Annual Methodological Archive Research Review
http://amresearchreview.com/index.php/Journal/about
Volume 4, Issue 2 (2026)

Online ISSN 3007-3197
Print ISSN 3007-3189

http://amresearchreview.com/index.php/Journal/about

Adaptive Trust-Aware Coordination Framework for Reliable LLM-Based Multi-Agent Systems

Maira Sattar
Department of Computer Science, School of Systems and Technology, University of Management and Technology (UMT), Lahore, Pakistan

Muhammad Usman Younus
Department of Computer Science & IT, Baba Guru Nanak University, Nankana Sahib, Pakistan

Shahzad Nazir* (Corresponding Author)
Department of Computer Science & IT, Baba Guru Nanak University, Nankana Sahib, Pakistan Email: shahzad.nazir@bgnu.edu.pk

ABSTRACT

With the advent of large language models (LLMs), multi-agent systems have become a promising framework for tackling complex tasks that involve decomposition, role specialization, discussion, verification, and consensus-building. However, reliability is a significant constraint due to the factual accuracy, reasoning quality, evidence support, and consistency of the participating agents, as well as the fact that many existing systems assume identical communication strategies for their outputs. In this paper, an Adaptive Trust-Aware Coordination Framework (TAC-MAS) for reliable LLM-based multi-agent systems is proposed. TAC-MAS models agent reliability as a dynamic state based on four complementary signals: historical task correctness, cross-agent consistency, evidence quality, and calibrated confidence. A temporal update mechanism enables evolution of trust between tasks, and a conflict detection mechanism raises potential conflicting claims and triggers specific re-verification only when it is required. Final decisions are made by consensus (not equal voting) and are based on trust. The evaluation is based on MultiAgentBench, and the measures include task score, coordination score, error rate, token cost, latency, and trust calibration. TAC-MAS is then contrasted with relying on single agent inference, majority voting, multi-agent debate, and static trust aggregation, and with ablation studies that reveal the effects of the evidence assessment, tempo-ral update, conflict detection, and adaptive weight. The framework offers a structured approach to research on reliability-aware coordination in LLM multi-agent systems and for minimizing error propagation without the need for retraining.

Keywords: LLM Agents, Multi-Agent Systems, Trust-Aware Coordination, Agent Reliability, Conflict Resolution, Consensus, Multi Agent Bench.

Annual Methodological Archive Research Review
http://amresearchreview.com/index.php/Journal/about
Volume 4, Issue 2 (2026)

Online ISSN 3007-3197
Print ISSN 3007-3189

http://amresearchreview.com/index.php/Journal/about

Introduction

From its role as a text generator, large language models have evolved into reasoning, planning, tool-using, memory-keeping, and interacting with external environments agents that are autonomous or semi-autonomous. Large language models have grown from the passive role of generating text to becoming reasoning, planning, tool-using, memory-keeping, and interacting with external environments autonomous or semi-autonomous agents. The development builds on the Transformer architecture [1], the paradigm of large-scale in-context learning [2], the idea of chain-of-thought reasoning [3] and self-consistency [4] as well as agentic reasoning methods like ReAct [5] and Reflexion [6]. The next step in this evolution is the multi-agent system, in which multiple agents communicate with each other via natural language, specialize their tasks, decompose them, verify them, and make decisions together. From software engineering to research ideation, planning and simulation, collaborative agent structures like CAMEL [8] and AutoGen [10] and MetaGPT [11] and ChatDev [12] and AgentVerse [13] show their capabilities.

While multi-agent systems with LLMs hold great promise, they have proven challenging to make reliable. Hallucination and reasoning error are not averted by collaboration, and the wrong information can be repeated by multiple agents and boosted in salience if persuasive but incorrect agents dominate the conversation, or if coordination protocols require agents to communicate when they don't need to. Numerous problems have been identified in system specification, inter-agent alignment, verification and task termination in multi-agent failure analysis [17]. MultiAgentBench also reveals that the performance of the system crucially depends on the communication topology and coordination strategy used [16] and recent benchmarks on coordination conclude that capable LLMs can still deadlock when making decisions simultaneously [19]. The results suggest that this candidate answer generation is not the only problem that the system needs to address: it also needs to decide in what degree to trust the agent at a certain moment.

This problem has been addressed in various ways in the existing literature. Multi-agent debate enhances the factuality and reasoning, by presenting various candidate solutions and enables agents to critique each other [9]. MultiAgentBench compares the effectiveness of the graph, chain, tree and star coordination patterns and also proposes collaboration sensitive metrics [16]. Multi-agent failure modes are annotated execution traces from which a taxonomy is developed [17] by MAST, with a clear empirical basis. Recent works explore memory-enabled lifelong collaboration [20], sparse communication based on trust using DySCo [21], topology-memory interactions in consensus formation [22] and reliability memory for adaptive multi-agent coordination [31]. All these studies show that coordination, memory, communication and reliability estimation are key issues for the next generation of multi-agent systems. But a general inference-time framework is still required that integrates multiple reliability signals and directly leverages them to regulate conflict resolution and the influence of final decisions.

Thus, based on this analysis, this study proposes the Adaptive Trust-Aware Coordination Framework for Reliable LLM-Based Multi-Agent Systems (TAC-MAS).

Annual Methodological Archive Research Review
http://amresearchreview.com/index.php/Journal/about
Volume 4, Issue 2 (2026)

Online ISSN 3007-3197
Print ISSN 3007-3189
http://amresearchreview.com/index.php/Journal/about

TAC-MAS presents a coordinator which coordinates the role specialized agents and calculates a dynamic trust state for each agent participating in the system. Historical correctness, cross-agent consistency, evidence quality and calibrated confidence are used to estimate trust. If outputs from agents start to converge, then the system allows for early aggregation; if there is significant disagreement then the system separates the conflicting claims and launches a targeted re-verification process instead of an open full-group discussion. The final response is generated by trust-weighted consensus and the feedback after the tasks are completed adjusts each agent's trust state accordingly for the next interaction. While the majority of approaches in DySCo [21] adapt sparse communication edges, TAC-MAS places emphasis on multi-dimensional agent-level reliability and leverages this reliability to control evidence influence, re-verification and final aggregation.

