

Explainable Decision Graphs for Trustworthy Autonomous Multi-Agent Systems

Siddhi Hon

siddhi.hon_24uds@sanjivani.edu.in

sanjivani university

Research Article

Keywords: Explainable Artificial Intelligence, Multi-Agent Systems, Explainable Decision Graphs, Trustworthy AI, Autonomous Agents, Decision Traceability, AI Safety, Accountability

Posted Date: October 19th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11396017/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: The authors declare no competing interests.

Explainable Decision Graphs for Trustworthy Autonomous Multi-Agent Systems

Siddhi Hon

Department of Artificial Intelligence and Data Science
Sanjivani University, Maharashtra, India
Email: siddhihon25@gmail.com

Abstract—Autonomous Multi-Agent Systems (MAS) are increasingly being considered for critical applications such as healthcare, finance, cybersecurity, and autonomous transportation. These systems consist of multiple intelligent agents that can independently perceive information, reason about their environment, communicate with other agents, and execute decisions. Although multi-agent architectures provide flexibility and scalability, their distributed decision-making process can make the overall system difficult to understand, audit, and verify.

Existing Explainable Artificial Intelligence (XAI) techniques such as Local Interpretable Model-Agnostic Explanations (LIME) and SHapley Additive exPlanations (SHAP) primarily focus on explaining individual machine-learning predictions. They provide limited support for representing collaborative reasoning, inter-agent communication, conflicts, and collective decisions. This paper proposes an Explainable Decision Graph (EDG) framework for representing the reasoning process of autonomous multi-agent systems in a structured and interpretable form.

The proposed framework models agents, observations, evidence, communications, intermediate decisions, conflicts, verification events, and final actions as interconnected graph components. A verification layer enables decision pathways to be examined against predefined safety and policy constraints. The resulting graph can provide different levels of explanations for end users, developers, and auditors.

The paper defines an experimental methodology using multi-agent execution traces and proposes evaluation through explanation fidelity, completeness, decision traceability, human comprehensibility, decision accuracy, explanation consistency, and computational overhead. Empirical numerical results are intentionally reserved for the implementation and evaluation stage.

Index Terms—Explainable Artificial Intelligence, Multi-Agent Systems, Explainable Decision Graphs, Trustworthy AI, Autonomous Agents, Decision Traceability, AI Safety, Accountability

I. INTRODUCTION

A. Background

Artificial Intelligence (AI) has developed from conventional prediction systems into autonomous systems capable of reasoning, planning, communication, and decision-making. One important development is the Autonomous Multi-Agent System (MAS), in which multiple intelligent agents collaborate to accomplish a common objective.

Each agent in a MAS can perform a specialized function such as perception, analysis, planning, risk assessment, verification, or execution. By distributing responsibilities among

agents, a MAS can solve complex problems more effectively than an isolated agent in certain applications.

Autonomous MAS are increasingly relevant to domains including healthcare, financial services, cybersecurity, autonomous transportation, and industrial automation. However, increased autonomy and collaboration introduce an important challenge: understanding how a final decision was generated.

A final system decision may depend on observations from multiple agents, messages exchanged between agents, intermediate decisions, conflicting recommendations, changing environmental conditions, and verification procedures. Therefore, examining only the final output does not provide a complete explanation of the underlying reasoning process.

Explainable Artificial Intelligence (XAI) attempts to address this challenge by making AI decisions more understandable to humans. However, many existing XAI methods are designed around individual machine-learning models rather than distributed multi-agent reasoning.

This research therefore investigates a graph-based explanation mechanism capable of representing the complete decision pathway of an autonomous multi-agent system.

B. Problem Statement

Autonomous Multi-Agent Systems can independently make intelligent decisions, but their reasoning processes can be difficult to interpret because decisions are distributed across multiple agents.

This lack of transparency can reduce user trust, complicate debugging, and make it difficult to determine whether a decision is safe, fair, consistent, and compliant with predefined requirements.

Existing explainability methods primarily focus on individual models and provide limited support for collaborative reasoning among multiple autonomous agents.

