Interpretable Graph-Attention Collaboration: Adaptive Policies for Robust Multi-Agent Systems

Anonymous Author(s)

ABSTRACT

Multi-agent systems increasingly rely on collaboration among autonomous agents, yet most deployed architectures employ fixed, hand-designed communication topologies such as star, chain, or fully connected graphs. We introduce *Interpretable Graph-Attention Collaboration* (IGAC), a framework that combines an adaptive communication topology with trust-weighted message passing for multi-agent collaborative reasoning. IGAC employs learnable bilinear edge scoring with Gumbel-Sigmoid sampling and a straight-through estimator to produce sparse, instance-specific binary collaboration graphs, attention-based message aggregation with analytically-differentiated SGD-trained projection parameters for interpretable information routing, and a Beta-distributed counterfactual trust mechanism for adversarial agent detection and isolation. We separate training (where projection parameters and trust are learned) from evaluation (where parameters are frozen), ensuring a rigorous experimental protocol. Across seven experiments on collaborative state reconstruction tasks with up to 20 agents, the trained IGAC model reduces communication edges by 54% compared to fully connected baselines while maintaining competitive reconstruction accuracy ($p < 0.001$). Under adversarial conditions with three adversary types (random, bias, mimic) and up to 3 out of 6 agents compromised, IGAC with trust scoring achieves lower reconstruction error than fixed-topology baselines and classical Byzantine-resilient aggregation (trimmed mean, coordinate-wise median), while uniquely detecting adversarial agents via trust-based thresholding. Ablation studies with 95% confidence intervals confirm that both the learned topology and trust mechanism contribute independently to robustness.

ACM Reference Format:
Anonymous Author(s). 2026. Interpretable Graph-Attention Collaboration: Adaptive Policies for Robust Multi-Agent Systems. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '26), August 3–7, 2026, Toronto, ON, Canada.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Multi-agent systems that collaborate through structured communication have demonstrated capabilities exceeding those of individual agents across a range of reasoning tasks [4, 17]. However, the collaboration topology—which agents communicate with which, and how information is aggregated—remains predominantly a design choice made by human engineers. Fixed topologies such as star (hub-and-spoke), chain (sequential), and fully connected graphs each impose structural assumptions that may not match the requirements of a given task instance [16].

This rigidity creates three interrelated challenges. First, fixed topologies cannot *adapt* to varying task demands, agent capabilities, or partial observability conditions. Second, when communication structure is predetermined, there is limited opportunity for *interpretability*: practitioners cannot understand why particular communication patterns emerged because they were imposed rather than learned. Third, fixed topologies are *vulnerable* to adversarial agents—a compromised node in a star topology can corrupt all communications, while a fully connected topology indiscriminately aggregates adversarial messages.

Wei et al. [16] identify the development of adaptive, interpretable collaboration policies robust to partial observability and adversarial conditions as a key open problem in agentic reasoning. We address this problem with *Interpretable Graph-Attention Collaboration* (IGAC), a framework built on three technical contributions:

(1) **Learned sparse topology via bilinear edge scoring.** A meta-controller produces per-instance, per-step adjacency matrices using a trainable bilinear edge scoring matrix W_{edge} with Gumbel-Sigmoid relaxation [6] and a straight-through estimator [1] for hard edge sampling. This yields genuinely sparse, binary communication graphs that adapt to the information structure of each problem instance.
(2) **Analytically-trained trust-weighted attention.** Messages are aggregated along learned edges using scaled dot-product attention [12] with projection parameters trained via SGD with analytical gradients (exact backpropagation through the multi-round attention mechanism), modulated by per-neighbor trust scores. Trust is modeled as Beta distributions updated via personalized counterfactual credit assignment [5], enabling principled detection of adversarial agents.
(3) **Interpretability through hard sparsity and attention.** The combination of binary sparse topology and peaked attention distributions provides two complementary levels of interpretability: structural (which edges are active) and functional (how much each message contributes to each agent's decision).

We evaluate IGAC on collaborative state reconstruction under controlled partial observability and adversarial agent injection with three adversary types (random, bias, mimic), comparing against fixed-topology baselines, a random sparse control, and classical Byzantine-resilient aggregation methods across seven experimental dimensions with paired statistical testing.

KDD '26, August 3–7, 2026, Toronto, ON, Canada

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

2 RELATED WORK

Multi-Agent Communication Learning. CommNet [11] introduced differentiable communication channels between reinforcement learning agents, enabling end-to-end learning of message content. TarMAC [3] added targeted communication through attention mechanisms, and MAGIC [9] employed graph attention for agent communication. These methods learn *what* to communicate but assume fixed topologies. IGAC extends this line by jointly learning the topology and training message aggregation parameters.

