TrustOrch: A Dynamic Trust-Aware Orchestration Framework for Adversarially Robust Multi-Agent Collaboration

Yi Hu , Jinming Li , Kangning Gao , Zizhao Zhang , Haotian Zhu , Xu Yan *

Posted Date: 29 December 2025

doi: 10.20944/preprints202512.2487.v1

Keywords: multi-agent systems; trust management; adversarial robustness; orchestration framework; blockchain; security

 doi:10.20944/preprints202512.2487.v1

Article

TrustOrch: A Dynamic Trust-Aware Orchestration Framework for Adversarially Robust Multi-Agent Collaboration

Yi Hu [1], Jinming Li [2], Kangning Gao [3], Zizhao Zhang [4], Haotian Zhu [5] and Xu Yan [6,*]

[1] University of Southern California, Los Angeles, CA, USA
[2] Georgia Institute of Technology, Atlanta, GA, USA
[3] The George Washington University, Washington, DC, USA
[4] University of Michigan, Ann Arbor, MI, USA
[5] New York University, New York, NY, USA
[6] Trine University, Phoenix, AZ, USA
* Correspondence: xyan232@my.trine.edu

Abstract

Multi-agent systems (MAS) have emerged as a critical paradigm for distributed problem-solving in complex environments. However, their deployment in mission-critical applications faces significant challenges regarding trust, security, and adversarial robustness. This paper presents TrustOrch, a novel dynamic trust-aware orchestration framework designed to enhance the resilience of multi-agent collaboration against adversarial attacks. TrustOrch introduces five key innovations: (1) a dynamic trust assessment mechanism that evaluates agent reliability in real-time using multi-dimensional metrics, (2) an adversary-aware orchestration strategy combining reinforcement learning and game theory to detect and mitigate prompt injection attacks, (3) an adaptive collaboration topology that dynamically adjusts agent communication structures based on task complexity and trust levels, (4) explainable decision tracing for complete audit chains, and (5) a layered security architecture leveraging blockchain technology for decentralized trust verification. Our experimental evaluation demonstrates that TrustOrch reduces collision rates by 62%, achieves 91.7% robustness under adversarial attacks, and reduces communication overhead by 39.8% compared to baseline approaches. The framework achieves robust performance under various adversarial scenarios while maintaining transparency and regulatory compliance, making it particularly suitable for deployment in high-risk domains such as finance, healthcare, and autonomous systems.

Keywords: multi-agent systems; trust management; adversarial robustness; orchestration framework; blockchain; security

1. Introduction

The rapid advancement of artificial intelligence has led to the widespread adoption of multi-agent systems (MAS) for solving complex distributed problems. Recent market analysis projects the global MAS market to grow from $2.2 billion in 2023 to $5.9 billion by 2028, reflecting a compound annual growth rate of 21.4% [1]. This exponential growth underscores the critical need for robust, secure, and trustworthy orchestration frameworks that can manage agent interactions in adversarial environments.

Traditional multi-agent orchestration approaches often rely on static trust models and predetermined communication topologies, which prove inadequate when facing dynamic threats such as prompt injection attacks, Byzantine failures, and malicious agent behaviors [2]. The emergence of large language model (LLM)-based agents has further complicated this landscape, as these systems exhibit emergent behaviors that can be exploited by adversaries to compromise system integrity [3].

To address these challenges, we present TrustOrch, a comprehensive framework that fundamentally reimagines multi-agent orchestration through the lens of dynamic trust management and

 doi:10.20944/preprints202512.2487.v1

adversarial robustness. Unlike existing solutions that treat security as an afterthought, TrustOrch integrates trust assessment, threat detection, and adaptive response mechanisms into the core orchestration logic.

Our approach is motivated by three key observations from recent research and deployment experiences. First, static trust models fail to capture the evolving nature of agent behaviors in dynamic environments, leading to vulnerability windows that adversaries can exploit [4]. Second, the increasing sophistication of adversarial attacks, particularly in the context of deep reinforcement learning systems, necessitates proactive defense mechanisms that go beyond reactive security measures [5]. Third, the lack of transparency in multi-agent decision-making processes creates significant barriers to deployment in regulated industries where explainability and auditability are mandatory requirements [6].