The problem addressed in this research is therefore:

How can the distributed reasoning process of autonomous multi-agent systems be represented in an understandable, traceable, and verifiable form?

C. Research Gap

Existing XAI techniques provide important foundations for interpreting machine-learning predictions. LIME explains individual predictions using locally interpretable approximations

[1]. SHAP provides feature contribution values based on Shapley-value concepts [2].

However, these approaches primarily describe the behavior of an individual predictive model. They do not naturally represent:

- which agent contributed to a decision;
- which agent influenced another agent;
- what evidence was exchanged;
- where agents disagreed;
- how conflicts were resolved;
- which verification rules were applied; and
- how the final action was produced.

Lipton highlighted that interpretability is not a single well-defined concept and can have different meanings depending on context [3]. Therefore, explainability for MAS should consider not only individual predictions but also interactions and system-level behavior.

The research gap can consequently be summarized as follows:

Model-level explanation → Agent-level explanation → System-level EDG explanation

D. Research Objectives

The major objectives of this research are:

- 1) To develop an Explainable Decision Graph framework representing the reasoning process of autonomous multi-agent systems.
- 2) To improve transparency and user understanding through structured visual explanations.
- 3) To represent relationships among agents, evidence, decisions, communications, and actions.
- 4) To enable users to trace final decisions back to their contributing agents and evidence.
- 5) To support verification of decisions against predefined safety and policy constraints.
- 6) To evaluate the proposed framework using quantitative and qualitative explainability metrics.

E. Research Contributions

The main contributions of this research are:

- 1) A graph-based representation for system-level explanation of multi-agent decisions.
- 2) An interaction-aware decision trace connecting agents, evidence, intermediate decisions, and final actions.
- 3) A verification-aware explanation mechanism.
- 4) Multi-level explanations for users, developers, and auditors.
- 5) An experimental methodology for evaluating EDG against existing explainability approaches.

II. RELATED WORK

A. LIME

Ribeiro et al. introduced Local Interpretable Model-Agnostic Explanations (LIME), a method for explaining individual predictions of machine-learning classifiers [1]. LIME creates a local approximation of a complex model around a

specific prediction. Its major advantage is that it can be applied without modifying the original predictive model.

However, LIME focuses on individual predictions. In a multi-agent system, the final outcome may depend on a sequence of decisions from several agents, which cannot be completely represented by one local explanation.

B. SHAP

Lundberg and Lee proposed SHAP, a unified framework for interpreting model predictions using Shapley-value concepts [2]. SHAP assigns contribution values to input features, helping users understand which features influenced a particular prediction.

Although feature attribution can improve transparency for individual agents, SHAP does not inherently model communication, coordination, or influence between autonomous agents.

C. Model Interpretability

Lipton examined different meanings of model interpretability and discussed the difficulty of defining interpretability as a single property [3]. This observation is important for multi-agent systems because an explanation may need to satisfy different users. For example, an end user may require a short explanation, while a developer may require detailed information about intermediate decisions.

D. Rigorous Evaluation of Interpretability

Doshi-Velez and Kim proposed a framework for developing a more rigorous science of interpretable machine learning [4]. Their work emphasizes that interpretability should be evaluated scientifically and according to application requirements and user needs. This motivates the evaluation methodology proposed for EDG.

E. Responsible XAI

Arrieta et al. provided a comprehensive survey of Explainable Artificial Intelligence, including concepts, taxonomies, opportunities, and challenges related to responsible AI [5]. Their work establishes the importance of explainability for transparency, accountability, and trustworthy AI systems.

F. Comparison of Existing Approaches

Most conventional XAI systems operate at the model or prediction level. They may provide feature importance, local explanations, or global model descriptions. However, autonomous MAS introduce an additional layer of complexity because several models or agents can interact before a final decision is produced.

A system-level explanation therefore requires information about both individual agent reasoning and inter-agent interactions.

TABLE I
CONCEPTUAL COMPARISON OF EXPLAINABILITY APPROACHES

Method	Model	Agent	Inter.	Trace	Verify
LIME	Yes	Partial	No	No	No
SHAP	Yes	Partial	No	No	No
Agent-level XAI	Partial	Yes	Partial	Partial	Partial
Proposed EDG	Yes	Yes	Yes	Yes	Yes

Table I represents the intended scope of the approaches rather than experimental performance.

