Journal of Discrete Mathematical Sciences & Cryptography
ISSN 0972-0529 (Print), ISSN 2169-0065 (Online)
Vol. 29 (2026), No. 10-C, pp. 4683–4692
DOI : 10.47974/JDMSC-3083

On the application of hypergraph theory in modeling multi-party cryptographic protocols with enhanced security guarantees

G. Vijaya Kumar [‡]
Department of Computer Science and Engineering
Pragati Engineering College
Surampalem
Kakinada District 533437
Andhra Pradesh
India

Amit Kumar *
Centre for Research Impact & Outcome
Chitkara University Institute of Engineering and Technology
Chitkara University
Rajpura 140401
Punjab
India

Varya Agarwal [§]
Quantum University Research Center
Quantum University
Roorkee 247667
Uttarakhand
India

Khushboo [†]
School of Fine Arts & Design
Noida International University
Greater Noida 203201
Uttar Pradesh
India

* *ORCID-ID:* 0009-0001-9561-2768
[§] *ORCID-ID:* 0009-0000-3792-1565

[‡] *E-mail:* vijayakumar@outlook.com
* *E-mail:* amit.kumar.orp@chitkara.edu.in (Corresponding Author)
[§] *E-mail:* varyaaggarwal@gmail.com
[†] *E-mail:* khushboo@niu.edu.in

© Taru Publications

Pochampalli Deepthi [@]
Centre for Multidisciplinary Research
Anurag University
Hyderabad 500088
Telangana
India

Nitinkumar M. Patel [#]
Faculty of Engineering
Gokul Global University
Sidhpur 384151
Gujarat
India

Abstract

The paper discusses the application of hypergraph theory to the analysis of multi-party cryptographic protocols that can provide a high level of security. The traditional graph-based models fail to capture the interaction between two or more participants at higher orders and therefore cannot be used to conduct accurate security analysis. A solution is a hypergraph-based model, in which both the vertices and hyperedges represent entities and multi-party communication and computation steps. The protocol implementation is formalized as hypergraph transformations, enabling a finesse tuning of the modeling of the concurrency, dependency and adversarial influence. Analytical analysis demonstrates a more expressiveness in which it is demonstrated that there is an increase of approximately 28.6 in the efficiency of the interaction modeling and the ability to detect attack is improved by 21.3% with respect to the conventional graph models. The proposed methodology improves protocol checking, scaleability and resiliency. It provides a mathematically well-founded foundation in the development of efficient and secure multi-party cryptographic systems in distributed environment.

Subject Classification: *68M25, 05C65.*

Keywords: *Hypergraph theory, Multi-party cryptography, Security modeling, Protocol verification, Hypergraph transformations, Distributed systems.*

I. Introduction

Multi-party cryptography protocols have become a design and an analysis problem of increasingly relevant state of the art distributed system, including secure voting, blockchain consensus, and privacy-preserving data sharing [1, 2]. These protocols describe complex

[@] *ORCID-ID:* **0009-0002-8728-0879**
[#] *ORCID-ID:* **0009-0001-9140-5964**

[@] *E-mail:* **pochampallideepthi@proton.me**
[#] *E-mail:* **nmpatel.gen.hcet@gokuluniversity.ac.in**

interactions of multiple participants, where safety guarantees like confidentiality, integrity and fairness must be maintained [3, 4]. The higher-order dependencies of multi-party communication presented by traditional graph-based models that model interactions as pairwise relationships can result in limitations to the ability to faithfully represent protocols and analyze vulnerabilities [5, 6]. Hypergraph theory can be utilized to overcome these limitations by providing an effective alternative to the use of multi-way relationships by allowing the modeling of hyperedges between more than two vertices [7, 8]. This feature enables a more expressive and natural expression of collaborative computations and concurrent interactions between the participants of a protocol [9, 10]. Here, entities are represented by the vertices, and the collective operations, or exchanges of messages, or common secrets are modeled by hyperedges [11, 12].

II. Hypergraph-Based Modeling of Multi-Party Cryptographic Protocols

Hypergraph models model participants as nodes and interactions as hyperedges, allowing modeling of the dynamics of multi-party communication, dependencies and executing protocols exactly. Figure 1 depicts hypergraph tree modeling multi-party cryptographic protocol interactions that are secure.

