)**.** *Fix a controller m and round t. Let n_t calibration trajectories reach round t, with states $s_1, \ldots, s_{n_t}$, and let s_{test} be the round-t state of a new trajectory conditional on that trajectory reaching round t. Assume that, conditional on the realized calibration-cohort size and on the test trajectory reaching round t, $Q_t^m(s_1), \ldots, Q_t^m(s_{n_t}), Q_t^m(s_{\text{test}})$ are exchangeable, and that π_m and all components of the score-generating procedure were fixed independently of the calibration sample. With k_t and ϵ_t^m defined in Stage 3,*

$$\mathbb{P}(Q_t^m(s_{\text{test}}) \le \epsilon_t^m \mid \mathcal{T}_{\text{test}} \text{ reaches } t,\ N_t = n_t) \ge 1 - \alpha. \tag{6}$$

Equivalently,

$$\mathbb{P}(\pi_m(s_{\text{test}}) \in \Gamma_t^m(s_{\text{test}}) \mid \mathcal{T}_{\text{test}} \text{ reaches } t,\ N_t = n_t) \ge 1 - \alpha.$$

If $k_t \le n_t$ and the pooled calibration and test scores are almost surely distinct, the probability in Equation 6 equals $k_t/(n_t + 1)$, and therefore lies in $\left[1 - \alpha,\ 1 - \alpha + \frac{1}{n_t+1}\right)$.

4 EXPERIMENTS

We evaluate QUORUM through four research questions: (1) if adaptive communication improves accuracy over no-discussion and round-robin baselines; (2) if distributed deliberation can match full-information reasoning; (3) how controller choice, actions, routing, and termination affect performance; and (4) if learned coordination generalizes to unseen tasks and heterogeneous agent groups.

Benchmarks and models. We evaluate QUORUM on the 65 hidden-profile decision problems in HiddenBench (Li et al., 2025) and on a separately generated benchmark following the task structure of GroupTravelBench (Cheng et al., 2026). Each task distributes evidence required for the correct answer across agents, which communicate only through public transcript. Because the original

Table 1: Task and vote accuracy for HiddenBench before conformal acceptance, across the evaluated language models. Each entry reports *task accuracy / vote accuracy*. Higher values are better.

Model	Hidden Pre	Hidden Post	Full Info	Rules	MLP	LLM	LLM+Vote
Qwen3-0.6B	.159/.202	.185/.203	.359/.364	**.374/.391**	.287/.295	.313/.306	.297/.314
Qwen3-1.7B	.062/.095	.092/.092	.267/.285	.559/.563	**.600/.599**	.554/.561	.564/.561
Qwen3-4B	.092/.095	.097/.124	.369/.406	.826/.829	**.831/.829**	.826/.831	.800/.789
Qwen3-8B	.062/.126	.149/.175	.631/.596	**.908/.913**	.856/.871	.877/.879	.805/.808
Qwen3-14B	.123/.161	.267/.266	.774/.752	.872/.879	.774/.783	.821/.841	**.928/.928**
Qwen3-32B	.108/.141	.282/.287	.826/.817	.918/.900	.841/.856	.913/.903	**.923/.918**
Qwen3-235B	.062/.123	.292/.306	.862/.841	**.928/.943**	.856/.846	.887/.879	.836/.848
Qwen2.5-32B	.046/.087	.297/.290	.908/.881	.897/.903	.867/.874	**.923/.926**	.918/.924
Llama-3.1-8B	.077/.177	.354/.354	.610/.599	**.733/.682**	.667/.655	.595/.572	.600/.590
Llama-3.3-70B	.046/.123	.436/.460	.862/.877	.928/.935	.928/.926	**.949/.953**	.908/.914
Phi-4	.077/.150	.195/.221	.877/.819	.923/.924	.826/.831	.903/.895	**.944/.937**
Mistral-24B	.072/.170	.467/.485	.908/.859	**.923/.924**	.841/.848	.903/.914	.923/.925
Gemma-3-27B	.097/.184	.349/.352	.846/.821	**.903/.883**	.872/.863	.856/.848	.897/.904
DeepSeek-V4-Flash	.092/.185	.384/.395	.907/.881	**.953/.957**	.784/.779	.953/.937	.908/.889
DeepSeek-V4-Pro	.077/.158	.431/.474	.954/.933	**1.00/1.00**	.939/.921	.969/.968	.969/.972

Table 2: Accuracies for the generated benchmark averaged across 12 models

Method	Task Accuracy	Vote Accuracy
Hidden Pre	0.2537	0.2475
Hidden Post	0.4982	0.4925
Full Info	0.3657	0.3358
Rules (Ours)	0.6544	0.6542
MLP (Ours)	0.5071	0.5025
LLM (Ours)	0.6150	0.6125
LLM+Vote (Ours)	**0.6682**	**0.6642**

GroupTravelBench data are unavailable, our generated benchmark is not a reproduction and results are not directly comparable; generation and validation details are in the Appendix. We test 15 instruction-tuned open-weight models spanning multiple families and scales (Touvron et al., 2023; Bai et al., 2023; Abdin et al., 2024; Xu et al., 2026; Jiang et al., 2023); model and serving details, together with details on supplementary closed-model experiments, are also provided in the Appendix.

4.1 Compared Discussion Protocols

We compare four QUORUM controllers (Rules, MLP, LLM, and LLM+Vote) with three references: Hidden-Pre, in which agents vote without communicating; Hidden-Post, which uses fixed round-robin discussion; and Full-Info, in which agents receive all evidence but do not discuss. Full-Info is an information-rich empirical reference, not a theoretical upper bound.

4.2 Evaluation Metrics

We report two complementary evaluation metrics computed from the agents' final votes. **Vote accuracy** measures the proportion of individual agent votes that match the ground-truth answer: $\mathrm{VoteAcc} = \frac{\sum_{d\in\mathcal{D}}\sum_{i=1}^{N_d}\mathbb{I}[v_{d,i}=y_d]}{\sum_{d\in\mathcal{D}}N_d}$, where $\mathcal{D}$ is the evaluation set, N_d is the number of agents for task d, $v_{d,i}$ is the final vote of agent i, and y_d is the correct answer.

Task accuracy measures whether the collective decision is correct. For each task, the final prediction is obtained by plurality voting, $\hat{y}_d = \arg\max_{a\in\mathcal{A}_d}\sum_{i=1}^{N_d}\mathbb{I}[v_{d,i}=a]$, with ties broken using the summed confidence of supporting agents when available, and otherwise by the benchmark's deter-

ministic ordering. Task accuracy is then $\mathrm{TaskAcc} = \frac{1}{|\mathcal{D}|} \sum_{d \in \mathcal{D}} \mathbb{I}[\hat{y}_d = y_d]$. It is our primary metric, while vote accuracy provides a complementary measure of consistent coverage to correct answer.

5 Results and Discussion

Table 1 reports the accuracy of each discussion protocol across the evaluated models. The hidden pre-discussion condition provides a lower-information baseline in which agents cannot exchange their evidence. Fixed round-robin discussion improves information availability, but does not explicitly control which evidence to be requested, which agent to provide it, or when discussion terminates.

Across the evaluated models, QUORUM consistently improves over these uncontrolled hidden-information protocols. The strongest controller obtains an average accuracy of **0.843**, compared with **0.285** for fixed round-robin discussion, corresponding to an absolute improvement of **0.558**. This result supports **RQ1:** the observed gains arise not merely from allowing agents to communicate, but from adapting the communication intervention to the evolving collective state.

5.1 Can Communication Compensate for Distributed Information?

We next compare controlled hidden-information deliberation with the full-information, no-discussion condition. This comparison separates two potential sources of performance: access to evidence and the ability to integrate that evidence effectively. For every model, at least one QUORUM controller reaches or exceeds Full Info. The strongest single controller on average, Rules, achieves **0.843** versus **0.731** for Full Info, a difference of **0.112**. In almost all cases, controlled deliberation performs better despite no individual agent having direct access to all hidden facts.

This result should not be interpreted as exceeding a theoretical upper bound. The full-information condition is an information-rich empirical baseline: language models may still overlook evidence, confuse relationships among facts, or commit prematurely to an answer. The result instead suggests that staged evidence sharing and explicit disagreement resolution can facilitate evidence integration more effectively than presenting all facts in a single context.

5.2 Effect of Controller Design

All controller variants share the same actions, routing rules, agents, and prompts, differing only in action selection. The rule-based controller achieves the highest average accuracy **0.843**, followed by the LLM **0.816**, LLM+Vote **0.815**, and MLP **0.785**, showing how collective-state variables support effective coordination even without learned control. The two LLM controllers behave differently across model scales as seen in Tab 1. The effect of adaptive voting varies across models and is not monotonic with model scale.