III. PROPOSED METHODOLOGY

A. System Overview

The proposed Explainable Decision Graph acts as an explanation and traceability layer around an autonomous multi-agent system. The general workflow is:

$$\begin{aligned}
 &\text{Environment} \rightarrow \text{Observations} \rightarrow \text{Agents} \\
 &\quad \rightarrow \text{Interactions} \rightarrow \text{Decisions} \\
 &\quad \rightarrow \text{EDG} \rightarrow \text{Verification} \rightarrow \text{Explanation.}
 \end{aligned} \tag{1}$$

The EDG does not replace the underlying agents. Instead, it records relevant execution events and transforms them into a structured representation.

B. System Architecture

The proposed system consists of four major modules:

- 1) Agent Execution and Monitoring
- 2) Inter-Agent Interaction Tracking
- 3) Explainable Decision Graph Construction
- 4) Verification and Explanation Generation

The conceptual architecture is illustrated in Fig. 1.

C. Module 1: Agent Execution and Monitoring

The first module monitors autonomous agent execution. For each relevant event, the framework records:

- agent identifier;
- observation or input;
- intermediate decision;
- confidence score;
- supporting evidence; and
- timestamp.

A generic decision event can be represented as:

$$E_i = (A_i, I_i, D_i, C_i, T_i), \tag{2}$$

where A_i denotes the agent, I_i its input, D_i its decision, C_i its confidence, and T_i the event timestamp.

D. Module 2: Inter-Agent Interaction Tracking

The second module records communication between agents. An interaction can be represented as:

$$I_{AB} = (A, B, M, T), \tag{3}$$

where A is the sender, B is the receiver, M is the message, and T represents the communication time.

The module also records conflicts. For example, if Agent A recommends D_1 and Agent B recommends D_2 , a conflict relationship is created in the decision graph.

E. Module 3: Explainable Decision Graph Construction

The decision graph is represented as:

$$G = (V, E), \tag{4}$$

where V represents nodes and E represents relationships. The node set is defined as:

$$V = V_A \cup V_I \cup V_E \cup V_D \cup V_V \cup V_{Act}, \tag{5}$$

where:

- V_A represents agent nodes;
- V_I represents input or observation nodes;
- V_E represents evidence nodes;
- V_D represents decision nodes;
- V_V represents verification nodes; and
- V_{Act} represents action nodes.

Edges represent relationships such as supports, influences, communicates, contradicts, verifies, and produces.

The graph enables the final system action to be traced backward through intermediate decisions, contributing agents, exchanged evidence, and verification events. This system-level representation extends individual model explanations by explicitly representing inter-agent dependencies and collaborative reasoning.

F. EDG Construction Algorithm

G. Module 4: Verification and Explanation Generation

The verification layer checks whether decisions satisfy predefined constraints.

A simplified verification condition is:

$$D \models R, \tag{6}$$

where D is the decision and R represents the applicable rules.

After verification, relevant graph paths are converted into human-readable explanations.

The system can provide:

- 1) **User-level explanation:** short explanation of the main reason for the decision.
- 2) **Developer-level explanation:** detailed agent, evidence, interaction, and verification information.
- 3) **Auditor-level explanation:** complete trace of decision events and verification results.