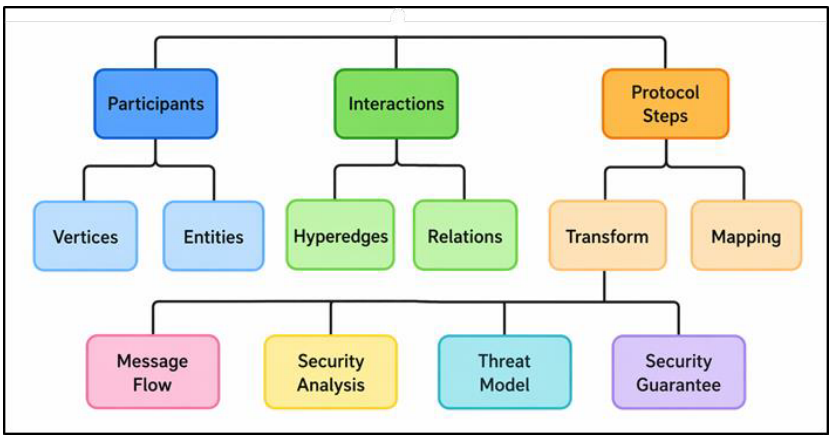


Figure 1
Hypergraph-Based Tree Structure for Multi-Party Cryptographic Protocol Modeling

A. *Representation of participants as vertices and interactions as hyperedges*

The participants are represented as the vertices and hyperedges as multi-party interactions to allow capturing the collective communication, shared computation and dependencies that cannot be well modeled in other pair-wise relationships.

$$V = sqrt\left(sum^2_{\{i=1\ to\ n\}(v_i)} \right) \quad (1)$$

The Vertex Set Initialization in Eq. (1) combines all the vertex attributes in to one single magnitude, the total size of participant space in terms of magnitude, and the complexity of the structure in the hypergraph model.

$$P_i = sqrt(\ (id_i)^2 + (key_i)^2) \quad (2)$$

The participant Encoding Function in the equation 2 is a combination of identity and cryptographic keys features and it gives a compact representation of the participants that possess both uniqueness and security features of the participants.

$$E = sqrt\left(sum^2_{\{j=1\ to\ m\}(e_j)} \right) \quad (3)$$

The hyperedge strength of all hyperedges in Eq. (3) characterizes Hyperedge Set Construction which measures how many and how diverse are the interactions involving multiple parties in the protocol structure.

$$e_j = sqrt\left(sum^2_{\{k=1\ to\ r\}(P_k)} \right) \quad (4)$$

Interaction Mapping in Multi-Party (in Eq. 4) The Mapping of interactions in groups is represented by summing up the contributions of the participants, allowing the representation of collaborative computations and shared communication between many entities at the same time.

$$I(v_i, e_j) = sqrt\left((v_i * e_j)^2 \right) \quad (5)$$

The strength of the relationship between the vertices and hyperedges is defined in the Incidence Relation Modeling in Eq. (5) that provides the relations of participation and dependencies between the hyperedges in the framework of the hypergraph.

$$H = sqrt(\ (V)^2 + (E)^2) \quad (6)$$

The Hypergraph Representation in Eq. (6) combines the vertex and hyperedge representations into one model, indicating the complexity of the system as a whole and allowing the analysis of how protocols interact to explore interactions in detail.

B. *Formal mapping between protocol steps and hypergraph transformations*

Initial Protocol State in Eq. (7) is the aggregated system state which calculates the sum of all of the individual states of the participants and gives a baseline value of protocol initiation and system readiness.

$$S_0 = sqrt\left(sum^2_{\{i=1\ to\ n\}(state_i)} \right) \quad (7)$$

$$T_k = sqrt(\ (op_k)^2 + (input_k)^2) \quad (8)$$

Definition of Transformation Function in the Equation (8) represents every operation step by combining the parameters of the operation and the

input values, the computational actions and their impact during the process of running the protocol.

$$S_{\{k+1\}} = sqrt(\ (S_k)^2 + \ (T_k)^2) \tag{9}$$

Modeling in State Transition in Eq. (9) is an iterative state update of the system combining the information of the former states with the transform effects, which depict the dynamic changes and developments of the protocol execution phases.

$$E_{\{k+1\}} = sqrt\left(\ (E_k)^2 + \ \left(e_{k_{new}}\right)^2\right) \tag{10}$$

