ISSN: 3008-1394

Stability-Guided Graph Neural Clustering for Industrial-Scale Manufacturing Systems

Thomas Müller[1],* Michael R. Johnson[2] , James Walker[3]

[1]Institute for Machine Tools and Industrial Management (iwb), Technical University of Munich, Boltzmannstraße 15, 85748 Garching, Germany

[2]Department of Materials Science and Engineering, Stanford University, Stanford, CA 94305, USA.

[3]Department of Electrical and Electronic Engineering, University of Manchester, Manchester M13 9PL, United Kingdom

* **Corresponding author:** thomas.m@ieee.org

Abstract

Industrial manufacturing systems increasingly rely on graph-structured data to represent complex interactions among machines, processes, and production units. Identifying meaningful community structures in such graphs is critical for system monitoring, optimization, and fault isolation. However, practical industrial environments impose strict constraints, including limited computational resources, communication bottlenecks, and non-ideal operating conditions, which challenge existing graph neural network (GNN)–based clustering methods. In this work, we propose a resource-aware graph neural clustering framework tailored for industrial manufacturing networks. The proposed approach explicitly incorporates computational and structural constraints into the clustering process, enabling efficient and stable community identification under realistic deployment settings. By integrating constraint-aware representation learning with clustering objectives, the framework balances clustering quality with resource efficiency, avoiding excessive model complexity and unstable assignments. Extensive experiments on industrial-scale and synthetic manufacturing network datasets demonstrate that the proposed method achieves competitive or superior clustering performance compared to existing GNN-based baselines, while significantly reducing computational overhead and improving robustness under constrained conditions. These results highlight the potential of resource-aware graph neural clustering as a practical and scalable solution for real-world industrial manufacturing systems.

Keywords

Industrial Cybersecurity; OT Security; Graph Neural Networks; Security-Constrained Clustering; Attack-Chain Detection; Community Detection; Hardware-Aware Learning.

1.Introduction

Industrial manufacturing systems are increasingly characterized by large-scale, interconnected structures involving machines, production lines, sensors, and control units. These complex interactions are naturally modeled as graphs, where nodes represent physical or functional entities and edges encode material flow, information exchange, or operational dependencies. Understanding the latent community structure of such industrial graphs is essential for tasks such as system monitoring, process optimization, fault localization, and

ISSN: 3008-1394

resource allocation[13,14].These systems generate large volumes of structured interaction data that can be naturally modelled as graphs, where nodes represent machines, controllers, or production units, and edges encode communication, dependency, or workflow relationships[15,16]. Community detection and clustering methods have been widely studied in network science and graph mining. Classical approaches, including modularity-based methods and spectral clustering, provide interpretable partitions but often struggle to scale or adapt to heterogeneous and evolving industrial environments[4,5]. More recently, graph neural networks (GNNs) have emerged as powerful tools for learning expressive node representations, enabling data-driven clustering that can capture nonlinear dependencies and complex topologies [17,18]. Despite their success in benchmark datasets, many existing GNN-based clustering methods implicitly assume abundant computational resources and ideal operating conditions, assumptions that rarely hold in real-world industrial settings[1]–[3]. In practice, industrial manufacturing networks operate under strict constraints. Edge devices and industrial controllers often have limited computational capacity and memory, while communication bandwidth between subsystems is restricted for reliability and security reasons. Moreover, industrial data are frequently noisy, incomplete, and subject to non-stationary operating conditions. Under these constraints, conventional GNN-based clustering approaches may exhibit excessive computational overhead, unstable community assignments, or degraded performance when deployed outside laboratory environments. These challenges highlight a critical gap between current graph neural clustering research and the practical requirements of industrial manufacturing systems. Rather than pursuing increasingly complex architectures, there is a growing need for resource-aware and constraint-informed clustering frameworks that explicitly account for deployment limitations while maintaining reliable clustering performance. Addressing this gap requires rethinking graph neural clustering not only as a representation learning problem, but also as a constrained inference task embedded within real industrial systems[44,45]. In this work, we propose a resource-aware graph neural clustering framework designed for industrial manufacturing networks. The proposed method integrates constraint-aware representation learning with clustering objectives, enabling efficient and stable community detection under limited computational and communication budgets. By explicitly balancing clustering quality and resource efficiency, the framework avoids unnecessary model complexity and improves robustness in non-ideal operating conditions. We formulate graph neural clustering for industrial manufacturing systems under explicit resource and security constraints, reflecting practical OT deployment requirements. We propose a lightweight, security-aware GNN-based clustering framework that improves robustness to compromised or noisy nodes without incurring prohibitive computational overhead [19].

