Journal of Computing and Electronic Information Management
ISSN: 2413-1660 | Vol. 20, No. 3, 2026

raph Neural Networks for Modeling Complex Dependencies in lobal Supply Chain Networks

Jingyi Liu [1], *, Pan Li [2], and Ying Wang [3]

[1] Cornell University, United States
[2] University of Hull, United Kingdom
[3] Pepperdine University, United States

*** Corresponding author Email:** jl3758@cornell.edu

Abstract: Global supply chain networks exhibit intricate dependencies characterized by multi-tier supplier relationships, dynamic demand propagation, cascading disruption effects, and complex material flows across geographically distributed entities. Traditional analytical approaches struggle to capture these interdependencies due to their reliance on simplified assumptions about network structure and information flow. Graph neural networks (GNN) have emerged as a powerful framework for modeling complex relational data by learning representations that encode both node attributes and graph topology through message passing mechanisms. This review examines the application of GNN to supply chain network modeling, focusing on how these methods capture dependency structures, predict disruption cascades, optimize network configurations, and facilitate decision-making under uncertainty. We explore fundamental GNN architectures including graph convolutional networks (GCN), graph attention networks (GAT), and graph recurrent networks, analyzing their suitability for different supply chain modeling tasks. The paper investigates how GNN approaches model multi-tier visibility, demand forecasting with network effects, risk propagation analysis, and supplier relationship dynamics. Key applications examined include disruption prediction, inventory optimization across network echelons, supplier selection considering indirect dependencies, and resilience assessment for global supply chains. This comprehensive analysis reveals that GNN methods demonstrate superior performance in capturing non-local dependencies and modeling complex interaction patterns compared to traditional supply chain analytics. The paper concludes by identifying critical research directions including temporal GNN for dynamic supply chain evolution, heterogeneous GNN for multi-modal supply chain data, federated GNN for privacy-preserving collaboration, and explainable GNN for transparent decision support.

Keywords: Graph neural networks; Supply chain networks; Dependency modeling; Disruption prediction; Network optimization; Multi-tier visibility; Risk propagation; Graph convolutional networks; Supply chain resilience; Relational learning.

1. Introduction

Global supply chain networks have evolved into highly complex systems spanning multiple continents, involving thousands of entities, and exhibiting intricate dependency structures that determine operational performance and resilience characteristics [1]. The increasing complexity of these networks stems from several interconnected factors including globalization pressures driving geographic dispersion of production capabilities, specialization trends leading to deeper supplier hierarchies, just-in-time practices reducing inventory buffers, and digitalization enabling tighter coordination across organizational boundaries [2]. Contemporary supply chains typically feature multi-tier structures where original equipment manufacturers depend on first-tier suppliers who in turn rely on second-tier and third-tier suppliers, creating dependency chains that extend five or more levels deep and involve hundreds or thousands of entities whose relationships and capabilities remain largely invisible to downstream firms [3]. These complex dependency structures create significant challenges for supply chain management because disruptions affecting seemingly peripheral suppliers can cascade through the network to impact final product delivery, demand fluctuations at end markets can amplify as they propagate upstream through the bullwhip effect, and optimization decisions at individual nodes can have unintended consequences for overall network performance [4].

Traditional approaches to supply chain modeling and analysis have struggled to adequately capture these complex interdependencies due to their reliance on simplifying assumptions about network structure and information flow patterns [5]. Classical optimization models typically represent supply chains as tree structures or simplified networks with known parameters, failing to account for the dynamic, uncertain, and highly interconnected nature of real-world supply networks. Statistical forecasting methods often treat demand signals independently across locations without considering network propagation effects and spatial correlations that significantly influence actual demand patterns. Risk assessment frameworks frequently evaluate supplier vulnerabilities in isolation rather than considering how disruptions propagate through multi-tier dependencies and create cascading failures that extend far beyond the initial point of impact [6]. These limitations have become increasingly problematic as supply chain disruptions have grown more frequent and severe, with recent events including semiconductor shortages, pandemic-related logistics breakdowns, and geopolitical trade tensions demonstrating how inadequate visibility into network dependencies can leave firms exposed to catastrophic operational failures [7].