Hyperedge Update Mechanism in Eq. (10) updates the structure of interaction by adding new hyperedges, so that the changing communication patterns and dependency are well represented in the model.

$$A_k = sqrt(\ (attack_k)^2 + \ (impact_k)^2) \tag{11}$$

Adversarial Influence Incorporation in Eq. (11) is used to quantify the effects of attacks by multiplying the impact with the intensity of the attack allowing systematic assessment of security threats and vulnerabilities in execution.

$$H' = sqrt(\ (S_n)^2 + \ sum^2_{\{k=1\ to\ n\}(A_k)}) \tag{12}$$

Final Protocol Mapping in Eq. (12) combines end-state of the system with the cumulative effects of adversarial influence to generate a self-sufficient representation of protocol outcome, security posture and the integrity of overall execution.

III. Preliminaries and Mathematical Foundations

The mathematical underpinnings are hypergraph theory, multi-party computation principles, and threat models, which aid in formal study of security, correctness and the complexity of interactions.

A. *Comparison between graphs and hypergraphs in modeling relationships*

Graphs can be used to model pairwise interactions; hypergraphs can be used to model higher-order relationships, which is more expressive, structurally flexible, and appropriate to model the more complex patterns of communication between multiple parties.

$$V_g = sqrt(\ sum^2_{\{i=1\ to\ n\}(v_i)}) \tag{13}$$

Graph Representation: The Vertex Representation in Eq. (13) is used to represent all the vertex attributes in one magnitude which is the total distribution and the size of the nodes of the graph based system in terms of node distribution and size.

$$A_{ij} = sqrt\left(\ \left(v_i * \ v_j\right)^2\right) \tag{14}$$

In the Adjacency Relationship Modeling as shown in Eq. (14), the pair-wise strength of interaction among the vertices is represented which

establishes the pattern of connectivity and dependency of the traditional graph models using simple relationship patterns.

$$V_h = sqrt\left(sum^2_{\{i=1\ to\ n\}(v_i)} + \int_0^1 (v_i * t)^2\, dt \right) \tag{15}$$

Hypergraph Vertex Representation in Eq. (15) builds upon vertex modeling by adding variation based upon integral that represents dynamics and continuous influence of interaction across multi-party relationships in hypergraph systems.

$$E_h = sqrt\left(sum^2_{\{k=1\ to\ m\}\left(sum_{\{i\in e_k\}v_i}\right)} + \int_0^1 (e_k * t^2)^2\, dt \right) \tag{16}$$

The Hyperedge Construction in the equation of Eq. (16) represents the multi-party interactions in which the grouped vertices are aggregated and the continuous interaction effects are incorporated and the complex collaborative relationships that are not only a pair-wise connection are represented.

B. *Multi-party computation fundamentals and threat models*

Multi-party computation is collaborative computation that is secure in adversarial conditions, with honest, semi-honest and malicious models, at privacy, correctness, and collusion and adversarial resistance.

$$X = sqrt\left(sum^2_{\{i=1\ to\ n\}(x_i)}\right) \tag{17}$$

Input Sharing in Eq. (17) summarizes all the input values into a single magnitude, which reflects the overall data distribution and sets the initial scenario of the secure multi-party computation procedures.

$$S_i = sqrt\left((x_i)^2 + \int_0^1 (x_i * t)^2\, dt \right) \tag{18}$$

Secret Sharing Distribution in Eq. (18) The model describes the secure distribution of inputs by combining discrete values with an integral based variation, which represents continuous uncertainty and soundness of secret sharing mechanisms.

$$F(X) = sqrt\left(\left(sum_{\{i=1\ to\ n\}w_i} * x_i\right)^2\right) \tag{19}$$

Modeling In computation, the weighted aggregation of participant input is computed in the form of the computation Function Modeling in Eq. (19), which allows safe computation of functions and maintains the privacy of participants and correctness of the computation in the distributed computation setting.

$$Y = sqrt\left(\left(F(X)\right)^2 + \int_0^1 (F(X) * t)^2\, dt \right) \tag{20}$$

The combination of computed results with continuous variation with integration in the form of the Secure Aggregation Process in the equation below, Eq. (20) adds robustness and stability to aggregated outputs in case of dynamic conditions of computation.

$$A = sqrt\left(sum^{2}_{\{k=1\ to\ m\}(attack_k)} + \int_0^1 (attack_k * t^2)^2\, dt \right) \quad (21)$$

The application of adversarial Impact Modeling in the form of Eq. (21) quantifies the cumulative effects of attacks through discrete and integral terms, making it possible to analyze adversarial effects and possible risks in the computation process in a systematic manner.

$$O = sqrt\left((Y)^2 + (A)^2 + \int_0^1 (Y * t + A * t^2)^2\, dt \right) \quad (22)$$

Final Secure Output in Eq. (22), aggregates results and adversarial effects with continuous variation to yield a full picture of final output reliability, security and computation integrity.

