Robust Multi-Agent LLMs under Byzantine Faults

Haejoon Lee [1 *] **Vincent-Daniel Yun** [2 *] **Hyeonho Oh** [2 *] **Dimitra Panagou** [1] **Sai Praneeth Karimireddy** [2]

Abstract

Large language model (LLM) agents increasingly collaborate over peer-to-peer networks to improve their reliability. However, these same interactions can also become a source of vulnerability, as unreliable or Byzantine agents may sway neighboring agents toward incorrect conclusions and degrade overall system performance. Existing methods rely on leader-based coordination or self-reported confidence, both of which are susceptible to adversarial manipulation. We study decentralized LLM multi-agent systems (LLM-MAS) and propose Self-Anchored Consensus (SAC), a fully decentralized iterative filter-and-refine protocol in which agents iteratively exchange responses, locally evaluate and filter unreliable messages, and refine their own outputs. We present $(F+1)$-robustness conditions for the communication graph that ensure honest agents preserve and propagate reliable information despite Byzantine influence. Experiments on mathematical and commonsense reasoning benchmarks show that SAC effectively suppresses Byzantine influence and consistently improves performance across diverse communication topologies, whereas prior methods degrade under adversarial conditions. We will release the implementation upon acceptance.

1. Introduction

Large language models (LLMs) are increasingly deployed as communicating agents in multi-agent systems (MAS), where multiple agents exchange intermediate responses, critique one another, and converge on a collective answer (Guo et al., 2024; Du et al., 2024). We consider a setting in which LLM agents are connected over a peer-to-peer communication network and aim to solve tasks with objectively verifiable correct answers through consensus.

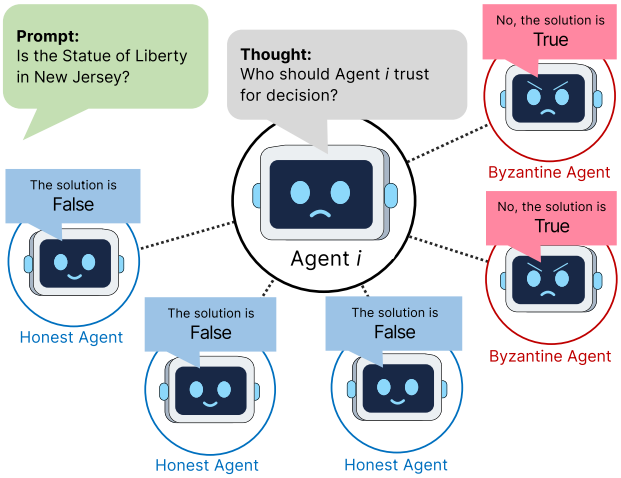

Figure 1. Decision ambiguity in the presence of Byzantine agents. An agent must decide which responses to trust despite adversarial and indistinguishable neighbor messages.

As these systems move into real-world deployment, a central challenge is *robustness* to faulty and adversarial agents. This concern parallels broader efforts to ensure trustworthy and reliable AI in high-stakes domains (Yun et al., 2024), where uncertainty and unreliability of model outputs directly affect downstream consequences. Individual LLM agents *make mistakes*: they hallucinate, miscalculate, and mis-apply safety judgments, and some may *behave adversarially*, injecting strategically crafted responses. Since each agent cannot locally distinguish trustworthy messages from Byzantine messages (shown in Fig. 1), such behavior can contaminate otherwise reliable agents. The practical question is therefore whether the system can *contain* the influence of unreliable agents so that reliable agents preserve their base capability.

A natural analogue arises through the classical *Byzantine-resilient consensus*, where a subset of nodes may behave arbitrarily while honest nodes must still reach correct agreement (Lamport et al., 1982; LeBlanc et al., 2013; Su & Vaidya, 2021). This framework captures the failure modes of an LLM agent that shares an incorrect response, whether unintentional (hallucinations, reasoning failures) or intentional (adversarial manipulation, jailbreaking); from the perspec-

[1]Department of Robotics, University of Michigan, Ann Arbor, USA [2]Thomas Lord Department of Computer Science, University of Southern California, USA. Correspondence to: Sai Praneeth Karimireddy <karimire@usc.edu>.

Published at the Second Workshop on Agents in the Wild: Safety, Security, and Beyond (AIWILD) at ICML 2026. Copyright 2026 by the author(s).

tive of other agents, such deviations are indistinguishable from Byzantine faults. Viewing LLM-MAS through this lens makes two structural questions unavoidable: (i) how should agents determine which neighbors' responses to trust in each round, and (ii) how should the communication graph be designed to limit the influence of Byzantine agents.

A recent approach, CP-WBFT (Zheng et al., 2026), has each agent attach a self-reported confidence score and aggregates neighbors' outputs by upweighting high-confidence agents. Two design choices limit its usage. First, "Byzantine" agents are still assumed to report honest confidence scores, departing from the standard Byzantine model in which agents may inject falsified confidence. Second, the communication graph is chosen ad hoc from a small set of generic topologies (e.g., chain, star, tree), without a principled connection to fault-tolerance guarantees, so performance varies significantly across topologies, with some yielding *worse* performance than no collaboration at all (Zheng et al., 2026).

We address these limitations by drawing on classical Byzantine-resilient consensus. The Mean-Subsequence-Reduced (MSR) family of algorithms (Kieckhafer & Azadmanesh, 1994) performs a *trimmed mean* over neighbor messages, while LeBlanc et al. (2013) establish r-robustness as a topological condition ensuring consensus under Byzantine faults. To our knowledge, these tools have not been brought to LLM-MAS. We do so here: we replace sender-reported confidence with receiver-side evaluations computed independently by each agent, and introduce an MSR-inspired filter-and-refine mechanism that selectively aggregates neighbor messages via re-prompting. We then design and analyze a communication network under which honest agents remain robust to Byzantine influence.

Contributions. Our contributions are twofold. First, we introduce Self-Anchored Consensus (SAC), an MSR-inspired algorithm that ensures honest LLM agents remain robust to a bounded number of Byzantine agents while iteratively refining their responses via receiver-side confidence evaluation and a filter-and-refine update tailored to natural-language interactions. Second, we leverage r-robustness to establish graph-theoretic conditions under which non-Byzantine agents contain Byzantine influence under SAC. We further show through simulations that our method preserves honest-agent capability under Byzantine faults across varying communication topologies, whereas baseline approaches collapse catastrophically.

2. Related Works

LLM multi-agent systems. Multiple LLM instances coordinated as agents have been shown to outperform single-agent inference through debate (Du et al., 2024; Liang et al., 2024) and role-based frameworks (Wu et al., 2023; Hong et al., 2024; Guo et al., 2024). A growing body of work studies the reliability cost when such agents behave incorrectly (Tian et al., 2023; Yu et al., 2024; Zhang et al., 2024). Our work takes this failure mode as its starting point and asks what aggregation rule and graph structure are needed to contain it.

Byzantine-robust consensus in LLM-MAS. Three recent methods are closest to ours. CP-WBFT (Zheng et al., 2026) relies on self-reported confidence to select high-confidence responses, implicitly assuming Byzantine agents would honestly report their confidence. It also evaluates their method across the six hand-picked graphs, with wildly varying performance. Trusted MultiLLMN (Luo et al., 2025) uses a leader-based BFT protocol in which consecutive Byzantine leaders force expensive re-elections. DecentLLMs (Jo & Park, 2025) removes the leader via a geometric-mean filter over evaluator scores, which is brittle under high-variance scoring and requires a fully-connected evaluator-worker graph. Our method differs on two axes at once: confidence is computed on the *receiver* side, so Byzantine agents cannot manipulate how others evaluate their output; and our communication topology is grounded on mathematically-driven robustness properties that limit Byzantine influence on any general graph.

Classical Byzantine-resilient algorithms. The Byzantine agreement problem (Lamport et al., 1982; Dolev et al., 1986) has been extensively studied to ensure reliability under arbitrary node failures, inspiring a broad line of work on fault-tolerant distributed computation. In particular, the *Weighted Mean-Subsequence-Reduced* (W-MSR) algorithm, together with r- and (r, s)-robustness conditions (LeBlanc et al., 2013), provides consensus guarantees despite Byzantine agents. These notions have since been extended beyond consensus to distributed optimization (Sundaram & Gharesifard, 2019; Yuan & Ishii, 2025) and distributed learning (Xie et al., 2023; Ye et al., 2024), with applications robotics and smart grids (Lee & Panagou, 2025; Yuan & Ishii, 2025). To our knowledge, this work is the first to bring MSR-type algorithms and robustness conditions into LLM-MAS.

