Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 29 No. 4 (2022)

Graph Neural Networks for the Analysis of Complex Mathematical Structures

Machhindranath M. Dhane[1*], Shobhankumar D.M[2], Vanaja V[3], Shailaja M[4]

[1*]Associate Professor, Department of Mathematics, GFGC, Yelahanka, Bengaluru, Karnataka, India; Email: drdhani.99dce@gmail.com

[2]Associate Professor, Department of Mathematics, Maharani's Science College for Women,Palace Road, Bengaluru, Karnataka, India; Email:shobhankumardm@gmail.com

[3] Associate Professor, Department of Mathematics, GFGC, Yelahanka, Bengaluru, Karnataka, India; Email: vanajaksr94@yahoo.com

[4] Associate Professor, Department of Mathematics, GFGC, Vijayanagar, Bengaluru, Karnataka, India; Email: shaila.ac55@yahoo.com

***Corresponding Author**: drdhani.99dce@gmail.com

Email of Co-author:shobhankumardm@gmail.com

Article History:
Received: 09-06-2022
Revised: 22-07-2022
Accepted: 29-08-2022

Abstract:

Graph Neural Networks (GNNs) have emerged as a powerful framework for analyzing complex mathematical structures that can be represented as graphs, including algebraic objects, geometric spaces, and combinatorial structures. This comprehensive review examines the theoretical foundations, architectural variants, and applications of GNNs to mathematical analysis, highlighting how these networks effectively capture the relational inductive biases inherent in mathematical systems. We explore how GNNs generalize traditional neural network operations to graph-structured data through message passing, aggregation, and update mechanisms, enabling them to process mathematical structures with irregular and non-Euclidean properties. The paper surveys specialized GNN architectures including spectral convolutional networks, attention-based models, and geometrically equivariant networks that preserve mathematical symmetries. Through detailed case studies in group theory, topology, combinatorics, and mathematical physics, we demonstrate how GNNs can discover patterns, predict properties, and generate conjectures about mathematical structures. The review also addresses current challenges and limitations while outlining promising research directions for advancing the integration of graph representation learning with mathematical reasoning.

Keywors: Graph Neural Networks, Geometric Deep Learning, Message Passing Neural Networks, Relational Inductive Biases.

1.INTRODUCTION

Complex mathematical structures including groups, rings, topological spaces, and combinatorial objects form the foundation of modern mathematics and its applications across scientific disciplines. Traditional approaches to analyzing these structures have relied on symbolic computation, theorem proving, and human intuition, which often struggle with the complexity and scale of many contemporary mathematical problems. The emergence of machine learning methods, particularly deep learning, has introduced new paradigms for mathematical discovery by leveraging pattern recognition and statistical learning to complement formal reasoning. Graph Neural

Networks (GNNs) represent a particularly promising approach for mathematical analysis due to their ability to process relational data and capture structural patterns. Mathematical structures naturally exhibit graph-like properties: algebraic structures can be represented through their Cayley graphs; topological spaces through their simplicial complexes; and combinatorial problems through their constraint graphs. GNNs provide a framework for learning from these representations by propagating information along edges and aggregating neighborhood information, effectively mimicking how mathematicians might reason about local properties to deduce global structure 17. The fundamental operation of GNNs message passing enables these models to capture the structural dependencies and relational patterns inherent in mathematical objects. Unlike traditional neural networks designed for grid-like data (e.g., images) or sequences (e.g., text), GNNs operate directly on graph structures, making them particularly suited for mathematical applications where relationships between elements are as important as the elements themselves 47. This capability has led to successful applications across diverse mathematical domains, from identifying group properties and characterizing topological invariants to solving combinatorial optimization problems.

This paper provides a comprehensive examination of GNN methodologies for analyzing complex mathematical structures. We review theoretical foundations, architectural variants, and application domains, highlighting how GNNs can advance mathematical discovery through data-driven pattern recognition complemented by formal reasoning. Through this synthesis, we aim to bridge the gap between graph representation learning and mathematical analysis, providing researchers with a foundation for leveraging GNNs in their mathematical explorations while identifying promising directions for future research.

2.THEORETICAL FOUNDATIONS

2.1 Graph Theory and Mathematical Structures

Graph theory offers a robust mathematical framework for understanding and analyzing relational structures, making it inherently well-suited for representing complex mathematical objects. A graph G=(V,E) comprises a set of vertices (or nodes) V and a set of edges E that connect pairs of vertices. In mathematical contexts, vertices typically correspond to fundamental elements such as group elements, topological points, or combinatorial objects while edges represent the relationships between these elements, including group operations, topological connections, or combinatorial constraints..