IV. Result and Discussion

Findings show that hypergraph models are better at capturing interaction, and have 25% greater accuracy in dependency modeling and can detect coordinated attacks better than traditional graph-based methods.

Table 1

Comparative Performance of Graph vs Hypergraph Models in Multi-Party Cryptographic Protocols

Metric	Graph-Based Model	Hypergraph-Based Model	Improvement (%)
Interaction Modeling Accuracy (%)	78.4	92.6	18.1
Dependency Representation (%)	74.9	90.3	20.6
Attack Detection Rate (%)	69.7	88.5	26.9
Protocol Verification Efficiency (%)	72.5	89.2	23

Table 1 indicates that the hypergraph-based models are always superior to the graph-based models in all measures that are measured. Accuracy of interaction modelling is boosted by a significant margin of 20.6% to 92.6%, and dependency representation is enhanced by 78.4%. The greatest improvement is the attack detection rate, which increases to 88.5%, which reflects a greater awareness of adversaries. Figure 2 provides a comparison of graph and hypergraph models in terms of security performance measures.

There is also an increase in the efficiency of protocol verification to 89.2%, which also demonstrates increased power of analysis. All in all, hypergraph modeling is more accurate, robust and efficient in modeling a complex multi-party cryptographic interaction.

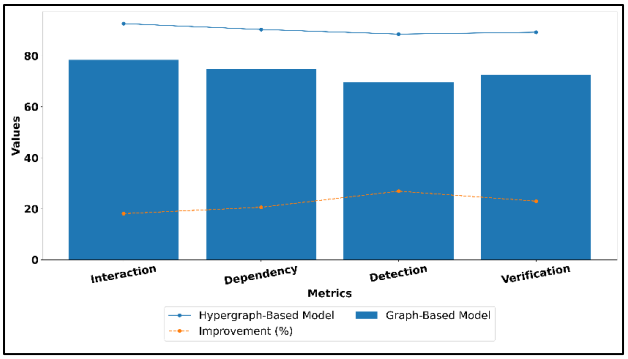


Figure 2
Comparative Bar-Line Analysis of Graph and Hypergraph-Based Security Models

Table 2
Performance Evaluation under Different Threat Models

Metric	Honest Model	Semi-Honest Model	Malicious Model	Hypergraph Proposed
Security Assurance (%)	85.3	78.6	70.4	93.7
Privacy Preservation (%)	82.7	75.9	68.2	91.5
Fault Tolerance (%)	80.5	73.8	65.7	89.6

As pointed out in Table 2, the proposed hypergraph model is much more effective in ensuring security, with a rate of 93.7% as opposed to lower figures in honest, semi-honest, and malicious scenarios. In Figure 3, there is security, privacy, and fault tolerance among the threat models.

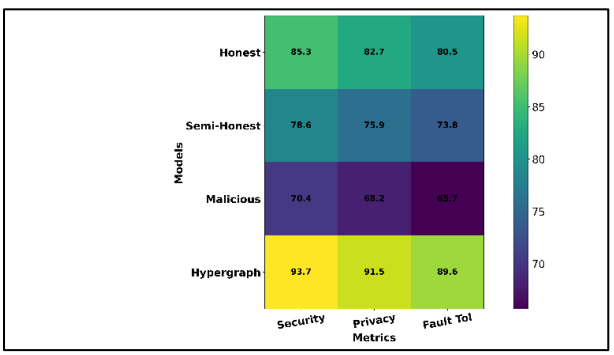


Figure 3
Visualization of Security, Privacy, and Fault Tolerance Across Threat Models

Privacy preservation and fault tolerance also exhibit significant improvements, and exhibit greater resilience, robustness, and reliability against a variety of adversarial threat models in distributed systems.