3. Problem Setup

3.1. LLM Multi-Agent System

We consider a multi-agent system of n LLM agents, indexed by $\mathcal{V} = \{1, \ldots, n\}$, collaboratively answering a query x drawn from a task distribution $\mathcal{D}$. Agents exchange messages over an undirected, time-invariant communication graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where an edge $(i, j) \in \mathcal{E}$ indicates that agents i and j can exchange their responses. The neighbor set of agent i is denoted by $\mathcal{N}_i = \{j \in \mathcal{V} \mid (i, j) \in \mathcal{E}\}$. We additionally consider a directed graph $\mathcal{G}' = (\mathcal{V}, \mathcal{E}')$, where

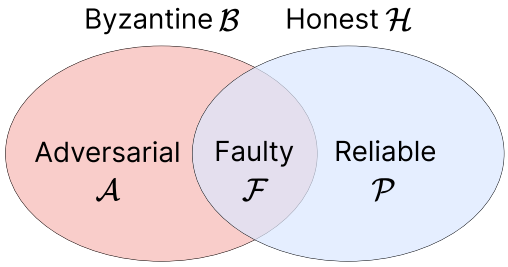

Figure 2. Venn diagram showing the partitioning of agents into reliable agents $\mathcal{P}$, faulty agents $\mathcal{F}$, and adversarial agents $\mathcal{A}$, as well as the resulting set of honest agents $\mathcal{H} = \mathcal{V} \setminus \mathcal{A}$.

a directed edge $(i, j) \in \mathcal{E}'$ indicates that agent j can receive responses from agent i. A directed graph contains a rooted out-branching if it has a node $p \in \mathcal{V}$ that can reach all other nodes. Let $\mathbb{Z}_{\geq 0}$ denote the set of non-negative integers.

At each time step $t \in \mathbb{Z}_{\geq 0}$, agents communicate with their neighbors. Each agent i is associated with an underlying LLM $\mathcal{M}_i$ and, given a query x, produces a response $r_i^{(t)} \in \mathcal{Y}$, where $\mathcal{Y}$ denotes the space of admissible responses (e.g., a numerical answer for mathematical reasoning or a binary safe/unsafe label for safety assessment). We assume that each query x admits a well-defined ground-truth answer $y^* \in \mathcal{Y}$, so that any response $r_i^{(t)}$ can be classified as correct ($r_i^{(t)} = y^*$) or incorrect ($r_i^{(t)} \neq y^*$) at each round t.

3.2. Threat Model

We consider the setting in which a subset of agents are Byzantine. We denote this set by $\mathcal{B} \subset \mathcal{V}$, and adapt the definition from LeBlanc et al. (2013) to LLM-MAS:

Definition 3.1 (Byzantine agent)**.** An agent $i \in \mathcal{B}$ is *Byzantine* if, in some round t, it deviates from the ideal behavior and provides unreliable information to its neighbors. This may occur in the following ways:

- *Faulty Byzantine:* the agent $b \in \mathcal{F} \subset \mathcal{B}$ follows the prescribed protocol, but produces incorrect or noisy responses due to internal errors, limited capability, or uncertainty.
- *Adversarial Byzantine:* the agent $b \in \mathcal{A} \subset \mathcal{B}$ does not follow the prescribed protocol by acting arbitrarily and may intentionally degrade system performance, e.g., by producing incorrect responses ($r_i^{(t)} \neq y^*$) or sending inconsistent information to different neighbors.

Note Byzantine agents include both faulty agents $\mathcal{F}$ and adversarial agents $\mathcal{A}$ such that $\mathcal{A} \cap \mathcal{F} = \emptyset$ and $\mathcal{A} \cup \mathcal{F} = \mathcal{B}$. These behaviors may arise from systematic errors (e.g. hallucinations, stochastic errors) as well as adversarial manipulation, which are indistinguishable to other agents.

The remaining agents $\mathcal{P} := \mathcal{V} \setminus \mathcal{B}$ are referred to as reliable agents. Furthermore, we refer non-adversarial Byzantine agents, i.e., $\mathcal{H} := \mathcal{V} \setminus \mathcal{A}$, to as honest agents, and the identities of adversarial Byzantine agents are unknown to them. The relationship among these agent classes is illustrated in Fig. 2. To quantify the scope of Byzantine (both faulty and adversarial) agents, we adopt the standard F-local model (LeBlanc et al., 2013):

Definition 3.2 (F-local (LeBlanc et al., 2013))**.** A set $\mathcal{S} \subset \mathcal{V}$ is F-local if every node outside $\mathcal{S}$ has at most F neighbors in $\mathcal{S}$, i.e., $|\mathcal{N}_i \cap \mathcal{S}| \leq F$ for all $i \in \mathcal{V} \setminus \mathcal{S}$.

3.3. Graph Robustness

We now introduce the graph-theoretic notion of r-robustness (LeBlanc et al., 2013):

Definition 3.3 (r-reachable (LeBlanc et al., 2013))**.** Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and a nonempty subset $\mathcal{S} \subset \mathcal{V}$, $\mathcal{S}$ is r-reachable if there exists $i \in \mathcal{S}$ such that $|\mathcal{N}_i \setminus \mathcal{S}| \geq r$.

Definition 3.4 (r-robust (LeBlanc et al., 2013))**.** A graph $\mathcal{G}$ is r-robust if, for every pair of nonempty, disjoint subsets $\mathcal{S}_1, \mathcal{S}_2 \subset \mathcal{V}$, at least one of them is r-reachable.

Higher robustness levels limit the influence of a larger number of Byzantine agents on reliable ones, but require denser communication graphs, which in the LLM setting translates into higher per-round inference cost.

These robustness notions represent stronger properties than other classical graph theoretic metrics such as connectivity or minimum degree (LeBlanc et al., 2013; Pirani et al., 2023), yet can achieve equivalent fault tolerance with sparser connectivity (Pirani et al., 2023). In particular, an r-robust graph is necessarily r-connected and has minimum degree of r. This sparsity is particularly valuable for LLM-MAS: each edge in $\mathcal{E}$ corresponds to an LLM-to-LLM message exchange per round and therefore directly governs per-round inference cost, so r-robust graphs that attain a target fault-tolerance level with fewer edges are inherently preferable in this setting. Now, we present one interesting property about the robustness from (LeBlanc et al., 2013):

Lemma 3.5. *Given an r-robust graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, let $\mathcal{G}' = (\mathcal{V}, \mathcal{E}')$ be a directed graph produced by removing upto k incoming edges of each agent in $\mathcal{G}$. Then, $\mathcal{G}'$ is $(r-k)$-robust.*

The above lemma shows that removing k edges from each agent decreases the robustness at most by k. In this work, we leverage these topological conditions to systematically design communication networks that contain Byzantine influence in LLM-based multi-agent systems.

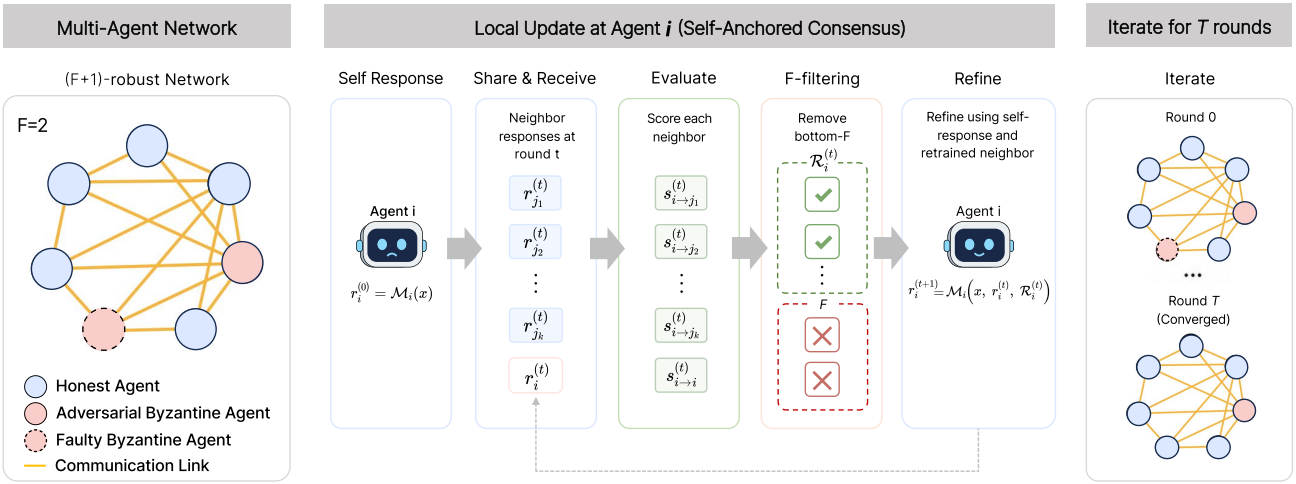

Figure 3. Overview of Self-Anchored Consensus (SAC) on an $(F+1)$-robust network. At each round, every agent generates its own response, exchanges responses with neighbors, scores them on the receiver side, filters out the bottom-F neighbors, and refines its response by re-prompting on the retained set. The procedure is iterated for T rounds.