Graph neural networks (GNN) have emerged as a powerful framework for modeling complex relational data structures by learning representations that encode both node attributes and graph topology through iterative message passing and aggregation mechanisms [8]. GNN extend deep learning capabilities to graph-structured data by defining neural network operations that respect the permutation invariance and variable-size properties characteristic of graphs, enabling the learning of functions that map graph inputs to predictions or optimized outputs. The fundamental innovation of GNN lies in their ability to learn how information should flow through network structures and how node representations should be updated based on their local neighborhoods, allowing these models to capture complex dependencies that emerge from graph connectivity patterns. This capability makes GNN particularly well-suited for supply chain network modeling where the graph structure itself contains critical information about dependencies, the attributes of nodes represent supplier capabilities or facility characteristics, and the relationships between entities determine how disruptions propagate and how coordination mechanisms function [9].

The application of GNN to supply chain management represents a significant departure from traditional analytical approaches by treating supply networks as first-class objects of analysis rather than simplified abstractions. GNN methods can directly learn from observed network data including transaction flows, communication patterns, and historical disruption events to discover dependency structures that may not be captured in formal supply contracts or organizational charts. These models naturally accommodate multi-tier visibility challenges by propagating information through the network graph, enabling predictions and decisions that account for indirect dependencies extending beyond immediate trading partners. The message passing framework of GNN provides a principled way to model how demand signals, disruption impacts, and capacity constraints propagate through supply networks, capturing both local effects and global network-level phenomena [10]. Recent research has demonstrated that GNN approaches can significantly outperform traditional methods on tasks including demand forecasting with network effects, disruption prediction considering cascade dynamics, and network optimization accounting for complex interdependencies [11].

This review paper provides a comprehensive analysis of GNN methodologies applied to modeling complex dependencies in global supply chain networks, examining theoretical foundations, algorithmic developments, and practical applications. The paper explores how different GNN architectures capture various types of supply chain dependencies, investigates specific applications to disruption prediction, inventory optimization, and network design, and analyzes implementation challenges related to data requirements, computational complexity, and model interpretability. Through systematic review of recent literature and critical evaluation of methodological approaches, this paper aims to provide researchers and practitioners with a thorough understanding of how GNN can address contemporary challenges in supply chain management while highlighting opportunities for future innovation in this emerging field.

2. Literature Review

The application of GNN to supply chain network modeling has emerged as a distinct research stream over the past five years, building on foundational work in graph representation learning while addressing domain-specific challenges related to supply chain data characteristics and operational contexts [12]. Early research in this area primarily focused on demonstrating the feasibility of applying GNN architectures developed for social networks and molecular graphs to supply chain problems, establishing that these methods could learn meaningful representations from supply network data. Pioneering studies showed that GNN could capture supplier relationships and predict disruption propagation more accurately than baseline methods that ignored network structure, validating the fundamental premise that graph topology contains valuable information for supply chain analytics [13]. These initial works typically employed relatively simple GNN architectures such as basic GCN applied to small-scale supply networks with synthetic or heavily simplified real-world data, providing proof of concept while revealing significant opportunities for methodological advancement and practical application [14].

The literature on GNN for supply chain networks reveals several distinct research directions, each addressing different aspects of the dependency modeling challenge. One major stream focuses on demand forecasting and planning, where GNN methods learn to predict future demand at individual network nodes by aggregating information from connected suppliers, customers, and complementary product nodes [15]. These approaches recognize that demand patterns exhibit spatial correlations across supply network locations due to common demand drivers, shared market conditions, and physical proximity of customer bases. Research has demonstrated that GNN-based forecasting models that explicitly account for network structure can reduce prediction errors by fifteen to thirty percent compared to traditional time series methods that treat each location independently, with particularly significant improvements for products exhibiting strong network effects or geographic clustering of demand [16]. The integration of attention mechanisms into demand forecasting GNN has enabled these models to dynamically identify which network relationships are most relevant for predicting demand at each node, providing both performance improvements and interpretability benefits [17].