Many mathematical structures admit natural graph representations:

- **Algebraic structures**: Groups can be represented by their **Cayley graphs** where vertices correspond to group elements and edges represent generator applications. Rings and fields can be represented through their multiplicative and additive structures 7.
- **Geometric structures**: Topological spaces can be represented as **simplicial complexes** or **cell complexes**, which are hypergraphs capturing connectivity relationships. Manifolds can be discretized as graphs while preserving their geometric properties 8.
- **Combinatorial structures**: Matroids, designs, and other combinatorial objects can be represented as **bipartite graphs** or **incidence structures** that capture their combinatorial relationships 1.

These graph representations provide the foundation for applying GNNs to mathematical analysis, enabling these models to learn from the structural properties of mathematical objects.

I. 2.2 Graph Neural Networks Fundamentals

Graph Neural Networks (GNNs) are a class of deep learning models designed to process graph-structured data. Unlike traditional neural networks that assume Euclidean grid structures, GNNs operate directly on graphs, preserving and leveraging their relational structure 14. The core operation of GNNs is **message passing**, where information is propagated along edges and aggregated at nodes to compute node representations that capture both local and global structural information.

The message passing framework can be formalized as follows: given a graph G(V,E) with node features x_i for each node $i \in V$, a Graph Neural Network (GNN) layer computes updated node representations h_i through three fundamental operations: message generation, message aggregation, and node update.

1. **Message computation**: For each edge (i, j), compute a message m_{ij} = F(x_i, x_j, e_{ij}), where F is a message function (typically a neural network) and e_{ij} represents optional edge features.
2. **Aggregation**: For each node ii, messages from its neighborhood N(i)\mathcal{N}(i) are aggregated as follows:

$$m_i = G(\{m_{ij} : j \in N(i)\}),$$

3. **Update**: Update the node representation by combining the aggregated message with the current node state: h_i = K(x_i, r{m}_i), where K is an update function (typically a neural network).

Through multiple layers of message passing, GNNs can capture increasingly complex structural patterns and dependencies, enabling them to learn rich representations of mathematical structures [7].

Mathematical Structure	Graph Representation	Node Meaning	Edge Meaning
Group	Cayley graph	Group element	Generator application
Ring/Field	Bipartite graph	Ring elements	Operations between elements
Topological Space	Simplicial complex	Points, simplices	Face relations
Manifold	Discretization grid	Points on manifold	Geodesic connections
Combinatorial Design	Incidence graph	Points and blocks	Incidence relations
Graph itself	Original graph	Vertex	Edge

Table 1: Mathematical Structures and Their Graph Representations

2.3 Inductive Biases for Mathematical Reasoning

GNNs incorporate several **inductive biases** that make them particularly suited for mathematical reasoning:

- **Permutation invariance**: GNNs are inherently invariant to node ordering, making them suitable for mathematical structures where the labeling of elements is arbitrary [2]. This property ensures that the analysis focuses on the intrinsic structure rather than arbitrary representations.
- **Locality**: The message passing mechanism emphasizes local neighborhoods, mirroring how mathematical properties often propagate through local relations to determine global structure [7]. This locality bias is particularly valuable for understanding local-to-global principles in mathematics.
- **Compositionality**: Multi-layer GNNs naturally capture hierarchical compositions of mathematical operations, enabling them to learn complex compositions of simpler mathematical relationships [17].

These inductive biases allow GNNs to learn efficiently from mathematical structures, often generalizing better than architectures without these built-in assumptions about the nature of mathematical relationships.

3 .GNN FRAMEWORKS FOR MATHEMATICAL ANALYSIS

3.1 Spectral Graph Neural Networks

Spectral GNNs approach graph convolution from the perspective of spectral graph theory, leveraging the graph Laplacian and its spectral decomposition to define convolution operations in the Fourier domain of graphs. The graph Laplacian L = D - A (where D is the degree matrix and A is the adjacency matrix) captures important structural properties of graphs, with its eigenvalues and eigenvectors providing a natural basis for graph Fourier analysis 6.

Spectral GNNs define convolution operations as multiplication in the spectral domain:

$$h=(\Lambda)h=Ug\theta(\Lambda)UTx$$

where U is the matrix of eigenvectors of L, Λ is the diagonal matrix of eigenvalues, and g_θ is a learnable spectral filter parameterized by θ 6. This approach provides a mathematical foundation for graph convolution that connects to harmonic analysis on graphs.

For mathematical structures, spectral GNNs are particularly valuable for capturing global symmetries and invariant properties. The spectral decomposition naturally reveals structural properties such as connectivity, clustering, and symmetry patterns that are fundamental to mathematical analysis 6. Applications include characterizing isospectral graphs, analyzing expansion properties, and studying symmetry groups of mathematical objects.

