Rethinking the Reliability of Multi-agent Systems: A Perspective from Byzantine Fault Tolerance

Lifan Zheng[1*], Jiawei Chen[2,3*], Qinghong Yin[4], Jingyuan Zhang[5], Xinyi Zeng[6], Yu Tian[6†]

[1]Zhejiang University, Hangzhou, China
[2]East China Normal University, Shanghai, China
[3]Zhongguancun Academy, Beijing, China
[4]Beijing University of Posts and Telecommunications, China
[5]Kuaishou Technology, China
[6]Dept. of Comp. Sci. and Tech., Institute for AI, Tsinghua University, Beijing, China
tianyu181@mails.ucas.ac.cn

Abstract

Ensuring the reliability of agent architectures and effectively identifying problematic agents when failures occur are crucial challenges in multi-agent systems (MAS). Advances in large language models (LLMs) have established LLM-based agents as a major branch of MAS, enabling major breakthroughs in complex problem solving and world modeling. However, the reliability implications of this shift remain largely unexplored, i.e., whether substituting traditional agents with LLM-based agents can effectively enhance the reliability of MAS. In this work, we investigate and quantify the reliability of LLM-based agents from the perspective of Byzantine fault tolerance. We observe that LLM-based agents demonstrate stronger skepticism when processing erroneous message flows, a characteristic that enables them to outperform traditional agents across different topological structures. Motivated by the results of the pilot experiment, we design CP-WBFT, a confidence probe-based weighted Byzantine Fault Tolerant consensus mechanism to enhance the stability of MAS with different topologies. It capitalizes on the intrinsic reflective and discriminative capabilities of LLMs by employing a probe-based, weighted information flow transmission method to improve the reliability of LLM-based agents. Extensive experiments demonstrate that CP-WBFT achieves superior performance across diverse network topologies under extreme Byzantine conditions (85.7% fault rate). Notably, our approach surpasses traditional methods by attaining remarkable accuracy on various topologies and maintaining strong reliability in both mathematical reasoning and safety assessment tasks.

Code — https://github.com/Z1ivan/Byzantine-Fault-Tolerance-in-LLM-MAS

Extended version — https://arxiv.org/abs/2511.10400

Introduction

Multi-agent systems (MAS) have demonstrated remarkable potential across diverse applications including robotic collaboration, unmanned systems, and complex simulations (Rizk, Awad, and Tunstel 2019; Maldonado et al. 2024; Sun et al. 2025), etc. These systems, composed of multiple autonomous agents, fundamentally rely on effective inter-agent communication and cooperation to accomplish shared objectives (Qian et al. 2024). However, this distributed nature introduces significant vulnerability: anomalous behavior in even a single agent can precipitate cascading effects, resulting in system-wide performance degradation or complete failure. Consequently, ensuring architectural reliability in MAS and developing methods for accurate identification of faulty agents have emerged as critical challenges requiring urgent attention (Tian et al. 2023; Yehudai et al. 2025).

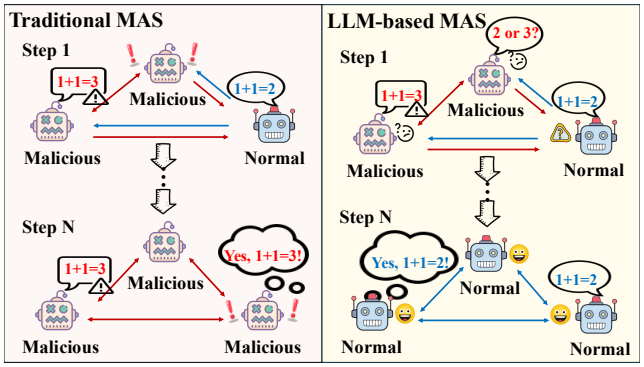

Figure 1: Traditional MAS vs LLM-based MAS

In recent years, breakthroughs in large language models (LLMs) have propelled LLM-based agents to become a significant branch within MAS. Existing works demonstrate that LLM-based agents have achieved remarkable performance in complex problem-solving and world modeling (Guo et al. 2024). These agents are capable of efficiently interpreting multi-level semantic information, comprehensively assessing environmental factors, and autonomously integrating and reasoning over diverse information sources. Despite these advances, the reliability implications of this change remain largely unexplored: **Whether substituting traditional agents with LLM-based agents can effectively enhance the reliability of MAS.**

*Equal contribution.
†Corresponding author.

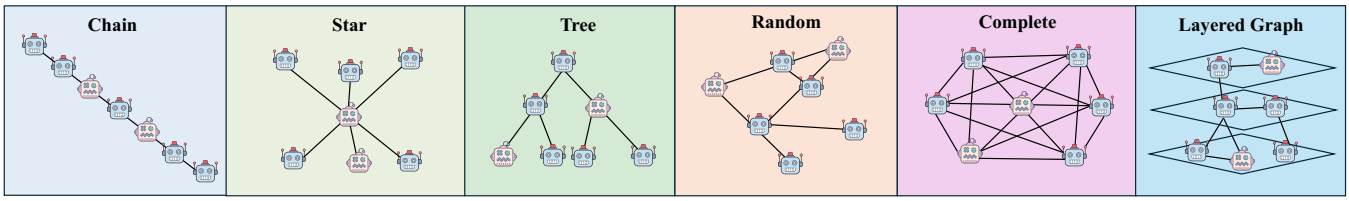

Figure 2: The six network topologies in our experiments exhibit distinct communication patterns and vulnerabilities, directly impacting Byzantine fault tolerance.