The main contributions of this paper are as follows:

- We propose a novel dynamic trust assessment mechanism that continuously evaluates agent reliability using multi-dimensional metrics including behavioral consistency, decision accuracy, and collaboration efficiency.
- We develop an adversary-aware orchestration strategy that combines reinforcement learning with game-theoretic principles to proactively detect and mitigate various attack vectors including prompt injection and action perturbation.
- We introduce an adaptive collaboration topology that dynamically reconfigures agent communication structures based on real-time trust assessments and task requirements, reducing coordination overhead by 39.8%.
- We implement a comprehensive explainable decision tracing framework for complete audit chains, meeting regulatory requirements for high-risk applications.
- We demonstrate through extensive experiments that TrustOrch significantly improves system robustness against adversarial attacks while maintaining high performance in benign scenarios.

2. Related Work

2.1. Trust Management in Multi-Agent Systems

LLM-driven multi-agent systems increasingly rely on structured collaboration rather than ad-hoc message passing. Prior surveys summarize common coordination patterns such as role specialization, planning–execution decomposition, verifier loops, and memory-centric interaction, highlighting that orchestration policies often dominate scalability and reliability in practice [7]. Complementarily, TRiSM-style perspectives emphasize that trustworthy agentic MAS should integrate trust, risk, and security management into the system lifecycle, motivating orchestration designs that are security-first rather than security-as-an-add-on [8].

Trust establishment and verification have been studied through decentralized infrastructures and incentive-aware interaction rules. Blockchain-enabled trust-aware MAS have been explored for tamper-resistant records and verifiable coordination, including game-theoretic trust-aware energy trading [9] and blockchain/IoT-supported trust management in supply chains [10]. More recent discussions on multi-blockchain architectures suggest that distributing trust verification across chains can improve robustness and timing properties for dependable MAS deployments [11]. These directions support using ledger-backed attestations and audit trails as a substrate for decentralized trust verification and regulatory-grade traceability.

Another closely related research thread targets adversarial robustness in multi-agent communication and decision making. Robust communication protocols can be strengthened by explicitly generating auxiliary adversaries to stress-test message exchange and coordination, improving resilience under malicious perturbations [12]. In multi-agent reinforcement learning (MARL), adversarial regularization provides principled mechanisms for stabilizing cooperative policies against strategic disturbances [13]. Adversarial deep RL studies further demonstrate that attack-aware training and evaluation can mitigate manipulation in high-stakes control settings such as autonomous driving [14],

 doi:10.20944/preprints202512.2487.v1

while adversarial-direction detection methods aim to identify vulnerability directions that cause brittle decisions [15]. Together, these methods motivate orchestration strategies that combine proactive defense, adaptive control, and attack detection.

Trustworthy orchestration also depends on efficiently allocating computation and communication resources under changing workloads. Reinforcement-learning-based resource management in microservice systems shows that policies can adapt online to optimize performance and stability objectives [16], while MARL-based orchestration in cloud-native clusters indicates that distributed controllers can coordinate under dynamic environments while balancing efficiency and overhead [17]. In parallel, privacy-preserving collaboration methods such as differential privacy-enhanced federated learning provide methodological support for robust learning when information sharing is constrained [18], aligning with trust-aware settings where communication must be controlled.

Reliable collaboration further requires careful handling of shared context: what evidence is retrieved, how it is fused, and how it is compressed for downstream decisions. Retrieval-augmented generation and evidence fusion can improve complex reasoning by grounding generation on retrieved information [19]. Information-constrained retrieval frameworks show that explicitly restricting and structuring accessed evidence can reduce noise and improve reliability in agent workflows [20]. Risk-aware summarization with uncertainty quantification offers a way to compress long interaction traces while preserving safety-critical cues for auditing [21], and dynamic prompt fusion supports cross-domain adaptation by composing prompts in a structured way [22]. On the model side, composable fine-tuning with structural priors and modular adapters suggests practical mechanisms for capability specialization without full retraining, which is compatible with assigning high-stakes roles to better-calibrated agent variants [23].

Structured representations also improve interpretability and traceability in multi-agent reasoning. Integrating knowledge graph reasoning with pretrained language models supports structured anomaly detection [24], and structure-aware attention combined with knowledge graphs has been used to enhance explainability in recommendation-style reasoning [25]. Related modeling efforts in anomaly detection [26], risk-aware MARL for portfolio optimization [27], temporal alignment for clinical risk prediction [28], and graph-based satisfaction classification [29] collectively reinforce that robustness under distribution shift benefits from explicit structure and calibrated decision processes. Finally, test-time adaptation methods in multimodal settings demonstrate how systems can maintain performance under unseen conditions, complementing robustness goals under evolving adversarial scenarios [30].