5.3 Generalization to Unseen Tasks

To evaluate held-out generalization, we train the MLP controller on five-task subsets and test it on the remaining tasks; the all-task MLP is retained only as an in-domain oracle-imitation diagnostic. The unseen-task MLP achieves **0.73** average accuracy, compared with **0.77** for the all-task model, indicating that most of the learned coordination behavior transfers to unseen discussion states. This suggests that the compact state captures recurring conditions such as disagreement, redundancy, and unsupported consensus rather than task-specific content. The remaining gap may reflect task-distribution shift, variation in answer-set size, or imperfect counterfactual reward targets. Full training and evaluation details are provided in the appendix.

5.4 Ablation Studies

We conduct five ablations that independently examine: (1) random sampling from conformal prediction sets, (2) action-conditional speaker routing, (3) the contribution of each communication action, (4) heterogeneous agent compositions, and (5) controller-state features. In each case, all other experimental components remain fixed. All detailed results are provided in appendix.

Conformal Prediction-Set Sampling tests whether conformally admissible actions also support successful deliberation. At each discussion round, we uniformly sample an action from the conformal prediction set rather than executing the controller's preferred action, and follow the resulting trajectory until termination. Randomly sampled actions maintain high task success across all controller variants, with only small differences between them. Thus, the calibrated sets generally contain multiple effective actions rather than a single fragile choice, providing useful alternative deliberation paths in addition to their finite-sample statistical guarantee.

Speaker-Routing Ablation compares action-conditional routing with generic policies while holding the discussion state and controller-selected action fixed. We measure average one-step objective improvement and normalized regret relative to the counterfactual optimal speaker. Action-conditional routing achieves the largest mean objective improvement while maintaining low regret. Random, most-disagreeing, and most-spoken routing produce negative mean improvements. Round-robin obtains marginally lower normalized regret, but substantially lower improvement, partly because near-optimal speakers can have similar outcomes when the counterfactual performance range is small. These results show that no single generic notion of speaker importance is sufficient. Matching disagreement-oriented actions to dissenting agents and evidence-seeking actions to less-participating agents yields more informative messages without jointly optimizing every action–speaker pair.

Action Leave-One-Out Ablation We remove each communication action for an entire discussion and measure the change in task accuracy. Removing ROUTE causes the largest average drop, highlighting the importance of engaging underrepresented agents, while removing SEEKEVIDENCE also substantially hurts performance, especially for LLM controllers. CHALLENGE has a modest effect, consistent with its targeted role in preventing premature consensus, whereas CLARIFY has the smallest effect and can slightly reduce accuracy, suggesting that routing and evidence seeking often resolve disagreement indirectly.

Heterogeneous-Agent Ablation We replace each agent with a different language model while keeping the controller fixed, introducing variation in reasoning ability, calibration, and instruction following. Results show that QUORUM remains effective across heterogeneous compositions. Strong and cross-family agent groups generally benefit most from adaptive coordination, whereas weaker compositions remain limited by their ability to extract evidence. This suggests that QUORUM can exploit complementary reasoning styles and primarily relies on observable discussion dynamics rather than model-family-specific interactions.

Controller-State Feature Ablation compares MLP controllers trained with different subsets of the controller-state representation. The complete state achieves the highest task accuracy and lowest discussion objective. Information-related features—redundancy, evidence gain, and premature consensus—produce comparable accuracy, showing that they provide the most direct signals for action selection, but their higher objective indicates less efficient or controlled discussions.

Removing communication cost from the combined belief-and-information state reduces accuracy and produces the worst objective. Conversely, belief-only features yield the shortest discussions but the lowest accuracy because uncertainty and disagreement do not reveal whether new evidence has been introduced or whether consensus is sufficiently supported. The feature groups are therefore complementary: information features guide intervention selection, belief features characterize collective reasoning, and cost discourages unnecessarily long discussions. Combining all three provides the most reliable balance between task success and discussion quality.

6 CONCLUSION

We introduced QUORUM, a closed-loop framework for adaptively selecting, certifying, and routing communication in hidden-profile multi-agent reasoning. Across two benchmarks and several language models, QUORUM consistently improves over uncontrolled discussion and often matches or exceeds full-information baselines, demonstrating that structured communication can be as important as information access itself. While our evaluation focuses on hidden-profile tasks, future work will explore broader collaborative reasoning settings.

Ethics Statement

This work evaluates communication and coordination among language-model agents in synthetic hidden-information decision tasks. The experiments do not involve human participants, personal data, or real-world deployment. QUORUM is intended as a research framework for studying how adaptive communication control can improve information sharing and collective reasoning in multi-agent systems. The conformal component provides a statistical guarantee on one-step surrogate regret under the assumptions stated in the paper, rather than a guarantee of final-answer correctness. We do not identify any direct ethical concerns arising from the experimental setup beyond the general considerations associated with deploying language-model agents in real-world applications.

Reproducibility Statement

We provide the methodological and experimental details needed to reproduce the reported results, including the full QUORUM control procedure, controller architectures, routing rules, conformal acceptance rule, evaluation metrics, and benchmark protocols. HiddenBench experiments use 65 tasks, with conformal evaluation performed on a fixed partition of 20 calibration tasks and 45 held-out evaluation tasks; calibration is performed separately for each evaluated configuration, including the controller, model, prompts, routing policy, termination rule, decoding procedure, and score-generating components. The Appendix reports the evaluated model families and controller configurations, and provides the complete prompt templates used for belief estimation, communication, controller decisions, and synthetic task generation. The synthetic benchmark construction procedure and its generation prompts are also documented in the Appendix. We will release the implementation, experiment configurations, benchmark-generation scripts, and evaluation code necessary to reproduce the main results upon publication.

AI Use Statement

In this work, we used generative AI tools for coding assistance and for editing and polishing prose. We also use language models as objects of study in this work: they are used as experimental agents, as described in Sections 3 and 4. We did not use generative AI tools to generate research ideas, to conduct the literature search, or to produce experimental results without verification. We tested and reviewed all AI-assisted code, and all reported numbers were produced by executed experiments rather than by a language model. We have reviewed all AI-assisted text and take full responsibility for the final content of this work, including its text, claims, and artifacts.

References

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J Hewett, Mojan Javaheripi, Piero Kauffmann, et al. Phi-4 technical report. *arXiv preprint arXiv:2412.08905*, 2024.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

Chi-Min Chan, Weize Chen, Yusheng Su, Jianxuan Yu, Wei Xue, Shanghang Zhang, Jie Fu, and Zhiyuan Liu. Chateval: Towards better llm-based evaluators through multi-agent debate. *arXiv preprint arXiv:2308.07201*, 2023.

Weize Chen, Yusheng Su, Jingwei Zuo, Cheng Yang, Chenfei Yuan, Chi-Min Chan, Heyang Yu, Yaxi Lu, Yi-Hsin Hung, Chen Qian, et al. Agentverse: Facilitating multi-agent collaboration and exploring emergent behaviors. In *International Conference on Learning Representations*, volume 2024, pp. 20094–20136, 2024.

Xiang Cheng, Yulan Hu, Lulu Zheng, Zheng Pan, Xin Li, and Yong Liu. Grouptravelbench: Benchmarking llm agents on multi-person travel planning. *arXiv preprint arXiv:2605.25200*, 2026.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B Tenenbaum, and Igor Mordatch. Improving factuality and reasoning in language models through multiagent debate. *arXiv preprint arXiv:2305.14325*, 2023.

Daniel Gigone and Reid Hastie. The common knowledge effect: Information sharing and group judgment. *Journal of Personality and social Psychology*, 65(5):959, 1993.

Sirui Hong, Mingchen Zhuge, Jonathan Chen, Xiawu Zheng, Yuheng Cheng, Jinlin Wang, Ceyao Zhang, Steven Yau, Zijuan Lin, Liyang Zhou, et al. Metagpt: Meta programming for a multi-agent collaborative framework. In *International Conference on Learning Representations*, volume 2024, pp. 23247–23275, 2024.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

Guohao Li, Hasan Abed Al Kader Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. Camel: Communicative agents for" mind" exploration of large language model society. *arXiv preprint arXiv:2303.17760*, 2023.

Yuxuan Li, Aoi Naito, and Hirokazu Shirado. Hiddenbench: Assessing collective reasoning in multi-agent llms via hidden profile tasks. *arXiv preprint arXiv:2505.11556*, 2025.

Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Shuming Shi, and Zhaopeng Tu. Encouraging divergent thinking in large language models through multi-agent debate. In *Proceedings of the 2024 conference on empirical methods in natural language processing*, pp. 17889–17904, 2024.

Jianhua Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information theory*, 37(1):145–151, 2002.