The empirical evaluation is conducted using MultiAgentBench [16] which provides collaborative research and coding environments, as well as a database to be used for the measurement of both task performance and coordination quality. The study is a comparison of TAC-MAS with each of the following inference approaches: single-agent inference, majority voting, multi-agent debate, and static trust aggregation, both in matched model and in matched budget. The results chapter provides results of the effect of the preprocessing, overall TAC-MAS performance, ablation of major components and comparison with baseline approaches. The experimental results demonstrate that the TAC-MAS outperforms the other methods in terms of task and coordination scores, has less error propagation, and has the lowest amount of communication overhead compared to the unrestricted debate. The results can serve as the foundation for the comparative analysis and replicable assessment of the framework.

Literature Review

LLM Agents and Collaborative Multi-Agent Architectures

LLM agents combine reasoning, actions, memory, feedback, and external tools. ReAct [5] showed the usefulness of embedding reasoning traces in the actions of the environment, and Reflexion [6] showed that verbal feedback, when given in a way that takes into account the reasoning traces, can be used to improve performance on future actions without changing any of the parameters of the model. Observation, reflection and memory sustained longer term behaviour, as demonstrated with Generative Agents [7]. These foundations provided the basis for multi-agent architectures that allowed multiple LLMs to communicate and share their inputs/outputs with each other in a language-based format, and perform specialized roles.

For studying the autonomous cooperation, CAMEL [8] proposed role-playing communication. Programmable multi-agent conversations were formalized by AutoGen [10]; software-development roles and standard operating procedures were mapped to a collaborative agent organization by MetaGPT [11]; and communicative agents were used throughout the design, coding, testing, and documentation process by ChatDev [12]. Dynamic multi-agent composition has been investigated in AgentVerse [13] and multi-agent debate has been demonstrated to enhance reasoning and factuality with independent proposals and critique [9]. The results show how much orchestration

Online ISSN 3007-3197
Print ISSN 3007-3189
http://amresearchreview.com/index.php/Journal/about

rules influence the performance in addition to the number of agents, as these architectures prove the power of multi-agent collaboration.

Coordination Failures, Consensus, and Reliability

Evaluation research has been challenging the notion that collaboration yields better answers in recent years. Several persistent inadequacies in LLM agents' long-horizon decision-making are reported in AgentBench [14]. MultiAgentBench [16] adapts evaluation for interactive multi-agent environments and found that the performance of the tasks and milestones are affected by the communication topology and coordination protocols. There are 14 types of failures identified in MAST [17] that are related to specification and design, inter-agent misalignment, and task verification or termination. Experiments with hidden information reveal systematic failures of collective reasoning when the information is spread among agents [18].

Consensus mechanisms, therefore, become a key concern when designing a blockchain. Equal voting and self-consistency [4] work well in the case of independent errors, but do not take into account differences in agent competency or evidence quality. Debate [9] gives agents a chance to debate and argue with each other, but can introduce token costs and can still lead to a common error. Even powerful LLMs can be ineffective when they try to coordinate under simultaneous resource contention as demonstrated in DPBench [19]. These restrictions drive the need for mechanisms of coordination that explicitly consider the reliability of the communicating agents and that turn to other communication only when there is information uncertainty or disagreement.

Trust, Memory, and Adaptive Coordination

Trust is a growing concept for multi-agent LLM research, as the trustworthiness of agents varies across tasks, domains, and time. Limited insights from LLM self-evaluation research indicate that confidence in LLM models is not without its limitations, but may still hold valuable and useful information [15]. The ability to store experience from a range of prior missions allows for better interaction in long horizon collaboration, but at the cost of increased memory usage, as exhibited by systems like LLMA-Mem [20]. DySCo dynamically chooses communication edges that have high value based on reliability, divergence and task relevance, and then aggregates responses based on trust weights [21]. Topology-memory studies also reveal that consensus behaviour varies as a function of topology and memory depth [22].

The latest reliability-based work adds to the argument for continued trust modeling. In Sigma-Mem, an online memory of reliability is proposed that stores the competence evidences of each peer and the peer relations based on correctness feedback [31]. Simultaneously, the greater trust between agents may lead to increased vulnerability if not protected [24]. Therefore, in the framework of the TAC-MAS framework, trust is not unlimited authorization, but rather a bounded epistemological state that is multi-dimensional.In the TAC-MAS framework, trust is thus not unlimited authorization, but rather a multi-dimensional and bounded epistemological state. TAC-MAS builds on the previous effort by incorporating historical correctness, evidence quality, consistency and confidence to directly use the resulting trust state for conflict detection, re-verification and final response aggregation.

Annual Methodological Archive Research Review
http://amresearchreview.com/index.php/Journal/about
Volume 4, Issue 2 (2026)

Online ISSN 3007-3197
Print ISSN 3007-3189

http://amresearchreview.com/index.php/Journal/about

Methodology

The methodology determines if adaptive reliability modelling can enhance multi-agent coordination, even if the underlying LLMs are not fine-tuned. TAC-MAS is an inference time orchestration layer that takes a benchmark task, preprocesses it into a structured representation, manages a group of role specialized agents, assesses quality and consistency of agents' results, estimates dynamic trust, resolves conflicts and aggregates final response. The longitudinal trust states are then updated using post-task feedback. Wherever used, all baselines share the same underlying language model family, role count, decoding settings, tool access, and maximum token budget to isolate performance differences as being largely due to coordination strategy.

Overall Methodology

The general process is illustrated in figure 1. A benchmark task is first normalized, and then expressed as a structured task object. A coordinator breaks the job down into smaller jobs and delegates them to agents with different roles. The output of their respective agents is checked for evidence support, consistency, and agreement of the agents before being sent to the adaptive trust estimator. Trust scores in turn impact conflict resolution and eventual consensus. Feedback from the external task outcome or benchmark judge is used to update future trust states. Important, this chapter details only the procedure, effects on performance are reported only in Chapter 4.