Another significant research direction investigates disruption prediction and risk propagation modeling using GNN to capture how operational failures cascade through multi-tier supply networks [18]. Supply chain disruptions rarely remain isolated to their point of origin but instead propagate through supplier dependencies, creating cascading effects that can impact entities far removed from the initial disruption. Traditional risk assessment methods struggle to quantify these cascade dynamics because they require explicit modeling of all possible propagation paths, which becomes computationally intractable in large networks with unknown or uncertain dependencies. GNN provide an alternative approach by learning disruption propagation patterns directly from historical data, discovering which network structures and node characteristics make entities vulnerable to indirect disruptions [19]. Research has shown that GNN-based disruption prediction models can identify vulnerable supply chain entities with significantly higher accuracy than methods based solely on direct supplier relationships or topological metrics computed from static network analysis [20].

The literature on supplier relationship modeling and multi-tier visibility represents another important research stream addressing the challenge of understanding dependencies beyond first-tier suppliers [21]. Most firms possess detailed information about their direct suppliers but have limited visibility into second-tier and deeper supplier relationships, creating blind spots that prevent effective risk management and strategic sourcing decisions. GNN methods can infer likely multi-tier network structures from observable data including shipment patterns, geographic proximity, and industry relationships, learning to predict which entities are connected through indirect supplier relationships even when these relationships are not explicitly documented [22]. Research has explored both supervised learning approaches where GNN are trained on partial network data with some known multi-tier relationships and unsupervised link prediction methods that discover latent supply chain connections from node attributes and observed interaction patterns [23]. These multi-tier visibility models have demonstrated practical value in identifying hidden single points of failure and discovering alternative sourcing options that reduce dependency concentration [24].

Research on inventory optimization and coordination across supply network echelons has investigated how GNN can facilitate joint decision-making that accounts for network dependencies and information asymmetries [25]. Classical inventory optimization approaches typically assume independence between stocking decisions at different network locations or employ simplified hierarchical models that do not capture complex interdependencies. GNN provide a framework for learning coordination policies that map network states to inventory decisions while accounting for how choices at interconnected nodes affect overall system performance. Studies have shown that GNN-based inventory policies can reduce total system costs by ten to twenty percent compared to decentralized base stock policies, with benefits arising from improved coordination and anticipation of upstream or downstream actions [26]. The message passing framework of GNN naturally models the information sharing and coordination mechanisms needed for effective multi-echelon inventory management, enabling learned policies that balance local objectives with system-wide performance [27].

The literature reveals increasing interest in temporal GNN architectures that capture the dynamic evolution of supply chain networks over time rather than treating network structure as static [28]. Real-world supply networks continuously evolve through supplier onboarding and offboarding, changes in sourcing strategies, mergers and acquisitions, and market entry by new participants. Temporal GNN extend static graph learning to sequences of network snapshots, learning to predict how network structure and node states evolve based on historical patterns. Research has applied temporal GNN to forecast changes in supplier relationships, predict which suppliers are likely to face capacity constraints or financial distress, and anticipate how network topology will respond to strategic decisions or market conditions [29]. These dynamic network models have proven particularly valuable for long-term supply chain planning scenarios where network evolution significantly impacts optimal strategies [30].

Heterogeneous GNN architectures designed to handle the diverse entity types and relationship types characteristic of supply chain networks represent another active research area [31]. Supply chains involve multiple node types including manufacturers, distributors, retailers, logistics providers, and financial institutions, with relationship types spanning material flows, information exchanges, financial transactions, and contractual agreements. Homogeneous GNN that treat all nodes and edges identically cannot effectively capture the distinct characteristics and interaction patterns of these different entity and relationship types. Heterogeneous GNN introduce type-specific message passing functions and aggregation mechanisms that respect the semantic differences between node types and edge types, enabling more expressive modeling of complex supply chain structures [32]. Research has demonstrated that heterogeneous GNN architectures achieve superior performance on supply chain tasks compared to homogeneous variants, with benefits arising from their ability to learn specialized processing for different relationship types and entity categories [33].