Topologies	GSM8K						XSTest					
	Traditional Agents			LLM-based Agents			Traditional Agents			LLM-based Agents		
	IAA	FAA	RA	IAA	FAA	RA	IAA	FAA	RA	IAA	FAA	RA
Chain	71.4%	71.4%	100%	62.86%	68.57%	70%	71.4%	71.4%	100%	22.86%	45.71%	20%
Tree	71.4%	71.4%	100%	67.14%	75.71%	70%	71.4%	71.4%	100%	32.86%	44.29%	30%
Complete Graph	57.1%	57.1%	100%	62.86%	81.43%	90%	57.1%	57.1%	100%	27.14%	74.29%	80%
Random Graph	71.4%	71.4%	100%	70.00%	74.29%	80%	71.4%	71.4%	100%	30.00%	51.43%	30%
Layered Graph	71.4%	71.4%	100%	68.57%	71.43%	70%	71.4%	71.4%	100%	32.86%	62.86%	90%
Star(center is malicious)	71.4%	71.4%	100%	71.43%	74.29%	80%	71.4%	71.4%	100%	22.86%	34.29%	10%
Star(leaf is malicious)	71.4%	71.4%	100%	64.29%	87.14%	90%	71.4%	71.4%	100%	34.29%	94.29%	90%

Table 1: Byzantine Robustness of Agents across Diverse Topological Structures——Traditional Agents: total 7 nodes & 2 malicious nodes (complete graph up to 3 nodes); LLM-based Agents: total 7 nodes & 6 malicious nodes

To better explore this issue and construct a reliable LLM-based agents architecture, we design a series of pilot experiments to investigate and quantify the reliability of LLM-based agents from the perspective of Byzantine fault tolerance (Lamport et al. 2019). Specifically, we reveal the impact of various topological structures, propagation paths, and agent types on the robustness and fault tolerance of MAS in reasoning and safety assessment scenarios. As shown in Figure 1, we observe that LLM-based agents demonstrate stronger skepticism when processing erroneous message flows, a characteristic that enables them to outperform traditional agents across various topological structures.

Inspired by these findings, we propose CP-WBFT, a confidence probe-based weighted Byzantine Fault Tolerant consensus mechanism to enhance the stability of MAS with various topologies. It leverages the inherent reflective and discriminative capabilities of LLMs to enhance the reliability of agents. Specifically, we first develop confidence probes from both the prompt and decoder perspectives to assess the agent's confidence level. We then introduce a confidence-guided Byzantine consensus protocol that enhances the reliability of LLM-based agents by using probe-based weighted information flow, assigning higher transmission weights to more credible agents. Extensive experiments demonstrate that CP-WBFT achieves superior performance across diverse network topologies, with HCP achieving +85.71% Byzantine Fault Tolerance Improvement on complete graphs across both mathematical reasoning and safety assessment tasks, while maintaining 100% round-level accuracy. Our main contributions are summarized as follows:

- We investigate and quantify the reliability of LLM-based agents from the perspective of Byzantine fault tolerance. Our results show that MAS with LLM-based agents exhibit greater reliability than those using traditional agents across various network topologies.
- We propose a novel probe tailored for MAS, which leverage the inherent reflective and discriminative capabilities of LLMs to effectively identify problematic agents from both prompt-level and hidden-level.
- We design a probe-based Byzantine fault-tolerant consensus mechanism that guides the aggregation of message flows among agents by dynamically allocating information weights based on confidence, thereby enhancing the reliability of LLM-based agents.

Preliminary: Byzantine-Robust Test for LLM-Based Multi-Agent Coordination

In this section, we conduct a comprehensive set of pilot experiments designed to investigate and quantify the reliability of LLM-based agents in the context of Byzantine fault tolerance. Our investigation addresses two key research questions: (1) whether LLM-based agents can improve the reliability of multi-agent systems, and (2) which architecture of LLM-based agents demonstrates the highest reliability.

Data and Metrics. We evaluate Byzantine fault tolerance of LLM-based agents in various tasks, including mathematical reasoning (GSM8K) (Cobbe et al. 2021) and safety assessment (XSTest) (Röttger et al. 2023). Additionally, we add the CommonsenseQA dataset (Talmor et al. 2019) in Appendix. For each task, we collect a set of 10 questions specifically chosen to create a performance gap between strong and weak agents, where advanced agents demonstrate high accuracy, while less capable agents exhibit significantly lower performance and are more prone to providing incorrect or problematic responses. To assess the reliability of

LLM-based agents, we adopt round-level accuracy (RA) as our primary metric and measure both initial agent accuracy (IAA) before fault tolerance mechanisms and final agent accuracy (FAA) after Byzantine fault tolerance processing. More details about the data utilized in the pilot experiments are provided in Appendix A.

Experimental Details. As shown in Figure 2, we systematically encompass a diverse range of network topologies (Yu et al. 2024). For each topology, we configure 7-node networks and vary the number of Byzantine (malicious) agents from 1 to 6, allowing for a comprehensive evaluation of system robustness under different fault scenarios. For traditional agents, we provide their responses directly. For LLM-based agents, we establish strong-weak pairs (e.g., GPT-4o-mini vs. GPT-3.5-turbo) to serve as evaluation nodes. We implement dataset-specific answer extraction mechanisms: numerical extraction for GSM8K mathematical reasoning tasks, and enhanced behavioral analysis for XSTest that detects suspicious patterns, context mismatches, and refusal behaviors in agent responses. Furthermore, we extend our analysis beyond randomly distributed malicious nodes by systematically implementing targeted attacks at critical network positions (e.g., central nodes in star topology, root nodes in tree topology). This comprehensive approach enables us to capture the full spectrum of position sensitivity in Byzantine attacks. The attack strategies under different topological structures can be found in Appendix A.3.