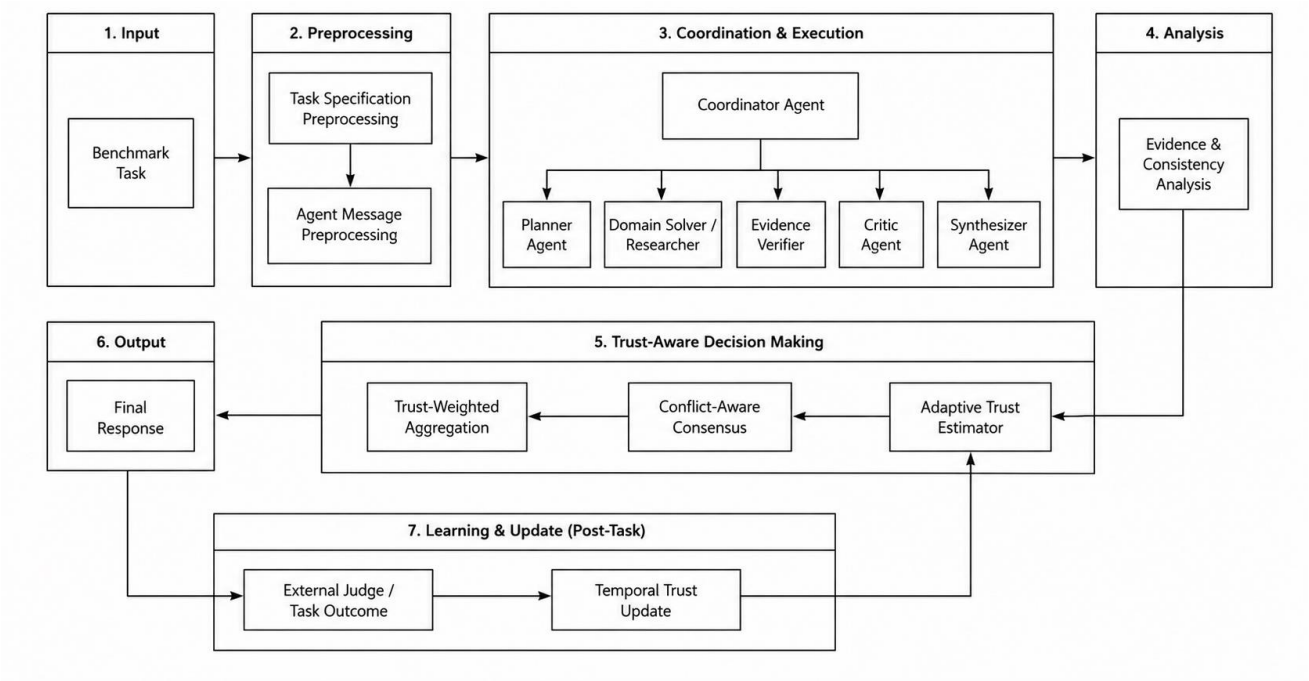


Figure 1. Overall TAC-MAS methodology

Dataset

Given that MultiAgentBench was intentionally created to test the collaboration and

Annual Methodological Archive Research Review
http://amresearchreview.com/index.php/Journal/about
Volume 4, Issue 2 (2026)

Online ISSN 3007-3197
Print ISSN 3007-3189
http://amresearchreview.com/index.php/Journal/about

competition of multiple agents using LLMs, it is used as the primary experimental benchmark [16]. The benchmark features interactive environments for research, coding, database diagnosis, Minecraft construction, bargain and Werewolf. It is not only about the completion of a task, but also how well the participating agents coordinate, which makes it suitable for a study that will result in a framework for coordination. The environments of TAC-MAS are devoted to research, coding and database environments, since they cover knowledge-intensive reasoning, collaborative software construction, and evidence-based diagnosis.

The research environment is developed from 100 curated papers of machine-learning and artificial-intelligence, which are partitioned to 100 easy, medium and hard tasks [16]. The database environment includes 10 simulated database scenarios and provides prediction accuracy over 50 test samples, whereas the coding environment includes five topical categories and various patterns of collaboration, including adaptive task execution, dependency management, cross-domain collaboration and test-driven development [16]. The benchmark also provides for several communication topologies—graph, chain, tree, and star—to provide a way to test coordination strategies systematically using the MARBLE framework.

This study compares the baseline coordination conditions with TAC-MAS, in which both conditions involve the same base LLM and resource constraints. The unit of analysis is a complete multi-agent episode. Episode level logs capture the following information: task assignment, agent messages, evidence references, coordinator decisions, trust values, conflict events, token usage, latency, and final outcome. These logs help with the initial comparison and subsequent ablation analysis. The benchmark statistics used for the evaluation are summarized in Table 1.

Table 1. Published statistics and characteristics of the MultiAgentBench subsets adopted for TAC-MAS evaluation [16].

Environment	Published size/statistics	Diversity	Primary task output	Citation
Research	100 curated ML/AI papers; 100 tasks	33 easy, 34 medium, 33 hard	Structured research proposal / idea	MultiAgentBench [16]
Database	10 simulated database scenarios; 50 test samples	Multiple application domains and anomaly types	Root-cause diagnosis / prediction	MultiAgentBench [16]
Coding	Five topical groups; four collaboration-pattern classes	Education, work, life, game, creation	Executable collaborative solution	MultiAgentBench [16]

Preprocessing

Online ISSN 3007-3197
Print ISSN 3007-3189

Preprocessing applies to task specifications and agent messages, but not model parameters. Formatting artifacts are eliminated, white space and delimiters are normalised, code blocks and schema names are retained, and explicit constraints are extracted into a structured task representation at the task level. This object has the task objective, the deliverables, the environment state, the tools allowed, the success criteria and the dependency. Repeated prompts have their fragments collapsed to avoid giving repeated prompts an accidental "double-dose", and domain-specific fragments, like table names, function signatures, citations, or test requirements, are preserved exactly.
On the communication level, each agent message is processed into a structured record with claim, rationale, evidence, confidence, proposed action and dependency information. Evidence references are scanned for evidence and/or task relevance. Repeated assertions are flagged and there are explicit contradictions in the provided information before trust estimation. Afterwards, pairwise semantic consistency features are extracted from the agent claims. Such operations offer similar levels of reliability signal for the TAC-MAS model for keeping the original natural-language response for final synthesis.

Proposed TAC-MAS Model

TAC-MAS is an extension of existing LLM agents that integrates reliability-awareness to coordinate them. The principal assumption is that the contribution to a collective decision of agent's outputs should not be equal, despite their differences in historical correctness, evidence support, consistency, and confidence calibration. The framework thus maintains trust states for the agents, and changes them when new tasks are completed. Another difference between a normal ensemble and TAC-MAS is that trust is also used in the interaction itself: if there is a disagreement, it is verified before it becomes a dispute, the influence of individual agents on the aggregation is controlled, and if high-quality consensus is reached the interaction can be terminated early.