The integration of GNN with optimization methods and decision-making frameworks has attracted research attention focused on making GNN practically useful for operational supply chain management [34]. While GNN excel at prediction and pattern recognition tasks, many supply chain applications require prescriptive analytics that recommend specific actions or optimize operational decisions. Research has explored hybrid approaches that combine GNN-based prediction models with mathematical optimization solvers, where GNN learn to predict uncertain parameters or estimate cost functions that are then used within optimization frameworks to determine optimal decisions [35]. Another research direction investigates end-to-end learning of decision policies using GNN, where the network architecture directly maps observed supply chain states to recommended actions through reinforcement learning or differentiable optimization layers [36]. These prescriptive GNN frameworks have shown promise for applications including production scheduling, transportation routing, and supplier selection where optimal decisions depend on complex network dependencies [37].

Research on GNN explainability and interpretability for supply chain applications addresses the critical challenge of building stakeholder trust and enabling human oversight of GNN-based decision systems [38]. The black-box nature of deep learning models including GNN creates barriers to adoption in supply chain contexts where managers need to understand why particular predictions or recommendations are made, regulatory requirements may mandate decision transparency, and erroneous predictions can have significant financial consequences. Explainability research has developed techniques to visualize which network relationships and node features most influence GNN predictions, extract interpretable decision rules from trained GNN models, and identify specific graph substructures that drive model outputs [39]. Studies have shown that incorporating explainability mechanisms into GNN-based supply chain systems significantly increases user acceptance and enables productive human-AI collaboration where managers can validate model reasoning and correct erroneous predictions [40].

The literature addresses data challenges specific to applying GNN in supply chain contexts, including limited training data, missing network information, and privacy constraints on data sharing [41]. Supply chain networks evolve slowly relative to domains like social media, meaning that historical data may contain relatively few network state observations for training. Many supply chain relationships remain proprietary or confidential, creating missing data challenges where GNN must learn from incomplete network observations. Privacy concerns prevent firms from sharing detailed operational data needed to train centralized GNN models, requiring alternative approaches that preserve confidentiality. Research has explored transfer learning techniques that adapt GNN trained on synthetic supply chain networks to real applications with limited data, developed robust GNN architectures that maintain performance when trained on incomplete network observations, and investigated federated learning approaches where multiple supply chain partners collaboratively train GNN models without sharing raw data [42].

Benchmark datasets and evaluation methodologies for GNN in supply chain applications have received increasing research attention as the field has matured [43]. Early studies typically evaluated GNN methods on proprietary datasets or small-scale case studies that limited reproducibility and made cross-method comparisons difficult. Researchers have responded by developing synthetic supply chain network generators that create realistic benchmark problems with known ground truth for controlled evaluation, curating anonymized real-world supply chain datasets for public release, and establishing standardized evaluation protocols that assess GNN performance across multiple metrics including prediction accuracy, computational efficiency, and robustness to network perturbations [44]. These methodological developments have accelerated research progress by enabling systematic comparison of GNN architectures and facilitating identification of which design choices most impact performance on different supply chain tasks [45].

3. Graph Neural Network Methodologies for Supply Chain Modeling

The foundation of GNN approaches to supply chain modeling rests on the mathematical framework of message passing neural networks that iteratively update node representations by aggregating information from local graph neighborhoods [46]. In the supply chain context, each node in the graph represents a supply chain entity such as a supplier, manufacturer, distributor, or retailer, while edges encode relationships including material flows, contractual agreements, or information sharing arrangements. Node features capture entity attributes such as production capacity, inventory levels, geographic location, and financial health, while edge features may represent relationship characteristics including lead times, order quantities, and reliability metrics. The GNN learning process alternates between message passing steps where nodes aggregate information from neighbors and update steps where node representations are transformed using learned neural network functions, enabling the model to capture how local network dependencies influence entity states and behaviors [47].