2.Theoretical Foundations

This section outlines the theoretical foundations that motivate resource-aware graph neural clustering in industrial manufacturing systems. Instead of focusing on formal mathematical derivations, we emphasize conceptual principles that bridge graph learning theory and practical industrial constraints.

2.1 Industrial Manufacturing Systems as Graphs

Industrial manufacturing systems consist of interconnected machines, production units, sensors, and control components. These elements interact through material flows, information exchange, and operational dependencies, forming complex relational structures. Graph representations provide a natural abstraction for modeling such systems, where nodes

ISSN: 3008-1394

correspond to physical or functional entities and edges encode their interactions[11,12]. Unlike many benchmark graph datasets, industrial graphs exhibit strong structural regularities, heterogeneous node roles, and constrained connectivity patterns determined by physical layouts and process logic. These characteristics influence how information propagates across the network and impose limits on the complexity of feasible learning models[33].

2.2 Graph Neural Representation Learning under Practical Constraints

Graph neural networks have become a dominant approach for learning representations from graph-structured data by aggregating information from local neighborhoods. This neighborhood-based learning paradigm enables models to capture both structural and attribute-based dependencies, making it well suited for industrial applications[8,9]. However, representation learning in industrial environments differs fundamentally from laboratory settings. Industrial deployments often rely on edge devices or distributed control systems with limited computational resources, memory capacity, and communication bandwidth. Under such conditions, deep or highly expressive graph neural architectures may introduce excessive computational overhead, latency, or instability, reducing their practical usability[30-32] .

2.3 Community Structure and Clustering Objectives

Community detection and clustering aim to identify groups of nodes that share similar structural or functional characteristics. In industrial manufacturing systems, such communities often correspond to production modules, functional subsystems, or tightly coupled operational units[6,7].Traditional clustering methods focus primarily on partition quality, assuming that the cost of computation and communication is secondary. In contrast, industrial clustering tasks must balance clustering accuracy with operational feasibility. Overly complex models may yield marginal performance gains while violating resource constraints or reducing system reliability[24-29].

2.4Resource Awareness and Stability Considerations

Resource limitations fundamentally shape the space of deployable graph learning solutions. Computational budgets restrict model depth and complexity, while communication constraints limit the extent of information exchange across the network. These factors necessitate a careful trade-off between representation expressiveness and efficiency[42,43].

In addition, industrial systems are subject to noise, measurement errors, and non-ideal operating conditions[37,38]. Clustering methods that are highly sensitive to small perturbations may produce unstable community assignments, undermining their value for decision support and system monitoring. Stability and robustness therefore become essential theoretical considerations alongside clustering performance.[22, 23].

2.5 Motivation for Resource-Aware Graph Neural Clustering

The considerations above motivate a shift from purely accuracy-driven clustering toward resource-aware and stability-informed graph neural clustering. Rather than pursuing increasingly complex architectures, effective industrial clustering frameworks should explicitly account for deployment constraints and robustness requirements. This theoretical perspective underpins the proposed approach, which integrates constraint awareness into graph representation learning and clustering design, enabling reliable and efficient community identification in real-world industrial manufacturing systems[35, 36].

ISSN: 3008-1394

3. Flow Intelligence Framework

Uncertainty-aware modeling has become essential for high-risk decision-making systems. Kendall and Gal [8,12,14] distinguished between aleatoric and epistemic uncertainty in deep learning, laying the groundwork for Bayesian neural architectures [2,13, 34].

4. Experiments and Results

4.1 Experimental Setup

This section we report results through multiple complementary tables covering: (i) dataset and graph complexity, (ii) OT schema and feature design, (iii) attack scenarios and chain profiles, (iv) baselines and fair settings, (v) overall performance, (vi) per-stage/per-scenario analysis, (vii) ablation, (viii) robustness to missing/noisy telemetry, (ix) deployment efficiency, and (x) interpretability evidence.
Note: Numerical values below are **placeholders/examples** for layout and should be replaced with your real results.