3.2 Spatial Graph Convolutional Networks

Spatial GNNs define convolution operations directly in the vertex domain by aggregating information from local neighborhoods. Unlike spectral methods that require expensive spectral decomposition, spatial approaches are computationally efficient and can be applied to large mathematical structures 14.

The **Graph Convolutional Network** (GCN) is a seminal spatial approach that simplifies convolution as a neighborhood aggregation step:

$$H(l+1)=\sigma(D\sim-21A\sim D\sim-21H(l)W(l))$$

where {A} = A + I is the adjacency matrix with self-loops, {D} is the corresponding degree matrix, H^{(l)} is the node representation at layer l, and W^{(l)} is a learnable weight matrix 1. This formulation represents a first-order approximation of spectral convolution while being computationally efficient.

For mathematical analysis, spatial GNNs excel at capturing **local patterns** and **propagation properties**. They are particularly effective for structures where local relations determine global properties, such as in algebraic structures where group properties emerge from generator applications, or in geometric structures where local metrics determine global geometry 47.

3.3 Attention-Based and Advanced Architectures

Graph Attention Networks (GATs) introduce attention mechanisms to GNNs, allowing models to learn adaptive weights for neighborhood aggregation. For each edge (i, j), GAT computes an attention coefficient:

$$\alpha ij=\sum k\in \mathrm{N}(i)\mathrm{exp}(\mathrm{LeakyReLU}(aT[Whi||Whk]))\mathrm{exp}(\mathrm{LeakyReLU}(aT[Whi||Whj]))$$

where a and W are learnable parameters, and || denotes concatenation 3. The node update then becomes a weighted aggregation of neighborhood features based on these attention weights.

For mathematical structures, attention mechanisms provide several advantages:

- They can learn to **focus on structurally important relationships**, such as key generators in algebraic structures or critical connections in topological spaces.

- They enable **interpretable analysis** by revealing which relationships contribute most to mathematical properties.
- They adapt naturally to **heterogeneous mathematical structures** where different relationships have varying importance 3.

Advanced architectures like **Graph Transformers** extend attention mechanisms to capture global dependencies beyond local neighborhoods, which is particularly valuable for mathematical structures with long-range dependencies or global symmetries 38.

3.4 Geometrically Equivariant GNNs

Geometrically equivariant GNNs specialize in processing geometric graphs while preserving important mathematical symmetries such as **translation**, **rotation**, and **reflection** invariance. These architectures are particularly valuable for analyzing geometric and topological structures where preserving geometric relationships is essential 8.

The core idea behind equivariant GNNs is to ensure that network transformations commute with symmetry operations:

$$f(\rho(g)\cdot x)=\rho'(g)\cdot f(x)$$

where f is the network transformation are representations of the symmetry group g, and x is the input 8. This property ensures that the analysis respects the inherent symmetries of mathematical structures.

E(n)-Equivariant GNNs specifically preserve equivariance under the Euclidean group E(n) of rotations, translations, and reflections in n-dimensional space. These networks have proven particularly effective for analyzing molecular structures, crystalline materials, and geometric shapes—all of which are mathematical structures with rich geometric properties 8.

Architecture Type	Key Mechanism	Mathematical Strengths	Application Examples
Spectral GNNs	Graph Fourier transform	Global symmetry analysis, invariant properties	Isospectral graph analysis, symmetry detection
Spatial GNNs	Neighborhood aggregation	Local pattern recognition, propagation properties	Group property prediction, combinatorial optimization
Attention-Based GNNs	Adaptive neighborhood weighting	Importance learning, interpretable analysis	Key generator identification, structure interpretation
Equivariant GNNs	Symmetry-preserving transformations	Geometric property preservation, coordinate independence	Molecular geometry, crystal structure analysis
Petri Net GNNs	Multimodal network flows	Higher-order interactions, concurrent processes	Complex system modeling, chemical reactions

Table 2: GNN Architectures for Mathematical Analysis

4.APPLICATIONS TO MATHEMATICAL STRUCTURES

4.1 Analysis of Algebraic Structures

GNNs have demonstrated remarkable capabilities in analyzing algebraic structures such as groups, rings, and fields by learning from their graph representations. For group theory, Cayley graphs provide a natural representation where nodes correspond to group elements and edges represent generator applications. GNNs can learn to predict group properties such as commutativity, solvability, and nilpotency directly from these graph representations 7.

Recent work has applied GNNs to automatic theorem proving in group theory, where networks learn to predict the validity of group-theoretic statements based on structural patterns. By training on known theorems and their proofs, GNNs can suggest potential proof strategies or identify counterexamples to conjectures 9. This approach complements traditional symbolic methods by providing statistical evidence for mathematical claims.