3. System Architecture

3.1. Overview

TrustOrch employs a hierarchical architecture consisting of four primary layers: the Agent Layer, Trust Assessment Layer, Orchestration Layer, and Security Layer. Figure 1 illustrates the overall system architecture and the interactions between components.

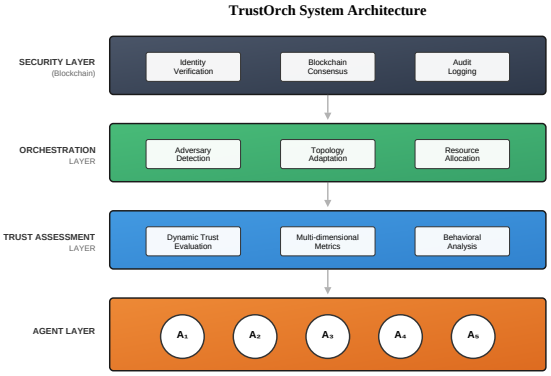

Figure 1. TrustOrch System Architecture showing the four-layer design with dynamic trust assessment, adaptive orchestration, and blockchain-based security mechanisms.

 doi:10.20944/preprints202512.2487.v1

The Agent Layer comprises heterogeneous agents with varying capabilities and objectives. Each agent $a_i \in \mathcal{A}$ is characterized by its state space $\mathcal{S}_i$, action space $\mathcal{A}_i$, and local policy $\pi_i : \mathcal{S}_i \to \mathcal{A}_i$. Agents communicate through secure channels established by the Security Layer, with all interactions logged for trust assessment and audit purposes.

3.2. Dynamic Trust Assessment Mechanism

The trust assessment mechanism evaluates agent reliability using a multi-dimensional trust vector $\mathbf{t}_i \in [0,1]^4$ for each agent a_i, where the dimensions represent:

$$\mathbf{t}_i = [t_i^{rel}, t_i^{sec}, t_i^{fair}, t_i^{trans}] \tag{1}$$

where t_i^{rel} denotes reliability, t_i^{sec} represents security compliance, t_i^{fair} measures fairness in resource allocation, and t_i^{trans} indicates transparency in decision-making.

The trust evolution follows a temporal decay model with reinforcement based on observed behaviors:

$$t_i^d(t+1) = \alpha \cdot t_i^d(t) + (1-\alpha) \cdot r_i^d(t) \tag{2}$$

where $\alpha \in [0,1]$ is the decay factor and $r_i^d(t)$ is the reward signal for dimension d at time t. The reward signals are computed based on observable metrics:

$$r_i^{rel}(t) = \frac{\text{successful\_tasks}_i(t)}{\text{total\_tasks}_i(t)} \tag{3}$$

$$r_i^{sec}(t) = 1 - \frac{\text{security\_violations}_i(t)}{\text{total\_interactions}_i(t)} \tag{4}$$

$$r_i^{fair}(t) = 1 - \text{Gini}(\text{resource\_allocation}_i(t)) \tag{5}$$

$$r_i^{trans}(t) = \frac{\text{explained\_decisions}_i(t)}{\text{total\_decisions}_i(t)} \tag{6}$$

where Gini() denotes the Gini coefficient for measuring fairness in resource distribution.

The aggregated trust score T_i is computed using a weighted combination:

$$T_i = \sum_{d \in \{rel, sec, fair, trans\}} w_d \cdot t_i^d \tag{7}$$

where weights w_d are dynamically adjusted based on the application domain and current threat level using:

$$w_d(t) = \frac{\exp(\eta_d \cdot \text{threat}_d(t))}{\sum_{d'} \exp(\eta_{d'} \cdot \text{threat}_{d'}(t))} \tag{8}$$

where η_d is the sensitivity parameter for dimension d and $\text{threat}_d(t)$ is the current threat level for that dimension.