3.4. Problem Statement

Given n LLM agents, a query x, and an F-local Byzantine set $\mathcal{B}$ consisting of faulty agents $\mathcal{F}$ and adversarial agents $\mathcal{A}$, our objective is twofold: (i) ensure that reliable agents $\mathcal{P}$ remain robust to Byzantine messages, and (ii) enable collaborative refinement of responses that helps all honest agents $\mathcal{H} = \mathcal{P} \cup \mathcal{F}$ move toward the correct answer in a fully decentralized, peer-to-peer network.

We approach this objective in two steps. First, we develop a fully decentralized protocol in which each honest agent iteratively refines its response using neighbors' messages while remaining robust to Byzantine messages. Second, we propose topological conditions on the communication network under which all honest agents can reliably refine their responses.

Remark 3.6. We distinguish between adversarial agents $\mathcal{A}$, which may deviate arbitrarily from the protocol, and faulty agents $\mathcal{F}$, which execute the protocol but may produce unreliable outputs. While both of them are considered Byzantine agents, our framework additionally aims to improve the quality of responses from faulty agents through iterative peer-to-peer refinement despite their initially wrong responses. In this sense, faulty agents can be viewed as *temporary* Byzantine agents: although their initial responses may be unreliable, iterative correction can improve their behavior and effectively transform them into reliable agents.

4. Method

We propose a four-stage pipeline. In Stage 1, each agent broadcasts its current response to its neighbors. In Stage 2, each agent computes confidence scores at the *receiver* side based on incoming messages, removing the assumption in prior work (Zheng et al., 2026) that agents honestly report their own confidence. In Stage 3, each agent filters responses based on the evaluation scores. In Stage 4, each agent refines its response by conditioning its LLM on its current output and the filtered set of neighbor responses.

4.1. Self-Anchored Consensus (SAC)

Given query x, every agent $i \in \mathcal{V}$ first produces an initial response

$$r_i^{(0)} \;=\; \mathcal{M}_i(x). \tag{1}$$

At each subsequent round $t \geq 0$, every agent $i \in \mathcal{V}$ performs the following steps.

Broadcast responses. Every agent i broadcasts its current response $r_i^{(t)}$ to its neighbors.

Score neighbors. Upon receiving neighbor responses $\{r_j^{(t)}\}_{j \in \mathcal{N}_i}$, agent i computes, for each neighbor $j \in \mathcal{N}_i$, an evaluation score

$$s_{i \to j}^{(t)} \;=\; \phi_i\big(x,\, r_j^{(t)}\big) \;\in\; [0,1], \tag{2}$$

where ϕ_i is a scoring function implemented by agent i's own LLM. In our implementation, ϕ_i is realized via a verification prompt that asks $\mathcal{M}_i$ to judge the correctness of $r_j^{(t)}$ with respect to x, returning a calibrated score in $[0,1]$; the full template is given in Section B.

The key distinction from prior confidence-based protocols (Zheng et al., 2026) is that $s_{i \to j}^{(t)}$ depends only on agent i's own computation applied to the content of $r_j^{(t)}$, and in particular does *not* rely on any score reported by agent j. This

removes the possibility for a Byzantine agent to manipulate its own reliability by reporting an inflated self-confidence.

Filter. Let $\mathcal{L}_i^{(t)} \subseteq \mathcal{N}_i$ be the set of neighbors whose score is strictly below agent i's self-score,

$$\mathcal{L}_i^{(t)} \;=\; \{\, j \in \mathcal{N}_i \;\mid\; s_{i\to j}^{(t)} < s_{i\to i}^{(t)} \,\}. \tag{3}$$

Agent i then discards the $\min(F, |\mathcal{L}_i^{(t)}|)$ lowest-scoring elements of $\mathcal{L}_i^{(t)}$, yielding the retained set

$$\mathcal{R}_i^{(t)} \;=\; \mathcal{N}_i \setminus \mathrm{bottom}_F(\mathcal{L}_i^{(t)}), \tag{4}$$

where $\mathrm{bottom}_F(\cdot)$ returns the F lowest-scoring elements of its argument, or the entire argument when $|\mathcal{L}_i^{(t)}| \le F$.

Refine. Agent i refines its response by conditioning $\mathcal{M}_i$ on its current response and the retained neighbor responses, weighted by the corresponding receiver-side scores:

$$r_i^{(t+1)} \;=\; \mathcal{M}_i\Big(x,\; r_i^{(t)},\; \big\{(r_j^{(t)},\, s_{i\to j}^{(t)})\big\}_{j\in\mathcal{R}_i^{(t)}}\Big). \tag{5}$$

The prompt template for Eq. (5) instructs $\mathcal{M}_i$ to prefer high-scored responses while retaining the option to keep its own response if none of the retained neighbor responses improves upon it; see Section B for the full template.

After T rounds (a user-specified parameter), the final response of agent $i \in \mathcal{V}$ is $r_i^{(T)}$. The overall procedure for each agent $i \in \mathcal{V}$ is summarized in Algorithm 1.

Algorithm 1 Self-Anchored Consensus (SAC)

Require: query x; adversary bound F; number of rounds T

1: each agent i computes $r_i^{(0)} = \mathcal{M}_i(x)$
2: **for** $t = 0, 1, \ldots, T-1$ **do**
3: each agent i broadcasts $r_i^{(t)}$ to $\mathcal{N}_i$
4: compute $s_{i\to j}^{(t)}$ for all $j \in \mathcal{N}_i \cup \{i\}$ via Eq. (2)
5: $\mathcal{L}_i^{(t)} \leftarrow \{j \in \mathcal{N}_i : s_{i\to j}^{(t)} < s_{i\to i}^{(t)}\}$
6: $\mathcal{R}_i^{(t)} \leftarrow \mathcal{N}_i \setminus \mathrm{bottom}_F(\mathcal{L}_i^{(t)})$
7: update $r_i^{(t+1)}$ via Eq. (5)
8: **end for**
9: **return** $r_i^{(T)}$

Remark 4.1. Note that Algorithm 1 is fully decentralized; each LLM agent operates using only local information from its neighbors and does not require knowledge of the global network topology $\mathcal{G}$ or the identities of Byzantine agents.

4.2. Robust Topological Conditions

While Algorithm 1 is fully decentralized, its effectiveness depends on the connectivity of the underlying communication graph. Since the filtering step in line 6 of SAC removes up to F neighbors per round, an agent with $|\mathcal{N}_i| \le F$ may have no incoming information after filtering, reducing the algorithm to repeated self-updates. We therefore design the topology to satisfy $(F+1)$-robustness, which ensures two key properties.

Property 4.2. Consider an LLM-MAS connected through a communication graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with an F-local Byzantine set $\mathcal{B} \subset \mathcal{V}$. Let $\mathcal{S}_i^{(t)} = \{\phi_i(x, r_j^{(t)}) \mid j \in \mathcal{P}\}$ denote the set of evaluation scores assigned by agent i at time t to responses from reliable agents. If $\mathcal{G}$ is $(F+1)$-robust, then for every $k \in \mathcal{R}_i^{(t)}$ and time $t \in \mathbb{Z}_{\ge 0}$, the evaluation score $s_{i\to k}^{(t)}$ is at least as high as the minimum evaluation score assigned to any reliable agent's response, i.e.,

$$s_{i\to k}^{(t)} \ge \min \mathcal{S}_i^{(t)}, \quad \forall k \in \mathcal{R}_i^{(t)}. \tag{6}$$

Proof. By Lemma 5 of LeBlanc et al. (2013), since $\mathcal{G}$ is $(F+1)$-robust, we have $|\mathcal{N}_i| \ge F+1$ for all $i \in \mathcal{V}$. Since the filtering step of SAC removes at most F neighbor indices corresponding to the lowest evaluation scores, and $|\mathcal{N}_i| \ge F+1$, at least one neighbor remains in $\mathcal{R}_i^{(t)}$, i.e., $\mathcal{R}_i^{(t)} \neq \emptyset$. Furthermore, under the F-local model, and since at most F indices are removed from the lowest end of the ordered scores, $\mathcal{R}_i^{(t)}$ contains only agent indices whose responses have scores no lower than the minimum score among reliable agents. □

Property 4.2 guarantees that, after filtering, each agent retains at least one response whose quality is comparable to that of a reliable agent, *even if the response does not originate from a reliable agent.* This prevents Byzantine responses from adversely affecting the refinement process, allowing honest agents to safely refine their outputs using information of comparable quality to that produced by reliable agents.