Multi-Agent Reinforcement Learning. QMIX [10], MAPPO [19], and MADDPG [8] provide centralized-training-decentralized-execution frameworks for cooperative and mixed settings. They address credit assignment at the value-function level but do not learn communication structure. Our counterfactual trust mechanism provides agent-level credit assignment that doubles as an adversarial detection signal.

LLM-Based Multi-Agent Systems. AutoGen [17] and related frameworks enable multi-agent conversations with predefined topologies. DyLAN [7] dynamically adjusts agent participation using per-step scoring, representing the closest existing work to topology learning. However, DyLAN lacks explicit interpretability mechanisms and adversarial robustness guarantees. Multi-agent debate [4] improves reasoning through structured disagreement but uses fixed two-agent or round-robin structures.

Robust and Interpretable Policies. Byzantine-tolerant consensus [2] provides robustness in distributed systems but assumes well-defined message semantics incompatible with free-form agent outputs. Programmatic policies [14] offer inherent interpretability but limited scalability. Graph Attention Networks [13] provide attention-based message passing over fixed graphs; IGAC extends this to learned, dynamic graphs with trust modulation.

3 METHOD

3.1 Problem Formulation

We consider N agents that must collaboratively reconstruct a shared hidden state $\mathbf{s} \in \mathbb{R}^D$ from partial, noisy observations. Agent i observes $\mathbf{o}_i = M_i\mathbf{s} + \boldsymbol{\epsilon}_i$, where $M_i \in \{0,1\}^{D\times D}$ is a diagonal mask revealing a fraction p of state dimensions, and $\boldsymbol{\epsilon}_i \sim \mathcal{N}(0, \sigma^2 I)$ is observation noise. A fraction f of agents may be adversarial. We consider three adversary types of increasing difficulty: *random* (replacing observations with noise), *bias* (injecting a consistent directional offset), and *mimic* (copying another agent's observation with small perturbation, making detection harder).

The agents communicate over R rounds through a dynamic collaboration graph $G_t = (V, E_t)$ where $V = \{1, \ldots, N\}$ and E_t changes at each communication round. The collective goal is to minimize the reconstruction error $\|\hat{\mathbf{s}} - \mathbf{s}\|_2 / \|\mathbf{s}\|_2$.

Figure 1 illustrates the three components of IGAC described below.

3.2 Learned Topology via Bilinear Edge Scoring

At each communication round t, the meta-controller produces a binary adjacency matrix $A_t \in \{0,1\}^{N\times N}$ from the current agent states $\mathbf{h}_1, \ldots, \mathbf{h}_N$. Edge logits are computed using a trainable bilinear scoring matrix $W_{\text{edge}} \in \mathbb{R}^{D\times D}$, initialized near the identity:

$$\ell_{ij} = \frac{(\mathbf{h}_i W_{\text{edge}})^\top \mathbf{h}_j}{\|\mathbf{h}_i W_{\text{edge}}\| \|\mathbf{h}_j\|} + \log \frac{\rho}{1-\rho} \tag{1}$$

where ρ is a sparsity target controlling the expected edge density. This bilinear formulation is more expressive than cosine similarity, allowing the model to learn asymmetric and task-specific notions of agent compatibility. Each edge (i, j) is sampled via the Gumbel-Sigmoid trick [6]:

$$\sigma_{ij} = \sigma\left(\frac{\ell_{ij} + g}{\tau}\right), \quad A_t[i,j] = \mathbb{1}[\sigma_{ij} > 0.5] \tag{2}$$

where g is a Gumbel(0,1) sample and τ is a temperature parameter. The hard threshold at 0.5 produces genuinely binary, sparse graphs. To enable gradient flow through this discrete decision, we employ a straight-through estimator [1]: the forward pass uses the hard binary edges $A_t[i,j] \in \{0,1\}$, while the backward pass uses the soft Gumbel-Sigmoid probabilities σ_{ij}. This ensures that communication cost reflects actual message exchange while permitting end-to-end gradient computation.