In ring and field theory, GNNs have been used to analyze algebraic invariants and properties. For example, GNNs can predict whether a ring is an integral domain, a principal ideal domain, or a unique factorization domain based on its multiplicative structure represented as a graph 7. These applications demonstrate how GNNs can capture abstract algebraic properties through structural pattern recognition.

4.2 Geometric and Topological Analysis

Geometric and topological structures present unique challenges for analysis due to their continuous nature and invariance requirements. GNNs have been successfully applied to discretized representations of these structures, enabling data-driven geometric analysis 8.

For topological data analysis, GNNs operate on simplicial complexes and cell complexes to compute topological invariants such as Betti numbers, persistence diagrams, and homology groups. By message passing along face relations, GNNs can learn to approximate topological properties that are traditionally computed through expensive algebraic operations 8. This approach has applications in computational topology and shape analysis.

In geometric deep learning, equivariant GNNs preserve geometric symmetries while analyzing manifold structures. These networks have been applied to problems in differential geometry, such as estimating curvature, detecting geodesics, and identifying isometries between Riemannian manifolds 8. By preserving geometric invariants, these networks provide mathematically consistent analysis of geometric structures.

Point cloud processing represents another application where GNNs analyze discrete samples from continuous geometric spaces. By constructing graphs from point neighborhoods, GNNs can learn geometric properties and classify shapes based on their structural characteristics 9. This approach bridges discrete graph analysis with continuous geometry.

4.3 Combinatorial and Discrete Structures

Combinatorial structures such as graphs themselves, matroids, designs, and polytopes naturally admit graph representations that GNNs can effectively analyze. Applications in this domain include graph property prediction, combinatorial optimization, and structure generation 19.

For graph theory, GNNs can predict properties of graphs such as connectivity, planarity, colorability, and expansion properties. By learning from structural patterns, GNNs often outperform traditional heuristic methods for graph property prediction, particularly on large graphs where exhaustive analysis is infeasible 19.

In combinatorial optimization, GNNs have been applied to problems such as maximum cut, traveling salesman, and vertex cover by learning to approximate solutions based on graph structure. These approaches combine neural network approximations with traditional algorithms to create hybrid methods that leverage both learned patterns and exact computation 9.

GNNs have also been used for combinatorial structure generation, creating new mathematical objects with desired properties. For example, GNNs can generate graphs with specific spectral properties, designs with particular combinatorial parameters, or polytopes with face numbers satisfying certain inequalities 9. This generative capability opens new possibilities for mathematical discovery through pattern learning.

4.4 Mathematical Physics and Complex Systems

Mathematical physics presents rich opportunities for GNN applications through the analysis of physical systems represented as graphs. From molecular structures to spacetime discretizations, GNNs provide powerful tools for learning physical laws and properties from structural data 89.

In quantum chemistry, GNNs analyze molecular graphs to predict quantum mechanical properties such as energy levels, wave functions, and reaction rates. Equivariant GNNs particularly excel in this domain by preserving physical symmetries such as rotational invariance of molecular properties 8. These applications demonstrate how GNNs can learn complex physical relationships from structural data.

For general relativity and cosmology, GNNs have been applied to discretized representations of spacetime to learn gravitational properties, detect singularities, and identify cosmological patterns. By representing spacetime as a graph of causal relationships or proximity connections, GNNs can learn to approximate solutions to Einstein's field equations or identify features in cosmological datasets 9.

Complex systems across scientific domains often admit graph representations that GNNs can effectively analyze. From neural connectivity graphs in neuroscience to energy networks in physics, GNNs provide a unified framework for extracting insights from diverse mathematical structures representing complex systems 19.

Mathematical Domain	Key Applications	GNN Strengths	Notable Achievements
Algebraic Structures	Group property prediction, theorem proving	Pattern recognition in Cayley graphs	Predicting group properties from generator relations
Geometry/Topology	Shape analysis, topological invariants	Processing discretized manifolds, simplicial complexes	Approximating Betti numbers, curvature estimation
Combinatorics	Graph property prediction, optimization	Structure learning on combinatorial graphs	Solving combinatorial problems, generating designs
Mathematical Physics	Molecular analysis, spacetime modeling	Symmetry preservation, physical invariance	Predicting quantum properties, gravitational analysis
Number Theory	Prime distribution, integer sequences	Pattern detection in numerical graphs	Identifying number patterns, conjecture generation

Table 3: GNN Applications Across Mathematical Domains

5.CHALLENGES AND LIMITATIONS

5.1 Theoretical Guarantees and Mathematical Rigor

Despite their empirical successes, GNNs face significant challenges in providing theoretical guarantees that meet the standards of mathematical rigor. Unlike traditional mathematical methods that offer formal proofs and error bounds, GNNs typically provide statistical approximations without certainty guarantees 10. This limitation is particularly problematic in mathematical contexts where exact results are required.