Li Lu, Y Connie Yuan, and Poppy Lauretta McLeod. Twenty-five years of hidden profiles in group decision making: A meta-analysis. *Personality and Social Psychology Review*, 16(1):54–75, 2012.

Stefan Schulz-Hardt and Andreas Mojzisch. How to achieve synergy in group decision making: Lessons to be learned from the hidden profile paradigm. *European Review of Social Psychology*, 23(1):305–343, 2012.

Garold Stasser. Computer simulation as a research tool: The discuss model of group decision making. *Journal of experimental social psychology*, 24(5):393–422, 1988.

Garold Stasser and William Titus. Pooling of unshared information in group decision making: Biased information sampling during discussion. *Journal of personality and social psychology*, 48 (6):1467, 1985.

Garold Stasser, Laurie A Taylor, and Coleen Hanna. Information sampling in structured and unstructured discussions of three-and six-person groups. *Journal of personality and social psychology*, 57(1):67, 1989.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. *Algorithmic learning in a random world*. Springer, 2005.

Qineng Wang, Zihao Wang, Ying Su, Hanghang Tong, and Yangqiu Song. Rethinking the bounds of llm reasoning: Are multi-agent discussions the key? In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 6106–6131, 2024.

Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, et al. Autogen: Enabling next-gen llm applications via multi-agent conversations. In *First conference on language modeling*, 2024.

Kai Xiong, Xiao Ding, Yixin Cao, Ting Liu, and Bing Qin. Examining inter-consistency of large language models collaboration: An in-depth analysis via debate. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 7572–7590, 2023.

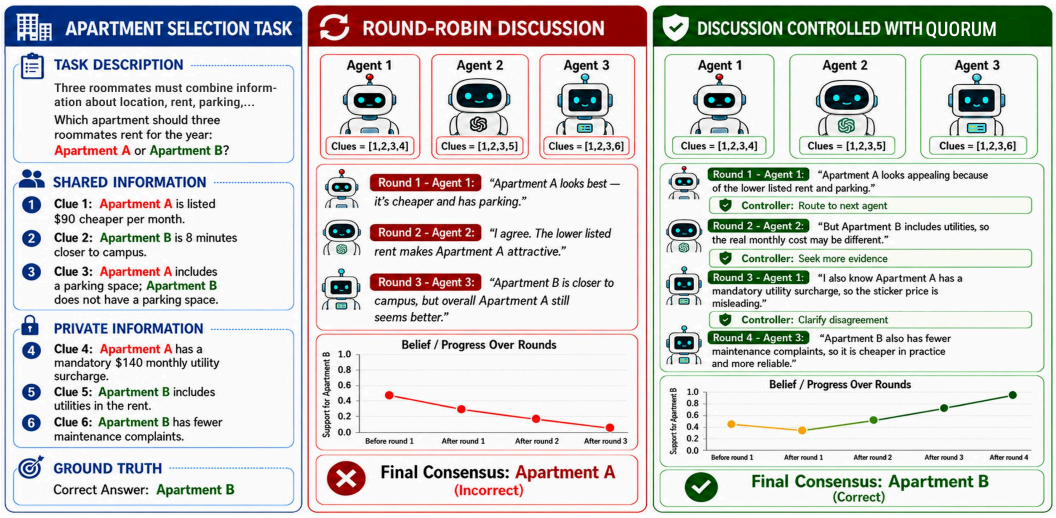

Figure 2: Three agents choose between Apartments A and B. Shared evidence initially favors Apartment A, but complementary private clues collectively identify Apartment B as the correct choice. Round-robin discussion repeats common evidence further suppresses support for B (as Agent 1 has no clue about any utility surcharges for Apartment B in Round 1), and the final decision converges on A. *QUORUM* instead adaptively selects interventions and speakers to surface unshared evidence, reverse premature consensus, and recover B.

Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, et al. Deepseek-v4: Towards highly efficient million-token context intelligence. *arXiv preprint arXiv:2606.19348*, 2026.

Kunlun Zhu, Hongyi Du, Zhaochen Hong, Xiaocheng Yang, Shuyi Guo, Daisy Zhe Wang, Zhenhailong Wang, Cheng Qian, Robert Tang, Heng Ji, et al. Multiagentbench: Evaluating the collaboration and competition of llm agents. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8580–8622, 2025.

A Interpretation, Scope, and Proof of Proposition 1

Coverage of the controller proposal. Proposition 1 provides a round-wise marginal coverage statement for the one-step regret of the controller's proposed action. Let N_t denote the random number of calibration trajectories that reach discussion round t, and let n_t denote its realized value. For a fixed controller m and round t, the proposition states that

$$\Pr\Big(\mathcal{R}_t\big(s_t^{\text{test}}, \pi_m(s_t^{\text{test}})\big) \le \epsilon_t^m \;\Big|\; \mathcal{T}_{\text{test}} \text{ reaches round } t,\ N_t = n_t\Big) \ge 1-\alpha. \tag{7}$$

Equivalently, with probability at least $1-\alpha$, the controller's proposal belongs to the round-specific conformal action set:

$$\pi_m(s_t^{\text{test}}) \in \Gamma_t^m(s_t^{\text{test}}). \tag{8}$$

The probability in Equation 7 is marginal over the calibration trajectories and the new test trajectory, conditional on the test trajectory reaching round t and on $N_t = n_t$. It is not, in general, a coverage guarantee conditional on one fixed realized calibration sample.

Deterministic property of the executed action. The probabilistic coverage statement concerns the controller's raw proposal rather than the action ultimately executed by QUORUM. The executed

action is

$$a_t = \begin{cases} \pi_m(s_t), & \mathcal{R}_t(s_t, \pi_m(s_t)) \le \epsilon_t^m, \\ \mathrm{Select}(\Gamma_t^m(s_t)), & \mathcal{R}_t(s_t, \pi_m(s_t)) > \epsilon_t^m. \end{cases} \tag{9}$$

Provided that the conformal action set is evaluated using all admissible counterfactual branches and that

$$\mathrm{Select}(\Gamma_t^m(s_t)) \in \Gamma_t^m(s_t),$$

the executed action satisfies

$$\mathcal{R}_t(s_t, a_t) \le \epsilon_t^m \tag{10}$$

for every evaluated state s_t.

Equation 10 is an algorithmic consequence of the acceptance rule, rather than an additional probabilistic conformal guarantee. If the proposal is accepted, it satisfies the threshold by definition. If it is rejected, the fallback returns an element of $\Gamma_t^m(s_t)$, every member of which has regret at most ϵ_t^m. The conformal set is nonempty because at least one admissible action minimizes the realized counterfactual next-state loss and therefore has zero one-step regret.

This deterministic statement assumes that the counterfactual branch used to evaluate $\mathcal{R}_t(s_t, a_t)$ is the branch committed to the public trajectory. In particular, the selected speaker message and the resulting belief updates must not be resampled after acceptance. Otherwise, the realized next state may differ from the evaluated counterfactual state, and Equation 10 need not hold for the resampled outcome.

Scope of the guarantee. The proposition bounds one-step regret relative to the admissible action producing the lowest realized counterfactual next-state loss under the surrogate discussion objective J. It does not identify a unique ground-truth communication action, guarantee that the surrogate-optimal action improves answer correctness, establish global policy optimality, or guarantee that the final collective answer is correct.

The proposition also does not provide simultaneous coverage over all rounds of a discussion trajectory. In particular, the collection of round-wise guarantees

$$\Pr\big(Q_t^m(s_t^{\mathrm{test}}) \le \epsilon_t^m \,\big|\, \mathcal{T}_{\mathrm{test}} \text{ reaches round } t,\ N_t = n_t\big) \ge 1 - \alpha$$

does not imply

$$\Pr\big(Q_t^m(s_t^{\mathrm{test}}) \le \epsilon_t^m \text{ for every reached round } t\big) \ge 1 - \alpha.$$

A trajectory-level guarantee would require an additional simultaneous or sequential calibration argument, which is outside the scope of Proposition 1.

Round-specific exchangeability. The marginal coverage statement requires exchangeability of the calibration and test proposal scores at the round under consideration. More precisely, conditional on $N_t = n_t$ and on the test trajectory reaching round t, the scores

$$Q_t^m(s_t^1), \ldots, Q_t^m(s_t^{n_t}), Q_t^m(s_t^{\mathrm{test}})$$

must be exchangeable. This condition holds under the proposition's assumptions when calibration and test trajectories are generated by the same fixed, policy-induced trajectory distribution and when the controller and all components of the score-generating procedure are fixed independently of the calibration sample.

Because communication control is sequential, exchangeability at a later round does not follow automatically from exchangeability at the initial round. In particular, if an earlier conformal replacement changes the distribution of later test states relative to the calibration trajectories, exchangeability of the later-round scores is an additional assumption. The guarantee should therefore be understood as conditional on round-specific exchangeability under the deployed policy.