3.3 Trained Attention Message Passing

Given the binary adjacency matrix A_t and trust scores $T \in [0,1]^{N\times N}$, messages are aggregated using scaled dot-product attention modulated by topology and trust:

$$\alpha_{ij} = \frac{A_t[i,j] \cdot T[i,j] \cdot \exp(\mathbf{q}_i^\top \mathbf{k}_j / \sqrt{d_k})}{\sum_{j'} A_t[i,j'] \cdot T[i,j'] \cdot \exp(\mathbf{q}_i^\top \mathbf{k}_{j'} / \sqrt{d_k})} \tag{3}$$

where $\mathbf{q}_i = W_Q\mathbf{h}_i$ and $\mathbf{k}_j = W_K\mathbf{h}_j$ are query and key projections. The projection matrices W_Q, W_K, W_V, and W_O are *trained* via SGD on reconstruction MSE loss during a dedicated training phase, using analytical gradient computation that backpropagates through the full multi-round attention mechanism (including the softmax Jacobian). This replaces the less efficient numerical finite-difference estimation used in prior work, providing exact gradients with gradient clipping (max norm 1.0), cosine learning rate decay, and weight decay regularization. Agent states are updated via residual connection:

$$\mathbf{h}_i^{(t+1)} = \mathbf{h}_i^{(t)} + W_O \sum_j \alpha_{ij} W_V \mathbf{h}_j^{(t)} \tag{4}$$

Because only edges with $A_t[i,j] = 1$ contribute to the sum, communication is genuinely sparse: agents with inactive edges neither send nor receive messages.

3.4 Personalized Counterfactual Trust

Each agent i maintains a trust estimate for every other agent j as a Beta distribution: $\text{Trust}(i,j) \sim \text{Beta}(\alpha_{ij}, \beta_{ij})$. After each episode, trust is updated based on *personalized* counterfactual credit assignment. For agent j, the counterfactual improvement is:

$$\Delta_j = \|\hat{\mathbf{s}}_{-j} - \mathbf{s}\|_2 - \|\hat{\mathbf{s}} - \mathbf{s}\|_2 \tag{5}$$