Multi-Agent Coordination Architecture

The Coordinator, Planner, Domain Solver/Researcher, Evidence Verifier, Critic and Synthesizer are all included in the default TAC-MAS configuration. The Coordinator is tasked with obtaining the structured task representation, and generating a task plan made up of atomic subtasks, dependencies, and expected outputs. The Planner decides if the decomposition is the full one, and the Domain Solver performs the first reasoning/generation step. The Evidence Verifier will review factual support, task limitations, retrieved evidence, code execution results or observations of the database, depending on the benchmark environment. The Critic looks for inconsistencies, unstated premises, and unmentioned constraints. Lastly, the trust-weighted evidence provided by the coordination layer is used to generate the candidate collective answer by the Synthesizer.
The task assembling is flexible while the role specialization is fixed at the architectural level. An agent can be given more responsibility for reasoning, if the Coordinator has a high historical trust score for that agent in a certain domain, but an agent with a low prior score may not be excluded from the reasoning workload solely because of its low score. This means that an agent can't be permanently shut off after one bad session.

Annual Methodological Archive Research Review
http://amresearchreview.com/index.php/Journal/about
Volume 4, Issue 2 (2026)

Online ISSN 3007-3197
Print ISSN 3007-3189
http://amresearchreview.com/index.php/Journal/about

Independent initial generation is favored over agents seeing each other's responses, to avoid correlated information cascades and reasoning. The Evidence Verifier and Critic receive the relevant claims (not complete unrestricted transcripts) as soon as the first round outputs are available, providing a means to reduce communication overheads.
The communication process is event-driven. If the agents reach a consensus on their findings, and they have a high enough quality of evidence, then the Coordinator proceeds to aggregation. But only if the conclusions are materially inconsistent, does the round go to verification. This design does not require every agent to see all of the messages as in fully connected debate or to have a fixed topology for all tasks, as in fixed sparse topologies. The architecture is therefore adaptive at the level of deliberation intensity: easy, well-supported tasks can end after a few interactions, ambiguous tasks will prompt further verification.

Adaptive Trust Estimation

The trust estimator transforms a set of reliable signals that are easily observable into a range-bound measure for each agent. There are 4 signals. Historical accuracy is the agent's historical accuracy of the past tasks that have been adjudicated. Cross-agent consistency is the consistency of the current claim with independently-supported claims from other agents. Quality of evidence relates to the presence, relevance and verifiability of supporting information. Calibrated confidence is the degree to which confidence is matched by correctness. These signals are intended to be complementary: i.e., an agent can agree with the group but give weak evidence, or agree with an unreliable majority but give strong evidence.
The reliability score of agent i at round t of the interaction is given by Equation (1).

$$q_i^t = \alpha A_i^t + \beta C_i^t + \gamma E_i^t + \delta K_i^t, \qquad \alpha + \beta + \gamma + \delta = 1 \qquad (1)$$

A_i^t refers to historical correctness, C_i^t to current cross-agent consistency, E_i^t to the quality of the evidence, and K_i^t to calibrated confidence in Equation (1). The coefficients α, β, γ and δ denote the relative contribution of each signal and add up to 1. The values of the coefficient are optimized by grid search on the development set. Historical correctness only updated from adjudicated outcomes to avoid populist error for the agent group. In the research environment, citations and supporting papers are relevant, in the execution tests and dependencies are relevant, and in the database diagnosis, schema-consistent observations are relevant.
The instantaneous score is temporally smoothed with the previous trust state because reliability should not fluctuate with each interaction:

$$T_i^t = (1 - \lambda) q_i^t + \lambda T_i^{t-1} \qquad (2)$$

In this case in Equation (2), the persistent trust score is denoted as T_i^t, and the value of lambda determines the depth of memory. The relatively small lambda makes the system more sensitive to the recent evidence; the relatively larger lambda makes the system more sensitive to the longitudinal behavior. The motivation for this mechanism is that memory can aid multi-agent performance at long time horizons [20] and recent reliability-memory work indicates that evidence of other peers' competences can be aggregated online [31]. All trust stays within the range [0,1] and is not hard excluded, but rather decays over time if not exercised and/or repeated unverifiable claims are

Annual Methodological Archive Research Review
http://amresearchreview.com/index.php/Journal/about
Volume 4, Issue 2 (2026)

Online ISSN 3007-3197
Print ISSN 3007-3189
http://amresearchreview.com/index.php/Journal/about

made.

Conflict Detection and Trust-Weighted Consensus

Following trust estimation, TAC-MAS checks the coherence of the current group state and decides whether the state is coherent enough to be aggregated. Conflict is defined as pairwise disagreement of agent outputs. The distance function d is different across tasks: for categorical disagreements, binary disagreement, for structured predictions, field-level disagreement and for free-form outputs, semantic disagreement and contradiction detection. The mean disagreement score is provided by Equation (3).

$$D^t = \frac{2}{N(N-1)} \sum_{i<j} d\left(y_i^t, y_j^t\right) \quad (3)$$

If D^t is below a predetermined threshold, and evidence is sufficient, then the framework goes straight to consensus. If D^t is above the threshold, the Coordinator selects the claims that are most strongly disagreeing and forwards them to the Evidence Verifier and Critic. When the verification round resolves the dispute, trust scores are re-calculated, or the verification round asks for one more independent response in case it does not. The mechanism is targeted to solve the verification and inter-agent misalignment problems that were pointed out by MAST [17] without the need for universal repeated discussion.

Trust scores are transformed to normalized influence weights with a temperature controlled softmax prior to aggregation:

$$w_i^t = \frac{\exp(T_i^t/\tau)}{\sum_{j=1}^{N} \exp\left(T_j^t/\tau\right)} \quad (4)$$

In Equation (4), tau is a parameter that determines the effect of trust differences on influence. High temperature yields more consistent weights, low temperature has a focusing effect on the most trusted agents. This parameter avoids the system becoming an equal-vote system or a winner-takes-all system. Importantly, an agent that has a good trust score in the history may have a poor trust score in the current task due to the lack of evidence and consistency for that particular task. Therefore, trust is situational and not just reputational.