Property 4.3. Consider an LLM-MAS connected through a communication graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with an F-local Byzantine set $\mathcal{B} \subset \mathcal{V}$. Let $\mathcal{G}' = (\mathcal{V}, \mathcal{E}')$ be a directed graph obtained by removing up to any F incoming edges of each agent in $\mathcal{G}$. If $\mathcal{G}$ is $(F+1)$-robust, then $\mathcal{G}'$ contains a rooted-out branching.

Property 4.3 is a direct consequence of Lemma 3.5 and Lemma 7 of (LeBlanc et al., 2013). It guarantees that F-filtering does not disconnect the network, thereby preventing fragmentation into isolated subgroups that may converge to different or incorrect solutions. Therefore, under $(F+1)$-robustness, the subgraph induced by reliable agents remains connected even after filtering out potentially malicious messages.

Together, these properties explain why $(F+1)$-robustness is a natural design choice: it provides *local resilience* against

Byzantine neighbors through MSR-style filtering and *global information-propagation guarantees* across the network.

Remark 4.4. The practical implications of the above properties depend on (1) the fidelity of the scoring function ϕ_i in identifying high-quality responses and (2) the agent's ability to integrate neighbor information during refinement. Since SAC uses each agent's own output as an "anchor" for filtering, a key failure mode arises when a faulty agent is overconfident and assigns low scores to correct neighbor responses, thereby discarding useful information and remaining trapped in error despite reliable neighbors. While our results show that SAC consistently improves faulty agents in practice, this coupling between reasoning and evaluation is a limitation. Decoupling these roles (e.g., via cross-agent verification) is an interesting direction for future work.

5. Experimental Results

5.1. Evaluation Metrics

Let $y^\star$ denote the ground-truth answer to query $x \sim \mathcal{D}$, and $r_i^{(t)}$ the response of agent i at round t. We partition honest agents $\mathcal{H}$ into strong and weak groups, where weak agents are treated as faulty (non-adversarial Byzantine) agents. Following Zheng et al. (2026), we report the following metrics (higher is better).

Initial / Final Agent Accuracy (IAA / FAA). Average per-agent accuracy before communication ($t = 0$) and after T consensus rounds ($t = T$).

Byzantine Fault Tolerance Improvement (BFTI). $\mathrm{BFTI} = \mathrm{FAA} - \mathrm{IAA}$ measures the gain or degradation introduced by the consensus protocol.

Round-level Accuracy (RA). The fraction of queries for which the majority vote across all agents in $\mathcal{V}$ at round T matches $y^\star$.

Group-wise Accuracy (W_IAA→FAA, S_IAA→FAA). Accuracy change of the weak and strong honest groups before and after consensus.

Honest Majority (H-Majority). The fraction of queries for which the majority vote among honest agents $\mathcal{H}$ at round T matches $y^\star$.

5.2. Experimental Setups

Datasets. We extend the evaluation protocol of Zheng et al. (2026), which evaluates Byzantine fault tolerance on 10 hand-curated GSM8K problems. Since GSM8K consists of grade-school arithmetic and the original 10-problem subset is further selected to amplify the strong–weak capability gap, the resulting benchmark may understate the difficulty of preserving reliable agents' base capability under adversarial influence. We therefore evaluate on two more challenging reasoning benchmarks. MATH500 (Lightman et al., 2023), restricted to Level 4 problems that require multi-step symbolic reasoning, and Commonsense170k (Hu et al., 2023), which spans heterogeneous question formats. From MATH500 we sample 30 Level 4 instances, and from Commonsense170k we sample 30 instances spanning the different question types (BoolQ true/false, multiple-choice commonsense, and sentence-completion). The full list of sampled problems is provided in Appendix C.

Network. Each network consists of $n = 7$ agents: 4 strong honest, 2 weak honest (faulty Byzantine), and 1 adversarial Byzantine. We set $F = 3$: the filter may then discard weak honest agents' responses, and a successful method must still preserve reliable-agent capability. We instantiate $\mathcal{G}$ as one of three $(F+1)$-robust topologies — a γ-MERG with $\gamma = \lceil n/2 \rceil = 4$ (Lee & Panagou, 2026), a complete graph which is 4-robust (LeBlanc et al., 2013), and an Erdős–Rényi random graph (Erdős & Rényi, 1961) with Zhang et al. (2015) — thereby satisfying Properties 4.2 and 4.3. More detailed descriptions of the topologies are given in Appendix A.

Models. On MATH500 we use `gpt-3.5-turbo` as the weak model and `gpt-4o-mini` as the strong model, following the standard CP-WBFT setup. On Commonsense170k we use `gpt-4o` as the weak model and `gpt-5` as the strong model.

Byzantine model. Adversarial Byzantine agents return weak-model responses generated on an out-of-distribution dataset with a falsified self-confidence score of 1.0. This simulates the strongest dishonest-confidence attack against CP-WBFT while remaining indistinguishable from reliable agents at the message level. In contrast, faulty Byzantine agents follow the same protocol as reliable agents.

5.3. Experimental Results

MATH500. CP-WBFT yields consistently negative BFTI across all three topologies (-10.4% to -24.8%; Table 1), indicating that collaboration degrades agent performance under Byzantine influence. While the weak group may show superficial gains in some cases (e.g., $25 \rightarrow 58\%$ on MERG), this comes at the cost of a substantial drop in the strong group (e.g., $84 \rightarrow 49\%$), resulting in degraded overall reliability. Per-round results in Table 2 show that this degradation occurs immediately: after Rnd 1, both weak and strong groups collapse to a contaminated state (e.g., $58.3/49.2$ on MERG) and remain unchanged thereafter. This behavior stems from the adversarial Byzantine agent's falsified self-confidence dominating the aggregation, after which all subsequent rounds operate on corrupted responses.

In contrast, SAC achieves positive BFTI across all topologies ($+3.8\%$ to $+7.6\%$) while preserving the strong group

Table 1. Comparison of CP-WBFT and SAC across three communication topologies on MATH500 Level 4 with $n = 7$ agents (4 strong honest, 2 weak honest (faulty Byzantine), 1 adversarial Byzantine, $F = 3$, and $r = 4$. The adversarial Byzantine agent reports a falsified self-confidence of 1.0. W IAA→FAA and S IAA→FAA denote the per-agent accuracy change for the weak and strong honest groups, and H-Majority is the fraction of queries on which the majority answer among honest agents matches the ground truth at round T.

Method	Topology	IAA	FAA	BFTI	RA	W IAA→FAA	S IAA→FAA	H-Majority
CP-WBFT	MERG	55.2%	44.8%	−10.4%	30.0%	25→58%	84→49%	40.0%
	Complete	56.2%	31.4%	−24.8%	33.3%	28→37%	84→37%	36.7%
	Erdős-Rényi	56.2%	40.0%	−16.2%	43.3%	25→45%	86→48%	46.7%
SAC (Ours)	MERG	54.8%	62.4%	+7.6%	90.0%	23→43%	84→88%	96.7%
	Complete	55.7%	59.5%	+3.8%	83.3%	28→40%	83→84%	83.3%
	Erdős-Rényi	54.3%	60.5%	+6.2%	80.0%	28→45%	81→83%	86.7%

Table 2. Per-round weak / strong honest accuracy (W/S, in %) on MATH500 Level 4 across three communication topologies. Each cell shows the accuracy of the weak honest group (gpt-3.5-turbo, 2 agents) and the strong honest group (gpt-4o-mini, 4 agents) at the corresponding round.

Method	Round	MERG (W/S)	Complete (W/S)	Erdős-Rényi (W/S)
CP-WBFT	Init	25.0 / 84.2	28.3 / 84.2	25.0 / 85.8
	Rnd 1	58.3 / 49.2	36.7 / 36.7	45.0 / 47.5
	Rnd 2	58.3 / 49.2	36.7 / 36.7	45.0 / 47.5
	Rnd 3	58.3 / 49.2	36.7 / 36.7	45.0 / 47.5
	Rnd 4	58.3 / 49.2	36.7 / 36.7	45.0 / 47.5
	Rnd 5	58.3 / 49.2	36.7 / 36.7	45.0 / 47.5
	Rnd 6	58.3 / 49.2	36.7 / 36.7	45.0 / 47.5
SAC (Ours)	Init	23.3 / 84.2	28.3 / 83.3	28.3 / 80.8
	Rnd 1	53.3 / 87.5	36.7 / 83.3	45.0 / 86.7
	Rnd 2	50.0 / 86.7	31.7 / 84.2	40.0 / 85.8
	Rnd 3	50.0 / 86.7	40.0 / 88.3	41.7 / 81.7
	Rnd 4	45.0 / 86.7	40.0 / 88.3	36.7 / 87.5
	Rnd 5	53.3 / 85.0	48.3 / 85.0	38.3 / 85.8
	Rnd 6	43.3 / 87.5	40.0 / 84.2	45.0 / 83.3

and improving the weak group. For example, on MERG, the strong group improves from $84 \rightarrow 88\%$ while the weak group increases from $23 \rightarrow 43\%$. H-Majority correspondingly rises to 83.3–96.7%, indicating reliable consensus among honest agents. Per-round results show that most gains occur at Rnd 1 and are sustained thereafter, without progressive degradation. Notably, the strong group remains stable across rounds, demonstrating that SAC prevents Byzantine signals from corrupting high-quality responses.