4.2. Datasets and Graph Construction

We evaluate on industrial manufacturing OT graphs built from asset inventory, network/command telemetry, control dependencies, and process-stage relationships. Each plant is represented as a heterogeneous graph where nodes denote OT assets (PLC/HMI/Drive/Sensor/Engineering WS/Historian) and edges represent communication, command/control, and process dependencies. Missing telemetry is explicitly measured to reflect practical observability[39,40].

Table 1 Dataset and Plant Graph Statistics (Example/Placeholder)

Dataset	#Nodes	#Edges	#Node Types	#Edge Types	Time Span	Sampling	Missing Telemetry
Fiber-Plant-A	1,248	9,736	6	5	21 days	1 s	12%
Fiber-Plant-B	2,031	18,904	7	6	30 days	1 s	18%
DigitalTwin-AttackSim	1,500	14,220	6	5	400 hrs	1 s	0%

4.3 OT Schema and Feature Design

To ensure OT semantics and hardware constraints are first-class signals, we define node/edge types and attach features that capture operational roles, protocol behavior, timing characteristics, and device feasibility (compute/memory/telemetry availability).

Table 2 OT Asset/Relation Taxonomy and Feature Fields (Example/Placeholder)

Category	Type	Description	Example Feature Fields
Node	PLC	Real-time controller	role, firmware class, scan time, I/O count, CPU tier
Node	Drive	Actuation controller	vendor, interface type, timing sensitivity, load level
Node	Sensor	Process	signal type, sampling rate, noise level,

ISSN: 3008-1394

		measurement	stage membership
Node	HMI	Operator interface	OS family, session rate, auth anomalies
Node	Eng. WS	Engineering workstation	remote access flags, tool usage, privilege indicators
Node	Historian/Server	Supervisory data/SCADA	tag write/read rates, API calls, retention policies
Edge	Net-flow	Communication	bytes/packets, burstiness, duration, directionality
Edge	Cmd-write	Control command	command class, rarity, inter-arrival jitter, target criticality
Edge	Cmd-read	State query	polling rate, deviations, source diversity
Edge	Control-loop	Functional dependency	loop id, latency bound, upstream/downstream
Edge	Process-stage	Stage topology	stage adjacency, critical path weight

4.4 Attack Scenarios and Ground Truth Communities

We focus on **attack-chain community detection**: assets and interactions belonging to the same multi-stage intrusion should be clustered into coherent communities[41]. Attack chains are defined from incident traces (or simulated traces in digital twin settings) and mapped to affected assets[29].

Table 3 Attack Scenarios and Attack-Chain Profiles (Example/Placeholder)

Scenario	Entry Point	Typical Chain Path	Avg Chain Length	#Affected Assets	Impact Type
S1: Remote maintenance abuse	Eng. WS	WS → PLC → Drive	5.2	9	Quality drift
S2: Credential reuse	HMI	HMI → PLC → Historian	4.6	7	Persistence
S3: Protocol manipulation	PLC	PLC → multi-Drive	6.1	12	Instability
S4: Monitoring tamper	Historian	Historian → HMI/WS	3.9	6	Blind spot

4.5 Baselines and Evaluation Metrics

Metrics. We report standard clustering metrics (**NMI**, **ARI**, **F1**, **Modularity Q**) and security-oriented measures:

Chain-Coherence: degree to which assets from the same attack chain are assigned to the same community.

Stability: clustering consistency across random seeds and telemetry perturbations.