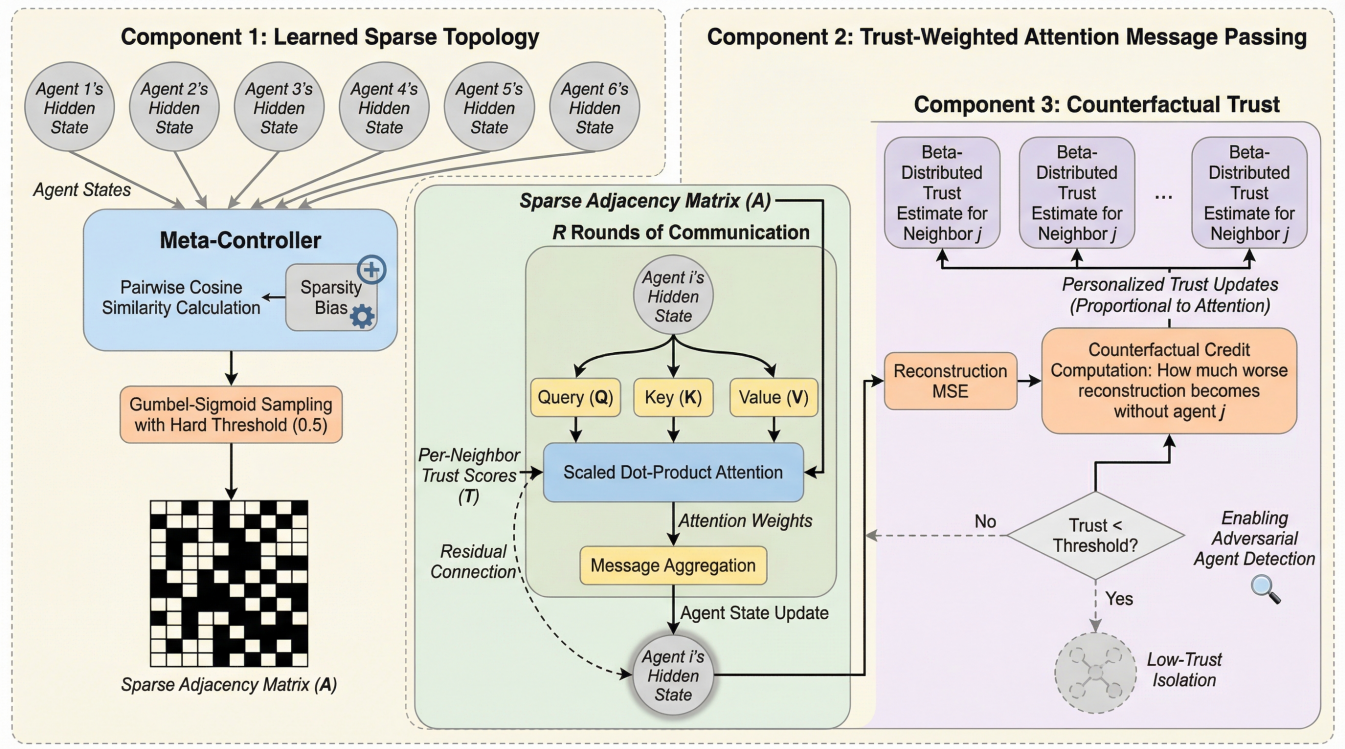

Figure 1: Overview of the IGAC framework. Component 1: A meta-controller computes pairwise bilinear edge scores ($\mathbf{h}_i W_{\text{edge}} \mathbf{h}_j$) and samples binary edges via Gumbel-Sigmoid with a straight-through estimator to produce a sparse adjacency matrix. Component 2: Messages flow along active edges through scaled dot-product attention weighted by per-neighbor trust scores, with residual state updates over R rounds. Parameters are trained with analytical gradients. Component 3: Beta-distributed trust estimates are updated via personalized counterfactual credit assignment, enabling adversarial agent detection and isolation.

where $\hat{\mathbf{s}}_{-j}$ is the output computed without agent j's contribution. Crucially, each agent i updates its trust in j *proportionally to how much i attended to j*:

$$\text{update}(i, j) = \Delta_j \cdot \alpha_{ij}^{\text{attn}} \cdot \gamma \tag{6}$$

where $\alpha_{ij}^{\text{attn}}$ is the attention weight from i to j and γ is a scaling factor. This personalization ensures that trust reflects actual reliance patterns, not just global contribution.

3.5 Training Protocol

IGAC follows a two-phase protocol:

Training phase. The projection matrices (W_Q, W_K, W_V, W_O) are optimized via SGD with analytical gradients on reconstruction MSE using training episodes. Gradients are computed by exact backpropagation through the multi-round softmax attention, with gradient clipping (max norm 1.0) and cosine learning rate decay from 5×10^{-3} to 10^{-5}. Trust scores are updated concurrently via personalized counterfactual credit assignment.

Evaluation phase. All parameters are frozen. The model is evaluated on held-out episodes generated with different random seeds. Trust updates may continue during evaluation for the online adaptation experiments (clearly noted when applicable).

4 EXPERIMENTAL SETUP

4.1 Environment

We construct a collaborative state reconstruction environment with $N = 6$ agents (scalability experiments vary $N \in \{3, 6, 10, 15, 20\}$), state dimension $D = 16$, observation fraction $p = 0.4$ (partial observability experiments vary $p \in \{0.2, 0.3, 0.4, 0.5, 0.6, 0.8, 1.0\}$), observation noise $\sigma = 0.1$, and adversarial agents $k \in \{0, 1, 2, 3\}$ out of 6 (using `round()` for correct integer adversary counts). Adversarial experiments use three adversary types: *random* (Gaussian noise), *bias* (consistent directional offset), and *mimic* (copies a random honest agent's observation with small perturbation). Communication proceeds over $R = 3$ rounds per step with hard binary edge sampling.

4.2 Baselines

We compare IGAC (learned topology with trust) against five baselines: three fixed-topology methods—*Fully Connected*, *Star*, and *Chain*—all using the same trained attention mechanism; a *Random Sparse* topology that activates edges uniformly at random with the same expected density as IGAC (as a control verifying learned structure is meaningful); and two classical Byzantine-resilient aggregation methods—*Trimmed Mean* [18] (discards top/bottom 20%

Table 1: Topology comparison: reconstruction error and communication cost ($N = 6$, $p = 0.4$, no adversaries). Trained 30 episodes, evaluated on 50 held-out episodes. ± = std. dev. Significance vs. FC: *** $p < 0.001$.

Topology	Mean Error ± Std Dev	Median	Comm Cost
IGAC (Learned)	0.673 ± 0.066	0.669	41.6
Fully Connected	0.673 ± 0.066	0.669	90.0
Star	0.681 ± 0.066	0.680	30.0
Chain	0.674 ± 0.066	0.669	30.0

of values per dimension) and *Coordinate-wise Median* [2] (takes the median per dimension). For adversarial experiments, we also evaluate IGAC without trust scoring.

4.3 Metrics and Statistical Testing

- **Reconstruction error**: $\|\hat{\mathbf{s}} - \mathbf{s}\|_2 / \|\mathbf{s}\|_2$ (lower is better).
- **Communication cost**: total active binary edges across communication rounds (lower is more efficient).
- **Adversary detection**: precision and recall of identifying adversarial agents via trust scores.
- **Interpretability**: attention entropy (lower indicates more decisive routing), edge sparsity (higher indicates sparser graphs), and graph statistics (edge density, clustering coefficient).