Commonsense reasoning. A similar trend is observed on the commonsense benchmark (Table 3). CP-WBFT exhibits severe degradation under Byzantine influence, with BFTI ranging from -32.4% to -54.7%. Both weak and strong groups collapse sharply after Rnd 1 (Table 4), converging to low-accuracy states (e.g., $16.7/16.7$ on complete graph) that persist across all rounds. This confirms that confidence-weighted aggregation is highly vulnerable to adversarial manipulation in this setting.

In contrast, SAC maintains near-zero or positive BFTI (-2.8% to $+0.0\%$) while consistently improving the weak group (e.g., $73 \rightarrow 77\%$ on MERG) and preserving the strong group (e.g., $83 \rightarrow 78\%$). H-Majority reaches 76.7–83.3% across all topologies, indicating robust consensus. As in MATH500, most improvements occur in early rounds and remain stable, with no evidence of Byzantine-driven drift. These results demonstrate that SAC generalizes across reasoning domains, maintaining robustness without sacrificing strong-agent performance.

5.4. Ablation Study

We probe SAC along two axes: the composition of the Byzantine set and the network size n.

Varying byzantine-to-weak composition. Table 5 fixes the strong honest count at 4 and sweeps $(b, w) \in \{(1, 2), (2, 1), (3, 0)\}$ at n=7. CP-WBFT yields negative BFTI in all 12 configurations (-10.4% to -32.4%), showing that even a single adversary suffices to dominate confidence-weighted aggregation. SAC maintains non-negative BFTI in 11 of 12 cells, with H-Majority in 80–96.7% across all (b, w) settings. BFTI does not strictly degrade as b grows, consistent with $(F+1)$-robustness bound-

Table 3. Comparison of CP-WBFT and SAC across three communication topologies on 30 sampled commonsense reasonings with $n = 7$ agents (4 strong honest, 2 weak honest (faulty Byzantine), 1 adversarial Byzantine), promoted models (strong = gpt-5, weak = gpt-4o), $F = 3$, and $r = 4$. W IAA→FAA and S IAA→FAA denote the per-agent accuracy change for the weak and strong honest groups, and H-Majority is the fraction of queries on which the majority answer among honest agents matches the ground truth at round T.

Method	Topology	IAA	FAA	BFTI	RA	W IAA→FAA	S IAA→FAA	H-Majority
CP-WBFT	MERG	69.5%	37.1%	−32.4%	26.7%	77→52%	83→39%	26.7%
	Complete	69.0%	14.3%	−54.7%	16.7%	72→17%	85→17%	16.7%
	Erdős-Rényi	69.0%	20.5%	−48.5%	23.3%	75→25%	83→23%	23.3%
SAC (Ours)	MERG	68.6%	66.7%	−1.9%	76.7%	73→77%	83→78%	80.0%
	Complete	69.5%	69.5%	+0.0%	83.3%	77→80%	83→82%	83.3%
	Erdős-Rényi	69.0%	66.2%	−2.8%	73.3%	73→75%	84→78%	76.7%

Table 4. Per-round weak / strong honest accuracy (W/S, in %) on 30 sampled commonsense reasonings across three communication topologies with promoted models (strong = gpt-5, weak = gpt-4o). Each cell shows the accuracy of the weak honest group (gpt-4o, 2 agents) and the strong honest group (gpt-5, 4 agents) at the corresponding round.

Method	Round	MERG (W/S)	Complete (W/S)	Erdős-Rényi (W/S)
CP-WBFT	Init	76.7 / 83.3	75.0 / 85.0	75.0 / 83.3
	Rnd 1	51.7 / 39.2	13.3 / 13.3	25.0 / 23.3
	Rnd 2	51.7 / 39.2	13.3 / 13.3	25.0 / 23.3
	Rnd 3	51.7 / 39.2	13.3 / 13.3	25.0 / 23.3
	Rnd 4	51.7 / 39.2	13.3 / 13.3	25.0 / 23.3
	Rnd 5	51.7 / 39.2	13.3 / 13.3	25.0 / 23.3
	Rnd 6	51.7 / 39.2	13.3 / 13.3	25.0 / 23.3
SAC (Ours)	Init	73.3 / 83.3	75.0 / 83.3	73.3 / 84.2
	Rnd 1	83.3 / 67.5	83.3 / 66.7	83.3 / 62.5
	Rnd 2	73.3 / 79.2	73.3 / 80.0	70.0 / 77.5
	Rnd 3	81.7 / 74.2	83.3 / 73.3	81.7 / 70.0
	Rnd 4	80.0 / 78.3	80.0 / 76.7	76.7 / 75.0
	Rnd 5	80.0 / 71.7	76.7 / 70.0	78.3 / 67.5
	Rnd 6	76.7 / 78.3	80.0 / 83.3	75.0 / 78.3

ing adversarial influence per round rather than per adversary, so long as the F-local assumption holds.

Scaling to larger networks. Table 6 scales to n=9 with a $6/2/1$ split at the same adversary bound F=3. CP-WBFT degrades more severely than at n=7 (BFTI -12.6% to -32.3%), indicating that a larger honest population does not dilute adversarial influence under self-reported confidence. Here, we add a preferential-attachment graph consisting of a complete graph of 7 nodes, with the remaining nodes each connected to any 4 of the 7 nodes. By Theorem 5 of LeBlanc et al. (2013), a preferential-attachment graph is 4-robust. SAC achieves positive BFTI on three of four topologies (mild regression of -2.2% on Erdős-Rényi) and H-Majority of 86.7–90.0%, showing that the robustness guarantee carries over without retuning.

6. Discussion

Our experiments expose a clear divide between sender-reported and receiver-side trust, and clarify the role of topology in containing Byzantine influence.

Self-reported confidence is an attack surface. A single adversarial agent with falsified self-confidence is sufficient to collapse CP-WBFT across all evaluated topologies, with BFTI ranging from -10.4% to -24.8% on MATH500 (Table 1) and -32.4% to -54.7% on Commonsense170k (Table 3). Per-round traces (Tables 2, 4) show that contamination occurs immediately at Rnd 1 and persists unchanged thereafter, indicating that the failure stems from the aggregation rule rather than from insufficient communication rounds. This pattern holds even though honest agents form a clear majority (6 of 7), showing that majority correctness alone does not confer robustness when the trust signal itself is manipulable.

Receiver-side evaluation contains Byzantine influence. By construction, SAC's filter-and-refine update bounds adversarial influence at the local level: under $(F+1)$-robustness, each honest agent retains at least one response no worse than the minimum-scored reliable response (Property 4.2), and the filtered network preserves a rooted out-branching (Property 4.3). Empirically, SAC yields non-negative BFTI in 11 of 12 configurations in the composition ablation (Table 5) and achieves H-Majority of 76.7%–

Table 5. Ablation over byzantine-to-weak composition at n=7 on MATH500 Level 4. The byzantine agent count (b) and weak honest count (w) are varied while the strong honest count is fixed at 4 (F=3, T=6, r=4). Strong/weak agents use `gpt-4o-mini`/`gpt-3.5-turbo`.

		IAA (%)			FAA (%)			BFTI (%)			RA (%)			H-Majority (%)		
Method	**Topology**	b=1 w=2	b=2 w=1	b=3 w=0	b=1 w=2	b=2 w=1	b=3 w=0	b=1 w=2	b=2 w=1	b=3 w=0	b=1 w=2	b=2 w=1	b=3 w=0	b=1 w=2	b=2 w=1	b=3 w=0
CP-WBFT	MERG	55.2	48.1	44.3	44.8	31.4	29.0	−10.4	−16.7	−15.3	30.0	40.0	50.0	40.0	43.3	50.0
	Complete	56.2	51.9	49.0	31.4	30.5	34.3	−24.8	−21.4	−14.7	33.3	43.3	60.0	36.7	43.3	60.0
	Erdős-Rényi	56.2	52.9	48.1	40.0	31.4	29.5	−16.2	−21.5	−18.6	43.3	40.0	50.0	46.7	40.0	53.3
SAC (Ours)	MERG	54.8	49.5	47.1	62.4	52.9	48.1	+7.6	+3.4	+1.0	90.0	83.3	86.7	96.7	83.3	83.3
	Complete	55.7	51.0	47.6	59.5	54.8	52.4	+3.8	+3.8	+4.8	83.3	86.7	93.3	86.7	83.3	93.3
	Erdős-Rényi	54.3	51.4	46.7	60.5	57.1	47.1	+6.2	+5.7	+0.4	80.0	93.3	83.3	86.7	90.0	86.7

Table 6. Scalability to larger networks (n=9 agents) on MATH500 Level 4. Configuration: 6 strong honest, 2 weak honest, 1 adversarial Byzantine (F=3, T=6); strong/weak agents use `gpt-4o-mini`/`gpt-3.5-turbo`.