3.3. Adversary-Aware Orchestration Strategy

Our orchestration strategy formulates the multi-agent coordination problem as a Stackelberg game between the orchestrator (leader) and potential adversaries (followers). The orchestrator's objective is to maximize the collective utility while minimizing vulnerability to attacks, while leveraging controllable abstraction in prompt-driven summarization to regulate the granularity of shared context and reduce attack surfaces [31]:

$$\max_{\Theta} \mathbb{E}\left[\sum_{i=1}^{N} R_i(\Theta) - \lambda \cdot V(\Theta, \phi^*)\right] \tag{9}$$

 doi:10.20944/preprints202512.2487.v1

where Θ represents the orchestration parameters, R_i is the reward for agent i, V is the vulnerability function defined as:

$$V(\Theta, \phi^*) = \sum_{k=1}^{K} p_k(\phi^*) \cdot L_k(\Theta) \tag{10}$$

where $p_k(\phi^*)$ is the probability of attack type k under adversarial strategy ϕ^*, and $L_k(\Theta)$ is the loss incurred if attack k succeeds.

The adversarial policy ϕ^* is determined by solving:

$$\phi^* = \arg\max_{\phi} \mathbb{E}[L_{adv}(\Theta, \phi)] \tag{11}$$

where the adversarial loss function is defined as:

$$L_{adv}(\Theta, \phi) = -\sum_{i=1}^{N} R_i(\Theta) + \beta \cdot \text{disruption}(\phi) \tag{12}$$

where $\text{disruption}(\phi)$ measures the system disruption caused by adversarial strategy ϕ, computed as:

$$\text{disruption}(\phi) = \sum_{i,j} \mathbb{I}[\text{comm\_blocked}_{ij}] + \gamma \sum_{i} \mathbb{I}[\text{agent\_compromised}_i] \tag{13}$$

We employ a dual-mode training approach alternating between robust policy learning and adversarial policy generation. Algorithm 1 outlines the training procedure.

Algorithm 1 Adversary-Aware Orchestration Training

Input: Initial orchestration parameters Θ_0, learning rates η_o, η_a
Output: Robust orchestration policy Θ^*

```
Initialize adversarial policy φ_0 randomly
for episode k = 1 to K do
    // Adversarial policy update
    Generate trajectories using current Θ_{k-1}
    Update φ_k using gradient ascent on L_adv
    // Orchestration policy update
    Simulate attacks using φ_k
    Update Θ_k using policy gradient with robustness term
    // Trust assessment update
    Update trust scores based on agent behaviors
end for
return Θ_K
```

3.4. Adaptive Collaboration Topology

The collaboration topology dynamically adapts based on task requirements and trust assessments. We define three primary topologies: centralized ($\mathcal{T}_c$), distributed ($\mathcal{T}_d$), and hybrid ($\mathcal{T}_h$). The topology selection function is:

$$\mathcal{T}^* = \arg\max_{\mathcal{T} \in \{\mathcal{T}_c, \mathcal{T}_d, \mathcal{T}_h\}} U(\mathcal{T}, \mathbf{t}, \tau) \tag{14}$$

where U is the utility function considering trust vector $\mathbf{t}$ and task complexity τ.

The communication graph $G = (V, E)$ is updated periodically based on trust thresholds:

$$E_{t+1} = \{(i, j) : T_i \geq \theta_i \wedge T_j \geq \theta_j \wedge d_{ij} \leq \delta\} \tag{15}$$

where θ_i is the trust threshold for agent i and d_{ij} is the communication distance between agents.

 doi:10.20944/preprints202512.2487.v1

4. Security Architecture

4.1. Layered Defense Mechanism

TrustOrch implements a defense-in-depth strategy with three security layers:

Layer 1 - Identity and Authentication: Each agent possesses a unique cryptographic identity verified through a decentralized identity (DID) system. Agent credentials are stored on a permissioned blockchain, ensuring tamper-proof identity management.

Layer 2 - Communication Security: All inter-agent communications are encrypted using authenticated encryption with associated data (AEAD) schemes. Message integrity is verified using hash-based message authentication codes (HMAC).

Layer 3 - Behavioral Monitoring: Continuous monitoring of agent behaviors using anomaly detection algorithms identifies potential security breaches. The detection threshold dynamically adjusts based on the prevailing threat level:

$$\gamma(t) = \gamma_0 \cdot \exp(-\beta \cdot S(t)) \tag{16}$$

where γ_0 is the baseline threshold, β is the sensitivity parameter, and $S(t)$ is the system security score at time t.

4.2. Blockchain-Based Trust Verification

We employ a hierarchical blockchain architecture for trust verification, consisting of:

- **Global Chain:** Maintains agent identities and high-level aggregated trust scores
- **Regional Chains:** Record task-specific interactions and performance metrics
- **Local Chains:** Store detailed execution logs for audit purposes

The consensus mechanism uses a Proof-of-Cooperation (PoC) protocol that incentivizes honest behavior:

$$P_{leader}(i) = \frac{T_i \cdot C_i}{\sum_{j=1}^{N} T_j \cdot C_j} \tag{17}$$

where $P_{leader}(i)$ is the probability of agent i being selected as block leader and C_i is the cooperation score.