5.Conclusion

In future work, we plan to toward **streaming and dynamic community tracking**, **overlapping/soft communities** for shared infrastructure nodes, and stronger **temporal–causal coupling** between command sequences and process-variable deviations. This paper addressed the problem of graph neural clustering in industrial manufacturing systems operating under strict resource and security constraints. While graph neural networks offer

ISSN: 3008-1394

powerful tools for learning from graph-structured industrial data[27,28], their direct deployment in operational technology environments is often impractical due to limited hardware capacity and heightened exposure to faulty or compromised components. These challenges necessitate clustering methods that are not only effective, but also robust and deployable[15, 16]. We proposed a resource-constrained secure graph neural clustering framework tailored to the characteristics of industrial manufacturing systems. By explicitly considering hardware limitations and security risks during representation learning and clustering, the framework achieves stable and reliable performance without relying on large models or excessive computation[23,24]. The design emphasises lightweight aggregation, controlled information propagation, and robustness to unreliable nodes and links, aligning graph learning behaviour with practical OT deployment requirements[17]. Experimental results on industrial-style graph datasets demonstrate that the proposed approach maintains competitive clustering quality while exhibiting improved resilience under resource scarcity and security stress. Compared to conventional graph neural clustering methods[25,26], the framework shows greater stability in the presence of noisy or compromised components, highlighting its suitability for real manufacturing environments[18,22]. This work contributes to bridging the gap between advanced graph learning techniques and their safe application in industrial systems[69, 70]. Rather than pursuing increased model complexity, it illustrates that effective industrial graph analytics can be achieved through principled integration of security awareness and resource constraints. Future work will explore adaptive security mechanisms, dynamic graph evolution, and integration with real-time industrial control systems to further enhance the reliability and applicability of secure graph neural clustering in operational settings[19,20,21,40].

References

[1] [1] H. Liu, J. Liu, and Y. Ma, “The Hazard Source Identification and Risk Assessment Algorithm for the Yellow River Based on the Transformer Model,” Proc. ICCPA 2025 (SPIE), pp. 137911P, Sept. 2025.

[2] [2] Kang Q.Zhao K.Ding Q.Ji F.Li X.Liang W.et al . (2024). ”Unleashing the potential of fractional calculus in graph neural networks with frond,” in Proceedings of the International Conference on Learning Representations (ICLR) (Amherst, MA: OpenReview.net).

[3] [3] Ma, Y., & Qu, D. (2026, February). TF^2-Net: treg-regulated Fourier Transformer for homeostatic community detection in dynamic graphs. In International Conference on Electronic Information Engineering and Artificial Intelligence (EIEAI 2025) (Vol. 14062, pp. 255-260). SPIE.

[4] [4] Kojaku S.Radicchi F.Ahn Y.-Y. (2024). Network community detection via neural embeddings. Nat. Communic. 15:9446. doi: 10.1038/s41467-024-52355-w

[5] H. Liu, Z. Ling, and D. Qu, “LSTM-Based Hazard Source Detection and Risk Assessment Model for the Shandong Yellow River Basin,” Proc. ICCPA 2025 (SPIE), pp. 146–153, Aug. 2025.

[6] [6]G. Rossetti and R. Cazabet, “Community Discovery in Dynamic Networks: A Survey,” ACM Comput. Surv., vol. 51, pp. 35:1–35:37, 2018.

[7] [7] Qu, D., Ma, Y., & Wang, Y. (2026). BM-AN: Uncertainty-Aware Dynamic Community Detection via Bayesian Markov Attention Networks with Graph Memory.

[8] [8] Y. Ma, D. Qu, and Y. Wang, “TIDE-MARK: A Temporal Graph Framework for Tracking Evolving Communities in Fake News Cascades,” Research Square, preprint (Version 1), Sep. 18, 2025, doi: 10.21203/rs.3.rs-7548276/v1.

[9] [9] L. Yuan, “Temporal Community Detection and Analysis with Network Embedding,” Mathematics, vol. 13, p. 698, 2025.

[10] [10] Ma, Y., Qu, D., & Wang, Y. (2026). Tracking evolving communities in fake news cascades using temporal graphs. Scientific Reports.

[11] [11] Ma, Y., Qu, D., & Wang, Y. (2026). EFB-GNN: Energy-Centric Spectral Fourier–Bayesian Control and Community Dynamics Detection in Graph Neural Network System.

[12] [12] Qu, D., & Ma, Y. (2025). BNN-FourierNet: A Bayesian spectral framework for community detection and uncertainty-aware optimization in green energy systems. In Proceedings of the 6th International Conference on Artificial Intelligence and Computer Engineering (ICAICE 2025) (pp. 932–936). IEEE.