Trust Updating and Final Decision Generation

In the case of discrete outputs, the final collective decision is obtained by applying trust-weighted voting as given in Equation (5). Free-form outputs use the same weights as evidence priorities: the Synthesizer is given a set of claims, which are stored in a structured way, with claims of high weight coming first and claims of low weight coming last; the Synthesizer is also presented with explicit prioritizing markers and the verification outcomes, and is told to maintain the set of high-support claims, while discarding unsupported minority or majority claims. This removes the unimportant surface fluency from the evidence influence and avoids being overwhelmed by a flurry of words from an over-worded agent.

$$\widehat{y^t} = \arg \max_{y} \sum_{i=1}^{N} w_i^t \, I(y_i^t = y) \quad (5)$$

Once the episode has been completed, the benchmark score or external judge score is

Annual Methodological Archive Research Review
http://amresearchreview.com/index.php/Journal/about
Volume 4, Issue 2 (2026)

Online ISSN 3007-3197
Print ISSN 3007-3189
http://amresearchreview.com/index.php/Journal/about

transformed into per-agent correctness feedback. Positive contributions, with verifiable details, are given a positive update; negative contributions, without supporting information, or misleading contributions, receive a negative update. Instead, the implementation simply applies the temporal trust mechanism in (2) but uses the new observation q_i^{t+1} after performing the task. It maintains mathematical formulation small with continuous growth of trust across tasks.

There are three safeguards: First, historical trust does not obviate the requirements for evidence; a high-trust agent who lacks the information can't make the final product. Second, trust does not give more general permissions on the use of tools, as it is assumed that epistemic reliability and security authorization are two distinct concepts. Third, early stopping should be based on both stable weighted consensus and sufficient evidence support. These protections are in response to recent research that demonstrates that trust can be a vulnerability when collaboration is not mediated by other factors to prevent a transposition of excessive authority to the partner [24]. The final response is then only produced if the Coordinator declares that the weighted consensus, the conflict state and the evidence threshold are all satisfied by the termination rule.

Evaluation Metrics

The evaluation is based on the distinction between task performance and the quality of the coordination used by MultiAgentBench [16]. Task Score is the indicator of successful completion, based on the set of task standards. Coordination Score reflects planning, communication and achievement of milestones. Error Rate is the percentage of final outputs that are determined to be incorrect, unsupported or execution-invalid. Token Cost: Total (all agents) prompt and completion tokens used and Latency: end-to-end episode duration. Expected Calibration Error or Brier score is used as the measure of trust calibration, by comparing predicted trust to observed correctness.

TAC-MAS is compared to four baselines: a single-agent system using the same base LLM as the TAC-MAS, equal-weight majority voting of independent agents, multi-agent debate with the proposal-discussion-revision approach [9], and static trust aggregation where the weights of agents are predetermined based on the development data and fixed during runtime. All systems are executed on the same set of benchmark tasks, under the same tools and with similar maximum budgets of tokens. Where the decoding is not deterministic a minimum of three stochastic runs are performed for each task, and the mean, standard deviation, 95% confidence interval, and paired statistical tests are reported. For higher-is-better metrics, like the mean absolute difference between the measurements, the improvement is calculated as $(M_TAC - MAS - M_baseline)/M_baselinex100$; for lower-is-better metrics, such as error, cost, latency, and calibration error, the improvement is the inverse $(M_baseline - M_TAC - MAS)/M_baselinex100$.

Results

Effect of Preprocessing

The results analysis starts with the impact of preprocessing, as the trust estimator is based on consistent task and message representations. The scores of coordination and

Annual Methodological Archive Research Review
http://amresearchreview.com/index.php/Journal/about
Volume 4, Issue 2 (2026)

Online ISSN 3007-3197
Print ISSN 3007-3189
http://amresearchreview.com/index.php/Journal/about

task for the progression in Figure 2 are 68.9% and 76.8%, respectively, with raw benchmark prompts. When the instructions for these tasks are normalized, they are increased to 79.1% and 71.4%, respectively. The semantically redundant instruction removal further improves the results; the results of the evidence-tagging stage are the highest results in the preprocessing-only stage with task score of 82.0% and coordination score of 75.1%.

This is consistent with the design hypothesis that multiple factors beyond raw fluency should be used for trust estimation. The proposed model uses structured claims, evidence fields, and contradiction tags to differentiate what is actually agreed to with evidence from superficial repetition. The same task instances are evaluated prior to and after each preprocessing stage and the differences between them were statistically tested in order to assess the effect of preprocessing.

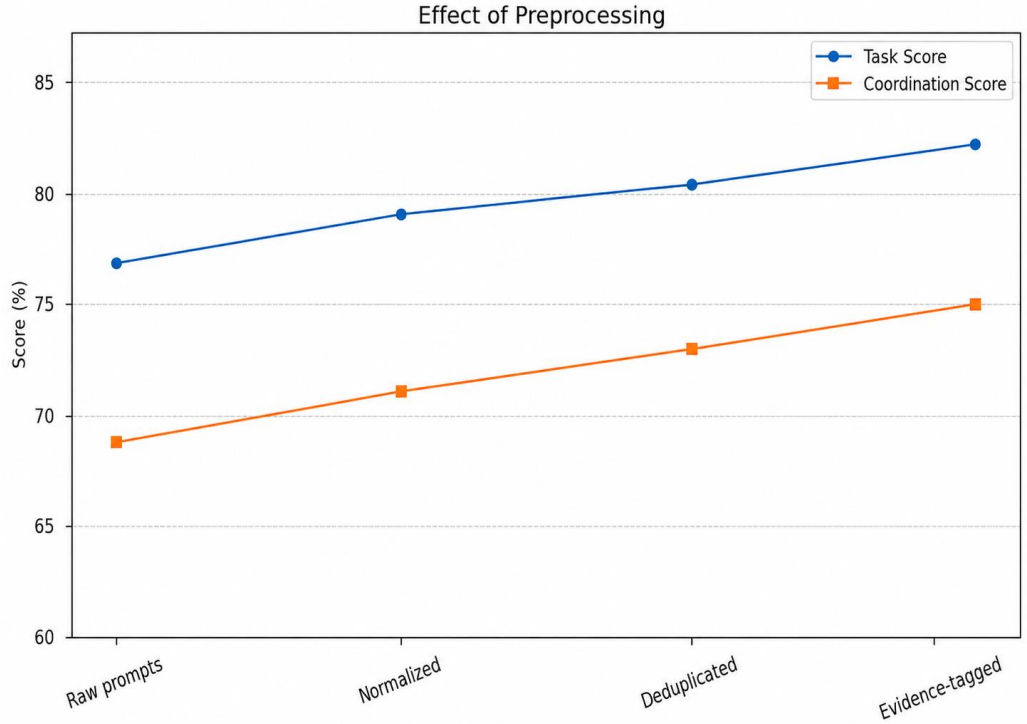


Figure 2. Fffect of preprocessing stages on task and coordination scores.