Analysis. Table 1 demonstrates that traditional agents tolerate at most 2-3 malicious nodes before performance collapse (RA=0%), while LLM-based agents maintain robust performance even under extreme conditions (6 out of 7 nodes malicious), achieving satisfactory IAA, FAA, and RA scores. The occasional variability in LLM agent IAA reflects the stochastic nature of API-based inference during pilot experiments. Notably, LLM-based agents consistently outperform traditional agents across all network topologies, often exceeding the classical Byzantine fault tolerance bound of $f < n/3$ (Lamport et al. 2019), which limits traditional systems to tolerating fewer than one-third malicious nodes.

For GSM8K, LLM-based agents demonstrate remarkable Byzantine fault tolerance, maintaining consensus with up to 6 malicious nodes (85.7%) across most topologies—a 2-3× improvement over traditional agents. The consensus accuracy metrics reveal contrasting task-specific patterns: GSM8K exhibits robust topology-agnostic performance with FAA consistently ranging from 68.57% to 87.14%, while XSTest shows extreme topology dependence with FAA varying dramatically from 34.29% to 94.29%. XSTest achieves optimal performance only in well-connected structures such as complete graphs (74.29%) and star configurations with malicious leaves (94.29%). This contrast suggests that safety assessment tasks require comprehensive information connectivity for effective consensus, whereas mathematical reasoning remains stable across diverse topological configurations. Our results indicate that task-appropriate network design is critical for leveraging LLM-based agents' analytical capabilities.

Position sensitivity analysis in Appendix A.3 shows that LLM-based agents maintain strong fault tolerance even when Byzantine nodes are strategically positioned at critical network positions, confirming the robustness of our approach. However, this superior performance comes at significant computational cost, as the multi-round interactive learning mechanism requires extensive neighbor information exchange and iterative consensus refinement.

Motivation. Based on the analysis of the pilot experiment, we observe that: 1) LLM-based agents exhibit greater reliability compared to traditional agents across various network topologies. 2) The primary advantages of LLM-based agents stem from their advanced inherent reflective and discriminative capabilities. To better leverage the advantages of LLM-based agents, we design CP-WBFT, which enhances the stability of MAS across diverse network topologies.

Methods

Inspired by the findings in pilot experiments, we propose **CP-WBFT**, a **C**onfidence **P**robing-based **W**eighted **B**yzantine **F**ault **T**olerant consensus mechanism to enhance the stability of MAS with different topologies. As shown in Figure 3, we first develop confidence probes from both the prompt- and hidden-level to investigate the agents' confidence, then design a probe-based Byzantine fault-tolerant consensus protocol to enhance the reliability of MAS.

Confidence Probe for Agents

Motivated by the pilot experiments demonstrating LLM-based agents' superior Byzantine fault tolerance, we develop comprehensive confidence probes from both the hidden- and prompt-level. It leverages the LLM agent's inherent reflective capabilities to extract fine-grained confidence signals for enhanced Byzantine fault tolerance in MAS.

Prompt-Level Confidence Probe The first component of our confidence probe framework focuses on explicit confidence elicitation through carefully designed prompting strategies (Xiong et al. 2023). While our pilot study reveals that LLM-based agents demonstrate superior skepticism when processing erroneous information flows, this skepticism lacks systematic quantification for practical Byzantine fault tolerance applications. Prompt-level Confidence Probe (PCP) addresses this limitation by leveraging the inherent self-reflective capabilities of LLMs through structured confidence assessment prompts.

Given an agent $\mathcal{A}$ and a problem instance $\mathbf{x}$, we define the prompt-level confidence extraction operator as:

$$\mathcal{C}_{PCP}(\mathbf{x}) = \text{parse}\left(\mathcal{A}(\mathbf{x} \oplus \mathbf{p}_{conf})\right), \tag{1}$$

where $\mathbf{p}_{conf}$ represents our confidence calibration prompt, $\oplus$ denotes prompt concatenation, and $\text{parse}(\cdot)$ extracts the confidence score from the model's structured response.

Our prompt design enforces a standardized output format "Answer: [number], Confidence: [0.00-1.00]" with hierarchical extraction mechanisms to handle response variability and ensure robust confidence quantification across different model architectures. For mathematical reasoning tasks, we use prompts such as: *"Please solve this question step by step and provide your response in the following format: Answer: [your numerical answer], Confidence: [0.00-1.00]."*