Accordingly, calibration is performed separately for every configuration for which coverage is claimed, including the controller, agent model, prompting configuration, admissible action space, routing policy, termination rule, decoding procedure, and score-generating components. Any learned component must be fixed before calibration or trained using data independent of the calibration sample.

Finite-sample quantile convention. Let

$$k_t = \lceil (n_t + 1)(1 - \alpha) \rceil .$$

When $k_t \leq n_t$, the threshold ϵ_t^m is the k_t-th smallest calibration score. When $k_t = n_t + 1$, including the case $n_t = 0$, we use the standard convention

$$\epsilon_t^m = +\infty.$$

This convention yields valid but uninformative coverage because every admissible action belongs to the resulting conformal set.

When ties occur among the pooled calibration and test scores, the nonrandomized procedure may be conservative. The exact coverage value $k_t/(n_t + 1)$ stated in Proposition 1 applies when $k_t \leq n_t$ and the pooled scores are almost surely distinct.

Proof of Proposition 1.

Proof. Condition on the event

$$\mathcal{E}_t = \{\mathcal{T}_{\text{test}} \text{ reaches round } t,\ N_t = n_t\} .$$

For notational convenience, define

$$Q_i = Q_t^m(s_t^i), \qquad i = 1, \ldots, n_t,$$

and

$$Q_{n_t+1} = Q_t^m(s_t^{\text{test}}).$$

By assumption,

$$Q_1, \ldots, Q_{n_t}, Q_{n_t+1}$$

are exchangeable conditional on $\mathcal{E}_t$.

By the definitions of Q_t^m and Γ_t^m,

$$\pi_m(s_t^{\text{test}}) \in \Gamma_t^m(s_t^{\text{test}}) \quad \Longleftrightarrow \quad Q_{n_t+1} \leq \epsilon_t^m. \tag{11}$$

First suppose that $k_t = n_t + 1$. By convention, $\epsilon_t^m = +\infty$, and hence

$$\Pr(Q_{n_t+1} \leq \epsilon_t^m \,|\, \mathcal{E}_t) = 1 = \frac{k_t}{n_t + 1}.$$

Now suppose that $k_t \leq n_t$, so that ϵ_t^m is the k_t-th smallest calibration score. Under conditional exchangeability, the standard split-conformal order-statistic argument gives

$$\Pr(Q_{n_t+1} \leq \epsilon_t^m \,|\, \mathcal{E}_t) \geq \frac{k_t}{n_t + 1}. \tag{12}$$

This inequality remains valid in the presence of ties; ties may make the nonrandomized conformal procedure conservative.

From the definition of k_t,

$$\frac{k_t}{n_t + 1} = \frac{\lceil (n_t + 1)(1 - \alpha) \rceil}{n_t + 1} \geq 1 - \alpha. \tag{13}$$

Combining this inequality with Equation 12 proves the round-wise lower coverage bound. Equation 11 then gives the equivalent action-set statement.

Finally, suppose that $k_t \leq n_t$ and that the $n_t + 1$ pooled calibration and test scores are almost surely distinct. Conditional exchangeability implies that the rank of the test score among the pooled scores is uniformly distributed over

$$\{1, \ldots, n_t + 1\}.$$

Under distinctness,

$$Q_{n_t+1} \leq \epsilon_t^m$$

Table 3: Task and vote accuracy on HiddenBench (Li et al., 2025) after conformal acceptance. QUORUM results are evaluated on 45 held-out tasks after using 20 tasks for conformal calibration. Hidden Pre, Hidden Post, and Full Info are reproduced 65-task benchmark references and are shown for context rather than as matched-split comparisons. Each entry reports task accuracy / vote accuracy. Higher values are better.

Model	Hidden Pre	Hidden Post	Full Info	Rules	MLP	LLM	LLM+Vote
Qwen3-0.6B	.159/.202	.185/.203	**.359/.364**	.289/.309	.267/.280	.333/.331	.222/.234
Qwen3-1.7B	.062/.095	.092/.092	.267/.285	**.644/.629**	.578/.583	.556/.571	.556/.560
Qwen3-4B	.092/.095	.097/.124	.369/.406	.733/.743	.778/.800	.778/.783	**.844/.846**
Qwen3-8B	.062/.126	.149/.175	.631/.596	**.911/.926**	.867/.880	.911/.903	.867/.880
Qwen3-14B	.123/.161	.267/.266	.774/.752	.889/.891	.778/.794	.867/.863	**.933/.931**
Qwen3-32B	.108/.141	.282/.287	.826/.817	**.933/.937**	.889/.897	.889/.886	.867/.869
Qwen2.5-32B	.046/.087	.297/.290	.908/.881	.911/.897	.867/.863	**.956/.960**	.933/.914
Llama-3.1-8B	.077/.177	.354/.354	.610/.599	.800/.743	**.822/.811**	.711/.680	.622/.623
Phi-4	.077/.150	.195/.221	.877/.819	**.933/.914**	.822/.823	.911/.897	.911/.914
Mistral-24B	.072/.170	.467/.485	.908/.859	**.956/.943**	.867/.863	.889/.880	.867/.891
DeepSeek-v4-Flash	.092/.185	.384/.395	.907/.881	**.956/.947**	.844/.840	.867/.863	.955/.931
DeepSeek-v4-Pro	.077/.158	.431/.474	.954/.933	**1.00/1.00**	.867/.874	.956/.954	**1.00/1.00**

holds if and only if the pooled rank of Q_{n_t+1} is at most k_t. Consequently,

$$\Pr(Q_{n_t+1} \le \epsilon_t^m \mid \mathcal{E}_t) = \frac{k_t}{n_t+1}. \tag{14}$$

Using

$$\lceil x \rceil < x + 1,$$

we obtain

$$\frac{k_t}{n_t+1} = \frac{\lceil (n_t+1)(1-\alpha) \rceil}{n_t+1} \tag{15}$$

$$< 1 - \alpha + \frac{1}{n_t+1}. \tag{16}$$

Together with the lower bound, this proves

$$1 - \alpha \le \frac{k_t}{n_t+1} < 1 - \alpha + \frac{1}{n_t+1},$$

completing the proof. □

B Additional Experimental Details

Models. We evaluate QUORUM on 15 instruction-tuned open-weight language models spanning multiple families and scales, including Qwen3 (0.6B, 1.7B, 4B, 8B, 14B, 32B, and 235B), Qwen2.5-32B, Mistral-24B, Phi-4, Llama-3.1-8B, Llama-3.3-70B, Gemma-3-27B, DeepSeek-v4-Flash, and DeepSeek-v4-Pro. The conformal evaluation is available for a 12-model subset, while the generated benchmark is evaluated on a 12-model subset as reported in the corresponding tables.

For LLM-based controller, the same model additionally serves as the controller responsible for selecting communication actions, whereas the learned-controller variants replace this decision process with a lightweight MLP while retaining the evaluated LLM for belief estimation, message generation, transcript analysis, and final voting.

B.1 Conformal Acceptance Results

HiddenBench. Table 3 reports HiddenBench performance after conformal acceptance. The 65 tasks are partitioned into 20 calibration tasks and 45 held-out evaluation tasks. The reported QUORUM results therefore measure performance on the held-out test split after applying the round-wise conformal acceptance procedure.

Table 4: Task and vote accuracy for the generated benchmark across the evaluated language models. Each entry reports *task accuracy / vote accuracy*. Higher values are better.

Model	Hidden Pre	Hidden Post	Full Info	Rules	MLP	LLM	LLM+Vote
Qwen3-0.6B	.228/.240	.236/.240	.204/.230	.228/.230	.220/.220	.209/.210	**.248/.240**
Qwen3-1.7B	.204/.210	.272/.270	.248/.240	**.312/.310**	.264/.260	.252/.250	.264/.260
Qwen3-4B	.224/.230	.292/.290	.292/.270	.456/.440	.428/.430	**.480/.480**	.476/.470
Llama-3.1-8B	.224/.230	.312/.310	.280/.270	.509/.530	.384/.370	**.524/.520**	.504/.490
Qwen3-8B	.248/.230	.324/.300	.288/.270	.684/.690	.528/.540	.652/.650	**.716/.710**
Qwen3-14B	.256/.240	.400/.400	.336/.310	.696/.690	.544/.540	.684/.690	**.744/.750**
Mistral-24B	.292/.280	.576/.570	.432/.370	.872/.870	.496/.480	.624/.620	**.832/.840**
Gemma-3-27B	.244/.250	.730/.720	.444/.400	.806/.800	.549/.560	.750/.740	**.858/.850**
Qwen3-32B	.256/.240	.668/.660	.376/.360	.660/.650	.384/.370	.592/.580	**.716/.710**
Qwen2.5-32B	.284/.280	.684/.670	.496/.440	.884/.890	.752/.740	.880/.880	**.900/.900**
Llama-3.3-70B	.276/.270	.728/.720	.520/.440	.889/.890	.848/.830	**.924/.920**	.912/.910
Qwen3-235B	.308/.270	.756/.760	.472/.430	**.857/.860**	.688/.690	.809/.810	.848/.840

Across the evaluated models, the rule-based controller achieves the highest average task accuracy of 0.830, followed by the LLM controller at 0.802, LLM+Vote at 0.798, and the MLP controller at 0.771. For reference, the reproduced Full-Info condition has an average task accuracy of 0.699. Performance varies across models, with no single controller dominating in every case. Task and vote accuracy are generally closely aligned, indicating that improvements in the final collective decision typically coincide with broader agreement among the individual agents.