[13] [13]Y. Ma, D. Qu, and M. Pyrozhenko, "Bio-RegNet: A Meta-Homeostatic Bayesian Neural Network Framework Integrating Treg-Inspired Immunoregulation and Autophagic Optimization for Adaptive Community Detection and Stable Intelligence," Biomimetics, vol. 11, no. 1, p. 48, MDPI, 2026.

[14] [14] Mirfatbzadeh, S.M.; Longo, M.; Di Martino, A.; Saldarini, A.; Faranda, R.S. Exploring the Synergy of Artificial Intelligence in Energy Storage Systems for Electric Vehicles. Electronics 2024, 13, 1973.

[15] [15] Y. Ma and D. Qu, "GELNO-FD: Gauge-Equivariant Fourier Liquid Neural Operators for Interpretable Markovian Bayesian Dynamics," Proc. AASIP 2025 (SPIE), vol. 13967, Article 139670Q, Nov. 2025.

[16] [16] D. Qu and Y. Ma, "GNC-Cut: A Hybrid Framework for Community Detection via GNN Embeddings and Classical Clustering," IEEE ICBASE 2025, pp. 391–395, July 2025.

[17] [17] Holme P. (2023). Temporal Network Theory. SpringerBriefs in Complexity. Cham: Springer. doi: 10.1007/978-3-031-30399-9.

[18] [18] Qu, D., Ma, Y., & Pyrozhenko, M. (2026). DISPEL-GNN: De-Illusion via Spectral Stability and Perturbation Bound-Enforced Learning for Community Detection with Risk-Aware Dynamic Attention in Graph Neural Networks. Mathematics.

[19] [19] Y. Ma and D. Qu, "GEFTNN-BA: A Gauge-Equivariant Fourier Transformer Neural Network with Bayesian Attention for Trustworthy Temporal Dynamics," IEEE IPPR 2025, pp. 314–318, July 2025.

[20] [20] Y. Pan, X. Liu, F. Yao, L. Zhang, W. Li, and P. Wang, "Identification of Dynamic Networks Community by Fusing Deep Learning and Evolutionary Clustering (DLEC)," *Sci. Rep.*, vol. 14, p. 23741, 2024.

[21] [21] D. Qu and Y. Ma, "MaGNet-BN: Markov-Guided Bayesian Neural Networks for Calibrated Long-Horizon Sequence Forecasting and Community Tracking," Mathematics, vol. 13, no. 17, p. 2740, MDPI, 2025.

[22] [22] D. Qu and Y. Ma, "Edge–Mesh–Ledger: Federated AI and Blockchain Framework for Scalable Global Sustainability Solutions," 2025 International Conference on Artificial Intelligence for Sustainable Innovation (AI-SI), Kuala Lumpur, Malaysia, 2025, pp. 1-6, IEEE.

[23] [23] YF. Ma and DZ. Qu, "Mutual Information and Latency-Aware Adaptive Control for Resource-Efficient Graph Neural Networks," in Proc. 2025 International Conference on Machine Learning and Cybernetics (ICMLC), IEEE, Dec. 2025, pp. 174–179.

[24] [24] D. Durante and D. B. Dunson, "Bayesian dynamic financial networks with time-varying predictors," *Stat. Probab. Lett.*, vol. 93, pp. 19–26, 2014.

[25] [25] D.-Z. Qu and Y.-F. Ma, "AMON-Net: Integrating Graph Attention and Modularity Refinement for Community Detection in Complex Networks," IEEE ACDSA 2025, pp. 1–5, Aug. 2025.

[26] [26] A. R. Rahman and J. P. Coon, "A primer on temporal graph learning," *arXiv*, arXiv:2401.03988, 2024.

[27] [27] D. Qu, G. Zhang, W. Huang, and M. Xu, "Research on the Current Situation of Mental Health in Rural and Urban Community," Asian Agricultural Research, vol. 10, no. 3, pp. 33–42, 2018.

[28] [28] W. Pang, X. Wang, Y. Sun, H. Zhang, J. Li, R. Chen, Q. Liu, T. Zhao, K. Yang, M. Zhou, *et al.*, "Bayesian spatio-temporal graph transformer network (b-star) for multi-aircraft trajectory prediction," in *Proc. ACM MM*, Lisboa, Portugal, Oct. 10–14, 2022, pp. 3979–3988.