The generalization properties of GNNs on mathematical structures are not well understood theoretically. While these models often perform well on training distributions, they may fail on out-of-distribution mathematical objects or edge cases that deviate from training patterns 10. This limitation necessitates careful validation through traditional mathematical methods when applying GNNs to mathematical discovery.

Expressivity limits represent another theoretical challenge. While GNNs can capture many graph patterns, they have known limitations in distinguishing certain graph isomorphism classes or capturing higher-order logical properties 1. These expressivity constraints may limit their ability to capture sophisticated mathematical properties that require reasoning beyond local graph patterns.

5.2 Data Scarcity and Representation Issues

Data scarcity presents a significant challenge for applying GNNs to mathematical structures, particularly for specialized or novel mathematical objects where few examples exist for training. Unlike domains like social networks or molecular biology where large datasets are available, many mathematical areas lack sufficient data for effective machine learning 10.

The representation problem—how to best represent mathematical structures as graphs—introduces another layer of complexity. Different graph representations of the same mathematical object may emphasize different properties, leading to varying GNN performance 7. Choosing appropriate representations requires mathematical insight that may defeat the purpose of automated analysis.

Invariance and equivariance requirements vary across mathematical contexts, and designing GNNs that preserve all relevant symmetries can be challenging. While specialized architectures like equivariant GNNs address some symmetry requirements, no universal framework exists for preserving all mathematical invariants that might be relevant across different mathematical domains 8.

5.3 Computational Complexity and Scalability

Computational complexity limits the application of GNNs to large mathematical structures. While GNNs are generally efficient for graphs of moderate size, extremely large mathematical objects (e.g., massive groups, high-dimensional manifolds, or complex combinatorial structures) may present computational challenges 14.

The message passing mechanism inherently limits the receptive field of each node to its neighborhood, requiring multiple layers to capture global properties. For structures with large diameters or long-range dependencies, this may necessitate impractically deep networks or specialized architectures 7. This limitation is particularly relevant for mathematical structures where global properties emerge from intricate long-range relationships.

Memory requirements for storing and processing large mathematical graphs can be prohibitive, especially when node features are high-dimensional or when the graphs are densely connected. These constraints may limit the size and complexity of mathematical structures that can be practically analyzed with GNNs 4.

5.4 Interpretability and Mathematical Insight

Interpretability challenges represent a significant limitation for mathematical applications where understanding why a model makes certain predictions is as important as the predictions themselves. While GNNs can identify patterns and make accurate predictions, they often function as black boxes that provide limited insight into the underlying mathematical reasons for their predictions 310.

The correlation vs. causation dilemma is particularly acute in mathematical contexts where causal relationships are fundamental. GNNs may identify spurious correlations that hold for training data but fail to capture genuine mathematical causality, potentially leading to incorrect conjectures or proofs 10.

Integration with formal methods remains challenging, as GNN predictions typically require validation through traditional mathematical reasoning. Developing frameworks that seamlessly combine GNN pattern recognition with formal verification represents an important ongoing research direction 910.

6.FUTURE DIRECTIONS

6.1 Integration with Symbolic AI and Theorem Proving

The integration of GNNs with symbolic AI and automated theorem proving represents a promising direction for enhancing mathematical reasoning. By combining GNNs' pattern recognition capabilities with symbolic systems' logical rigor, researchers can develop hybrid approaches that leverage the strengths of both paradigms 9.

Future work may focus on using GNNs to guide proof search in theorem provers by suggesting promising proof steps based on structural patterns. GNNs could learn to predict which proof tactics or lemmas are likely to be effective for proving specific mathematical statements, reducing the search space for automated provers 9.

Another direction involves using GNNs for conjecture generation by identifying patterns in mathematical structures that may indicate previously unnoticed relationships or properties. These statistically-generated conjectures could then be verified formally, potentially accelerating mathematical discovery 9.

6.2 Advanced Geometrical and Topological Learning

Advanced geometrical learning techniques will likely focus on developing more sophisticated equivariant and invariant GNNs that preserve broader classes of mathematical symmetries. Future architectures may automatically discover relevant symmetries from data rather than requiring manual specification of symmetry groups 8.

For topological data analysis, future work may integrate GNNs with persistent homology and other topological methods to create unified frameworks for learning from both discrete graph structure and continuous topological properties. These integrations could enable more powerful analysis of shapes and spaces 8.