B.2 Generated Benchmark Results

Table 4 presents a different performance profile. Hidden pre-discussion obtains an average task accuracy of 0.254, while uncontrolled discussion increases it to 0.498. Interestingly, hidden post-discussion outperforms the full-information condition, which achieves only 0.366. This suggests that merely placing all evidence in one context does not guarantee that the model will identify and combine the relevant facts; interaction can help expose, repeat, and reconcile evidence that may otherwise be overlooked. Among the controlled methods, LLM+Vote achieves the highest average task accuracy at 0.668, followed closely by Rules at 0.654, LLM at 0.615, and MLP at 0.507. Relative to uncontrolled discussion, LLM+Vote improves task accuracy by 0.170, whereas the MLP improves it by only 0.009. Adaptive termination is therefore especially useful on this benchmark, where continued discussion may reinforce an incorrect answer after the relevant evidence has already been surfaced.

Performance on the generated benchmark also depends strongly on the capability of the participating model. The smallest Qwen3 models receive limited benefit from communication control, whereas Qwen2.5-32B and Llama-3.3-70B exceed 0.88 task accuracy under the strongest controllers. The MLP controller is less consistent under this setting, performing substantially below Rules and the LLM-based controllers for models such as Mistral-24B and Qwen3-32B. This pattern suggests that the fixed learned mapping from collective-state features to actions is more sensitive to changes in benchmark structure or discussion dynamics, while an LLM controller can adapt its action choice using the current transcript and beliefs.

Comparison across benchmarks. Across the 12 models shared by the non-conformal HiddenBench evaluation in Table 1 and the generated benchmark in Table 4, controller performance is generally lower on the generated benchmark. Relative to HiddenBench, average task accuracy decreases by 0.160 points for Rules, 0.261 points for MLP, and 0.170 points for LLM, but by only 0.115 points for LLM+Vote. Thus, LLM+Vote exhibits the smallest cross-benchmark degradation, while the MLP exhibits the largest. Together, these results indicate that explicit communication control consistently improves hidden-profile reasoning, but the most effective controller depends on the discussion distribution: simple state-dependent rules are particularly effective on Hidden-

Bench, whereas transcript-aware control with adaptive voting transfers more robustly to the generated benchmark.

B.3 Generalization to Held-Out Tasks

The standard MLP controller is trained using exhaustive action outcomes collected from all benchmark tasks. It therefore measures how accurately a lightweight learned controller can imitate the one-step action oracle within the task distribution used to generate its supervision. However, because the same tasks contribute to both controller training and downstream evaluation, this condition should be interpreted primarily as an in-domain oracle-imitation diagnostic rather than as a strict test of task-level generalization.

We therefore construct a separate held-out evaluation. HiddenBench is partitioned into a five-task training subset and a disjoint 60-task test set. For each training task, the exhaustive action explorer follows the current discussion trajectory and, at every reached state, executes each of the four available communication actions: Challenge, Clarify, SeekEvidence, and Route.

The resulting training examples pair the six-dimensional collective-state representation with a four-dimensional vector containing the observed reward of every communication action. The MLP is trained only on examples generated from the five training tasks. Its feature-normalization statistics, reward-normalization statistics, and network parameters are therefore estimated exclusively from the training split. After training, the controller is frozen and evaluated on the remaining 60 tasks without any additional fitting or access to their counterfactual action outcomes. At each test-time discussion state, the MLP predicts the expected immediate reward of the four actions and executes the action with the highest predicted value. Agent prompting, speaker routing, belief estimation, and final voting remain unchanged.

This evaluation measures whether the mapping learned from abstract collective-state variables to communication actions transfers across decision problems. The controller does not observe task identifiers or raw private clues directly; it acts on quantities such as group-belief entropy, inter-agent disagreement, message redundancy, communication cost, evidence gain, and premature consensus. Consequently, successful transfer would indicate that these variables capture recurring coordination conditions rather than merely memorizing the content or trajectory of individual benchmark tasks.

Across the 13 models for which held-out results are available, the held-out MLP obtains an average task accuracy of **0.73**. The corresponding all-task MLP average is **0.77**. Thus, withholding the evaluation tasks during controller training is associated with a descriptive decrease of approximately 0.04. The relatively small aggregate gap suggests that a substantial portion of the learned action-selection behavior transfers to discussion trajectories that were not observed during training.

Transfer is particularly strong for medium and large language models. 8 of the 13 evaluated models achieve at least 0.83 held-out task accuracy. Qwen3-32B, Qwen3-235B, Llama-3.3-70B, and Mistral-24B each reach 0.86 task accuracy, while Qwen3-8B, Qwen3-14B, Qwen2.5-32B, and Gemma-3-27B reach 0.83. These results indicate that the controller trained on only a small subset of tasks can still select useful communication actions when paired with agents capable of producing stable beliefs and informative responses.

The largest degradations relative to the all-task MLP occur for Qwen3-1.7B and Phi-4, whose task accuracies decrease by 0.17 and 0.16 respectively. Qwen3-0.6B also decreases by 0.09 points. This pattern suggests that held-out transfer is less reliable when the participating agents produce noisier belief distributions or when their discussion states differ substantially from those represented in the controller-training trajectories. Because the MLP observes only the collective-state summary, changes in the quality or calibration of agent beliefs can produce a state distribution that is difficult to interpret using a controller trained from a small number of tasks.

Several models show comparable or slightly higher accuracy under the held-out controller. For example, held-out task accuracy increases by 0.06 points for Qwen3-14B, 0.07 points for Llama-3.1-8B, and 0.03 points for both Qwen3-32B and Mistral-24B. These increases should not be interpreted as evidence that using less training data improves the controller. The two conditions are evaluated over different task sets, and the deliberations themselves involve stochastic language-model outputs. Small positive differences can therefore arise from variation in task difficulty, generated messages, belief estimates, or action trajectories.

Task and vote accuracy remain closely aligned under held-out evaluation. Their mean absolute difference across models is approximately 0.02 accuracy points, and for Qwen3-8B, Qwen3-14B, and Llama-3.3-70B the two values are identical. This agreement indicates that held-out performance is generally supported by the individual agents' final judgments rather than arising only from plurality aggregation or confidence-based tie-breaking. Larger discrepancies for a few models nevertheless show that communication control can affect both the correctness of the group decision and the distribution of support among agents.

Overall, the held-out results provide evidence that the controller learns reusable coordination behavior from the collective-state representation. The performance decrease relative to the in-domain MLP is modest on average, and high accuracy is retained across most medium and large models despite training on only five tasks. At the same time, the larger losses for some weaker or behaviorally distinct models show that transfer is not uniform. The learned controller generalizes most reliably when the test-time discussion dynamics resemble the state distributions encountered during training.

For a strictly controlled estimate of the generalization gap, the all-task MLP should additionally be evaluated on the same 60 held-out tasks. The current comparison uses the 65-task result for the standard MLP and the 60-task result for the held-out MLP; consequently, the reported 0.03 and 0.04 point differences combine controller-training effects with a small difference in the evaluation task set. Evaluating both checkpoints on the identical 60-task test split would isolate the effect of withholding task-level training supervision.

C HETEROGENEOUS-AGENT ANALYSIS.

Table 5 evaluates whether QUORUM remains effective when the participating agents use different underlying language models. Overall, heterogeneous groups can achieve performance comparable to strong homogeneous groups, but their results are substantially more sensitive to model composition, agent assignment, and controller choice. Among the heterogeneous four-agent configurations, the rule-based controller reaches its highest task accuracy of 0.966 on both Position Rot-4 and Ladder L2, while the LLM and LLM+Vote controllers reach 0.948 on Ladder L4. These results show that model heterogeneity does not inherently prevent successful coordination when the group contains sufficiently capable and complementary agents.