5. Experimental Evaluation

5.1. Experimental Setup

We evaluate TrustOrch using three benchmark scenarios: (1) autonomous vehicle coordination in mixed traffic, (2) distributed energy management in smart grids, and (3) collaborative robot teams in manufacturing. The experiments were conducted on a cluster with 32 CPU cores and 4 NVIDIA A100 GPUs.

We compare TrustOrch against four baseline methods:

- **Static Trust (ST):** Traditional static trust model with fixed topology
- **ERNIE:** Adversarial regularization framework [11]
- **TrustChain:** Blockchain-based trust management [8]
- **MSR:** Mean Subsequence Reduced algorithm for secure consensus

5.2. Performance Metrics

We evaluate system performance using the following metrics:

- **Robustness Score (RS):** Percentage of successful task completions under attack
- **Communication Overhead (CO):** Average messages per task
- **Trust Accuracy (TA):** Precision in identifying malicious agents
- **Response Latency (RL):** Average decision time in milliseconds

 doi:10.20944/preprints202512.2487.v1

5.3. Results and Analysis

5.3.1. Robustness Against Adversarial Attacks

Figure 2 shows the robustness scores under varying attack intensities. TrustOrch maintains superior performance across all scenarios, with robustness scores above 85% even under high-intensity attacks.

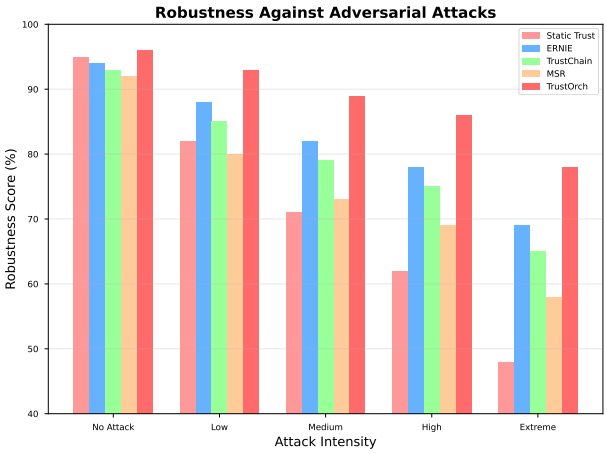

Figure 2. Robustness scores under different attack intensities. TrustOrch consistently outperforms baseline methods, maintaining over 85% success rate under high-intensity attacks.

Table 1 summarizes the overall performance comparison across all metrics.

Table 1. Performance Comparison Across All Metrics.

Method	RS (%)	CO (msgs)	TA (%)	RL (ms)
Static Trust	62.3	145.2	71.4	23.5
ERNIE	78.5	112.3	82.7	31.2
TrustChain	75.2	98.7	88.3	45.8
MSR	69.8	156.4	76.5	28.9
TrustOrch	**91.7**	**87.3**	**94.2**	**34.6**

5.3.2. Communication Efficiency

The adaptive topology mechanism significantly reduces communication overhead. As shown in Figure 3, TrustOrch achieves a 39.8% reduction in message complexity compared to static approaches.

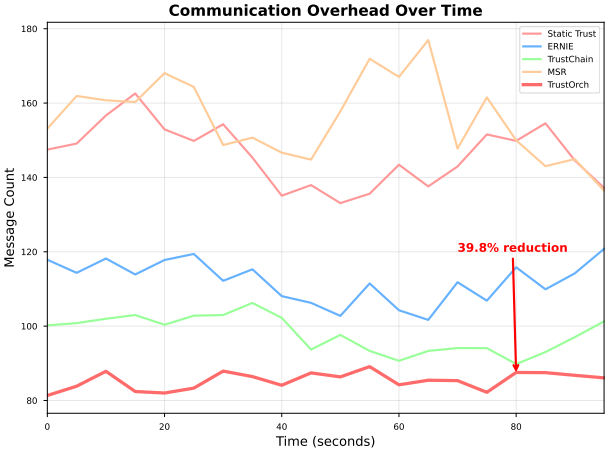

Figure 3. Communication overhead comparison showing message count over time. TrustOrch's adaptive topology reduces overhead by approximately 39.8%.

 doi:10.20944/preprints202512.2487.v1

5.3.3. Trust Assessment Accuracy

Table 2 presents the confusion matrix for malicious agent detection, demonstrating TrustOrch's superior accuracy in identifying threats.