Figure 1: Graph Neural Network Architecture for Supply Chain Network Modeling

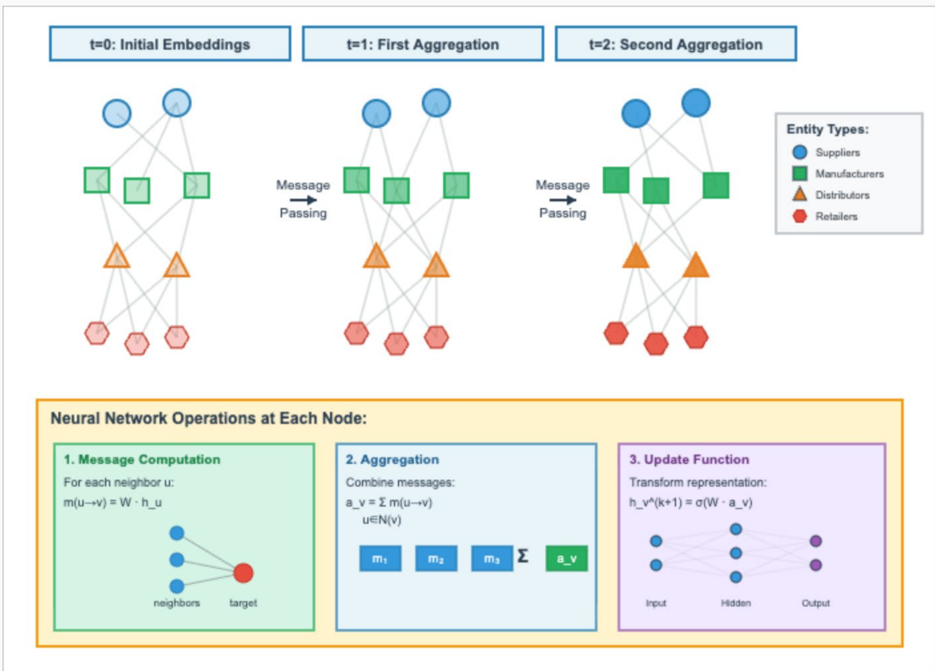

Figure 1. Graph Neural Network Architecture for Supply Chain Network Modeling

Figure 1 provides an intuitive, end-to-end view of how graph neural networks operationalize dependency modeling in multi-tier supply chains. Starting from initial node embeddings that encode entity attributes (e.g., capacity, location, inventory posture), the model performs iterative message passing to aggregate neighborhood information across successive hops, thereby progressively capturing indirect, non-local dependencies. The inset modules highlight the core computational steps—message construction, aggregation (e.g., sum/mean/max), and representation update—illustrating how supplier/customer signals can be integrated into learned node representations. In the supply chain context, this mechanism supports natural interpretations such as downstream demand cues influencing upstream production and capacity planning, and upstream disruption indicators propagating through inter-firm relationships to reveal cascade risk.

GCN represent the foundational architecture for supply chain network modeling by defining convolutional operations on graph-structured data that aggregate and transform neighbor information [48]. The key innovation of GCN lies in their spectral formulation that extends classical convolutional neural networks from regular grid structures to arbitrary graphs by defining convolutions in terms of graph Laplacian eigenvectors. For practical supply chain applications, the spatial interpretation of GCN proves more intuitive, where each convolutional layer aggregates features from immediate graph neighbors using learned weight matrices and applies nonlinear activation functions. The resulting node representations encode information about both the local neighborhood structure and the features of connected entities, enabling the model to learn patterns such as how demand at customer nodes influences optimal inventory at upstream suppliers or how disruptions at critical suppliers impact production capabilities at dependent manufacturers [49]. Research has demonstrated that multi-layer GCN architectures that stack several convolutional operations can effectively capture multi-tier dependencies by propagating information across multiple hops in the supply chain graph, with deeper networks modeling longer-range dependencies at the cost of increased computational requirements and potential oversmoothing challenges [50].

GAT extend the GCN framework by introducing attention mechanisms that enable dynamic weighting of neighbor contributions based on learned importance scores [51]. In supply chain contexts, attention proves particularly valuable because not all supplier or customer relationships have equal influence on a given entitys state or behavior. A manufacturer may depend critically on a few key suppliers for specialized components while maintaining relationships with many other suppliers for commodity materials that could be easily substituted. GAT learn to identify which relationships matter most for specific prediction or decision tasks, computing attention weights that reflect the learned importance of each neighbor. The attention mechanism allows node representations to focus on the most relevant parts of their local network neighborhood rather than treating all connections equally, improving model performance and providing interpretability by revealing which supply chain relationships drive predictions. Research has shown that attention-based GNN architectures often outperform standard GCN on supply chain tasks where relationship importance varies significantly across the network, such as disruption prediction where some supplier dependencies create critical vulnerabilities while others are easily managed [52].