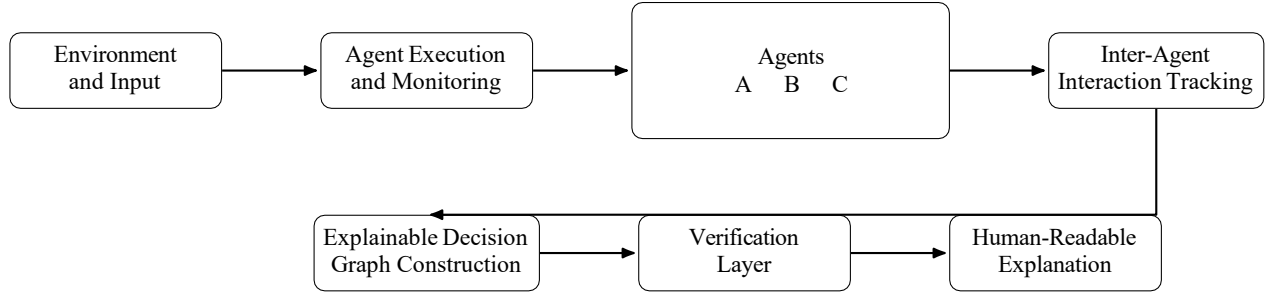


Fig. 1. Conceptual architecture of the proposed Explainable Decision Graph framework. Agent execution is monitored, inter-agent interactions are recorded, decision graphs are constructed, decision pathways are verified, and human-readable explanations are generated.

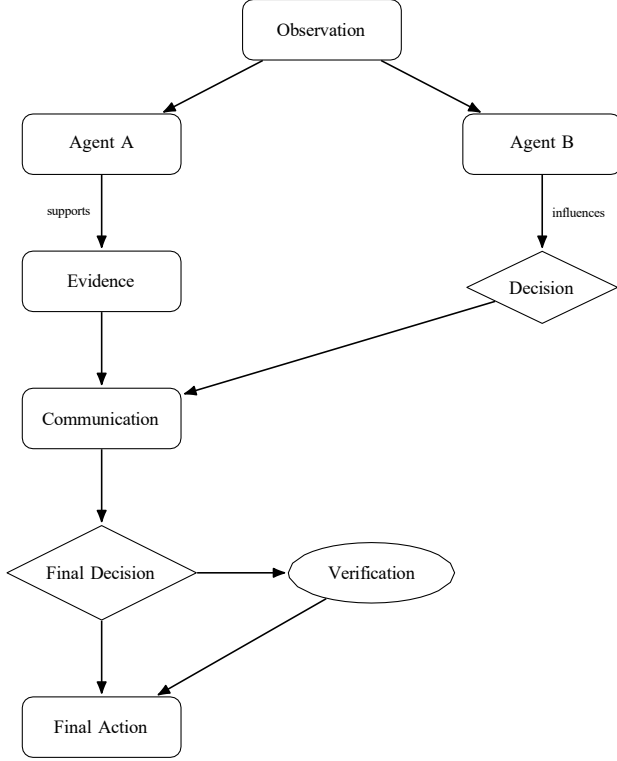


Fig. 2. Example Explainable Decision Graph showing observations, agent contributions, evidence, communication, decision formation, verification, and final action.

H. Mathematical Formulation

Let the set of autonomous agents be:

$$A = \{a_1, a_2, \dots, a_n\}. \quad (7)$$

At time t , agent i receives observation o_i^t and information m_i^t from other agents.

The agent produces a decision:

$$d_i^t = \pi_i(o_i^t, m_i^t), \quad (8)$$

where π_i is the policy of agent i .

The collective system decision can be represented as:

$$D^t = F(d_1^t, d_2^t, \dots, d_n^t). \quad (9)$$

Algorithm 1 Explainable Decision Graph Construction

Require: Multi-agent system MAS , rules R

Ensure: Decision graph G , explanation X

- 1: Initialize $G = (V, E)$
- 2: Start MAS
- 3: **while** MAS is executing **do**
- 4: Observe agent event
- 5: Record event attributes
- 6: Add event node to G
- 7: Record communication
- 8: Add interaction edge
- 9: Link decision to evidence
- 10: Detect and link conflicts
- 11: Apply rules R
- 12: Record verification result
- 13: **end while**
- 14: Identify final decision
- 15: Generate explanation X
- 16: Return G, X

The explanation associated with the final decision is:

$$X(D^t) = \{P(D^t), E(D^t), A(D^t), V(D^t)\}, \quad (10)$$

where $P(D^t)$ is the decision pathway, $E(D^t)$ is supporting evidence, $A(D^t)$ is the set of contributing agents, and $V(D^t)$ contains verification information.