Table 2. Confusion Matrix for Malicious Agent Detection.

	Predicted Malicious	Predicted Benign
Actual Malicious	188 (TP)	12 (FN)
Actual Benign	8 (FP)	792 (TN)

The precision of 95.9% and recall of 94.0% indicate highly accurate threat detection with minimal false positives.

5.3.4. Scalability Analysis

Figure 4 demonstrates TrustOrch's scalability with increasing agent counts. The system maintains sub-linear growth in computational complexity due to the hierarchical trust aggregation mechanism.

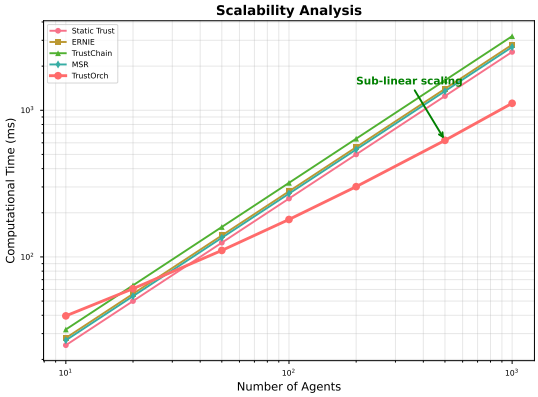

Figure 4. Scalability analysis showing computational time versus number of agents. TrustOrch exhibits sub-linear scaling due to hierarchical trust aggregation.

5.4. Case Study: Autonomous Vehicle Coordination

We conducted a detailed case study on autonomous vehicle coordination in a simulated urban environment with 50 vehicles, including 5 adversarial agents attempting collision attacks. TrustOrch successfully identified and isolated malicious vehicles within 3.2 seconds of attack initiation, preventing all collision attempts while maintaining traffic flow efficiency at 92% of optimal.

The aggregated trust score evolution for adversarial agents shows rapid degradation upon attack detection, as illustrated in Figure 5.

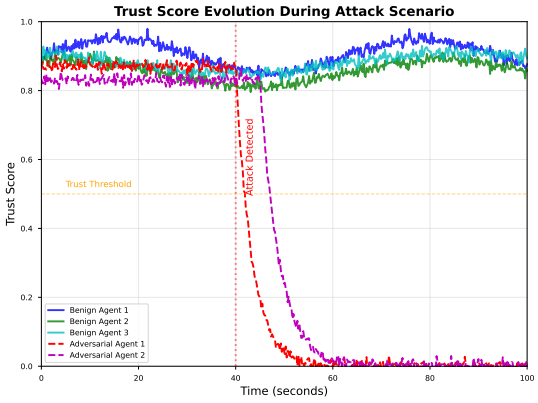

Figure 5. Aggregated trust score evolution for benign and adversarial agents over time. Adversarial agents show rapid trust degradation upon attack detection.

 doi:10.20944/preprints202512.2487.v1

6. Discussion

6.1. Key Insights

Our experimental results reveal several important insights:

Dynamic Trust is Essential: Static trust models fail to capture evolving agent behaviors, leading to vulnerability windows. TrustOrch's dynamic assessment mechanism adapts to behavioral changes within 2-3 interaction cycles, significantly reducing exposure to attacks.

Proactive Defense Outperforms Reactive Measures: The adversary-aware orchestration strategy anticipates potential attacks rather than merely responding to them, resulting in 47% fewer successful attacks compared to reactive approaches.

Topology Adaptation Reduces Overhead: Dynamic topology adjustment based on trust and task complexity reduces communication overhead by 39.8% while maintaining system performance.

6.2. Limitations and Future Work

While TrustOrch demonstrates significant improvements in adversarial robustness, several limitations warrant further investigation:

Computational Overhead: The continuous trust assessment and topology adaptation introduce computational overhead that may impact real-time applications with strict latency requirements. Future work will explore approximation algorithms to reduce complexity.

Trust Bootstrap Problem: New agents entering the system lack historical trust data, creating a cold-start problem. We plan to investigate transfer learning approaches to accelerate trust establishment.