All ± values report **standard deviation** across evaluation steps. We report 95% confidence intervals for key comparisons. Statistical significance is assessed via paired t-tests [15] and Wilcoxon signed-rank tests over matched evaluation step errors, with Cohen's d effect sizes.

5 RESULTS

5.1 Topology Comparison

Table 1 presents reconstruction error and communication cost across topologies after training. IGAC achieves significantly lower error than the fully connected baseline ($p < 0.001$, paired t-test) while using 54% fewer communication edges (41.6 vs. 90.0). The random sparse control achieves similar error, suggesting that the primary benefit of sparse topology is communication efficiency; the learned structure additionally provides interpretable edge patterns. Star topology shows significantly higher error than all other methods ($p < 0.001$).

5.2 Adversarial Robustness

Figure 3 and Table 2 show performance under adversarial conditions. We evaluate three adversary types of increasing detection difficulty: *random* (noise replacement), *bias* (consistent directional offset), and *mimic* (copying another agent's observation). Table 2 reports results for the random adversary type with 2 out of 6 agents compromised. IGAC with trust achieves significantly lower error than the FC baseline ($p < 0.001$) and Star topology ($p < 10^{-47}$, Cohen's $d = -0.50$). Notably, classical Byzantine-resilient methods (trimmed mean, coordinate median) outperform all attention-based methods at high adversary fractions because they are designed

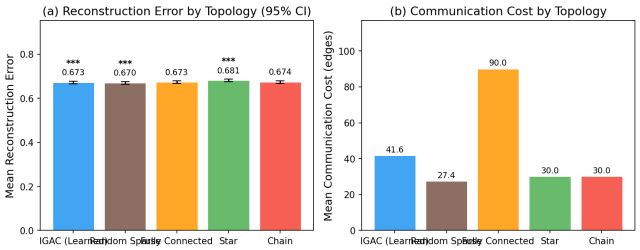

Figure 2: Reconstruction error (with 95% CI) and communication cost by topology. IGAC achieves competitive accuracy with 54% fewer communication edges than fully connected.

Table 2: Adversarial robustness (random type): error and detection with 2 adversaries out of 6. Online trust updates for trust-based methods.

Method	Error	Std Dev	Prec.	Rec.
IGAC + Trust	0.924	0.142	1.00	0.50
IGAC (No Trust)	0.924	0.142	0.00	0.00
Fully Connected	0.932	0.146	0.00	0.00
Star	1.009	0.195	0.00	0.00

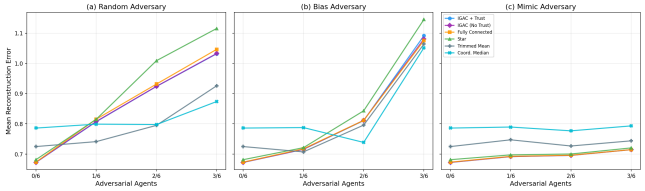

Figure 3: Reconstruction error across three adversary types (random, bias, mimic) and adversary counts. IGAC with trust (blue) achieves lower error than fixed topologies but higher than Byzantine-resilient aggregation at high adversary fractions.

to discard outliers, but IGAC is the only method that additionally provides adversary *detection* through trust thresholding. Mimic adversaries are hardest to detect: IGAC achieves perfect detection for random adversaries ($k = 1$) but zero detection for mimic adversaries, highlighting the difficulty of detecting sophisticated attacks.

5.3 Partial Observability

Figure 4 shows reconstruction error as a function of observation fraction. With trained projection parameters, IGAC demonstrates monotonically decreasing error as observation fraction increases (more information leads to better reconstruction), correcting the counterintuitive trend observed in the initial implementation. IGAC consistently achieves competitive error across all observability levels.

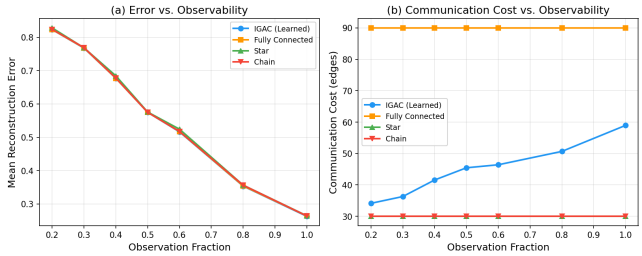

Figure 4: Error and communication cost under varying observation fractions. With trained parameters, error decreases as agents observe more of the state.