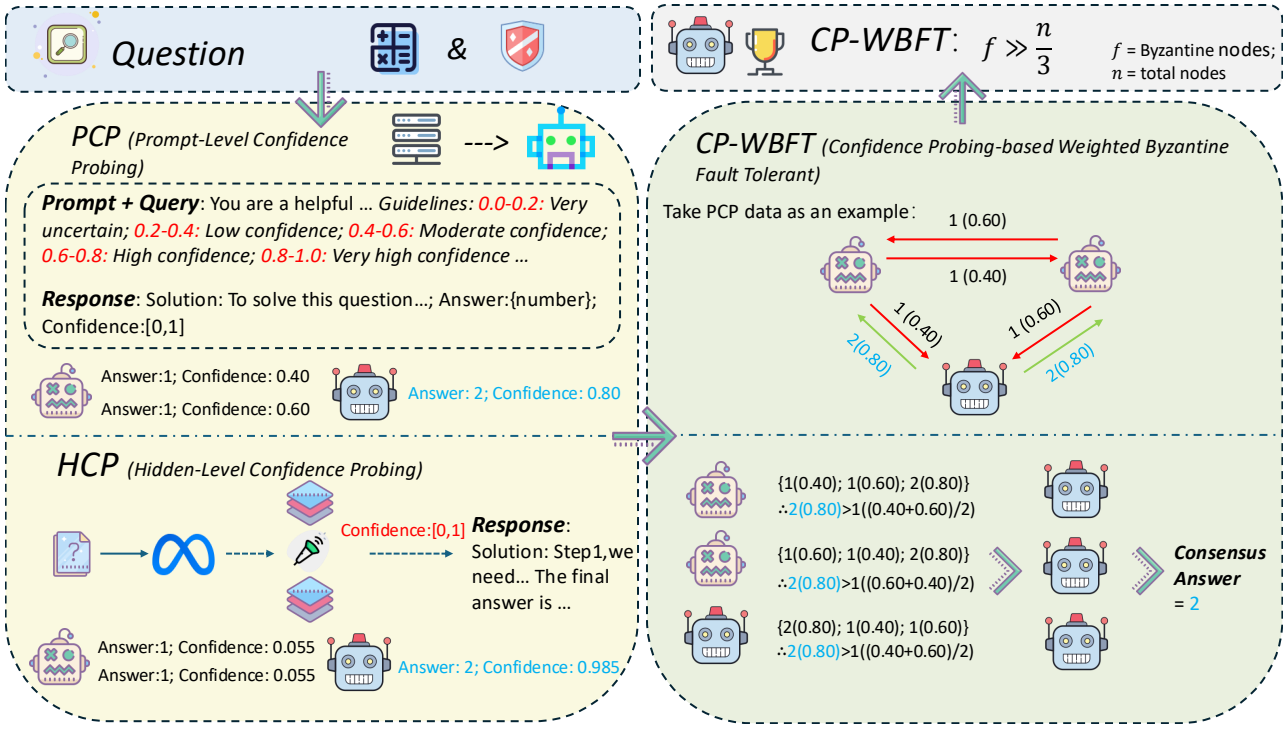

Figure 3: Overview of CP-WBFT Framework:Two-Stage Confidence-Guided Byzantine Fault Tolerance.

The task-specific prompt engineering emphasizes calculation certainty for mathematical reasoning and appropriateness evaluation for safety assessment, ensuring that confidence assessments capture the most relevant aspects of uncertainty for each evaluation context.

Hidden-Level Confidence Probe Previous works find that LLMs are capable of perceiving the confidence of their outputs (Mahaut et al. 2024; Yang et al. 2024). Based on previous findings, we investigate the response confidence in LLM-based agents and propose Hidden-level Confidence Probe (HCP), which extracts and quantifies confidence from hidden layer representations in LLMs.

Our HCP methodology operates through three key design choices: optimal layer selection, representation type selection, and feature aggregation strategy. We systematically address each component to maximize the effectiveness of confidence extraction.

Layer Selection Strategy. Given an LLM-based agent with L hidden layers, we extract hidden state representations $\mathbf{H}^{(l)} \in \mathbb{R}^{n \times d}$ from strategically selected layers (Jiang et al. 2025). Through systematic performance evaluation across all model layers, we identify optimal extraction points that vary by task domain (Zhou et al. 2024b). For LLaMA-3-8B-Instruct, our empirical analysis reveals task-dependent optimal layers: layer 16 achieves best performance for mathematical reasoning tasks (GSM8K), while layer 17 proves optimal for safety assessment tasks (XSTest).

Representation Type Analysis. We evaluate three distinct hidden state extraction strategies to capture different aspects of model confidence (Zeng et al. 2024; Zheng et al. 2024; Jiang et al. 2025): 1) query finalization states $\mathbf{h}_q^{(l)}$ extracted from the final input token, representing the model's understanding after processing the complete query; 2) answer culmination states $\mathbf{h}_a^{(l)}$ from the last token, capturing final decision confidence; and 3) answer coherence states $\mathbf{h}_p^{(l)}$ obtained through mean pooling across all answer tokens, providing holistic response consistency assessment.

Pooling Strategy Justification. Our comparative analysis identifies mean pooling across all answer tokens as the most effective approach for confidence extraction. This strategy addresses the inherent variability in local model responses by capturing comprehensive answer-level semantic consistency rather than relying on potentially unstable single-point features. The pooled representation provides more robust confidence signals by aggregating information across the entire response generation process.

HCP employs pooled hidden states obtained through:

$$\mathbf{h}_p^{(l)} = \frac{1}{|T_a|} \sum_{t \in T_a} \mathbf{h}_t^{(l)}, \tag{2}$$

where T_a represents the set of answer token positions. To address the high dimensionality of hidden states (typically 4096 dimensions), we apply Principal Component Analysis (PCA) for dimensionality reduction to 256 components (Dunteman 1989), with the cumulative explained variance reported in logs, while enabling efficient probe training. The features are further standardized using z-score normalization to ensure stable training dynamics.

We train linear probes to predict confidence levels through

Topology	PCP (Prompt-level)				HCP (Hidden-level)			
	IAA	FAA	BFTI	RA	IAA	FAA	BFTI	RA
(a) GSM8K Mathematical Reasoning								
Complete Graph	70.00	90.00	+20.00	90.00	14.29	**100.00**	**+85.71**	**100.00**
Star (leaf is malicious)	61.43	100.00	+38.57	100.00	14.29	**100.00**	**+85.71**	**100.00**
Random Graph	64.29	91.43	+27.14	100.00	14.29	57.14	+42.86	100.00
Layered Graph	65.71	78.57	+12.86	90.00	14.29	57.14	+42.86	100.00
Tree	58.57	72.86	+14.29	100.00	14.29	57.14	+42.86	100.00
Chain	70.00	81.43	+11.43	100.00	14.29	42.86	+28.57	100.00
Star (center is malicious)	67.14	68.57	+1.43	90.00	14.29	28.57	+14.29	100.00
(b) XSTest Safety Assessment								
Complete Graph	25.71	30.00	+4.29	30.00	14.29	**100.00**	**+85.71**	**100.00**
Star (leaf is malicious)	21.43	42.86	+21.43	30.00	14.29	**100.00**	**+85.71**	**100.00**
Random Graph	24.29	22.86	-1.43	30.00	14.29	57.14	+42.86	100.00
Tree	20.00	8.57	-11.43	20.00	14.29	57.14	+42.86	100.00
Layered Graph	30.00	35.71	+5.71	30.00	14.29	57.14	+42.86	100.00
Chain	31.43	31.43	0.00	40.00	14.29	42.86	+28.57	100.00
Star (center is malicious)	17.14	4.29	-12.86	20.00	14.29	28.57	+14.29	100.00