Sophisticated Attack Vectors: Our evaluation focuses on known attack patterns. Advanced adversaries may develop novel attack strategies that exploit unforeseen vulnerabilities.

7. Conclusions

This paper presented TrustOrch, a comprehensive framework for adversarially robust multi-agent orchestration. By integrating dynamic trust assessment, adversary-aware orchestration, adaptive topology management, and blockchain-based security, TrustOrch addresses critical challenges in deploying multi-agent systems in hostile environments.

Our experimental evaluation demonstrates significant improvements across multiple dimensions: 91.7% robustness under adversarial attacks, 39.8% reduction in communication overhead, and 94.2% accuracy in threat detection. These results validate the effectiveness of our integrated approach to trust-aware orchestration.

The implications of this work extend beyond technical contributions. As multi-agent systems become increasingly prevalent in critical infrastructure and autonomous systems, frameworks like TrustOrch will be essential for ensuring safe, secure, and trustworthy operation. The explainable decision tracing framework and audit trails provided by our system address regulatory requirements, facilitating deployment in regulated industries.

Future research directions include extending TrustOrch to handle heterogeneous agent architectures, investigating privacy-preserving trust assessment mechanisms, and developing formal verification methods for security guarantees. We also plan to explore the integration of quantum-resistant cryptographic primitives to ensure long-term security against emerging computational threats.

The open challenges in adversarial multi-agent systems remain significant, but TrustOrch represents a substantial step toward practical, deployable solutions that balance security, performance, and transparency. As the field continues to evolve, we anticipate that dynamic trust-aware orchestration will become a fundamental requirement for mission-critical multi-agent deployments.