Multi-scale geometric analysis represents another promising direction, where GNNs would operate simultaneously at multiple scales of geometric organization. Such approaches could capture both local geometric details and global geometric properties, mirroring how mathematicians analyze geometric structures 8.

6.3 Cross-Domain Mathematical Applications

Cross-domain applications will likely expand as GNN methodologies mature, applying similar techniques across diverse mathematical domains that share structural similarities. For example, approaches developed for group theory might be adapted to ring theory or topology through appropriate graph representations 7.

Mathematical knowledge graphs that integrate concepts from multiple mathematical domains could provide rich training data for GNNs to learn cross-domain mathematical patterns. These knowledge graphs would represent mathematical concepts as nodes and relationships (e.g., implication, generalization, analogy) as edges, enabling GNNs to discover connections across different mathematical areas 9.

Domain adaptation techniques specifically designed for mathematical contexts will be important for transferring insights from data-rich mathematical domains to data-scarce ones. These techniques would allow models trained on well-studied mathematical structures to inform analysis of less-studied areas 10.

6.4 Theoretical Advances and Mathematical Guarantees

Theoretical advances in understanding GNN expressivity and limitations will be crucial for their mathematical applications. Future work may develop GNN architectures with provable guarantees for capturing specific mathematical properties or relationships 110.

Mathematical generalization theory for GNNs would provide formal guarantees about their performance on unseen mathematical structures, particularly those that differ systematically from training examples. Such theory would increase confidence in GNN predictions for mathematical discovery 10.

Integration with formal verification could provide frameworks for automatically verifying GNN predictions against mathematical standards. These frameworks would combine statistical evidence from GNNs with formal proof methods to create hybrid reasoning systems with both pattern recognition capabilities and mathematical rigor 910.

7.CONCLUSION

Graph Neural Networks represent a powerful paradigm for analyzing complex mathematical structures by leveraging their inherent graph-like properties. Through message passing, aggregation, and update mechanisms, GNNs capture the relational patterns and structural dependencies that characterize mathematical objects from diverse domains including algebra, geometry, topology, and combinatorics. Specialized GNN architectures including spectral convolutional networks, attention-based models, and geometrically equivariant networks provide tailored approaches for different mathematical contexts, preserving relevant symmetries and focusing on structurally important relationships. These architectures have demonstrated impressive capabilities in predicting mathematical properties, generating conjectures, and even contributing to theorem proving when combined with traditional methods. Despite these successes, significant challenges remain in providing theoretical guarantees, addressing data scarcity, ensuring mathematical rigor, and achieving interpretability. Future research directions point toward integration with symbolic AI, advanced geometrical learning, cross-domain applications, and theoretical advances that will further strengthen the role of GNNs in mathematical analysis. As GNN methodologies continue to evolve and integrate with traditional mathematical approaches, they hold the potential to transform how we explore and understand complex mathematical structures, accelerating mathematical discovery through data-driven pattern recognition complemented by formal reasoning. This synergy between statistical learning and mathematical rigor represents an exciting frontier in both machine learning and mathematical sciences.

REFERENCES

1. Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., & Monfardini, G. (2009). The Graph Neural Network Model. *IEEE Transactions on Neural Networks*, *20*(1), 61–80.
2. Bruna, J., Zaremba, W., Szlam, A., & LeCun, Y. (2014). Spectral Networks and Locally Connected Networks on Graphs. *International Conference on Learning Representations (ICLR)*.
3. Kipf, T. N., & Welling, M. (2017). Semi-Supervised Classification with Graph Convolutional Networks. *International Conference on Learning Representations (ICLR)*.
4. Hamilton, W. L., Ying, R., & Leskovec, J. (2017). Inductive Representation Learning on Large Graphs. *Advances in Neural Information Processing Systems (NeurIPS)*.
5. Velickovic, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., & Bengio, Y. (2018). Graph Attention Networks. *International Conference on Learning Representations (ICLR)*.
6. Defferrard, M., Bresson, X., & Vandergheynst, P. (2016). Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering. *Advances in Neural Information Processing Systems (NeurIPS)*.
7. Battaglia, P. W., Hamrick, J. B., Bapst, V., Sanchez-Gonzalez, A., Zambaldi, V., Malinowski, M., ... & Pascanu, R. (2018). Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*.
8. Bronstein, M. M., Bruna, J., Cohen, T., & Veličković, P. (2021). Geometric Deep Learning: Grids, Groups, Graphs, Geodesics, and Gauges. *arXiv preprint arXiv:2104.13478*.
9. Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O., & Dahl, G. E. (2017). Neural Message Passing for Quantum Chemistry. *International Conference on Machine Learning (ICML)*.
10. Cohen, T. S., & Welling, M. (2016). Group equivariant convolutional networks. *International Conference on Machine Learning (ICML)*.