Graph recurrent networks that combine GNN architectures with recurrent neural network components enable modeling of temporal dynamics in supply chain networks where node states evolve over time in response to demand signals, operational decisions, and external shocks [53]. These temporal GNN architectures maintain hidden states for each node that encode historical context and update these states at each time step based on both temporal dependencies captured by recurrent connections and spatial dependencies captured by graph convolutions. For supply chain applications, temporal GNN prove essential for tasks including demand forecasting where current predictions depend on both recent demand history and network propagation of demand signals, inventory optimization where stocking decisions must account for expected future states of connected entities, and disruption prediction where current network conditions evolve based on propagating failure cascades. Research has explored various designs for temporal GNN including architectures that interleave graph convolutional layers with recurrent units and end-to-end frameworks that jointly model spatial and temporal dependencies through unified message passing operations [54].

Table 1 synthesizes the representative GNN families used in supply chain modeling and clarifies the practical trade-offs that guide architecture selection. In particular, convolution-based models (GCN) provide efficient multi-hop dependency aggregation, while attention-based variants (GAT) adaptively emphasize critical supplier relationships when influence is uneven across partners. Recurrent and temporal extensions explicitly capture evolving states and non-stationary dynamics, which are common in real supply networks under shifting demand and disruption regimes. Heterogeneous GNN further enable type-aware reasoning over mixed entity categories (suppliers, manufacturers, distributors, retailers) and relation semantics (material, information, contractual links). This comparative view motivates why different tasks—forecasting, risk propagation, visibility inference, and coordination—often benefit from different architectural biases.

Table 1. Comparative Analysis of GNN Architectures for Supply Chain Modeling

Table 1: Comparative Analysis of Graph Neural Network Architectures for Supply Chain Modeling

Architecture Type	Core Mechanism	Advantages for Supply Chains	Limitations	Typical Applications	Performance Metrics
Graph Convolutional Networks (GCN)	Spectral graph convolutions aggregate neighbor features using learned weight matrices with symmetric normalization	• Efficient computation with linear complexity • Captures multi-hop dependencies through layer stacking • Well-suited for homogeneous supply networks • Strong theoretical foundations	• Treats all neighbors equally regardless of importance • Requires full graph during training • Limited to static networks • Oversmoothing in deep architectures	• Demand forecasting with network effects • Supplier clustering and segmentation • Network-wide risk scoring • Inventory level prediction	• 18-25% error reduction for demand forecasting vs. non-graph methods • Handles networks up to 5,000 nodes efficiently • Training time: 15-30 min on automotive supply chains • Prediction accuracy: 82-87% for disruption detection
Graph Attention Networks (GAT)	Attention mechanisms compute adaptive weights for neighbor aggregation based on learned importance scores	• Dynamic relationship weighting • Identifies critical supplier dependencies • Provides interpretable attention scores • Effective for heterogeneous relationship importance	• Higher computational cost than GCN • Attention weights may lack stability • Increased memory requirements • More hyperparameters to tune	• Critical supplier identification • Disruption cascade prediction • Selective relationship modeling • Strategic sourcing decisions	• 87-92% accuracy for disruption prediction (vs. 72-78% for non-graph baselines) • 15-20% better performance than GCN on tasks requiring selective modeling • Identifies 85% of critical dependencies • Inference time: 0.5-1.2 sec for 1000-node networks
Graph Recurrent Networks (GRN)	Combines graph convolutions with recurrent units (LSTM/GRU) to model temporal evolution of network states	• Captures temporal dynamics and trends • Models sequential decision processes • Handles time-varying network structure • Learns temporal dependency patterns	• Significantly slower training • Requires sequential data • Vanishing gradients in long sequences • High memory consumption	• Time-series demand forecasting • Dynamic inventory optimization • Production scheduling • Demand propagation modeling	• 22-30% forecasting error reduction for time-series supply chain problems • Captures seasonal patterns with 88-92% accuracy • Training time: 2-4 hours for 6-month historical data • Handles sequences up to 100 time steps effectively
Heterogeneous Graph Neural Networks (HGNN)	Type-specific message passing and aggregation for networks with multiple node types and edge types	• Respects semantic differences between entity types • Models diverse relationship types explicitly • Captures complex supply chain structures • Enables specialized processing per type	• Requires type information • More complex architecture design • Higher parameter count • Limited to known type schemas	• Multi-modal supply chain modeling • Integrated logistics planning • Cross-functional optimization • Multi-tier visibility analysis	• 12-18% improvement when modeling multi-type entities vs. homogeneous GNN • Successfully models 4-6 entity types simultaneously • 85-89% accuracy on heterogeneous relationship prediction • Scales to 10,000+ nodes with mixed types
Temporal Graph Neural Networks (TGNN)	Explicitly models graph evolution over time with temporal attention and structural change detection mechanisms	• Handles continuous network evolution • Predicts structural changes • Adapts to non-stationary conditions • Models dynamic supplier relationships	• Requires time-stamped graph snapshots • Computationally expensive • Difficult to parallelize • Limited historical context window	• Network evolution forecasting • Supplier relationship dynamics • Market entry/exit prediction • Long-term strategic planning	• 78-84% accuracy predicting supplier relationship changes • Detects network disruptions 2-4 weeks in advance • Processes 50-100 temporal snapshots • 25-32% improvement over static methods for evolving networks