IV. EXPERIMENTAL SETUP

A. Dataset

The proposed EDG framework requires datasets containing multi-agent interactions and decision traces. A suitable benchmark for the proposed evaluation is the MAESTRO MAS Benchmark, which provides execution traces from multiple multi-agent systems. The benchmark contains data for 12 MAS systems and provides trace-level information including agent names, timestamps, execution duration, status, communication metadata, events, run identifiers, and system example names.

The benchmark provides a systems-level view of multi-agent behavior and is therefore suitable for constructing decision-trace representations.

The EDG preprocessing stage maps available trace information into the following conceptual event categories:

- agent observations;
- agent actions;
- communication events;
- intermediate decisions;
- evidence;
- conflicts, where identifiable;
- verification outcomes; and
- final system actions.

1) *Dataset Overview*: The currently available MAESTRO dataset reports 12 MAS systems, approximately 117K rows in its trace subset, and 439,957 rows across the consolidated dataset. The dataset is distributed in Parquet format and includes execution traces and hardware-metric records.

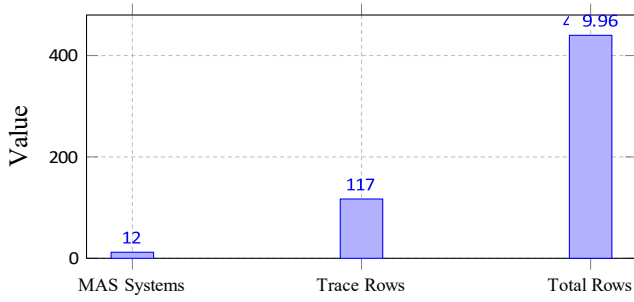


Fig. 3. Overview of the selected MAESTRO MAS benchmark. The first bar represents 12 MAS systems. The second and third bars represent approximately 117K trace rows and 439,957 total rows, respectively, plotted in thousands of rows.

The trace subset and total dataset sizes are reported by the dataset provider; they should not be interpreted as the number of independent decision scenarios.

B. Data Preprocessing

The raw execution logs are transformed into structured events.

The preprocessing pipeline includes:

- 1) timestamp normalization;
- 2) agent identification;
- 3) communication extraction;
- 4) decision-event extraction;
- 5) evidence association;
- 6) conflict identification; and
- 7) graph node and edge generation.

The resulting event records are transformed into graph nodes and relationships according to the EDG formulation.

C. Experimental Environment

The final implementation should report:

- operating system;
- Python version;
- machine-learning framework;
- graph-processing library;
- CPU;

- GPU, if used;
- RAM; and
- software versions.

These values must be filled using the actual experimental environment rather than estimated hardware specifications.

D. Baseline Methods

The proposed EDG framework should be compared against:

- 1) No-explanation baseline;
- 2) LIME;
- 3) SHAP;
- 4) Agent-level explanation method; and
- 5) Proposed EDG.

LIME and SHAP can be applied within individual agents where their underlying models support such explanations. They should not be treated as direct system-level MAS explanations without defining how individual-agent explanations are aggregated.

E. Evaluation Metrics

The following metrics are proposed:

- Explanation Fidelity;
- Explanation Completeness;
- Decision Traceability;
- Human Comprehensibility;
- Decision Accuracy;
- Explanation Consistency; and
- Computational Overhead.

V. DATA ANALYSIS AND VISUALIZATION

A. Exploratory Data Analysis

The analysis should examine:

- number of agents;
- number of decisions;
- number of communication events;
- number of conflicts;
- verification failures;
- graph nodes;
- graph edges;
- decision depth; and
- explanation generation time.

B. Graph Complexity

As the number of participating agents increases, the number of interactions and decision dependencies can increase the size of the resulting decision graph. Graph complexity should therefore be measured using graph nodes and edges for different numbers of participating agents.

Figure 4 presents the graph-complexity analysis derived from the available execution traces. The reported values are dataset-derived and are not treated as fabricated experimental results.