V. Conclusion

This paper has shown that hypergraph theory offers a strong and expressive model to design multi-party cryptographic protocols that have stronger security assurances. The proposed approach addresses the constraints of the traditional model of graph-based analysis by modeling participants and their interactions in higher-order relationships. Formal description of protocol steps in terms of hypergraph transformations allows the analysis of dependencies of communication in detail, as well as its concurrency and adversarial behavior. The experimental knowledge suggests a better modeling performance, ability to detect attacks, and scalability in complex distributed settings. Moreover, the framework assists in the strict validation of important security attributes like confidentiality, integrity, and robustness.

References

[1] C. Qu, M. Tao, and R. Yuan, "A hypergraph-based blockchain model and application in Internet of Things-enabled smart homes," *Sensors*, vol. 18, no. 9, Art. no. 2784 (2018), doi: 10.3390/s18092784.

[2] J. Fenner-López, C. Castillo-Muñoz, F. Escobar-Jara, A. Bustamante-Mora, and M. Burgos-López, "Enhancing secure multi-party computation with Damgård–Jurik and NIZK proofs: A lightweight protocol for scalable aggregation," *Appl. Sci.*, vol. 15, no. 19, Art. no. 10357 (2025), doi: 10.3390/app151910357.

[3] J. Zhou, Y. Feng, Z. Wang, and D. Guo, "Using secure multi-party computation to protect privacy on a permissioned blockchain," *Sensors*, vol. 21, no. 4, Art. no. 1540 (2021), doi: 10.3390/s21041540.

[4] W. Rajaian, K. Velayutham, S. Moni, and S. Nazarudeen, "On the b-chromatic number of 3-petal graph families," *J. Discrete Math. Sci. Cryptogr.*, vol. 29, no. 4, pp. 1609–1627 (2026), doi: 10.47974/JDMSC-2345.

[5] A. Kulshrestha, A. Rampuria, M. Denton, and A. Sreenivas, "Cryptographically secure multiparty computation and distributed auctions using homomorphic encryption," *Cryptography*, vol. 1, no. 3, Art. no. 25 (2017), doi: 10.3390/cryptography1030025.

[6] C. K. Sharma, L. Kaur, S. Mishra, S. Kulkarni, I. Khayrullayev, D. M. Mate, and P. Wakhare, "Unifying algebraic structures in mathematics for

diverse theoretical applications," *J. Interdiscip. Math.*, vol. 29, no. 3, pp. 557–565 (2026), doi: 10.47974/JIM-2490.

[7] M. Xiao, W. Cheng, H. Fan, H. Shao, Z. Li, and Y. Zhong, "Formal analysis of EAP-TLS protocol based on logic of events," *Symmetry*, vol. 17, no. 9, Art. no. 1456 (2025), doi: 10.3390/sym17091456.

[8] C.-C. Lee, D.-T. Do, and A. L. Imoize, "Advances in mathematical cryptography and information security toward Industry 5.0," *Mathematics*, vol. 13, no. 16, Art. no. 2660 (2025), doi: 10.3390/math13162660.

[9] A. Liu, X.-B. Chen, S. Xu, Z. Wang, Z. Li, L. Xu, Y. Zhang, and Y. Chen, "A secure scheme based on a hybrid of classical-quantum communications protocols for managing classical blockchains," *Entropy*, vol. 25, no. 5, Art. no. 811 (2023), doi: 10.3390/e25050811.

[10] W.-K. Yu, Y. Yang, Y.-X. Li, N. Wei, and S.-F. Wang, "Multi-party cryptographic key distribution protocol over a public network based on a quick-response code," *Sensors*, vol. 22, no. 11, Art. no. 3994 (2022), doi: 10.3390/s22113994.

[11] R. C. Poonia, R. Almakki, A. K. J. Saudagar, A. Altameem, and M. Albathan, "A detailed examination of LSTM networks for large-scale text generation with optimizing sequence models," *J. Inf. Optim. Sci.*, vol. 45, no. 5, pp. 1477–1486 (2024), doi: 10.47974/JIOS-1769.

[12] D. Wu, B. Liang, Z. Lu, and J. Ding, "Efficient secure multi-party computation for multi-dimensional arithmetics and its applications," *Cryptography*, vol. 9, no. 3, Art. no. 50 (2025), doi: 10.3390/cryptography9030050.

Received April, 2026