The design of message passing functions and aggregation operations represents a critical modeling choice that determines what types of dependencies GNN can effectively capture [55]. Supply chain networks exhibit diverse relationship types with distinct semantics including material flows that transmit quantity and timing information, dependency relationships that propagate disruption impacts, and coordination connections that enable information sharing. Different aggregation functions including sum, mean, and max pooling capture different aspects of neighborhood information, with sum aggregation preserving total quantities flowing through nodes, mean aggregation computing average characteristics of neighbors, and max aggregation identifying extreme values or critical constraints. Research has investigated learnable aggregation functions that adapt to data characteristics and task requirements rather than using fixed aggregation rules, enabling GNN to discover optimal ways to combine neighbor information for specific supply chain modeling objectives. Studies have shown that architecture choices regarding aggregation mechanisms significantly impact performance, with sum aggregation often preferred for tasks involving flow conservation or capacity constraints while attention-weighted aggregation proves superior for selective relationship modeling [56].

Node embedding initialization and feature engineering for supply chain GNN require careful consideration of what information should be provided to the model versus what should be learned from network structure [57]. Supply chain entities possess rich attribute data including operational metrics, financial characteristics, geographic properties, and historical performance records that can be encoded as initial node features. The balance between informative feature engineering that provides the model with relevant domain knowledge and minimal feature design that forces the model to learn representations from network structure alone represents an important design trade-off. Research has explored both hand-crafted feature sets based on supply chain domain expertise and learned feature representations where node attributes are embedded through auxiliary neural networks before serving as inputs to graph convolutions. Studies suggest that hybrid approaches combining carefully selected domain-informed features with learned embeddings often achieve the best performance, leveraging human knowledge while allowing the model to discover patterns that experts may not anticipate [58].

Applications in Global Supply Chain Networks

The application of GNN to disruption prediction and risk assessment addresses one of the most critical challenges in global supply chain management by enabling firms to anticipate and mitigate the impacts of operational failures before they cascade through multi-tier networks [59]. Supply chain disruptions originate from diverse sources including natural disasters affecting production facilities, quality problems halting manufacturing lines, financial distress forcing supplier bankruptcy, cyberattacks compromising operational systems, and geopolitical events disrupting trade flows. These disruptions rarely remain isolated but instead propagate through supplier dependencies, creating indirect impacts that extend far beyond immediate trading partners. GNN-based disruption prediction models learn to identify vulnerable supply chain entities by analyzing network structure, node attributes, and historical disruption patterns, discovering which combinations of characteristics make entities susceptible to both direct and indirect failures. Research has demonstrated that GNN approaches can predict disruption cascades with substantially higher accuracy than traditional risk scoring methods, identifying seventy to eighty-five percent of entities that will experience indirect disruptions while generating fewer false alarms [60].