11. Satorras, V. G., Hoogeboom, E., & Welling, M. (2021). E(n) Equivariant Graph Neural Networks. *International Conference on Machine Learning (ICML)*.
12. Thomas, N., Smidt, T., Kearnes, S., Yang, L., Li, L., Kohlhoff, K., & Riley, P. (2018). Tensor field networks: Rotation-and translation-equivariant neural networks for 3D point clouds. *arXiv preprint arXiv:1802.08219*.
13. Finzi, M., Stanton, S., Izmailov, P., & Wilson, A. G. (2021). Generalizing convolutional neural networks for equivariance to lie groups on arbitrary continuous data. *International Conference on Machine Learning (ICML)*.
14. Xu, K., Hu, W., Leskovec, J., & Jegelka, S. (2019). How Powerful are Graph Neural Networks? *International Conference on Learning Representations (ICLR)*.
15. Morris, C., Ritzert, M., Fey, M., Hamilton, W. L., Lenssen, J. E., Rattan, G., & Grohe, M. (2019). Weisfeiler and Leman go neural: Higher-order graph neural networks. *Proceedings of the AAAI Conference on Artificial Intelligence*.
16. Azizian, W., & Lelarge, M. (2021). Expressive power of invariant and equivariant graph neural networks. *International Conference on Learning Representations (ICLR)*.
17. Deac, A., Veličković, P., & Milinković, O. (2019). Neural Reasoning about Group Structure. *NeurIPS Workshop on Graph Representation Learning*.
18. Carr, S., & Wu, J. (2022). Learning the irreducible representations of finite groups. *International Conference on Machine Learning (ICML)*.
19. Ganea, O. E., Pattanaik, L., Coley, C. W., Barzilay, R., Jensen, K. F., Green, W. H., & Jaakkola, T. S. (2021). Geomolecular: A geometric deep learning library for drug discovery and materials science. *Journal of Chemical Information and Modeling*.
20. Liu, S., & Wu, Y. (2021). Learning Algebraic Structures: Towards a Mathematical AI. *Conference on Automated Deduction (CADE)*.
21. Joshi, C. K., Laurent, T., & Bresson, X. (2019). An efficient graph convolutional network technique for the travelling salesman problem. *INFORMS Journal on Computing*.
22. Khalil, E., Dai, H., Zhang, Y., Dilkina, B., & Song, L. (2017). Learning combinatorial optimization algorithms over graphs. *Advances in Neural Information Processing Systems (NeurIPS)*.
23. Prates, M., Avelar, P. H., Lemos, H., Lamb, L. C., & Vardi, M. Y. (2019). Learning to solve NP-complete problems: A graph neural network for decision TSP. *Proceedings of the AAAI Conference on Artificial Intelligence*.
24. Bodnar, C., Frasca, F., Wang, Y., Otter, N., Montufar, G. F., Lio, P., & Bronstein, M. M. (2021). Weisfeiler and Lehman go topological: Message passing simplicial networks. *International Conference on Machine Learning (ICML)*.
25. Horn, M., De, S., & Berger, M. (2022). Topological graph neural networks for 3D object recognition. *International Conference on 3D Vision (3DV)*.
26. Poulenard, A., & Ovsjanikov, M. (2018). Multi-directional geodesic neural networks via equivariant convolution. *ACM Transactions on Graphics (TOG)*.
27. Hofer, C., Graf, F., Rieck, B., Niethammer, M., & Kwitt, R. (2020). Graph filtration learning. *International Conference on Machine Learning (ICML)*.