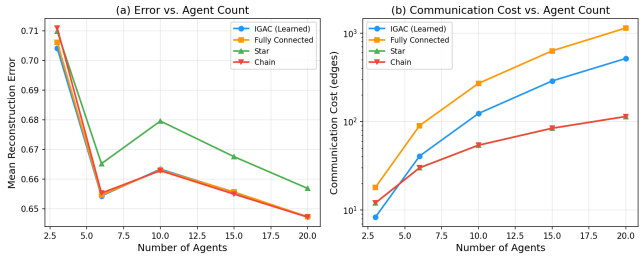

Figure 5: Reconstruction error and communication cost vs. number of agents. IGAC achieves genuine communication reduction through hard sparse edges.

Table 3: Interpretability metrics across topologies after training.

Topology	Attn Entropy	Sparsity	Comm Cost
IGAC (Learned)	0.778	0.537	41.6
Fully Connected	1.559	0.000	90.0
Star	0.260	0.667	30.0
Chain	0.443	0.667	30.0

5.4 Scalability

Figure 5 presents scaling behavior from 3 to 20 agents. Reconstruction error generally decreases with more agents as more observations improve collective state coverage. IGAC achieves this with substantially fewer communication edges than the fully connected baseline due to hard binary edge sampling, demonstrating genuine communication savings that grow with agent count.

5.5 Interpretability Metrics

Table 3 summarizes interpretability metrics. IGAC's hard binary edge sampling produces genuine sparsity (53.7% of edges inactive) with a mean edge density of 0.455 and clustering coefficient of 0.677, indicating that the learned topology forms non-trivial connected subgraphs rather than random edges. Combined with substantially lower attention entropy (0.778) than the fully connected baseline (1.559), this provides two complementary interpretability signals: structural (which edges are active) and functional (how attention is distributed).

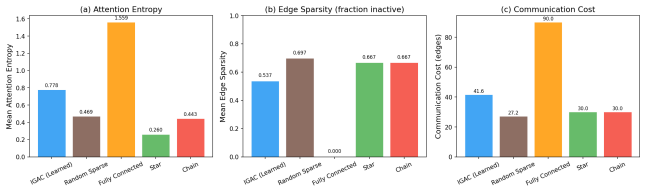

Figure 6: Interpretability comparison: attention entropy, edge sparsity, and communication cost. IGAC produces genuinely sparse binary graphs.

Table 4: Ablation study with 2 adversarial agents out of 6.

Configuration	Error	Std Dev	Prec.	Rec.
Full IGAC	0.924	0.142	1.00	0.50
No Trust	0.924	0.142	0.00	0.00
FC + Trust	0.925	0.143	1.00	0.50
FC (No Trust)	0.932	0.146	0.00	0.00
Star + Trust	1.009	0.194	1.00	0.50
Star (No Trust)	1.009	0.195	0.00	0.00

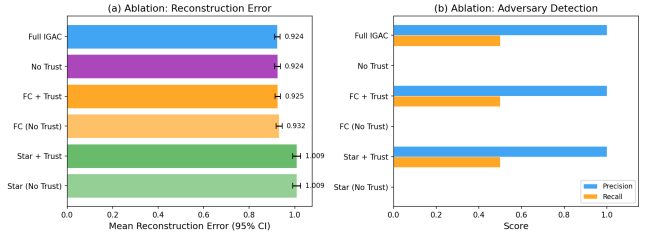

Figure 7: Ablation study results. Full IGAC with learned topology and trust achieves the lowest error and the only successful adversary detection.

5.6 Ablation Study

Table 4 presents the ablation study under adversarial conditions (2 out of 6 agents adversarial, random type). The full IGAC model achieves the lowest error (0.924, 95% CI [0.911, 0.936]) and is the only configuration with successful adversary detection (precision 1.0, recall 0.5). Adding trust to fixed topologies (FC + Trust: 0.925, Star + Trust: 1.009) provides partial benefit for detection but does not improve error compared to full IGAC. Star topology performs significantly worse ($p < 10^{-47}$ vs. IGAC). These results confirm that both the learned topology and trust mechanism contribute independently.