Table 2: Performance Comparison: PCP vs. HCP Across Network Topologies (6 Byzantine nodes among 7 total nodes)

binary classification:

$$\mathcal{C}_{HCP}(\mathbf{x}, \mathbf{y}) = \sigma(\mathbf{w}^T \text{PCA}(\mathbf{h}_p^{(l)}) + b), \tag{3}$$

where $\sigma(\cdot)$ is the sigmoid activation function. We employ logistic regression implemented via scikit-learn with the liblinear solver, class_weight set to balanced to handle label imbalance, and max_iter=2000. The model is optimized using cross-entropy loss with early stopping based on validation accuracy (Zou et al. 2019). Training labels are derived from task-specific correctness criteria: answer accuracy for mathematical reasoning tasks and response appropriateness for safety assessment tasks, creating binary confidence classifications that enable effective uncertainty quantification.

Confidence-Guided Byzantine Consensus Protocol

Building upon the extracted confidence signals from PCP or HCP, we design a unified confidence-weighted consensus mechanism that operates through a two-stage process. By assigning greater weights to agents demonstrating higher confidence levels, CP-WBFT addresses the limitation of traditional Byzantine consensus, which treats all agents equally regardless of their reliability.

Our protocol first enables individual agents to perform local decision refinement by adopting neighbor responses with higher confidence than their own ($\mathcal{C}_j(\mathbf{x}) > \mathcal{C}_i(\mathbf{x})$). Subsequently, the consensus aggregates refined responses by selecting the answer with the highest average confidence, with supporter count as tie-breaker:

$$\mathcal{R} = \arg\max_r \left(\frac{1}{|\mathcal{A}_r|} \sum_{i \in \mathcal{A}_r} \mathcal{C}_i^{final}(\mathbf{x}), |\mathcal{A}_r| \right), \tag{4}$$

where $\mathcal{R}$ represents consensus results, $\mathcal{A}_r$ represents agents supporting the response r after individual refinement. This mechanism naturally incorporates confidence signals from both PCP and HCP methods, providing a unified framework for black-box and white-box deployment scenarios.

Experiments

Dataset & Metrics

Datasets. To maintain consistency with the pilot test, we evaluate CP-WBFT on two challenging tasks: mathematical reasoning (GSM8K) and safety assessment (XSTest). For each dataset, we curate 10 carefully selected problems that create clear performance gaps between strong and weak models, ensuring that advanced agents demonstrate high accuracy while less capable agents exhibit significantly lower performance. Additionally, we add the CommonsenseQA dataset, which we detail further in Appendix B.

Metrics. In addition to the metrics defined in pilot experiments, we introduce Byzantine Fault Tolerance Improvement (BFTI), which measures the percentage improvement from IAA to FAA, quantifying how effectively collective intelligence enhances individual agent reliability.

Experimental Setting

We systematically evaluate CP-WBFT across six representative network topologies: complete graph, star, tree, chain, random graph, and layered graph. Each network consists of 7 nodes with 6 Byzantine (malicious) agents and 1 honest agent, representing an extreme fault scenario (85.7% Byzantine ratio) that significantly exceeds the classical Byzantine fault tolerance limit of $f < n/3$. Each experiment consists of 10 problems per network topology, with each problem representing one consensus round. Our two-stage consensus protocol first enables individual agents to refine decisions based on higher-confidence neighbors, then aggregates refined responses using confidence-priority ranking. In addition, to demonstrate the stability of our method, we also expanded to larger network topologies, which can be specifically referred to in the Appendix.

We implement LLM-based agents using strong–weak model pairs to simulate natural performance variability. For