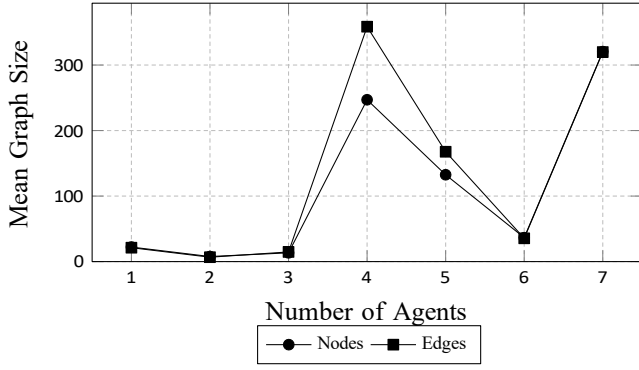


Fig. 4. Dataset-derived graph complexity versus the number of participating agents. Mean node and edge counts are calculated from the available execution traces. No value is reported for eight agents because no corresponding trace was available under the analysis procedure.

C. Statistical Summary

For every evaluation metric, mean, median, standard deviation, minimum, maximum, and sample size should be calculated.

The sample mean is:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \quad (11)$$

The sample standard deviation is:

$$s = \frac{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2}}{n - 1} \quad (12)$$

D. Feature Analysis

Potential variables affecting explanation quality include:

- number of agents;
- number of interactions;
- number of graph nodes;
- number of graph edges;
- decision depth;
- confidence;
- number of conflicts; and
- verification status.

E. Explanation Latency

In addition to explanation quality, the computational cost of constructing and presenting decision explanations should be evaluated. Explanation latency is measured as the time required to transform recorded execution events into the corresponding explanation representation.

Because latency values depend on the actual implementation and hardware environment, the following figure is provided as a measurement-ready visualization rather than as fabricated results.

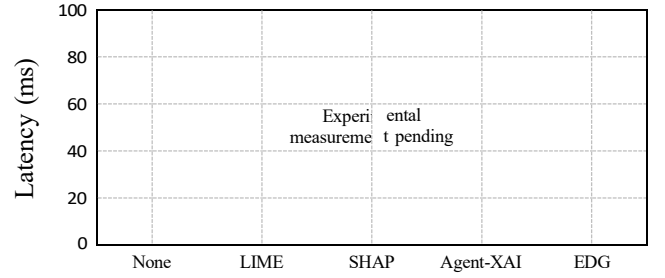


Fig. 5. Measurement-ready visualization for explanation generation latency. Numerical values are intentionally omitted because latency must be measured from the implemented methods and hardware environment.

VI. RESULTS

A. Performance Results

The final experimental evaluation should report decision accuracy, explanation fidelity, traceability, and explanation latency.

At the current research stage, numerical values are not inserted because they require actual execution of the EDG framework on the selected dataset.

TABLE II
EXPERIMENTAL PERFORMANCE COMPARISON

Method	Accuracy	Fidelity	Traceability	Latency
No Explanation	—	—	—	—
LIME	—	—	—	—
SHAP	—	—	—	—
Agent-level XAI	—	—	—	—
EDG	—	—	—	—

The dashes in Table II indicate measurements that must be obtained experimentally. They are deliberately not replaced with estimated values.

B. Decision Classification Analysis

A confusion matrix can be used only after a specific classification task has been defined and ground-truth labels have been established. For example, if the verification layer classifies a decision as valid or invalid, the resulting predictions can be compared against the ground-truth verification labels.

The matrix below is therefore a measurement-ready representation. Its values must be populated from the actual experiment.

		Predicted Valid Predicted Invalid	
Actual Class	TP Pending	FN Pending	
	FP Pending	TN Pending	

Fig. 6. Measurement-ready confusion matrix for the binary decision verification task. TP, TN, FP, and FN must be replaced by measured counts after ground-truth evaluation.

The confusion matrix should be used to calculate standard classification measures such as accuracy, precision, recall, and F1-score when a binary ground-truth verification task is defined.

C. Comparative Analysis

The primary objective of the comparison is to determine whether EDG improves system-level traceability and explanation completeness compared with individual model-level explanations.

Prediction accuracy should also be monitored to ensure that the explanation layer does not negatively affect the underlying decision system.