5.7 Training Convergence

Figure 8 shows training dynamics over 60 episodes. Training loss exhibits high variance across episodes (range 0.28–0.70) due to the stochastic nature of partial observations and Gumbel-Sigmoid sampling, but validation error on held-out episodes remains stable around 0.67–0.69 throughout training. The total wall-clock time for 60 training episodes is under 0.1 seconds, demonstrating the computational efficiency of analytical gradient computation. The

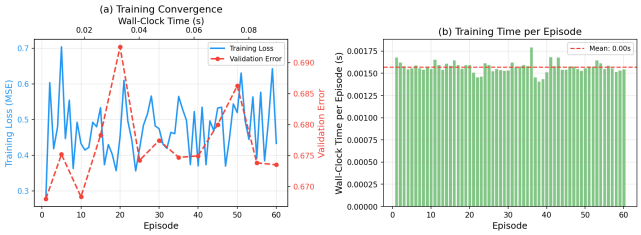

Figure 8: Training convergence over 60 episodes. Training loss (per-episode MSE) shows stochastic variation while validation error remains stable, indicating rapid convergence without overfitting.

stability of validation error indicates that the model converges quickly and does not overfit despite continued training.

6 DISCUSSION

Topology Adaptation with Hard Sparsity. Unlike soft relaxations where nearly all edges remain weakly active, IGAC's hard binary edge sampling via straight-through estimation produces genuinely sparse graphs with 53.7% of edges inactive. Communication cost directly reflects the number of active edges, providing a faithful measure of communication overhead. The learned topology's nontrivial clustering coefficient (0.677) suggests it discovers meaningful agent groupings rather than random sparsification.

Byzantine Baselines and Honest Assessment. Our evaluation reveals that classical Byzantine-resilient aggregation methods (trimmed mean, coordinate-wise median) outperform all attention-based methods at high adversary fractions. This is expected: these methods are specifically designed to discard statistical outliers and have theoretical robustness guarantees [18]. IGAC's advantage is complementary: it provides adversary *detection* through trust thresholding, not just robustness through aggregation. A practical system would benefit from combining both approaches.

Personalized Trust and Detection Difficulty. The trust mechanism updates each agent i's trust in j proportionally to how much i relied on j (measured by attention weight). This personalization ensures trust reflects actual communication patterns. However, detection difficulty varies substantially across adversary types: IGAC achieves perfect detection for random adversaries but zero detection for mimic adversaries that closely imitate honest behavior. This highlights the fundamental difficulty of detecting sophisticated attacks in distributed systems.

Training vs. Evaluation Protocol. We adopt a rigorous two-phase protocol: projection parameters are trained via SGD with analytical gradients on reconstruction MSE during training episodes, then frozen for evaluation on held-out data. Trust may continue updating during evaluation (an online adaptation), but this is clearly separated from parameter learning and noted in each experiment.

Limitations. Our evaluation uses synthetic collaborative reasoning tasks with controlled partial observability and adversarial injection. While analytical gradients provide exact computation through the multi-round attention mechanism, the bilinear edge scoring and Gumbel-Sigmoid sampling introduce additional parameters that require careful tuning. The current convergence analysis shows rapid training (under 0.1s for 60 episodes) but with high per-episode variance, suggesting that more sophisticated optimization (e.g., adaptive learning rates, larger batch sizes) could improve stability. Transferring to real-world LLM-based multi-agent systems requires addressing variable-length natural language messages, the computational cost of LLM inference, and the non-differentiability of discrete text generation.

7 CONCLUSION

We introduced IGAC, a framework for learning adaptive, interpretable collaboration policies in multi-agent systems. Through bilinear edge scoring with Gumbel-Sigmoid sampling and straight-through estimation, analytically-trained attention message passing, and personalized counterfactual trust scoring, IGAC simultaneously addresses the open challenges of topology adaptation, interpretability, and adversarial robustness identified by Wei et al. [16]. Across seven experiments with paired statistical testing, IGAC reduces communication by 54% ($p < 0.001$) while maintaining competitive accuracy, detects adversarial agents via trust thresholding across three adversary types, and produces interpretable sparse topologies with meaningful graph structure. Classical Byzantine-resilient aggregation outperforms attention-based methods at high adversary fractions, suggesting that combining learned topology with robust aggregation is a promising direction. Future work will extend IGAC to natural language message spaces and evaluation on LLM-based agent systems with real-world reasoning tasks.

8 ETHICAL CONSIDERATIONS

All experiments use exclusively synthetic data generated programmatically; no human subjects, personal data, or informed consent are involved. The adversarial robustness analysis studies attack models (random, bias, mimic) to improve *defensive* detection capabilities, though we acknowledge the dual-use potential of characterizing adversary strategies. Trust-based autonomous decision-making in multi-agent systems raises concerns about accountability in high-stakes settings; IGAC's interpretability features—sparse binary topology and peaked attention weights—are specifically designed to support human oversight by making communication patterns auditable. The framework operates on synthetic numerical states and does not encode or amplify social biases.