Method	**Topology**	r	**IAA (%)**	**FAA (%)**	**BFTI (%)**	**RA (%)**	**H-Majority (%)**
CP-WBFT	MERG	5	63.7	37.0	−26.7	20.0	23.3
	Complete	5	61.9	29.6	−32.3	33.3	33.3
	Preferential	4	64.8	44.1	−20.7	26.7	26.7
	Erdős-Rényi	4	63.0	50.4	−12.6	36.7	43.3
SAC (Ours)	MERG	5	62.6	64.8	+2.2	90.0	90.0
	Complete	5	60.4	62.2	+1.8	86.7	86.7
	Preferential	4	62.6	65.6	+3.0	90.0	90.0
	Erdős-Rényi	4	62.6	60.4	−2.2	86.7	86.7

96.7% across all topologies and benchmarks, in contrast to CP-WBFT's 16.7%–46.7%.

Strong agents are preserved, weak agents are lifted. Group-wise accuracy reveals a consistent asymmetry between the two protocols. Under SAC, strong-group accuracy on MATH500 remains within a few percentage points of its initial value across all topologies (84→88% on MERG, 83 → 84% on complete graph; Table 1), while the weak group improves (e.g., 23 → 43% on MERG). CP-WBFT exhibits the opposite pattern: superficial weak-group gains (e.g., 25→58%) come with substantial strong-group degradation (84 → 49%), so the protocol effectively averages capabilities downward rather than upward.

Topology matters within the robustness class. All three evaluated graphs satisfy $(F+1)$-robustness, yet SAC's BFTI on MATH500 spans +3.8% to +7.6% (Table 1), indicating that the robustness condition is necessary but not sufficient for optimal performance. More structured constructions (MERG) tend to better preserve strong-group accuracy than less structured ones (Erdős–Rényi), and the scaling experiment at n=9 (Table 6) shows the same ordering, suggesting that practical topology design should consider degree regularity and information-propagation properties beyond the minimum robustness threshold.

Limitations. Our evaluation is limited to networks of $n \in \{7, 9\}$, a single adversarial behavior (cached weak-model replay with falsified confidence), and single-turn responses. Larger populations, dynamic topologies, multi-turn reasoning, and tool use are outside the present scope. SAC's robustness also rests on the quality of the receiver-side scoring function ϕ_i; an adversary that crafts responses appearing correct to ϕ_i but factually wrong is not addressed by the current filtering mechanism, and we leave this attack vector to future work.

7. Conclusion

We studied robust decision-making in LLM-based multi-agent systems under Byzantine faults, and showed that existing confidence-weighted aggregation is fundamentally vulnerable to adversarial manipulation of self-reported scores. We proposed Self-Anchored Consensus (SAC), a fully decentralized filter-and-refine protocol that replaces sender-reported confidence with receiver-side evaluation, and established $(F+1)$-robustness conditions under which honest agents reliably refine their responses despite Byzantine influence. Experiments on mathematical and commonsense reasoning benchmarks show that SAC preserves strong-agent performance while uplifting weaker agents across diverse topologies, whereas prior methods collapse under the same conditions. These results point to receiver-side evaluation and robustness-aware topology design as essential ingredients for reliable decentralized LLM collaboration.

References

Dolev, D., Lynch, N. A., Pinter, S. S., Stark, E. W., and Weihl, W. E. Reaching approximate agreement in the presence of faults. *J. ACM*, 33(3):499–516, May 1986.

ISSN 0004-5411. doi: 10.1145/5925.5931.

Du, Y., Li, S., Torralba, A., Tenenbaum, J. B., and Mordatch, I. Improving factuality and reasoning in language models through multiagent debate. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235, pp. 11733–11763. PMLR, 2024.

Erdős, P. and Rényi, A. On the strength of connectedness of a random graph. *Acta Mathematica Hungarica*, 12(1-2): 261–267, 1961.

Guo, T., Chen, X., Wang, Y., Chang, R., Pei, S., Chawla, N. V., Wiest, O., and Zhang, X. Large language model based multi-agents: A survey of progress and challenges. In *Proceedings of the 33rd International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 8048–8057, 2024.

Hong, S., Zheng, X., Chen, J., Cheng, Y., Wang, J., Zhang, C., Wang, Z., Yau, S. K. S., Lin, Z., Zhou, L., Ran, C., Xiao, L., Wu, C., and Schmidhuber, J. MetaGPT: Meta programming for a multi-agent collaborative framework. In *International Conference on Learning Representations (ICLR)*, 2024.

Hu, Z., Wang, L., Lan, Y., Xu, W., Lim, E.-P., Bing, L., Xu, X., Poria, S., and Lee, R. K.-W. LLM-Adapters: An adapter family for parameter-efficient fine-tuning of large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 5254–5276, 2023.

Jo, Y. and Park, C. Byzantine-robust decentralized coordination of LLM agents. *arXiv preprint arXiv:2507.14928*, 2025.

Kieckhafer, R. and Azadmanesh, M. Reaching approximate agreement with mixed-mode faults. *IEEE Transactions on Parallel and Distributed Systems*, 5(1):53–63, 1994. doi: 10.1109/71.262588.

Lamport, L., Shostak, R., and Pease, M. The Byzantine generals problem. *ACM Transactions on Programming Languages and Systems*, 4(3):382–401, 1982.

LeBlanc, H. J., Zhang, H., Koutsoukos, X., and Sundaram, S. Resilient asymptotic consensus in robust networks. *IEEE Journal on Selected Areas in Communications*, 31 (4):766–781, 2013. doi: 10.1109/JSAC.2013.130413.

Lee, H. and Panagou, D. Distributed resilience-aware control in multi-robot networks. pp. 3868–3875, 2025. doi: 10.1109/CDC57313.2025.11312021.

Lee, H. and Panagou, D. Minimal construction of graphs with maximum robustness. *arXiv preprint arXiv:2507.00415*, 2026. v2, February 2026.

Liang, T., He, Z., Jiao, W., Wang, X., Wang, Y., Wang, R., Yang, Y., Tu, Z., and Shi, S. Encouraging divergent thinking in large language models through multi-agent debate. *arXiv preprint arXiv:2305.19118*, 2024.

Lightman, H., Kosaraju, V., Burda, Y., Edwards, H., Baker, B., Lee, T., Leike, J., Schulman, J., Sutskever, I., and Cobbe, K. Let's verify step by step. *arXiv preprint arXiv:2305.20050*, 2023.

Luo, H., Sun, G., Liu, Y., Zhao, D., Niyato, D., Yu, H., and Dustdar, S. A weighted Byzantine fault tolerance consensus driven trusted multiple large language models network. *arXiv preprint arXiv:2505.05103*, 2025.

Pirani, M., Mitra, A., and Sundaram, S. Graph-theoretic approaches for analyzing the resilience of distributed control systems: A tutorial and survey. *Automatica*, 157: 111264, 2023. ISSN 0005-1098. doi: https://doi.org/10.1016/j.automatica.2023.111264.

Su, L. and Vaidya, N. H. Byzantine-resilient multiagent optimization. *IEEE Transactions on Automatic Control*, 66 (5):2227–2233, 2021. doi: 10.1109/TAC.2020.3008139.

Sundaram, S. and Gharesifard, B. Distributed optimization under adversarial nodes. *IEEE Transactions on Automatic Control*, 64(3):1063–1076, 2019. doi: 10.1109/TAC.2018.2836919.

Tian, Y., Yang, X., Zhang, J., Dong, Y., and Su, H. Evil geniuses: Delving into the safety of LLM-based agents. *arXiv preprint arXiv:2311.11855*, 2023.

Usevitch, J. and Panagou, D. Determining r-and (r, s)-robustness of digraphs using mixed integer linear programming. *Automatica*, 111:108586, 2020.

Wu, Q., Bansal, G., Zhang, J., Wu, Y., Li, B., Zhu, E., Jiang, L., Zhang, X., Zhang, S., Liu, J., Awadallah, A. H., White, R. W., Burger, D., and Wang, C. AutoGen: Enabling next-gen LLM applications via multi-agent conversation. 2023.

Xie, Y., Mou, S., and Sundaram, S. Communication-efficient and resilient distributed q-learning. *IEEE Transactions on Neural Networks and Learning Systems*, 35 (3):3351–3364, 2023.