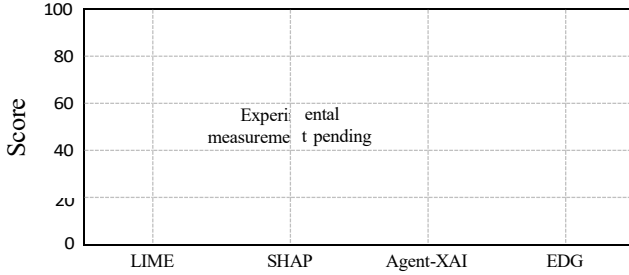


Fig. 7. Measurement-ready comparison of accuracy, explanation fidelity, and decision traceability. Numerical values are intentionally omitted until the evaluation experiment is executed.

Table II provides the corresponding numerical format for the comparative evaluation.

D. Ablation Study

An ablation study is performed to determine the contribution of individual EDG components.

Four progressively richer configurations are considered:

- 1) Agent information only;
- 2) Agent information + interactions;
- 3) Agents + interactions + evidence;
- 4) Full EDG + verification.

The resulting evaluation should determine whether interactions, evidence, and verification improve system-level explanation quality.

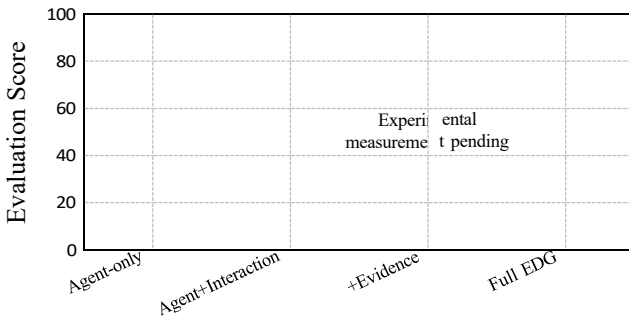


Fig. 8. Measurement-ready ablation-study visualization for progressively richer EDG configurations. Numerical values are intentionally omitted until the ablation experiment is executed.

E. Error Analysis

Errors should be categorized into:

- missing agent information;
- incorrect evidence association;
- missed communication;
- incorrect conflict identification;
- incorrect verification;
- incomplete decision path; and
- misleading natural-language explanation.

TABLE III
ERROR CATEGORIES FOR EDG EVALUATION

Error Category	Count
Missing agent information	—
Incorrect evidence association	—
Missed communication	—
Incorrect conflict identification	—
Incorrect verification	—
Incomplete decision path	—
Misleading explanation	—

The error counts must be obtained from actual evaluation traces and should not be estimated.

VII. STATISTICAL ANALYSIS

A. Descriptive Statistics

For every metric, the final study should report:

- Mean;
- Median;
- Standard deviation;
- Minimum;
- Maximum; and
- Sample size.

B. Statistical Tests

A paired *t*-test can be used when comparing two methods on the same test cases and when the required assumptions are satisfied.

When normality assumptions are not appropriate, the Wilcoxon signed-rank test can be considered.

For comparisons involving more than two methods, ANOVA or a suitable non-parametric alternative can be applied.

C. Confidence Intervals

A confidence interval for the sample mean can be represented as:

$$CI = \bar{x} \pm t_{\alpha/2} \sqrt{\frac{s}{n}} \quad (13)$$

D. Statistical Interpretation

Statistical significance should be interpreted together with effect size and practical relevance. A higher explanation score should not automatically be considered evidence of superiority unless the observed difference is statistically supported and practically meaningful.

VIII. DISCUSSION

A. Expected Findings

The proposed EDG framework is designed to provide information that is difficult to obtain from isolated model explanations.

Specifically, it represents:

- participating agents;
- supporting evidence;
- communication;
- decision influence;
- conflicts;
- verification; and
- final actions.

This creates a system-level representation of collaborative decision-making.

B. Comparison with Existing Work

LIME and SHAP provide valuable model-level explanations [1], [2]. The proposed EDG framework does not necessarily replace these methods. Instead, they can be integrated inside individual agents.

For example, SHAP can identify important features for Agent A, while the EDG can represent how Agent A's resulting decision influenced Agent B and eventually contributed to the final system action.

Therefore, EDG can operate as a higher-level explanation layer.