The position-rotation experiments reveal that assigning the same four models to different agent positions can materially alter performance. For example, rule-based accuracy varies from 0.879 in the documented assignment to 0.966 in Position Rot-4, while LLM+Vote varies from 0.828 to 0.931 across the rotations. Thus, performance depends not only on which models are present, but also on which private evidence each model receives. Using the stronger Qwen3-32B controller further improves the documented heterogeneous group, raising MLP accuracy from 0.776 to 0.931, LLM accuracy from 0.845 to 0.897, and LLM+Vote accuracy from 0.879 to 0.931. This indicates that controller capability can compensate for some of the variability introduced by heterogeneous participants.

Group composition also has a clear effect. Configurations composed primarily of capable models, such as Family-Qwen, Spread-Low, and Ladder L4, remain strong across several controllers. In contrast, groups containing several small models perform substantially worse: Star and Ladder L1 obtain only 0.586 task accuracy under LLM+Vote. The Weak-Link configuration shows that introducing a single weak agent does not necessarily cause failure, particularly for Rules, but can reduce the effectiveness of adaptive voting. Among homogeneous groups, Qwen3-32B with LLM+Vote achieves the highest overall task accuracy of 0.983, while the other strong homogeneous groups also remain consistently competitive. Taken together, the results suggest that QUORUM generalizes to heterogeneous teams, but reliable performance depends on both the capability distribution within the group and the controller's ability to route communication across differently capable agents.

D ABLATIONS

State-feature ablation. Table 6 evaluates how different subsets of the collective-state representation affect the learned MLP controller. Using the complete feature set achieves the best overall trade-off, with 82.22% task success and the lowest final objective value of 0.0239. The information-

Table 5: Performance of heterogeneous and homogeneous multi-agent groups on the corresponding HiddenBench agent-count partitions. Four-agent configurations are evaluated on the 58-task four-agent partition, while Documented-3Agent is evaluated on the seven-task three-agent partition. Each controller entry reports *task accuracy / vote accuracy*. The orchestration controller is Qwen3-8B unless marked with †, where Qwen3-32B is used. Model abbreviations are: Q0.6, Q1.7, Q4, Q8, Q14, and Q32 for Qwen3 at the corresponding parameter scale; Q2.5-32 for Qwen2.5-32B; P4 for Phi-4; M24 for Mistral-24B; G27 for Gemma-3-27B; L8 for Llama-3.1-8B; and L70 for Llama-3.3-70B.

Configuration	Agent Models	Rules	MLP	LLM	LLM+Vote
Heterogeneous groups					
Documented-4Agent	Q8, Q32, P4, M24	.879/.888	.776/.780	.845/.866	.879/.888
Documented-3Agent	Q8, Q32, P4	.714/.714	.571/.571	.714/.667	.714/.667
Position Rot-2	Q32, P4, M24, Q8	.897/.897	.897/.897	.828/.853	.931/.909
Position Rot-3	P4, M24, Q8, Q32	.914/.931	.897/.888	.879/.875	.897/.914
Position Rot-4	M24, Q8, Q32, P4	.966/.961	.879/.879	.897/.897	.828/.836
Controller-Strong†	Q8, Q32, P4, M24	.879/.897	.931/.892	.897/.901	.931/.927
Family-Cross	L8, P4, M24, G27	.914/.897	.776/.772	.845/.815	.914/.871
Family-Qwen	Q8, Q14, Q32, Q2.5-32	.897/.892	.862/.853	.914/.914	.914/.905
Spread-Low	P4, M24, G27, Q2.5-32	.948/.940	.793/.797	.845/.853	.948/.927
Spread-High	Q0.6, Q4, Q14, Q32	.793/.746	.845/.810	.741/.724	.724/.685
Weak-Link	Q0.6, Q32, P4, M24	.897/.853	.828/.802	.845/.819	.759/.759
Star	Q32, Q0.6, Q1.7, Q4	.759/.694	.655/.634	.621/.621	.586/.556
Ladder L1	Q0.6, Q1.7, Q4, Q8	.776/.746	.621/.616	.724/.711	.586/.603
Ladder L2	Q4, Q8, Q14, P4	.966/.966	.897/.888	.845/.853	.879/.871
Ladder L3	Q14, P4, M24, Q32	.897/.918	.828/.828	.793/.780	.879/.858
Ladder L4	Q32, Q2.5-32, G27, L70	.931/.927	.897/.892	.948/.948	.948/.935
Homogeneous groups					
Qwen3-32B	4×Q32	.948/.927	.862/.862	.897/.905	.983/.957
Mistral-24B	4×M24	.931/.927	.879/.879	.914/.905	.948/.948
Phi-4	4×P4	.914/.918	.879/.879	.931/.931	.879/.871
Qwen3-8B	4×Q8	.948/.944	.810/.841	.914/.922	.810/.819

only controller matches this success rate, but incurs slightly higher communication cost and a substantially worse final objective, indicating that the remaining state variables improve the quality and efficiency of the learned policy even when they do not change the final task accuracy.

Removing information-related features reduces performance. Both the belief-plus-information and belief-plus-cost variants achieve 77.78% success, while the belief-only controller performs worst at 75.56%. Although the belief-only representation produces the lowest communication cost, this reduction is accompanied by a 6.66 percentage-point drop in success relative to the full model. Among the reduced representations, belief plus cost yields a lower final objective than belief plus information, suggesting that explicit cost awareness helps the controller avoid inefficient deliberation. Overall, the results show that belief statistics alone are insufficient, while combining belief, information, and cost-related signals produces the most balanced controller.

Table 6: State-feature ablation for Qwen3-8B. Each row retrains the MLP using only the retained feature groups. Higher success is better; lower communication cost and final objective are better.

Features retained	Success ↑	Communication cost ↓	Final objective ↓
All features	**82.22%**	0.5228	**0.0239**
Information only	**82.22%**	0.5254	0.2277
Belief + information	77.78%	0.5285	0.3509
Belief + cost	77.78%	0.5253	0.1582
Belief only	75.56%	**0.4625**	0.2212

Table 7: Action ablation study on HiddenBench. Each row removes one communication action from the full controller. Entries report task accuracy (%), with the change relative to the corresponding full action set shown in parentheses. Negative values indicate performance degradation.

Configuration	Rules	MLP	LLM	LLM+Vote	Average
Full	92.3	87.7	89.2	83.1	**88.1**
– Challenge	90.8 (−1.5)	86.2 (−1.5)	87.7 (−1.5)	83.1 (0.0)	86.9 (−1.2)
– Clarify	93.8 (+1.5)	87.7 (0.0)	89.2 (0.0)	81.5 (−1.5)	88.1 (0.0)
– New Evidence	92.3 (0.0)	92.3 (+4.6)	81.5 (−7.7)	70.8 (−12.3)	84.2 (−3.8)
– Routing	87.7 (−4.6)	89.2 (+1.5)	87.7 (−1.5)	69.2 (−13.8)	**83.5 (**−4.6**)**
– Voting	92.3 (0.0)	NA	NA	83.1 (0.0)	87.7 (0.0)

Action ablation. Table 7 evaluates the contribution of each communication action by removing it while keeping the remaining controller unchanged. Across controllers, routing and new-evidence solicitation are the most consequential components. Removing routing produces the largest average drop, reducing task accuracy by 4.6 percentage points, while removing SEEKEVIDENCE decreases average accuracy by 3.8 points. These effects are especially pronounced for the LLM+Vote controller, where removing routing and new-evidence solicitation reduces accuracy by 13.8 and 12.3 points, respectively. This suggests that adaptive termination is particularly dependent on directing the discussion toward underrepresented agents and unshared evidence before voting occurs.

The effects vary across controller types. The rule-based controller depends most strongly on routing, whereas the standard LLM controller is most affected by removing SEEKEVIDENCE. Removing CHALLENGE causes a small but consistent degradation for Rules, MLP, and LLM, indicating that explicitly testing the leading hypothesis provides a modest benefit. In contrast, removing CLARIFY has no average effect, and removing voting leaves the Rules and LLM+Vote results unchanged in this ablation. Several removals improve performance—most notably removing SEEKEVIDENCE from the MLP controller—showing that individual actions are not uniformly useful and may interact with the controller's selection policy. Overall, the results indicate that the full action set is most valuable for providing complementary interventions, with routing and new-evidence acquisition contributing the strongest aggregate gains.

Speaker-routing ablation. Table 8 isolates the effect of speaker selection by holding the controller-selected communication action fixed. QUORUM's action-conditional router achieves the lowest mean objective change, $\Delta J = -0.0079$, indicating that its selected speakers produce the strongest average immediate improvement in the discussion state. Round-robin is the closest alternative, with $\Delta J = -0.0017$, while random routing slightly worsens the objective on average ($\Delta J = 0.0145$).