[29] [29] L. Zhu, D. Qu, and M. Xu, "Research on Agricultural Biotechnology Management Work,"

ISSN: 3008-1394

Journal of Anhui Agricultural Sciences, vol. 45, no. 29, pp. 221–223, Oct. 2017.

[30] [30] Y. Chen, L. Wu, and M. Zaki, "Iterative Deep Graph Learning for Graph Neural Networks: Better and Robust Node Embeddings," in *Proc. NeurIPS*, Online, Dec. 6–12, 2020, pp. 19314–19326.

[31] [31]Y. Ma, D. Qu and Y. Wang, "Quantum Walk–Inspired Fourier Operators for Markovian Dynamics and Modular Structure Detection," 2025 6th International Conference on Machine Learning and Computer Application (ICMLCA), Shenzhen, China, 2025, pp. 360-363, IEEE.

[32] [32]T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," IEEE Signal Processing Magazine, 2020.

[33] [33] Qu, D., & Ma, Y. (2025). Fourier–Transformer Mixer Network for Efficient Video Scene Graph Prediction. Engineering Proceedings, 120(1), 16.

[34] [34] D. Qu (2017), "Analysis and Research on the Changes in University Students' Employment Attitudes", Scientific Chinese, 3(Z), 140.

[35] [35] Qu, D., & Ma, Y. (2026, February). TM-GNN: treg-regulated Markov graph neural networks for stable community detection in dynamic systems. In International Conference on Computer Graphics, Artificial Intelligence, and Data Processing (ICCAID 2025) (Vol. 14113, pp. 244-249). SPIE.

[36] [36] Rossi E.Chamberlain B.Frasca F.Eynard D.Monti F.Bronstein M. (2020). "Temporal graph networks for deep learning on dynamic graphs," in International Conference on Learning Representations (ICLR) Workshop (Amherst, MA: OpenReview.net).

[37] [37] Y. Ma and D. Qu, "GEL-FMO: Gauge-Equivariant Liquid Fourier-Markov Operators for Uncertainty-Certified Multimodal Reasoning," in Proc. 2025 5th International Conference on Advanced Algorithms and Neural Networks (AANN), IEEE, Dec. 2025, pp. 604–607.

[38] [38] D. Qu, Y. Ma and S. Zhang, "OAMF: Optics-Accelerated Multimodal Learning with Markov Temporal Priors and Fourier Regularization," 2025 4th International Conference on Image Processing, Computer Vision and Machine Learning (ICICML), Chongqing, China, 2025, pp. 600-605.

[39] [39] Ma, Y., & Qu, D. (2025). FMD-GAN: Generating Realistic and Class-Preserving Time Series with Neural Networks via Fourier–Markov Diffusion.

[40] [40] Y. Chen, H. Wen, Y. Li and Y. Ma, "SyntheClean: Enhancing Large-Scale Multimodal Models via Adaptive Data Synthesis and Cleaning," 2025 5th International Conference on Artificial Intelligence and Industrial Technology Applications (AIITA), Xi'an, China, 2025, pp. 1769-1772, doi: 10.1109/AIITA65135.2025.11047850.

[41] Rong Y.Huang W.Xu T.Huang J. (2020). "Dropedge: towards deep graph convolutional networks on node classification," in Proceedings of the International Conference on Learning Representations (ICLR) (Amherst, MA: OpenReview.net) (Accessed September 21, 2025).

[42] Ma, Y., Qu, D., & Wang, Y. (2026). Dynamic community detection using class preserving time series generation with Fourier Markov diffusion. Scientific Reports.

[43] Ma, Y., & Qu, D. (2025). Machine Learning Based Markov Decision Framework for Optimizing Circular Economy Systems. Engineering Proceedings, 120(1), 44.

[44] Qu, D., & Ma, Y. (2026). F^2-CommNet: Fourier–Fractional Neural Networks with Lyapunov Stability Guarantees for Hallucination-Resistant Community Detection. Frontiers in Computational Neuroscience, 19, 1731452.

[45] N. S. Sattar, "Exploring temporal community evolution: Algorithmic comparison and parallel detection," Appl. Netw. Sci., vol. 8, p. 64, 2023.