C. Implications

The proposed approach has potential applications in:

- healthcare decision support;
- cybersecurity;
- autonomous transportation;
- financial risk analysis;
- industrial automation; and
- intelligent decision-support systems.

In these environments, traceability can be important for debugging, auditing, compliance, and human oversight.

D. Limitations

Several limitations must be considered.

First, decision graphs can become very large as the number of agents and interactions increases.

Second, storing every interaction may increase computational and memory overhead.

Third, an excessively detailed graph may become difficult for humans to interpret.

Finally, the framework depends on accurate logging of agent events. Missing or incorrect event information can result in incomplete explanations.

IX. CONCLUSION

This paper proposed an Explainable Decision Graph framework for trustworthy autonomous multi-agent systems. Autonomous MAS can provide powerful distributed decision-making capabilities, but their collaborative nature makes it difficult to understand how final decisions are produced.

Existing XAI methods such as LIME and SHAP primarily focus on individual model predictions. The proposed EDG approach extends the explanation scope by representing agents, observations, evidence, communication, intermediate decisions, conflicts, verification events, and final actions within a unified graph.

The framework provides a foundation for system-level traceability, transparency, accountability, and human understanding.

However, empirical validation is required before claims about performance improvements or statistical significance can be made. The proposed evaluation therefore uses multi-agent execution traces, baseline comparisons, quantitative explainability metrics, and ablation experiments.

X. FUTURE WORK

Future work will focus on:

- 1) Real-time decision graph construction.
- 2) Causal decision graphs that distinguish actual influence from correlation.
- 3) Integration with Large Language Model based autonomous agents.
- 4) Formal verification of agent decisions.
- 5) Adaptive self-correction mechanisms.
- 6) Human-in-the-loop evaluation.
- 7) Large-scale multi-agent environments.
- 8) Standardized benchmarks for multi-agent explainability.
- 9) Privacy-preserving decision traces.
- 10) Hierarchical graph representations for reducing explanation complexity.

A particularly important direction is integrating Explainable Decision Graphs with the broader VeriTrust AI framework. In such an architecture, EDG can act as an evidence layer connecting runtime monitoring, formal reasoning, and adaptive self-correction.

ACKNOWLEDGMENT

The authors would like to express their sincere gratitude to Sanjivani University and the Department of Computer Science and Engineering for providing the academic environment and resources required for this research.

The authors also thank their faculty mentors and project supervisors for their guidance, feedback, and support throughout the development of this work.

REFERENCES

- [1] M. T. Ribeiro, S. Singh, and C. Guestrin, "Why Should I Trust You?: Explaining the Predictions of Any Classifier," in *Proc. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 1135–1144.

- [2] S. M. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," in *Advances in Neural Information Processing Systems*, vol. 30, 2017.
- [3] Z. C. Lipton, "The Mythos of Model Interpretability," *Queue*, vol. 16, no. 3, pp. 31–57, 2018.
- [4] F. Doshi-Velez and B. Kim, "Towards A Rigorous Science of Interpretable Machine Learning," arXiv preprint arXiv:1702.08608, 2017.
- [5] A. Barredo Arrieta *et al.*, "Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges toward Responsible AI," *Information Fusion*, vol. 58, pp. 82–115, 2020.
- [6] S. M. Lundberg *et al.*, "From local explanations to global understanding with explainable AI for trees," *Nature Machine Intelligence*, vol. 2, pp. 56–67, 2020.
- [7] T. Miller, "Explanation in Artificial Intelligence: Insights from the Social Sciences," *Artificial Intelligence*, vol. 267, pp. 1–38, 2019.
- [8] R. Guidotti *et al.*, "A Survey of Methods for Explaining Black Box Models," *ACM Computing Surveys*, vol. 51, no. 5, 2018.
- [9] T. Ma, Y. Chen, V. Anand, A. Cornacchia, A. R. Faustino, G. Liu, S. Zhang, H. Luo, S. A. Fahmy, Z. A. Qazi, and M. Canini, "MAESTRO: Multi-Agent Evaluation Suite for Testing, Reliability, and Observability," arXiv preprint arXiv:2601.00481, 2026.