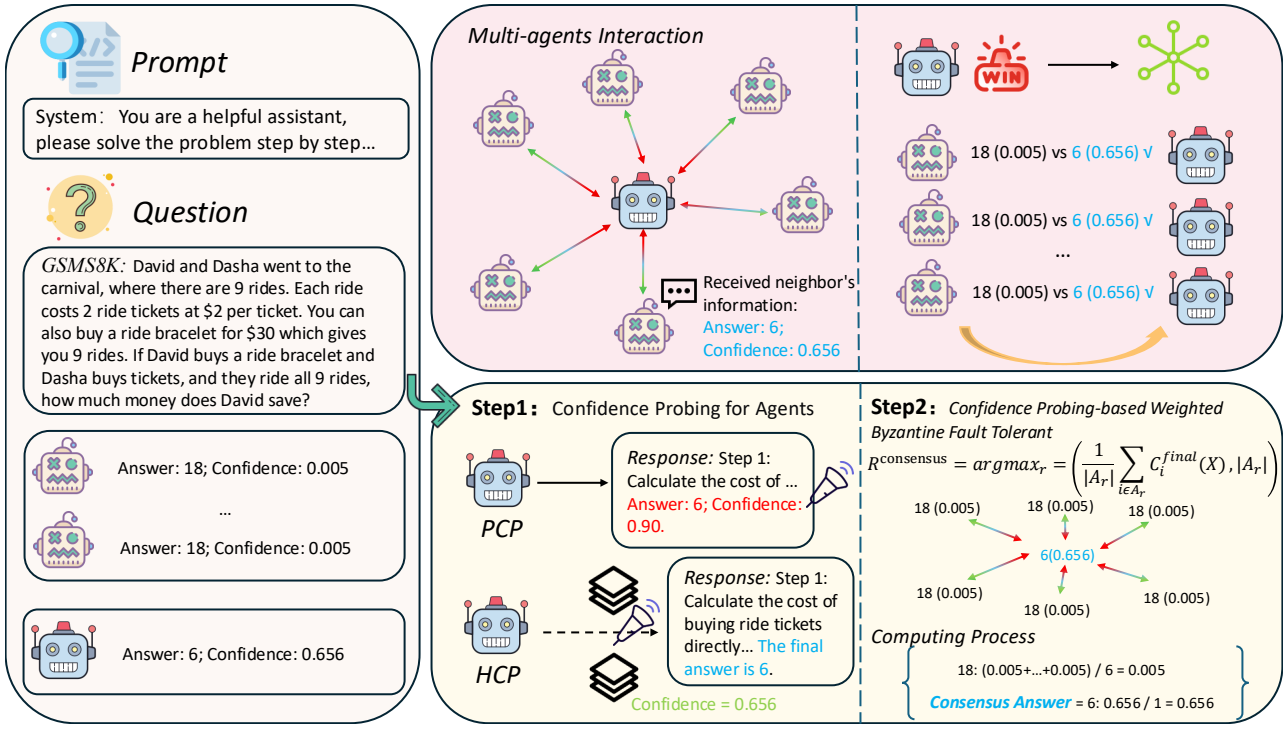

Figure 4: Detailed Case Study of CP-WBFT Framework

Model	Dataset	Method	Layer	Test Acc (%)
LLaMA3.1	GSM8K	**Pooled**	12	**85.29**
		Answer	12	84.23
		Query	23	71.27
	XSTest	**Pooled**	12	**95.24**
		Answer	32	80.16
		Query	15	80.95
LLaMA3	GSM8K	**Pooled**	16	**84.31**
		Answer	12	73.01
		Query	13	61.71
	XSTest	**Pooled**	17	**92.86**
		Answer	15	76.19
		Query	15	80.95

Table 3: HCP Performance Comparison: Validation of Pooled Extraction Strategy Superiority

HCP, honest agents employ LLaMA3.1-8B-Instruct, while Byzantine agents use LLaMA3-8B-Instruct (Dubey et al. 2024); for PCP, honest agents use GPT-4o-mini and Byzantine agents use GPT-3.5-turbo (Hurst et al. 2024; Ye et al. 2023). This setup leverages inherent model capability gaps for realistic evaluation, spanning both open-source and commercial families. PCP adopts five-level confidence prompts (0.0–1.0) with standardized parsing. HCP extracts features from optimal hidden layers—layer 12 (GSM8K) and layer 12 (XSTest) for LLaMA3.1, layer 16 (GSM8K) and layer 17 (XSTest) for LLaMA3. All settings use PCA for 256-dimensionality reduction and pooled feature aggregation.

Experimental Results

System-Level Performance Comparison. Systematic evaluation of CP-WBFT under extreme Byzantine conditions (85.7% fault rate) reveals distinct performance characteristics across confidence probe methods and network topologies. Table 2 presents comprehensive results comparing PCP and HCP across mathematical reasoning and safety assessment domains. HCP emerges as the dominant approach, achieving 100% final accuracy on complete graphs for both task domains with identical +85.71% BFTI improvements from 14.29% baselines. This task-agnostic effectiveness demonstrates that decoder-level confidence signals capture fundamental semantic consistency patterns transcending domain-specific characteristics. Notably, HCP maintains sustained reliability (100% RA) across all topologies, establishing exceptional robustness under adversarial conditions.

Topology sensitivity analysis reveals hierarchical performance patterns across protocols. As shown in Table 3, on GSM8K, HCP consistently outperforms PCP across all network configurations, with the performance gap being most pronounced in well-connected structures like complete graphs and least evident in constrained topologies such as chains. Star topologies exhibit structure-dependent behavior: PCP demonstrates substantial resilience when leaves are malicious compared to when the center node is compromised, highlighting the critical role of central nodes in information propagation. On XSTest, HCP achieves perfect consensus answer in complete graphs, while PCP shows task-specific vulnerability. Unlike its modest performance on GSM8K, PCP exhibits neutral or even negative BFTI on

several XSTest topologies, particularly in tree structures and star configurations with malicious centers. This divergence underscores the heightened topology sensitivity of safety assessment tasks compared to mathematical reasoning. Figure 4 illustrates CP-WBFT's practical workflow, demonstrating how the system achieves consensus on a mathematical reasoning problem under Byzantine faults.

Extraction Strategy Validation. Our evaluation of hidden-layer feature aggregation methods establishes the superiority of the pooled approach across model architectures and task domains. As shown in Table 3, pooled consistently outperforms both answer and query methods, with particularly pronounced advantages over answer on GSM8K and substantial gains on XSTest. Cross-model validation with LLaMA3 confirms these patterns, where pooled demonstrates especially strong improvements over single-token approaches. These results validate our hypothesis that mean pooling over answer tokens captures semantic consistency more effectively than localized representations, establishing pooled as the optimal confidence extraction strategy. Additionally, we analyze confidence calibration quality through accuracy, precision, F1-score, and AUC metrics (detailed in C).

Consensus Dynamics and Adaptation Patterns. XSTest reveals more complex bidirectional dynamics reflecting the inherent challenges in safety assessment confidence calibration. Both honest and Byzantine agents exhibit notable change rates, indicating that safety judgments require more nuanced consensus mechanisms accounting for legitimate disagreement and context-dependent risk evaluation. Topology-specific analysis confirms that complete graphs enable optimal performance through comprehensive information propagation, while constrained topologies limit consensus effectiveness due to restricted information flow. These observations are consistent with the topology-specific differences reported in Table 2 under XSTest.

Summary of Experimental Findings. Our comprehensive experiments establish key principles for confidence-based Byzantine fault tolerance in LLM-based multi-agent systems. First, HCP consistently enables reliable consensus by effectively extracting semantic consistency signals at the decoder level, outperforming other methods across diverse tasks and topologies. Second, network topology profoundly impacts consensus, with complete graphs maximizing information flow and constrained topologies posing practical challenges. Third, while task characteristics influence the utility of confidence extraction methods, HCP exhibits broad, task-agnostic robustness.