Ye, L., Figura, M., Lin, Y., Pal, M., Das, P., Liu, J., and Gupta, V. Resilient multiagent reinforcement learning with function approximation. *IEEE Transactions on Automatic Control*, 69(12):8497–8512, 2024. doi: 10.1109/TAC.2024.3409676.

Yu, M., Wang, S., Zhang, G., Mao, J., Yin, C., Liu, Q., Wen, Q., Wang, K., and Wang, Y. NetSafe: Exploring the topological safety of multi-agent networks. *arXiv preprint arXiv:2410.15686*, 2024.

Yuan, L. and Ishii, H. Resilient distributed economic dispatch in smart grids. *IEEE Transactions on Automatic Control*, 2025.

Yun, J., Abousamra, S., Li, C., Gupta, R., Kurc, T., Samaras, D., Van Dyke, A., Saltz, J., and Chen, C. Uncertainty estimation for tumor prediction with unlabeled data. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 6946–6954, 2024. doi: 10.1109/CVPRW63382.2024.00688.

Zhang, G., Yue, Y., Li, Z., Yun, S., Wan, G., Wang, K., Cheng, D., Yu, J. X., and Chen, T. Cut the crap: An economical communication pipeline for LLM-based multi-agent systems. *arXiv preprint arXiv:2410.02506*, 2024.

Zhang, H., Fata, E., and Sundaram, S. A notion of robustness in complex networks. *IEEE Transactions on Control of Network Systems*, 2(3):310–320, 2015.

Zheng, L., Chen, J., Yin, Q., Zhang, J., Zeng, X., and Tian, Y. Rethinking the reliability of multi-agent system: A perspective from Byzantine fault tolerance. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2026. arXiv:2511.10400.

Appendix

A. Graph Topologies

We evaluate our method on four graph topologies, which we describe below:

- γ-MERGs (Lee & Panagou, 2026): These graphs achieve maximum robustness, i.e., $r = \lceil n/2 \rceil$, for a given number of nodes n, while using the minimal set of edges possible. We distinguish two cases for constructing a γ-MERG:
  - **If n is odd:** Let $\mathcal{X} \subset \mathcal{V}$ be a set of $\gamma + 1$ nodes forming a complete subgraph. Connect each of the remaining $\gamma - 1$ nodes to any γ nodes in $\mathcal{X}$.
  - **If n is even:** Let $\mathcal{X} \subset \mathcal{V}$ be a set of γ nodes, each adjacent to all other nodes in $\mathcal{V}$. Then select $\lceil \frac{\gamma-2}{2} \rceil$ disjoint pairs of nodes in $\mathcal{X}$ and remove the edge between each pair.

  By Theorem 1 of (Lee & Panagou, 2026), the resulting graph is $\lceil n/2 \rceil$-robust.

- Complete graphs: In this case, every agent is connected to every other agent. By Lemma 4 of LeBlanc et al. (2013), a complete graph with n nodes is $r = \lceil n/2 \rceil$-robust.

- Preferential-attachment graphs: To construct r-robust graphs, we first form a complete graph with $2r - 1$ nodes. By Lemma 4 of LeBlanc et al. (2013), this subgraph is r-robust. The remaining $n - 2r + 1$ nodes are then each connected to any r nodes in the original set of nodes in the complete graph. By Theorem 5 of LeBlanc et al. (2013), the resulting graph is then r-robust.

- Erdős–Rényi random graphs (Erdős & Rényi, 1961): These graphs are parameterized by the number of nodes n and an edge probability p. The presence of each edge is independent of every other edge, and each edge is included independently with probability p. As shown in Theorem 3 of Zhang et al. (2015), choosing $p = \frac{\ln n + (r-1)\ln\ln n}{n}$ ensures that the graph is r-robust with high probability as $n \to \infty$. We sample graphs using this probability and verify r-robustness prior to simulation using the method from Usevitch & Panagou (2020).

B. Prompt Templates

We instantiate the receiver-side scoring function ϕ_i (Eq. (2)) and the refine operator (Eq. (5)) using the templates below. We present the MATH500 prompts in full. For Commonsense170k, we use the same overall structure, with only the task wording and answer-format instruction modified as described in Section B.3.

B.1. Scoring Prompt (ϕ_i)

The scoring prompt instructs $\mathcal{M}_i$ to independently solve the query x, compare its own reasoning against $r_j^{(t)}$, and finally output a calibrated confidence score in $[0, 1]$. This procedure makes $s_{i \to j}^{(t)}$ a receiver-side evaluation: the score depends solely on $\mathcal{M}_i$'s internal reasoning applied to the content of $r_j^{(t)}$, without relying on any confidence reported by agent j.

```
System: You are an expert evaluator. Respond using only a single number.
User:
You are evaluating the correctness of a proposed answer to a mathematical reasoning
problem.
Assume that up to f responses may be malicious or unreliable.
Reason carefully before producing the final score.
Problem: {x}
Proposed answer: {r_j^(t)}
Step 1: Solve the problem independently step by step.
Step 2: Evaluate the proposed answer.
Step 3: Output ONLY a single decimal number between 0.0 and 1.0.
0.0 = clearly incorrect, 0.5 = uncertain, 1.0 = clearly correct.
Score:
```

The first numeric token in $\mathcal{M}_i$'s response is extracted using the regular expression `([0-9]+(?:\.[0-9]+)?)` and clipped to the interval $[0, 1]$. If parsing fails or the LLM returns an invalid response, the score defaults to $s_{i\to j}^{(t)} = 0.5$, treating the response as maximally uncertain. The self-score $s_{i\to i}^{(t)} = \phi_i(x, r_i^{(t)})$ is computed using the same prompt with $r_j^{(t)}$ replaced by $r_i^{(t)}$.

B.2. Refine Prompt (Eq. (5))

The refine prompt provides $\mathcal{M}_i$ with its current response $r_i^{(t)}$ together with the retained neighbor responses $\{(r_j^{(t)}, s_{i\to j}^{(t)})\}_{j\in\mathcal{R}_i^{(t)}}$, sorted in descending order of reliability score. The instruction *"Prefer answers with higher reliability scores, but retain your current answer if you believe it is correct"* serves as the natural-language realization of Eq. (5), encouraging $\mathcal{M}_i$ to prioritize highly scored neighbor responses while preserving its own response as an anchor when appropriate.

System: You are a helpful assistant. Follow the required output format exactly.
User:
You are solving a mathematical reasoning problem. Review the information below and provide your best answer.
Assume that up to f responses may be malicious or unreliable.
Reason carefully before producing the final answer.
Problem: $\{x\}$
Your current answer: $\{r_i^{(t)}\}$
Other agents' answers with reliability scores (higher indicates greater reliability):
- Answer: $\{r_{j_1}^{(t)}\}$ (reliability score: $\{s_{i\to j_1}^{(t)}\}$)
- Answer: $\{r_{j_2}^{(t)}\}$ (reliability score: $\{s_{i\to j_2}^{(t)}\}$)
... (sorted in descending order of score, $j_k \in \mathcal{R}_i^{(t)}$)
Prefer answers with higher reliability scores, but retain your current answer if you believe it is correct.
Respond using the format:
Answer: [your numerical answer]

Reliability scores are formatted to two decimal places. The retained set $\mathcal{R}_i^{(t)}$ is computed prior to rendering the prompt, so neighbors removed by $\mathrm{bottom}_F(\cdot)$ never appear in the refinement stage. If $\mathcal{R}_i^{(t)} = \emptyset$, the refinement step is skipped and $r_i^{(t+1)} = r_i^{(t)}$. The final numerical answer is extracted using a dataset-specific parser; if extraction fails, the update again defaults to $r_i^{(t+1)} = r_i^{(t)}$.

B.3. Commonsense170k Prompt Adaptation

Since the sampled Commonsense170k instances span heterogeneous formats, including true/false, multiple-choice, and sentence-completion tasks, we replace the math-specific wording with a question-agnostic formulation. In particular, "mathematical reasoning problem" is replaced with "question", and the answer-format instruction becomes `Respond strictly using the answer format specified in the question.\n Answer: [your answer]`. The receiver-side scoring procedure and descending-score neighbor presentation remain unchanged.

B.4. Byzantine Agent Behavior

Adversarial Byzantine agents (Section 5) bypass both prompting procedures. At each round t, they (i) broadcast a cached weak-model response generated from an out-of-distribution query, and (ii) report a falsified self-confidence score of 1.0 when interacting with confidence-based baselines such as CP-WBFT (Zheng et al., 2026). This simulates a strong dishonest-confidence attack while remaining indistinguishable from honest agents at the message level. In contrast, faulty Byzantine agents follow the same prompting pipeline as reliable agents, and produce unreliable outputs solely due to the limited capability of the underlying weak model.