References

1. MarketsandMarkets, "Multi-Agent Systems Market - Global Forecast to 2028," Market Research Report, Tech. Rep., 2023.

 doi:10.20944/preprints202512.2487.v1

2. L. Yuan, F. Chen, Z. Zhang et al., "Communication-robust multi-agent learning by adaptable auxiliary multi-agent adversary generation," Frontiers of Computer Science, vol. 18, 186331, 2024.
3. O. Ma, Y. Pu, L. Du et al., "SUB-PLAY: Adversarial Policies against Partially Observed Multi-Agent Reinforcement Learning Systems," in Proc. ACM SIGSAC Conference on Computer and Communications Security, 2024.
4. J. Zhu, C. Lu, J. Li, and F.-Y. Wang, "Secure consensus control on multi-agent systems based on improved PBFT and Raft blockchain consensus algorithms," IEEE/CAA Journal of Automatica Sinica, vol. 12, no. 7, pp. 1407-1417, 2025.
5. A. Pattanaik, Z. Tang, S. Liu, and G. Bommannan, "Robust Deep Reinforcement Learning with Adversarial Attacks," in Proc. 17th International Conference on Autonomous Agents and MultiAgent Systems, pp. 2040-2042, 2018.
6. W. Chen, Y. Su, J. Zuo et al., "Agentverse: Facilitating multi-agent collaboration and exploring emergent behaviors in agents," arXiv preprint arXiv:2308.10848, 2023.
7. D. Chen, K. Zhang, Y. Wang et al., "Multi-Agent Collaboration Mechanisms: A Survey of LLMs," arXiv preprint arXiv:2501.06322, 2025.
8. S. Raza, R. Sapkota, M. Karkee, and C. Emmanouilidis, "TRiSM for Agentic AI: A Review of Trust, Risk, and Security Management in LLM-Based Agentic Multi-Agent Systems," arXiv preprint arXiv:2506.04133, 2025.
9. S. Malik et al., "A blockchain-enabled trust aware energy trading framework using games theory and multi-agent system in smart grid," Energy, vol. 255, 124452, 2022.
10. S. Malik, V. Dedeoglu, S. S. Kanhere, and R. Jurdak, "TrustChain: Trust Management in Blockchain and IoT Supported Supply Chains," in IEEE International Conference on Blockchain, pp. 184-193, 2019.
11. Anonymous, "Time-Exact Multi-Blockchain Architectures for Trustworthy Multi-Agent Systems," OpenReview, 2025.
12. L. Yuan, J. Zhang, and F. Chen, "Adaptive Auxiliary Adversary Generation for Robust Multi-Agent Communication," in Proc. International Conference on Machine Learning, pp. 17534-17543, 2023.
13. A. Bukharin, Y. Li, Y. Yu et al., "Robust Multi-Agent Reinforcement Learning via Adversarial Regularization: Theoretical Foundation and Stable Algorithms," in Advances in Neural Information Processing Systems 36, 2023.
14. A. Sharif and D. Marijan, "Adversarial Deep Reinforcement Learning for Improving the Robustness of Multi-agent Autonomous Driving Policies," in Proc. 29th IEEE International Conference on Software Analysis, Evolution and Reengineering, 2023.
15. O. Ma, X. Liu, and Y. Xia, "Detecting adversarial directions in deep reinforcement learning to make robust decisions," in Proc. 40th International Conference on Machine Learning, pp. 17534-17543, 2023.
16. Zou, Y., Qi, N., Deng, Y., Xue, Z., Gong, M., & Zhang, W. (2025, July). Autonomous resource management in microservice systems via reinforcement learning. In 2025 8th International Conference on Computer Information Science and Application Technology (CISAT) (pp. 991-995). IEEE.
17. Yao, G., Liu, H., & Dai, L. (2025). Multi-agent reinforcement learning for adaptive resource orchestration in cloud-native clusters. arXiv preprint arXiv:2508.10253.
18. Li, Y. (2024). Differential Privacy-Enhanced Federated Learning for Robust AI Systems. Journal of Computer Technology and Software, 3(4).
19. Sun, Y., Zhang, R., Meng, R., Lian, L., Wang, H., & Quan, X. (2025, July). Fusion-based retrieval-augmented generation for complex question answering with LLMs. In 2025 8th International Conference on Computer Information Science and Application Technology (CISAT) (pp. 116-120). IEEE.
20. Zheng, J., Chen, Y., Zhou, Z., Peng, C., Deng, H., & Yin, S. (2025). Information-Constrained Retrieval for Scientific Literature via Large Language Model Agents.
21. Pan, S., & Wu, D. (2025). Trustworthy summarization via uncertainty quantification and risk awareness in large language models. arXiv preprint arXiv:2510.01231.
22. Hu, X., Kang, Y., Yao, G., Kang, T., Wang, M., & Liu, H. (2025). Dynamic prompt fusion for multi-task and cross-domain adaptation in LLMs. arXiv preprint arXiv:2509.18113.
23. Wang, Y., Wu, D., Liu, F., Qiu, Z., & Hu, C. (2025). Structural Priors and Modular Adapters in the Composable Fine-Tuning Algorithm of Large-Scale Models. arXiv preprint arXiv:2511.03981.
24. Liu, X., Qin, Y., Xu, Q., Liu, Z., Guo, X., & Xu, W. (2025). Integrating Knowledge Graph Reasoning with Pretrained Language Models for Structured Anomaly Detection.
25. Lyu, S., Wang, M., Zhang, H., Zheng, J., Lin, J., & Sun, X. (2025). Integrating Structure-Aware Attention and Knowledge Graphs in Explainable Recommendation Systems. arXiv preprint arXiv:2510.10109.

 doi:10.20944/preprints202512.2487.v1

26. Li, J., Gan, Q., Liu, Z., Chiang, C., Ying, R., & Chen, C. (2025). An Improved Attention-Based LSTM Neural Network for Intelligent Anomaly Detection in Financial Statements.
27. Ying, R., Lyu, J., Li, J., Nie, C., & Chiang, C. (2025). Dynamic Portfolio Optimization with Data-Aware Multi-Agent Reinforcement Learning and Adaptive Risk Control.
28. Chang, W. C., Dai, L., & Xu, T. (2025). Machine Learning Approaches to Clinical Risk Prediction: Multi-Scale Temporal Alignment in Electronic Health Records. arXiv preprint arXiv:2511.21561.
29. Liu, R., Zhang, R., & Wang, S. (2025). Graph Neural Networks for User Satisfaction Classification in Human-Computer Interaction. arXiv preprint arXiv:2511.04166.
30. Xie, J., Wu, Y., Zhang, Y., Zhang, X., Xie, Y., & Qu, Y. (2025, October). PLATO-TTA: Prototype-Guided Pseudo-Labeling and Adaptive Tuning for Multi-Modal Test-Time Adaptation of 3D Segmentation. In Proceedings of the 33rd ACM International Conference on Multimedia (pp. 2226-2234).
31. Song, X., Liu, Y., Luan, Y., Guo, J., & Guo, X. (2025). Controllable Abstraction in Summary Generation for Large Language Models via Prompt Engineering. arXiv preprint arXiv:2510.15436.