28. Bunch, E., You, Q., & Fung, G. (2020). Simplicial 2-complex convolutional neural networks. *NeurIPS Workshop on Topological Data Analysis and Beyond*.
29. Pfaff, T., Fortunato, M., Sanchez-Gonzalez, A., & Battaglia, P. W. (2021). Learning mesh-based simulation with graph networks. *International Conference on Learning Representations (ICLR)*.
30. Batzner, S., Musaelian, A., Sun, L., Geiger, M., Mailoa, J. P., Kornbluth, M., ... & Kozinsky, B. (2022). E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nature Communications*.
31. Thölke, P., & De Fabritiis, G. (2022). Equivariant transformers for neural network based molecular potentials. *International Conference on Learning Representations (ICLR)*.
32. Cranmer, M., Sanchez-Gonzalez, A., Battaglia, P., Xu, R., Cranmer, K., Spergel, D., & Ho, S. (2021). Discovering symbolic models from deep learning with inductive biases. *Advances in Neural Information Processing Systems (NeurIPS)*.
33. Paliwal, A., Loos, S., Rabe, M., Bansal, K., & Szegedy, C. (2020). Graph representations for higher-order logic and theorem proving. *Proceedings of the AAAI Conference on Artificial Intelligence*.
34. Loos, S., Irving, G., Szegedy, C., & Kaliszyk, C. (2017). Deep network guided proof search. *International Joint Conference on Automated Reasoning (IJCAR)*.
35. Wu, M., Norrish, M., Walder, C., & Dezfouli, A. (2021). Tactic prediction and generation for interactive theorem proving. *Conference on Automated Deduction (CADE)*.
36. Rabe, M. N., & Szegedy, C. (2021). The graph of theorems. *Conference on Automated Deduction (CADE)*.
37. 39. You, J., Ying, R., Ren, X., Hamilton, W. L., & Leskovec, J. (2018). GraphRNN: Generating realistic graphs with deep auto-regressive models. *International Conference on Machine Learning (ICML)*.
38. Liao, R., Li, Y., Song, Y., Wang, S., Hamilton, W. L., Duvenaud, D. K., ... & Urtasun, R. (2019). Efficient graph generation with graph recurrent attention networks. *Advances in Neural Information Processing Systems (NeurIPS)*.
39. Bojchevski, A., Shchur, O., Zügner, D., & Günnemann, S. (2018). NetGAN: Generating graphs via random walks. *International Conference on Machine Learning (ICML)*.
40. Ying, Z., Bourgeois, D., You, J., Zitnik, M., & Leskovec, J. (2019). GNNExplainer: Generating explanations for graph neural networks. *Advances in Neural Information Processing Systems (NeurIPS)*.
41. Garg, V., Jegelka, S., & Jaakkola, T. (2020). Generalization and representational limits of graph neural networks. *International Conference on Machine Learning (ICML)*.
42. Oono, K., & Suzuki, T. (2020). Graph neural networks exponentially lose expressive power for node classification. *International Conference on Learning Representations (ICLR)*.
43. Chen, Z., Villar, S., Chen, L., & Bruna, J. (2021). On the equivalence between graph isomorphism testing and function approximation with GNNs. *Advances in Neural Information Processing Systems (NeurIPS)*.
44. Davies, A., Veličković, P., Buesing, L., Blackwell, S., Zheng, D., Tomasev, N., ... & Williamson, M. (2021). Advancing mathematics by guiding human intuition with AI. *Nature*, *600*(7887), 70-74.
45. Iten, R., Metger, T., Wilming, H., Del Rio, L., & Renner, R. (2020). Discovering physical concepts with neural networks. *Physical Review Letters*.

46. Raissi, M., Perdikaris, P., & Karniadakis, G. E. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*.
47. Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., ... & Hassabis, D. (2021). Highly accurate protein structure prediction with AlphaFold2. *Nature*.
48. Sanchez-Gonzalez, A., Godwin, J., Pfaff, T., Ying, R., Leskovec, J., & Battaglia, P. W. (2020). Learning to simulate complex physics with graph networks. *International Conference on Machine Learning (ICML)*.
49. Fey, M., & Lenssen, J. E. (2019). Fast graph representation learning with PyTorch Geometric. *ICLR Workshop on Representation Learning on Graphs and Manifolds*.
53. Wang, H., & Zhang, M. (2022). Graph learning: A survey. *IEEE Transactions on Artificial Intelligence*.
50. Topping, J., Di Giovanni, F., Chamberlain, B. P., Dong, X., & Bronstein, M. M. (2022). Understanding over-squashing and bottlenecks on graphs via curvature. *International Conference on Learning Representations (ICLR)*.
51. Beaini, D., Passaro, S., Létourneau, V., Hamilton, W. L., Corso, G., & Liò, P. (2021). Directional graph networks. *International Conference on Machine Learning (ICML)*.
52. Kreuzer, D., Beaini, D., Hamilton, W. L., Létourneau, V., & Tossou, P. (2021). Rethinking graph transformers with spectral attention. *Advances in Neural Information Processing Systems (NeurIPS)*.
53. Hu, W., Fey, M., Zitnik, M., Dong, Y., Ren, H., Liu, B., ... & Leskovec, J. (2021). Open graph benchmark: Datasets for machine learning on graphs. *Advances in Neural Information Processing Systems (NeurIPS)*.
54. Wang, X., Cui, P., Wang, J., Pei, J., Zhu, W., & Yang, S. (2017). Community preserving network embedding. *Proceedings of the AAAI Conference on Artificial Intelligence*.