The normalized-regret results are less decisive. Round-robin obtains the lowest regret at 0.4839, followed closely by action-conditional routing at 0.4868 and random routing at 0.4930. Thus, although the proposed router performs best under the mean objective-change metric, its advantage over simple round-robin and random selection is small, and it does not achieve the lowest normalized regret. In contrast, selecting the most-disagreeing or most-spoken agent performs substantially worse under both metrics, suggesting that disagreement or participation frequency alone is insufficient for identifying the most useful speaker. Overall, the results support action-aware routing over these stronger heuristic alternatives, but provide only limited evidence of a clear advantage over round-robin routing.

Conformal-sampling ablation. Table 9 compares the four controller variants over 400 trials with Qwen3-8B. The LLM and MLP controllers achieve the highest mean success rate, both reaching 82.75%, despite using different action-selection mechanisms. Their discussion lengths are also similar, averaging 9.54 and 9.87 rounds per trial, respectively. This indicates that the lightweight learned controller can match the success of the LLM controller, although it requires slightly more communication.

Table 8: Counterfactual speaker-routing ablation with Qwen3-8B. The controller-selected action is held fixed and only the speaker-selection policy is changed. Lower objective change is better, while lower normalized regret indicates routing decisions closer to the counterfactual optimum.

Routing policy	**Mean $\Delta J \downarrow$**	**Normalized regret $\downarrow$**
QUORUM (action-conditional)	**-0.0079**	0.4868
Random	0.0145	0.4930
Round-robin	-0.0017	**0.4839**
Most-disagreeing	0.1205	0.5735
Most-spoken	0.2815	0.7184

The rule-based controller terminates substantially earlier, after only 3.24 rounds on average, but its success decreases to 64.50%. The LLM+Vote variant also conducts shorter discussions, averaging 5.43 rounds, yet attains the lowest success rate of 63.00%. These results reveal a clear trade-off between communication length and task performance: controllers that allow longer deliberation are substantially more successful, whereas aggressive or adaptive termination can stop the discussion before complementary private evidence has been sufficiently integrated. Overall, conformal sampling preserves comparable performance for the LLM and MLP controllers, while the weaker results of the early-terminating variants suggest that calibrated action selection alone does not compensate for premature voting.

Table 9: Comparison of controller methods over 400 trials. Using Qwen3-8B model

Method	**Mean Success (%)**	**Mean Rounds/Trial**
LLM	**82.75**	9.54
MLP	**82.75**	9.87
Rules	64.50	3.24
LLM+Vote	63.00	5.43

E COMPLETE PROMPT TEMPLATES

This section reports the complete prompt templates used in our experiments. Text enclosed in angle brackets denotes a runtime placeholder populated separately for each task or deliberation state.

Prompt for initial belief estimation

```
You are <AGENT_NAME>, one participant in a multi-agent group decision task.

Task description:
<TASK_DESCRIPTION>

Possible answers:
<POSSIBLE_ANSWERS>

Shared information visible to everyone:
- <SHARED_INFORMATION_ITEM_1>
- <SHARED_INFORMATION_ITEM_2>
- ...

Information visible only to you:
- <PRIVATE_INFORMATION_ITEM_1>
- <PRIVATE_INFORMATION_ITEM_2>
- ...

Rules:
- You may use the information visible only to you.
- Share decision-relevant evidence when it helps the group.
- Do not invent facts.
- Be concise.

Public transcript so far:
(empty transcript)

Give your current belief over the possible answers. Return JSON only:
{
  "belief": {"OPTION": 0.5},
  "best_answer": "one option exactly as written",
  "confidence": 0.0,
  "one_sentence_evidence": "short evidence summary",
  "needs_more_information": true
}

Include every possible answer exactly and make the probabilities sum to 1.
Possible answers: <POSSIBLE_ANSWERS>
```

Prompt for speaker response generation

```
You are <AGENT_NAME>, one participant in a multi-agent group decision task.

Task description:
<TASK_DESCRIPTION>

Possible answers:
<POSSIBLE_ANSWERS>

Shared information visible to everyone:
- <SHARED_INFORMATION_ITEM_1>
- <SHARED_INFORMATION_ITEM_2>
- ...

Information visible only to you:
- <PRIVATE_INFORMATION_ITEM_1>
- <PRIVATE_INFORMATION_ITEM_2>
- ...

Rules:
- You may use the information visible only to you.
- Share decision-relevant evidence when it helps the group.
- Do not invent facts.
- Be concise.

Public transcript so far:
<PUBLIC_TRANSCRIPT>
```

Prompt for speaker response generation (continued)

```
Controller instruction:
<ACTION_SPECIFIC_INSTRUCTION>

Write your next message to the group. Share relevant evidence, clearly state
if it rules out an option, communicate naturally, and do not refer to it as
private information.
```

Controller instruction for CHALLENGE

```
The current leading option is <LEADING_OPTION>. State any evidence that could
support or challenge <LEADING_OPTION>. If your information rules out another
option, say so clearly.
```

Controller instruction for CLARIFY

```
Your belief differs from the group. Explain the evidence behind your view,
especially any fact that rules out an option.
```

Controller instruction for SEEKEVIDENCE

```
Share one new decision-relevant fact. Focus on facts that eliminate or support an option.
    Avoid repeating the transcript.
```

Controller instruction for ROUTE

```
Contribute one concise piece of decision-relevant evidence. If possible,
explain which option it supports or rules out.
```

Prompt for LLM controller action selection

```
You are the QUORUM controller. Choose the next controller action for a
multi-agent decision task. You must choose exactly one existing action type; do not
choose a speaker and do not write the agent's message.

Available actions:
- challenge: ask an agent to test the current leading option and surface contrary evidence
    .
- clarify: ask the agent whose belief differs most from the group to explain why.
- SeekEvidence: ask the least-heard agent for one new decision-relevant fact.
- route: ask the least-heard agent for concise evidence that supports or rules out an
    option.

Task:
<TASK_DESCRIPTION>

Possible answers:
<POSSIBLE_ANSWERS>

Public transcript so far:
<PUBLIC_TRANSCRIPT>

Current state:
<COLLECTIVE_STATE_JSON>

Current round:
<CURRENT_ROUND>

Current agent beliefs:
<AGENT_BELIEFS_JSON>

Times each agent has spoken:
<SPEAKER_COUNTS_JSON>
```

Prompt for LLM controller action selection (continued)

```
Return JSON only:
{
  "action": "one of: challenge, clarify, SeekEvidence, route",
  "reason": "brief reason for choosing this action",
  "expected_effect": "brief description of what this should improve"
}
```

Prompt for LLM controller with adaptive voting

```
You are the QUORUM controller. Choose the next controller action for a
multi-agent decision task. You must choose exactly one existing action type; do not
choose a speaker and do not write the agent's message.

Available actions:
- challenge: ask an agent to test the current leading option and surface contrary evidence
    .
- clarify: ask the agent whose belief differs most from the group to explain why.
- SeekEvidence: ask the least-heard agent for one new decision-relevant fact.
- route: ask the least-heard agent for concise evidence that supports or rules out an
    option.
- vote: end the discussion now and collect every agent's final vote.

Task:
<TASK_DESCRIPTION>

Possible answers:
<POSSIBLE_ANSWERS>

Public transcript so far:
<PUBLIC_TRANSCRIPT>

Current state:
<COLLECTIVE_STATE_JSON>

Current round:
<CURRENT_ROUND>

Current agent beliefs:
<AGENT_BELIEFS_JSON>

Times each agent has spoken:
<SPEAKER_COUNTS_JSON>

Return JSON only:
{
  "action": "one of: challenge, clarify, SeekEvidence, route, vote",
  "reason": "brief reason for choosing this action",
  "expected_effect": "brief description of what this should improve"
}
```

Prompt for message-level evidence and redundancy evaluation

```
Analyze a message in a multi-agent decision conversation.

Options: <POSSIBLE_ANSWERS>
Transcript before this message:
<PUBLIC_TRANSCRIPT_BEFORE_MESSAGE>

New message:
<NEW_MESSAGE>

Return JSON only:
{
  "new_evidence_score": 0.0,
  "redundancy_score": 0.0,
  "supports_options": ["option names exactly"],
  "attacks_or_rules_out_options": ["option names exactly"],
  "is_clarification": false,
  "short_explanation": "one sentence"
}

Use scores between 0 and 1. New evidence adds task-relevant information; redundancy
```

Prompt for message-level evidence and redundancy evaluation (continued)

```
means the message mostly repeats the earlier transcript.
```

E.1 SYNTHETIC BENCHMARK GENERATION PROMPTS

The additional travel-planning benchmark was generated using structured prompting followed by deterministic validation, automatic repair, and LLM-based semantic quality review. Text enclosed in angle brackets denotes a value populated at runtime.