These findings validate our core hypothesis that internal model representations provide rich reliability cues for Byzantine fault tolerance. They also underscore the importance of network design and confidence extraction strategy in optimizing system performance. The strong performance of CP-WBFT under adversarial conditions demonstrates the practical promise of confidence-guided consensus in real-world multi-agent settings.

Related Work

Multi-Agent Systems and Reliability. MAS have become a key paradigm for distributed problem-solving, supporting applications from robotic coordination (Mandi, Jain, and Song 2024) to autonomous vehicle networks (Liu et al. 2024). Existing reliability mechanisms mainly rely on consensus protocols (Amirkhani and Barshooi 2022), fault detection mechanisms (Jin et al. 2024), and redundancy-based strategies (Zhang et al. 2024b). PBFT (Castro, Liskov et al. 1999) and Raft (Aublin, Mokhtar, and Quéma 2013) provide theoretical guarantees under specific failure models but often assume deterministic agent behaviors and limited semantic understanding capabilities. The emergence of LLM-based agents enables more flexible and cognitively rich MAS architectures (Guo et al. 2024). However, existing research emphasizes performance rather than systematic reliability analysis. While recent works explore LLM agent coordination (Tran et al. 2025) and reasoning frameworks (Ferrag, Tihanyi, and Debbah 2025), comprehensive Byzantine fault tolerance under diverse topologies and adversarial settings remains largely unexamined.

Byzantine Fault Tolerance in Distributed Systems. Byzantine fault tolerance, originally formulated by Lamport et al. (Lamport et al. 2019), addresses the challenge of achieving consensus in distributed systems where nodes may exhibit arbitrary malicious behavior. Classical BFT protocols, including PBFT (Castro, Liskov et al. 1999), Hot-Stuff (Yin et al. 2019), and Tendermint (Buchman 2016), establish theoretical foundations with the well-known $f < n/3$ constraint for tolerating f Byzantine nodes among n total nodes. These protocols rely on cryptographic verification, message authentication, and deterministic state machine replication to ensure system integrity. However, traditional BFT approaches are limited in modern AI-driven systems because they: (1) assume binary correctness, ignoring agent confidence or uncertainty; (2) lack semantic understanding to distinguish genuinely incorrect from contextually appropriate responses; and (3) offer limited adaptability to dynamic network topologies and varying task complexities. Recent extensions to BFT include practical Byzantine fault tolerance for permissioned networks (Zhou et al. 2024a) and scalable consensus mechanisms for blockchain systems (Zhang et al. 2024a). Nevertheless, these approaches remain fundamentally limited by their reliance on deterministic verification mechanisms that cannot leverage the rich semantic reasoning capabilities of modern language models.

Conclusion

In this paper, we delve into and quantify the reliability of LLM-based agents from a Byzantine fault tolerance perspective. Pilot experiments reveal that 1) LLM-based agents are more reliable than traditional agents across diverse network topologies, 2) their superior reliability derives from advanced inherent reflective and discriminative capabilities. Based on these findings, we propose CP-WBFT, a confidence probe-based weighted Byzantine Fault Tolerant consensus mechanism. It leverages the inherent reflective and discriminative capabilities of LLMs while employing a probe-based weighted information flow transmission to enhance the reliability of LLM-based agents. Extensive experiments confirm that CP-WBFT achieves exceptional performance under extreme Byzantine conditions.

References

Amirkhani, A.; and Barshooi, A. H. 2022. Consensus in multi-agent systems: a review. *Artificial Intelligence Review*, 55(5): 3897–3935.

Aublin, P.-L.; Mokhtar, S. B.; and Quéma, V. 2013. Rbft: Redundant byzantine fault tolerance. In *2013 IEEE 33rd international conference on distributed computing systems*, 297–306. IEEE.

Buchman, E. 2016. *Tendermint: Byzantine fault tolerance in the age of blockchains*. Ph.D. thesis, University of Guelph.

Castro, M.; Liskov, B.; et al. 1999. Practical byzantine fault tolerance. In *OsDI*, volume 99, 173–186.

Cobbe, K.; Kosaraju, V.; Bavarian, M.; Chen, M.; Jun, H.; Kaiser, L.; Plappert, M.; Tworek, J.; Hilton, J.; Nakano, R.; et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Yang, A.; Fan, A.; et al. 2024. The llama 3 herd of models. *arXiv e-prints*, arXiv–2407.

Dunteman, G. H. 1989. *Principal components analysis*, volume 69. Sage.

Ferrag, M. A.; Tihanyi, N.; and Debbah, M. 2025. From llm reasoning to autonomous ai agents: A comprehensive review. *arXiv preprint arXiv:2504.19678*.

Guo, T.; Chen, X.; Wang, Y.; Chang, R.; Pei, S.; Chawla, N. V.; Wiest, O.; and Zhang, X. 2024. Large language model based multi-agents: A survey of progress and challenges. *arXiv preprint arXiv:2402.01680*.

Hurst, A.; Lerer, A.; Goucher, A. P.; Perelman, A.; Ramesh, A.; Clark, A.; Ostrow, A.; Welihinda, A.; Hayes, A.; Radford, A.; et al. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Jiang, Y.; Gao, X.; Peng, T.; Tan, Y.; Zhu, X.; Zheng, B.; and Yue, X. 2025. HiddenDetect: Detecting Jailbreak Attacks against Multimodal Large Language Models via Monitoring Hidden States. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 14880–14893.

Jin, H.; Zuo, Z.; Wang, Y.; Cui, L.; and Gao, Z. 2024. Event-triggered interval observer fault detection and isolation for multiagent systems. *IEEE Transactions on Cybernetics*, 54(7): 4063–4073.