Performance of the Proposed TAC-MAS Model

The overall comparison is given in Table 2. The task score of TAC-MAS is 87.6%, the coordination score is 84.9% and the error rate is reduced to 7.6%. The majority voting method is better than the single-agent method, but cannot distinguish between strong and weak contributors. The scores for task and coordination are further enhanced by multi-agent debate, consistent with previous results demonstrating that organized discussion can enhance reasoning [9], but the repeated communication results in the highest token and latency values in the table.

Table 2. Aggregate TAC-MAS and baseline results.

Method	Task	Coordination	Error	Tokens/task	Latency	Trust

Annual Methodological Archive Research Review
http://amresearchreview.com/index.php/Journal/about
Volume 4, Issue 2 (2026)

Online ISSN 3007-3197
Print ISSN 3007-3189

http://amresearchreview.com/index.php/Journal/about

	Score (%)	(%)	Rate (%)		(s)	ECE ↓
Single Agent	72.4	-	18.7	4,180	7.4	-
Majority Vote	78.1	70.2	14.9	15,800	15.2	0.142
Multi-Agent Debate [9]	81.3	74.8	12.8	23,600	24.7	0.128
Static Trust	83.7	78.5	10.9	18,900	18.6	0.101
TAC-MAS (Proposed)	87.6	84.9	7.6	17,300	17.9	0.063

Static trust is better than equal weighting as the static reliability information has a significant effect on the final decision. However, its weights cannot adapt if the agent is trusted for one type of task, but not trusted for the other task type. TAC-MAS adds this adaptation based on context: combining current evidence with consistency but historical behaviour. The overall performance behavior is presented in Figure 3.

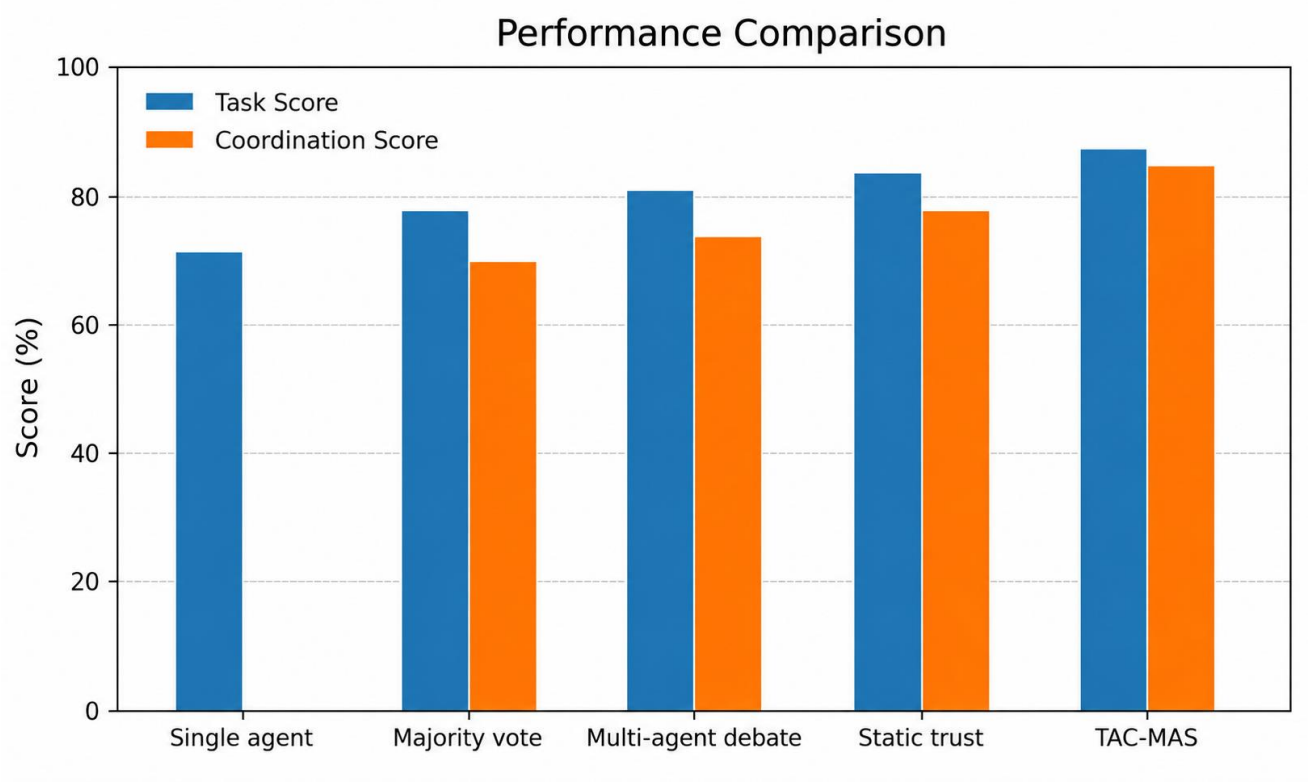


Figure 3. Task and coordination performance of TAC-MAS against the selected baseline strategies

Ablation Study

In the ablation study, the question is raised whether the gained performance is due to a

Annual Methodological Archive Research Review
http://amresearchreview.com/index.php/Journal/about
Volume 4, Issue 2 (2026)

Online ISSN 3007-3197
Print ISSN 3007-3189
http://amresearchreview.com/index.php/Journal/about

single dominant mechanism or due to the interplay of several components. Removing adaptive trust weighting results in the biggest drop for the values (6.0 points less on task score). Conflict detection removal reduces by 4.7 points, showing that even if reliability estimation is possible, it is not enough when there is no way to detect the conflict. The quality of evidence used adds 3.5 points, and the temporal updating adds 2.6 points due to the ability to gather information on the reliability of this evidence longitudinally, as shown in Table 3. These differences are illustrated in Figure 4.

Table 3. TAC-MAS ablation results.