System prompt for synthetic task generation

```
You design adversarial, realistic multi-agent travel-planning
benchmarks. Return exactly one valid JSON object and no markdown. Never reveal or hint at
     the
correct answer outside the declared correct_answer field. Use only the schema and
     operators
specified by the user.
```

User prompt for synthetic task generation

```
Create one HiddenBench-style group travel decision task.

Variation seed: <VARIATION_SEED>
Dataset position: <TASK_INDEX>
Avoid duplicating these accepted task signatures: <ACCEPTED_TASK_SIGNATURES>
Use exactly <NUMBER_OF_TRAVELERS> travelers.
Make option <TARGET_CORRECT_OPTION> the declared correct answer.
Use HiddenBench-style weighted preference aggregation:
- satisfied must = +2
- satisfied prefer = +1
- matched avoid = -1
- matched reject = -2
The declared answer must be the unique highest-scoring option. It may violate a strong
     preference
if it remains the best group compromise. Distractors do not need artificial, unique
     violation
patterns.

The task must:
- involve exactly 3 or 4 travelers with plausible roles and a concrete group-trip context;
- expose exactly four fully specified candidate plans (A, B, C, D) in public information;
- distribute private structured preferences across travelers using all tiers: must, prefer
     ,
  avoid, reject;
- make the declared answer uniquely best under the validation profile above;
- require evidence from multiple travelers to identify the answer;
- remain solvable by literal comparison only: no outside travel facts, arithmetic
     ambiguity,
  subjective interpretation, compromise, or unstated assumptions;
- make every candidate feasible in ordinary real-world terms;
- give every traveler at least one preference that matches some but not all candidates;
- avoid unsafe, discriminatory, medically implausible, or internally contradictory content
     .

Tier semantics:
- must: matching is a strong positive preference worth +2.
- reject: matching is a strong negative preference worth -2.
- prefer: matching adds +1 utility.
- avoid: matching subtracts 1 utility.

Allowed operators: eq, neq, in, not_in, contains, contains_all, contains_any, excludes,
     lte, gte.
For contains/excludes operators, candidate attributes must be arrays. `in` means a scalar
     or any
array item belongs to the preference value list; `not_in` means no scalar/array item
     belongs to
that list. For lte/gte, attributes must be numbers. Every preference must reference an
     attribute
present in every option.
```

User prompt for synthetic task generation (continued)

```
Every option attributes object must include at least these exact keys:
transport_mode, departure_window, duration_days, price_per_person_usd, hotel_category,
room_arrangement, step_free, daily_active_hours, max_walk_km_per_day, meal_options,
    activities.
You may add a small number of concrete keys such as arrival_time, layovers, or
    neighborhood.
All four options must use the same keys and comparable units.

Return this exact shape:
{
  "name": "short_unique_snake_case_name",
  "description": "shared scenario, dates or season, origin, destination, group
     relationship",
  "public_information": [
    "2-5 public facts that do not encode any preference or answer"
  ],
  "options": [
    {
      "id": "A",
      "title": "concise plan label",
      "attributes": {
        "transport_mode": "train",
        "departure_window": "morning",
        "duration_days": 4,
        "price_per_person_usd": 900,
        "hotel_category": "midrange",
        "room_arrangement": "two twin rooms",
        "step_free": true,
        "daily_active_hours": 7,
        "max_walk_km_per_day": 5,
        "meal_options": ["vegetarian", "seafood"],
        "activities": ["museum", "market"]
      }
    }
  ],
  "agents": [
    {
      "id": "traveler_1",
      "role": "realistic role in this group",
      "context": "one sentence explaining why the preferences are plausible",
      "preferences": [
        {
          "id": "t1_must_1",
          "tier": "must",
          "attribute": "step_free",
          "operator": "eq",
          "value": true,
          "statement": "A natural first-person statement with clear non-negotiable force."
        },
        {
          "id": "t1_reject_1",
          "tier": "reject",
          "attribute": "transport_mode",
          "operator": "eq",
          "value": "flight",
          "statement": "A natural first-person statement clearly refusing flights."
        }
      ]
    }
  ],
  "correct_answer": "one of A/B/C/D"
}

Use 2-4 preferences per traveler. Individual travelers may have only hard or only soft
     items, but
the complete task must use each of must, prefer, avoid, and reject. NEVER omit the "tier"
     key:
copy its lowercase value from the preference id. Before responding, evaluate every
     predicate
against A/B/C/D, calculate the aggregate score for all options, and verify that each
     traveler has
at least one predicate matching 1-3 options. It is acceptable for an individual traveler
     to rank
the declared answer first; at least two profiles must still contribute non-constant
     evidence to
the aggregate decision. The statement must faithfully express its predicate and tier. Do
     not add
```

User prompt for synthetic task generation (continued)

```
evaluation, rationale, violation summaries, or extra top-level fields.
```

Failure-conditioned regeneration instruction

```
A previous blueprint was abandoned after failed repairs. Create a genuinely new blueprint
    and
avoid these failures:
<PRIOR_VALIDATION_ISSUES>
```

E.1.1 Deterministic Validation and Repair

Generated tasks are first evaluated by a deterministic validator. Under the HiddenBench validation profile, a matched `must` contributes $+2$, a matched `prefer` contributes $+1$, a matched `avoid` contributes -1, and a matched `reject` contributes -2. The declared answer must be the unique highest-scoring candidate. Instances that fail validation are repaired using the following prompt.

Prompt for repairing a rejected synthetic task

```
Repair the rejected HiddenBench travel blueprint below in place.

Dataset position: <TASK_INDEX>
Do not duplicate accepted signatures: <ACCEPTED_TASK_SIGNATURES>

Change the minimum option attributes or preferences needed to fix every validator issue.
    Preserve
the scenario, traveler roles, four option ids, and all already-valid content. Return the
    complete
repaired blueprint as one JSON object, not a patch.

For an "expected exactly one hard-feasible option, found []" error in strict mode, repair
    in this order:
1. Read the detailed violation list for the declared correct_answer.
2. Change its public attributes, or the conflicting preference predicate and matching
    statement,
   until it violates zero MUST/REJECT preferences.
3. Re-evaluate every hard predicate literally against all four options.
4. Ensure each other option retains a different non-empty violation set.
Do not merely repeat the rejected blueprint.

Required invariants after repair:
- exactly four comparable plans and exactly 3 or 4 travelers;
- every preference contains id, tier, attribute, operator, value, and statement;
- tier is exactly one lowercase value: must, prefer, avoid, or reject;
- each traveler has at least two preferences and at least one predicate that matches some
    but not
  all plans;
- the task collectively uses all four tiers;
- using must=+2, prefer=+1, avoid=-1, reject=-2, correct_answer is the unique
  highest-scoring option;
- an individual traveler may rank correct_answer first, but at least two profiles must
    contribute
  non-constant option scores;
- plans may violate MUST or REJECT items; do not force a fully hard-feasible plan;
- every preference statement exactly matches its structured predicate.

Validator issues:
<VALIDATOR_ISSUES_JSON>

Rejected blueprint:
<REJECTED_BLUEPRINT_JSON>
```

System prompt for synthetic-task quality review

```
You are a strict dataset reviewer. Check a travel hidden-profile
benchmark for semantic realism and consistency. Return exactly one valid JSON object with
    keys
```

System prompt for synthetic-task quality review (continued)

```
accepted (boolean), issues (array of concise strings), and checks (object). Do not repair
     the task.
```

User prompt for synthetic-task quality review

```
Review this candidate benchmark after deterministic validation.

Reject it if any preference statement disagrees with its structured predicate; a plan
     attribute
is vague, incomparable, or implausible; a private fact leaks into public text; the
     declared answer
is hinted by names/order/wording; the scenario needs outside knowledge; constraints are
     contrived
instead of realistic; an entire traveler profile has no decision-relevant information; two
      options
are semantically indistinguishable; or the task could be solved from public information or
      one
profile alone. A realistic preference shared by all plans is allowed when that same
     traveler has
another preference that distinguishes plans.

Accept only if the exact four plans, realistic profiles, a unique weighted-utility answer
     under
must=+2, prefer=+1, avoid=-1, reject=-2, and multi-agent dependency are clear to a careful
      human.
Do not require every strong preference to be satisfied in HiddenBench mode; realistic
     compromise
plans are allowed.

The deterministic audit is authoritative for predicate evaluation, scoring, winner
     uniqueness,
and information dependency. Do NOT reject a task merely because any option fails a MUST,
     matches a
REJECT, or because one traveler locally ranks the final answer first. Do not reinterpret
     operators
or report option-level score violations already handled by the audit. Review only semantic
fidelity of statements to predicates, realism, ambiguity, leakage, and unsupported outside
      facts.

Deterministic audit:
<DETERMINISTIC_AUDIT_JSON>

Blueprint:
<CANDIDATE_BLUEPRINT_JSON>
```