C. Dataset

We list the 30 sampled MATH500 Level 4 instances and 30 sampled Commonsense170k instances used in our experiments. Each row reports the dataset-specific question identifier (`qid`), the question type, an abbreviated version of the question, and the ground-truth answer $y^\star$. Full problem statements can be retrieved using the corresponding `qid` from the original public releases of MATH500 and Commonsense170k.

C.1. MATH500 (Level 4)

The 30 sampled instances span the seven question categories present in MATH500 Level 4. We group rows by category for readability.

Table 7. Sampled MATH500 Level 4 problems used in our experiments, grouped by category. Question column gives an abbreviated statement; full problem text is recoverable from the public MATH500 release via the `qid`.

qid	Question	$y^\star$
Algebra		
`32448964`	Value of K for which $6x+4y=7,\ Kx+8y=7$ has no solution	12
`57cbe0b7`	Solve $(\sqrt{12x}+12)(\sqrt{3x}-6)=4(x+3)+x-34$	50
`984a9fb8`	Express $g^4+12g^2+9=c(g^2+p)^2+q$; find q	-27
`b0969f24`	Piecewise f with $f(-4)=-60/13$, $f(4)=3120$; find $a+b$	28
`d026abb8`	Integer x in arithmetic sequence $3^2,\ x,\ 3^4$	45
`9bb18dfb`	Minimum of $\lvert x-1\rvert+\lvert x-1.5\rvert+\lvert x-2\rvert$ over $x\in\mathbb{R}$	1
Intermediate Algebra		
`40c817ff`	$\log_y x+\log_x y=7$; find $(\log_y x)^2+(\log_x y)^2$	47
`b3f26b98`	Piecewise f invertible; find k	1
`b11209a5`	Geometric sequence with $a_5-a_4=576$, $a_2-a_1=9$; sum $\sum_{i=1}^{5}a_i$	1023
`10aef5c7`	Count of $n<1000$ such that $\lfloor\log_2 n\rfloor$ is a positive even integer	340
Prealgebra		
`2de720d4`	Number of times the digit 6 appears in integers from 1 to 100	20
`b6a36467`	Smallest integer >2 that leaves remainder 2 modulo 3, 4, 5, 6	62
`34e64136`	Five tests scored 87, 85, 87; last two differ by 3 with average 90; find max	97
`f3b85d7a`	Number of even perfect cubes less than 2008	6
`c740506c`	Box nesting: 4 large $\times$ 3 medium $\times$ 2 small; total number of boxes	40
`e067503f`	Unit conversion: 6 wallops $=$ 5 ballops, 3 ballops $=$ 11 fallops; wallops for 110 fallops	36
`7bfcd56a`	First odd year after 2006 whose digits split into 3-digit and 1-digit groups with common factor >1	2013
Number Theory		
`a10973bf`	Smallest integer with exactly 16 divisors, including 12 and 15	120
`27b01b01`	Base-7 cryptarithm $\overline{AB}_7+\overline{BA}_7=\overline{AA0}_7$; product $A\cdot B$	6
`583c9eaf`	Number of even positive divisors of 252	12
`37bab629`	Count of $n\in[1,29]$ for which $n/30$ has a repeating decimal expansion	20
Counting & Probability		
`0a3e457d`	20-member club, 3 distinct officers, with constraint Alex serves only if Bob does not	6732
`d790474f`	Jar with 4 red, 2 white marbles; swap-and-sample procedure; $P(\text{red})=11/18$	$11/18$
`b98d41f7`	Coefficient of x^2y^2 in $(x+y)^4+(x+2y)^4$	30
`03694fa9`	120 triangles formed by n vertices on a base; find n	16
`33b5135e`	10-member chess club, 900 games played; find N games per pair	20
`e5787bf4`	Distinct bracelets with 5 distinct beads under rotation and reflection equivalence	12
Geometry		
`1efe044e`	Maximum volume (cm^3) when right $\triangle$ with legs $3,\ 4$ rotated about a leg	50
`de4ec0fd`	Isosceles $\triangle$ with $AB=AC=14$, $BC=26$; shortest angle bisector	$8\sqrt{33}/3$
Precalculus		
`16be6141`	$\mathbf{v}=\mathrm{proj}_{\mathbf{a}}\mathbf{v}+\mathrm{proj}_{\mathbf{b}}\mathbf{v}$ for all $\mathbf{v}$; find $\mathbf{a}\cdot\mathbf{b}$	0

C.2. Commonsense170k

The 30 sampled Commonsense170k instances span four heterogeneous question formats present in the benchmark: true/false (BoolQ-style), two-option pronoun resolution, multiple-choice reasoning, and sentence-completion tasks. We group the examples by format. For non-true/false questions, the answer choices are listed below the question statement, and the ground-truth label corresponds to the option identifier provided in the original dataset.

Table 8. Sampled Commonsense170k problems used in our experiments, grouped by format. Each instance preserves its native answer format; ground-truth answers are reported using the corresponding label as released. For multi-option items we list the choices in the question column.

qid	Question (and choices, where applicable)	$y^\star$
True/False		
50816f9e	Is there a season 3 of *Wrecked*?	true
4ba7682f	Does Las Vegas have a professional football team?	true
17f50271	Can a person be jailed for civil contempt of court?	true
4d7cce65	Has Maroon 5 ever performed at a Super Bowl?	true
84e37998	Is *I Know Why the Caged Bird Sings* a memoir?	true
4bb2e906	Is *Varsity Blues* based on a true story?	false
433f24bc	Does it count if you hit the backboard on a free throw?	true
b4c59eee	Can you remove the venom glands from a snake?	true
0c7abae6	Is the Statue of Liberty in New Jersey?	false
a461f8a8	Can you send a letter without a return address?	true
f448271b	Is it illegal to pass on a solid yellow line?	false
28225993	Can you score an own goal from a direct free kick?	true
90b6886c	Do red, yellow, and orange peppers taste different?	true
a03e3147	Is there a fifth season of *Mom*?	true
07250c0c	Are *The Five Heartbeats* based on a real group?	false
Two-option pronoun resolution		
263ebb80	Embroidery: when threading the needle, the _ was too thick. *Choices:* (option1) needle; (option2) floss.	option2
c3a49074	"Felicia asked Katrina about new technology because _ was interested." *Choices:* (option1) Felicia; (option2) Katrina.	option1
60091082	Movers wanted to store the boxes in the offices, but the _ were too small. *Choices:* (option1) offices; (option2) boxes.	option1
2e8510fe	Elena loves to read books but Jessica does not; _ bought videos all the time. *Choices:* (option1) Elena; (option2) Jessica.	option2
08a09099	The public elected Jason over Randy, because _ delivered a less persuasive speech. *Choices:* (option1) Jason; (option2) Randy.	option2
Multiple choice		
11aa364a	Addison spent a month of lunches and finally found Carson a place; how would you describe Addison? *Choices:* (answer1) helpful; (answer2) meanspirited; (answer3) selfish.	answer1
bd59d9db	Austin got a PS4 Pro and bought a new TV; what will Austin want to do next? *Choices:* (answer1) play his new game console; (answer2) return the TV; (answer3) go watch a movie at the theatre.	answer1
2ce5f4f0	From which part of the plant does a bee get food? *Choices:* (answer1) flower; (answer2) seed; (answer3) stem; (answer4) root.	answer1
612a4066	Remy came up behind Jan and pushed her in the back very hard; how would you describe Remy? *Choices:* (answer1) physical; (answer2) ready to fight Jan; (answer3) very hostile towards Jan.	answer3
0cd100f4	The best way to improve future production yields on the farm is… *Choices:* (answer1) planting cabbage one year and spinach the next; (answer2) chemical fertilizers and salts; (answer3) rotating water schedules daily; (answer4) over-watering each field.	answer1
25040642	How do you treat period cramps? *Choices:* (solution1) drink some coffee; (solution2) take some Midol.	solution2
ee3abe29	Paper towel: what is it good for? *Choices:* (solution1) clean telescope; (solution2) operate telescope.	solution1
Sentence completion		
3abb847a	"How to start a giving fund" — after opening a bank account, what naturally follows? *Choices:* (ending1) choose checking vs. savings depending on payment frequency; (ending2) a separate account protects you from debt; (ending3) set up a line for the bank's payout the day before your event; (ending4) it isn't complicated — consult an individual bank.	ending1
873e845e	Javelin throw scene: as the man releases the javelin, in the background… *Choices:* (ending1) bleachers in the background, hot and sunny day; (ending2) large crowd standing around, kites flying; (ending3) a big splash, the man runs to rescue it; (ending4) another man in a black t-shirt easily catches it.	ending1
e47d4d28	Drumming and piano duet: a child drums while a woman plays piano along; they… *Choices:* (ending1) continue playing the drums and the music; (ending2) have a small audience watching them perform; (ending3) play till there is no longer a fist drumming in the background; (ending4) play and sing along intently for joy.	ending2