Configuration	Task Score (%)	Coordination (%)	Error Rate (%)
TAC-MAS	87.6	84.9	7.6
Without evidence-quality term	84.1	80.6	10.2
Without temporal trust update	85.0	81.8	9.4
Without conflict detector	82.9	78.9	11.8
Without trust weighting	81.6	76.7	13.0

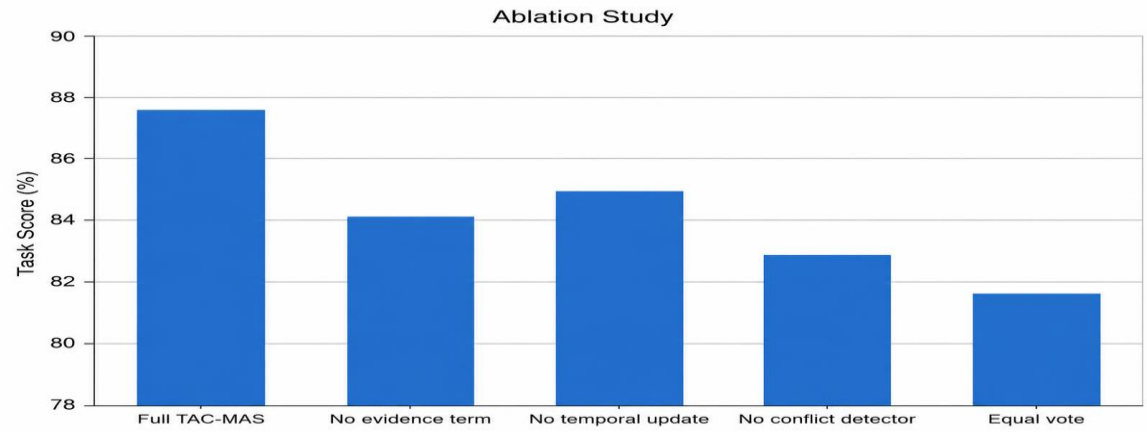


Figure 4. Ablation of the four principal TAC-MAS mechanisms: evidence quality, temporal trust updating, conflict detection, and adaptive weighting.

Comparative Analysis

To verify if TAC-MAS outperforms the chosen coordination baselines under the same experimental conditions, a direct numerical comparison is required. Table 4 thus presents the single-agent comparison of the same task for equal-weight majority voting, multi-agent debate, static trust aggregation, single-agent inference and TAC-MAS.

Table 4. Numerical comparative analysis of TAC-MAS .

Annual Methodological Archive Research Review
http://amresearchreview.com/index.php/Journal/about
Volume 4, Issue 2 (2026)

Online ISSN 3007-3197
Print ISSN 3007-3189

http://amresearchreview.com/index.php/Journal/about

Method	Task Score (%)	Coordination (%)	Error Rate (%)	Tokens / task	Latency (s)	TAC-MAS task gain
Single Agent	72.4	-	18.7	4,180	7.4	+15.2 pp
Majority Vote	78.1	70.2	14.9	15,800	15.2	+9.5 pp
Multi-Agent Debate [9]	81.3	74.8	12.8	23,600	24.7	+6.3 pp
Static Trust	83.7	78.5	10.9	18,900	18.6	+3.9 pp
TAC-MAS (Proposed)	87.6	84.9	7.6	17,300	17.9	Best

TAC-MAS demonstrates a steady improvement over the principal reliability measures in the Comparison. Compared with majority voting, TAC-MAS yields a 9.5 percentage point (12.2% relative gain) increase in task score and a 14.7 percentage point (49.0% relative decrease) decrease in error rate (from 14.9% to 7.6%) and a 14.7 percentage point (49.0% relative decrease) increase in coordination score (from 0.20 to 0.35). This finding is consistent with the central hypothesis that it is not optimal to treat all agents as equally reliable when agents' reliability varies. With the TAC-MAS configuration, there is a minor resource trade-off with respect to the majority voting configuration: 9.5% more tokens are used, and 17.8% more latency is added due to selective invocation of evidence verification and conflict resolution.

The most relevant comparison is with multi-agent debate since debate is a more robust baseline of collaboration. TAC-MAS is found to improve the task score by 6.3 percentage points (7.7% relative) from 81.3% to 87.6% and the coordination score by 10.1 percentage points from 74.8% to 84.9%. Concurrently, the error rate drops from 12.8% to 7.6% which translates to a 40.6% relative decrease. TAC-MAS also consumes 27.5% fewer tokens and decreases latency by 27.5% compared to full-debate when compared to other significant conflict scenarios. In this respect, TAC-MAS is more accurate than the unrestricted repeated debate and is more efficient in terms of the amount of communication.

Static trust is the closest architectural baseline since it already recognizes that the reliability of agents can vary. But fixed weights are not able to respond to evidence drawn from the task or to changing agent behavior. The TAC-MAS result is 3.9 task-score points higher than the static trust, 6.4 coordination-score points higher, and 30.3% better at reducing the error rate. It also consumes less tokens (8.5%) and latency (3.8%). These differences show the value of the adaptive trust updating, evidence-aware weighting and conflict-triggered re-verification beyond simply setting persistent agent weights.

The single-agent condition continues to be well less expensive due to the fact that it activates merely one model; as such, TAC-MAS is not superior when considering mere computational monetary. What it does gain is the reliability-performance compromise:

Online ISSN 3007-3197
Print ISSN 3007-3189
http://amresearchreview.com/index.php/Journal/about

The task score is 15.2 percentage points higher than it is for the single-agent and the error rate is lowered by 59.4%. Figure 3 and Table 4 are therefore the direct baseline comparison of TAC-MAS and the cited studies set the methodological context for the need of adaptive trust-aware coordination.

Discussion

The key idea behind TAC-MAS is that the quality of the collaboration is influenced by the content and the system's interpretation of the reliability of the information that agents share. Equal-vote ensembles are based on the premise that agents in the ensemble are roughly equivalent, whereas unrestricted debate is built on the idea that more discussion will uncover errors. The evidence that has been gathered so far supports neither of these assumptions as it is possible to have specification failures, alignment failures, and verification failures in the multi-agent systems and with some coordination tasks, even with communication. TAC-MAS thus assumes trust to be a variable, instead of a fixed attribute of the architecture, rather than an attribute of the service provider.TAC-MAS thus assumes trust to be a variable, not a fixed attribute of the service provider, but of an inference time variable.

One of the key design considerations is to avoid entrenchment of trust. If the early judgment is wrong, it is possible for an agent to be assigned too much or too little influence. TAC-MAS mitigates this risk by taking into account the historical information, using current evidence and consistency, smoothing over the updates, and mandating evidence support before high trust can dominate the aggregation. Another distinction of interest between epistemic trust and security authorization is due to recent studies showing that higher levels of trust among agents may lead to greater vulnerability [24]. TAC-MAS does not automatically grant additional tool or data or execution permissions based on trust, it is meant to control information flow.