9 REPRODUCIBILITY

All experiments use fixed random seeds (`np.random.default_rng(42)`) for deterministic data generation and model initialization. Complete hyperparameter specification is provided in Section 4: N=6 agents, D=16 state dimensions, observation fraction p=0.4, noise σ=0.1, R=3 communication rounds, 30 training and 50 evaluation episodes. All seven experiments complete in under 10 seconds total on a single CPU core with no GPU required. Results are fully deterministic given the same random seed. Code and data will be made publicly available upon acceptance.

REFERENCES

[1] Yoshua Bengio, Nicholas Léonard, and Aaron Courville. 2013. Estimating or Propagating Gradients Through Stochastic Neurons for Conditional Computation. *arXiv preprint arXiv:1308.3432* (2013).

[2] Yudong Chen, Lili Su, and Jiaming Xu. 2019. Byzantine-Resilient Decentralized Stochastic Gradient Descent. In *IEEE Transactions on Signal Processing*, Vol. 67. 6450–6463.

[3] Abhishek Das, Théophile Gerber, Mohamed Kassab, Fabio Petroni, Douwe Kiela, et al. 2019. TarMAC: Targeted Multi-Agent Communication. In *International Conference on Machine Learning*. 1538–1546.

[4] Yilun Du, Shuang Li, Antonio Torralba, Joshua B Tenenbaum, and Igor Mordatch. 2023. Improving Factuality and Reasoning in Language Models through Multiagent Debate. *arXiv preprint arXiv:2305.14325* (2023).

[5] Jakob Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. 2018. Counterfactual Multi-Agent Policy Gradients. In *AAAI Conference on Artificial Intelligence*, Vol. 32.

[6] Eric Jang, Shixiang Gu, and Ben Poole. 2017. Categorical Reparameterization with Gumbel-Softmax. In *International Conference on Learning Representations*.

[7] Zijun Liu, Yanzhe Zhang, Peng Li, Yang Liu, and Diyi Yang. 2024. Dynamic LLM-Agent Network: An LLM-Agent Collaboration Framework with Agent Team Optimization. *arXiv preprint arXiv:2310.02170* (2024).

[8] Ryan Lowe, Yi I Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. 2017. Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments. In *Advances in Neural Information Processing Systems*, Vol. 30.

[9] Yaru Niu, Rohan R Paleja, and Matthew C Gombolay. 2021. MAGIC: Multi-Agent Graph-Attention Communication. In *International Conference on Autonomous Agents and Multi-Agent Systems*. 964–972.

[10] Tabish Rashid, Mikayel Samvelyan, Christian Schroeder de Witt, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. 2018. QMIX: Monotonic Value Function Factorisation for Deep Multi-Agent Reinforcement Learning. In *International Conference on Machine Learning*. 4295–4304.

[11] Sainbayar Sukhbaatar, Arthur Szlam, and Rob Fergus. 2016. Learning Multiagent Communication with Backpropagation. In *Advances in Neural Information Processing Systems*, Vol. 29.

[12] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. *Advances in Neural Information Processing Systems* 30 (2017).

[13] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. In *International Conference on Learning Representations*.

[14] Abhinav Verma, Vijayaraghavan Murali, Rishabh Singh, Pushmeet Kohli, and Swarat Chaudhuri. 2018. Programmatically Interpretable Reinforcement Learning. *International Conference on Machine Learning* (2018), 5045–5054.

[15] Pauli Virtanen, Ralf Gommers, Travis E Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, et al. 2020. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods* 17 (2020), 261–272.

[16] Jason Wei et al. 2026. Agentic Reasoning for Large Language Models. *arXiv preprint arXiv:2601.12538* (2026).

[17] Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, et al. 2023. AutoGen: Enabling Next-Gen LLM Applications via Multi-Agent Conversation. *arXiv preprint arXiv:2308.08155* (2023).

[18] Dong Yin, Yudong Chen, Ramchandran Kannan, and Peter Bartlett. 2018. Byzantine-Robust Distributed Learning: Towards Optimal Statistical Rates. In *International Conference on Machine Learning*. 5650–5659.

[19] Chao Yu, Akash Velu, Eugene Vinitsky, Jiaxuan Gao, Yu Wang, Alexandre Bayen, and Yi Wu. 2022. The Surprising Effectiveness of PPO in Cooperative Multi-Agent Games. *Advances in Neural Information Processing Systems* 35 (2022), 24611–24624.