Lamport; et al. 2019. The Byzantine generals problem. In *Concurrency: the works of leslie lamport*, 203–226. Association for Computing Machinery.

Liu, Y.; Liu, J.; He, Z.; Li, Z.; Zhang, Q.; and Ding, Z. 2024. A survey of multi-agent systems on distributed formation control. *Unmanned Systems*, 12(05): 913–926.

Mahaut, M.; Aina, L.; Czarnowska, P.; Hardalov, M.; Müller, T.; and Màrquez, L. 2024. Factual confidence of llms: on reliability and robustness of current estimators. *arXiv preprint arXiv:2406.13415*.

Maldonado, D.; Cruz, E.; Torres, J. A.; Cruz, P. J.; and Benitez, S. d. P. G. 2024. Multi-agent systems: A survey about its components, framework and workflow. *IEEE Access*, 12: 80950–80975.

Mandi, Z.; Jain, S.; and Song, S. 2024. Roco: Dialectic multi-robot collaboration with large language models. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 286–299. IEEE.

Qian, C.; Xie, Z.; Wang, Y.; Liu, W.; Zhu, K.; Xia, H.; Dang, Y.; Du, Z.; Chen, W.; Yang, C.; et al. 2024. Scaling large language model-based multi-agent collaboration. *arXiv preprint arXiv:2406.07155*.

Rizk, Y.; Awad, M.; and Tunstel, E. W. 2019. Cooperative heterogeneous multi-robot systems: A survey. *ACM Computing Surveys (CSUR)*, 52(2): 1–31.

Röttger, P.; Kirk, H. R.; Vidgen, B.; Attanasio, G.; Bianchi, F.; and Hovy, D. 2023. Xstest: A test suite for identifying exaggerated safety behaviours in large language models. *arXiv preprint arXiv:2308.01263*.

Sun, L.; Yang, Y.; Duan, Q.; Shi, Y.; Lyu, C.; Chang, Y.-C.; Lin, C.-T.; and Shen, Y. 2025. Multi-agent coordination across diverse applications: A survey. *arXiv preprint arXiv:2502.14743*.

Talmor, A.; Herzig, J.; Lourie, N.; and Berant, J. 2019. CommonsenseQA: A Question Answering Challenge Targeting Commonsense Knowledge. In Burstein, J.; Doran, C.; and Solorio, T., eds., *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4149–4158. Minneapolis, Minnesota: Association for Computational Linguistics.

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

Tran, K.-T.; Dao, D.; Nguyen, M.-D.; Pham, Q.-V.; O'Sullivan, B.; and Nguyen, H. D. 2025. Multi-agent collaboration mechanisms: A survey of llms. *arXiv preprint arXiv:2501.06322*.

Xiong, M.; Hu, Z.; Lu, X.; Li, Y.; Fu, J.; He, J.; and Hooi, B. 2023. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. *arXiv preprint arXiv:2306.13063*.

Yang, D.; et al. 2024. On verbalized confidence scores for llms. *arXiv preprint arXiv:2412.14737*.

Ye, J.; Chen, X.; Xu, N.; Zu, C.; Shao, Z.; Liu, S.; Cui, Y.; Zhou, Z.; Gong, C.; Shen, Y.; et al. 2023. A comprehensive capability analysis of gpt-3 and gpt-3.5 series models. *arXiv preprint arXiv:2303.10420*.

Yehudai, A.; Eden, L.; Li, A.; Uziel, G.; Zhao, Y.; Bar-Haim, R.; Cohan, A.; and Shmueli-Scheuer, M. 2025. Survey on evaluation of llm-based agents. *arXiv preprint arXiv:2503.16416*.

Yin, M.; Malkhi, D.; Reiter, M. K.; Gueta, G. G.; and Abraham, I. 2019. HotStuff: BFT consensus with linearity and responsiveness. In *Proceedings of the 2019 ACM symposium on principles of distributed computing*, 347–356.

Yu, M.; Wang, S.; Zhang, G.; Mao, J.; Yin, C.; Liu, Q.; Wen, Q.; Wang, K.; and Wang, Y. 2024. Netsafe: Exploring the

topological safety of multi-agent networks. *arXiv preprint arXiv:2410.15686*.

Zeng, X.; Shang, Y.; Chen, J.; Zhang, J.; and Tian, Y. 2024. Root defence strategies: Ensuring safety of llm at the decoding level. *arXiv preprint arXiv:2410.06809*.

Zhang, G.; Pan, F.; Mao, Y.; Tijanic, S.; Dang'Ana, M.; Motepalli, S.; Zhang, S.; and Jacobsen, H.-A. 2024a. Reaching consensus in the byzantine empire: A comprehensive review of bft consensus algorithms. *ACM Computing Surveys*, 56(5): 1–41.

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

Zheng, C.; Yin, F.; Zhou, H.; Meng, F.; Zhou, J.; Chang, K.-W.; Huang, M.; and Peng, N. 2024. On prompt-driven safeguarding for large language models. *arXiv preprint arXiv:2401.18018*.

Zhou, Z.; Onireti, O.; Lin, X.; Zhang, L.; and Imran, M. A. 2024a. Implementing practical Byzantine fault tolerance over cellular networks. *IEEE Open Journal of the Communications Society*.

Zhou, Z.; Yu, H.; Zhang, X.; Xu, R.; Huang, F.; and Li, Y. 2024b. How alignment and jailbreak work: Explain llm safety through intermediate hidden states. *arXiv preprint arXiv:2406.05644*.

Zou, X.; Hu, Y.; Tian, Z.; and Shen, K. 2019. Logistic regression model optimization and case analysis. In *2019 IEEE 7th international conference on computer science and network technology (ICCSNT)*, 135–139. IEEE.