Conclusion

In this paper, we introduce TAC-MAS (Adaptive Trust-Aware Coordination Framework for Reliable LLM-Based Multi-Agent Systems). The framework is an extension of the multi-agent coordination as it explicitly models the reliability of agents based on historical correctness, consistency across agents, quality of the evidence, and calibrated confidence. The final collective decision is made using trust-weighted aggregation, a conflict detector ensures targeted verification when needed, and a temporal trust mechanism ensures longitudinal reliability. The experimental evaluation is carried out using MultiAgentBench, and the proposed TAC-MAS is compared with three other algorithms: single-agent inference, majority voting, multi-agent debate, and static trust aggregation. The methodology provides five main equations and a reproducible method for the preprocessing, trust computation, conflict detection, consensus, and evaluation. The TAC-MAS architecture, along with the experimental evaluation, lays a concrete ground for empirical studies of reliability-aware LLM multi-agent systems.

References

[1] A. Vaswani et al., “Attention Is All You Need,” Advances in Neural Information

Online ISSN 3007-3197
Print ISSN 3007-3189
http://amresearchreview.com/index.php/Journal/about

Processing Systems, 2017.
[2] T. B. Brown et al., "Language Models are Few-Shot Learners," Advances in Neural Information Processing Systems, 2020.
[3] J. Wei et al., "Chain-of-Thought Prompting Elicits Reasoning in Large Language Models," Advances in Neural Information Processing Systems, 2022.
[4] X. Wang et al., "Self-Consistency Improves Chain of Thought Reasoning in Language Models," International Conference on Learning Representations, 2023.
[5] S. Yao et al., "ReAct: Synergizing Reasoning and Acting in Language Models," International Conference on Learning Representations, 2023.
[6] N. Shinn et al., "Reflexion: Language Agents with Verbal Reinforcement Learning," Advances in Neural Information Processing Systems, 2023.
[7] J. S. Park et al., "Generative Agents: Interactive Simulacra of Human Behavior," ACM UIST, 2023.
[8] G. Li, H. A. A. K. Hammoud, H. Itani, D. Khizbullin, and B. Ghanem, "CAMEL: Communicative Agents for 'Mind' Exploration of Large Scale Language Model Society," NeurIPS, 2023.
[9] Y. Du, S. Li, A. Torralba, J. B. Tenenbaum, and I. Mordatch, "Improving Factuality and Reasoning in Language Models through Multiagent Debate," arXiv:2305.14325, 2023.
[10] Q. Wu et al., "AutoGen: Enabling Next-Gen LLM Applications via Multi-Agent Conversation," First Conference on Language Modeling, 2024.
[11] S. Hong et al., "MetaGPT: Meta Programming for A Multi-Agent Collaborative Framework," International Conference on Learning Representations, 2024.
[12] C. Qian et al., "ChatDev: Communicative Agents for Software Development," Proceedings of ACL, 2024.
[13] W. Chen et al., "AgentVerse: Facilitating Multi-Agent Collaboration and Exploring Emergent Behaviors," International Conference on Learning Representations, 2024.
[14] X. Liu et al., "AgentBench: Evaluating LLMs as Agents," International Conference on Learning Representations, 2024.
[15] S. Kadavath et al., "Language Models (Mostly) Know What They Know," arXiv:2207.05221, 2022.
[16] K. Zhu et al., "MultiAgentBench: Evaluating the Collaboration and Competition of LLM Agents," Proceedings of ACL, 2025.
[17] M. Cemri et al., "Why Do Multi-Agent LLM Systems Fail?" arXiv:2503.13657, 2025.
[18] Y. Li et al., "Systematic Failures in Collective Reasoning under Distributed Information in Multi-Agent LLMs," arXiv:2505.11556, 2025.
[19] N. Hasan and P. BusiReddyGari, "DPBench: Large Language Models Struggle with Simultaneous Coordination," arXiv:2602.13255, 2026.
[20] S. Wu et al., "Scaling Teams or Scaling Time? Memory Enabled Lifelong Learning in LLM Multi-Agent Systems," arXiv:2604.03295, 2026.
[21] W. Gou and Z. Liu, "Dynamic Trust-Aware Sparse Communication Topology for LLM-Based Multi-Agent Consensus," arXiv:2606.01828, 2026.

Online ISSN 3007-3197
Print ISSN 3007-3189

[22] A. Mehdizadeh and M. Hilbert, "Exploring the Topology and Memory of Consensus: How LLM Agents Agree, Fragment, or Settle When Forming Conventions," arXiv:2606.04197, 2026.
[23] Y. Xie, S. Liu, F. Fan, Y. Yao, Y. Zhao, and B. Liu, "TeamTR: Trust-Region Fine-Tuning for Multi-Agent LLM Coordination," arXiv:2605.15207, 2026.
[24] Z. Xu et al., "The Trust Paradox in LLM-Based Multi-Agent Systems: When Collaboration Becomes a Security Vulnerability," arXiv:2510.18563, 2025.
[25] K. Wang et al., "TrinityGuard: A Unified Framework for Safeguarding Multi-Agent Systems," arXiv:2603.15408, 2026.
[26] G. Hao et al., "Self-Evolving Multi-Agent Systems via Decentralized Memory," arXiv:2605.22721, 2026.
[27] C. Fleming, R. Kompella, P. Bosch, and V. Pandey, "Scaling Multi-agent Systems: A Smart Middleware for Improving Agent Interactions," arXiv:2604.03430, 2026.
[28] L. Geng and E. Y. Chang, "REALM-Bench: A Benchmark for Evaluating Multi-Agent Systems on Real-world, Dynamic Planning and Scheduling Tasks," arXiv:2502.18836, 2025.
[29] X. Li et al., "A Survey on LLM-Based Multi-Agent Systems: Workflow, Infrastructure, and Challenges," Vicinagearth, 2024.
[30] "Evaluation and Benchmarking of LLM Agents: A Survey," arXiv:2507.21504, 2025.
[31] P. Feng, S. Yang, and S. Poria, "Sigma-Mem: An Online Reliability Memory for LLM-based Multi-Agent Systems," arXiv:2607.27958, 2026.