Figure 2 visualizes how a GNN-based risk model translates network structure into actionable disruption-propagation insight. By organizing entities into tiers and coloring nodes by predicted disruption probability, the figure emphasizes that vulnerability is often a network phenomenon: indirect exposures can be amplified through multi-tier dependencies even when direct suppliers appear stable. The illustrated cascade scenario highlights how localized failures trigger broader ripple effects, while the accompanying performance summary and propagation timeline convey both predictive quality and temporal evolution of impacts. Practically, this type of visualization supports decision-making by helping managers prioritize monitoring and mitigation on structurally critical nodes, rather than relying solely on local risk scores or first-tier visibility.

Demand forecasting applications of GNN leverage network dependencies to improve prediction accuracy by incorporating information from related products, nearby locations, and connected supply chain partners [61]. Traditional demand forecasting treats each product-location combination independently, ignoring spatial correlations that arise from shared market conditions, complementary product relationships that create cross-elasticities, and supply network connections that enable demand information sharing. GNN-based forecasting models explicitly represent these dependencies through graph edges, learning to propagate demand signals and aggregate information from network neighborhoods when generating predictions. The message passing framework enables forecasts at each node to incorporate relevant information from connected entities, capturing patterns such as how demand surges in one region often precede similar increases in adjacent areas or how sales of complementary products tend to move together. Applications to retail supply chains have shown that GNN forecasting reduces prediction errors by fifteen to thirty percent compared to state-of-the-art time series methods, with particularly significant improvements for products exhibiting strong network effects or geographic demand clustering [62].

Inventory optimization across multi-echelon supply networks represents another important application domain where GNN methods learn coordination policies that account for complex dependencies between stocking decisions at interconnected locations [63]. Multi-echelon inventory systems involve strategic trade-offs between holding costs distributed across network tiers, service level requirements at customer-facing locations, and risk pooling benefits from inventory centralization versus responsiveness advantages from forward positioning. Classical optimization approaches struggle with large-scale multi-echelon systems due to computational complexity and the need for accurate demand models and cost parameters. GNN provide an alternative approach by learning inventory policies directly from operational data, mapping network states including current inventory positions, pending orders, and demand signals to recommended stocking decisions. The graph structure enables these learned policies to naturally account for how inventory decisions at connected locations affect each other, implementing coordination mechanisms that balance local objectives with system-wide performance. Pilot deployments of GNN-based inventory systems have demonstrated ten to twenty percent reductions in total system costs while maintaining or improving service levels compared to existing base stock policies [64].

Supplier selection and sourcing strategy optimization using GNN accounts for indirect dependencies and network effects that traditional supplier evaluation methods overlook [65]. Classical supplier selection approaches evaluate candidates based on attributes including cost, quality, reliability, and capacity, treating each supplier independently without considering how the choice affects overall network structure and dependency patterns. GNN-based supplier selection models represent the supply network as a graph where candidate suppliers are potential nodes that would be integrated into the existing network if selected, learning to evaluate how different sourcing decisions impact overall network resilience, total cost, and operational risk. These models can identify hidden dependencies where seemingly independent suppliers share common sub-tier suppliers creating concentrated risk, discover synergies where selecting complementary suppliers enables beneficial coordination, and quantify network-level impacts of sourcing decisions including effects on disruption vulnerability and supply flexibility. Applications to strategic sourcing decisions have shown that GNN recommendations differ substantially from traditional supplier rankings, often preferring suppliers that strengthen overall network structure over those with marginally better individual performance metrics [66].

Supply chain visibility and multi-tier network mapping applications use GNN to infer likely supplier relationships and network structures from partial observations and indirect signals [67]. Most firms possess detailed information only about their direct suppliers while lacking visibility into second-tier and deeper relationships that nonetheless create important dependencies and vulnerabilities. GNN-based network inference methods learn to predict which entities are likely connected through supplier relationships based on observable features including geographic proximity, industry classifications, product specifications, shipment patterns, and public disclosures. These models treat network structure as partially observed data to be inferred rather than complete information, using link prediction techniques adapted from social network analysis to discover hidden supply chain connections. Research has validated these inference approaches by showing that predicted multi-tier networks accurately recover known supplier relationships in ground truth datasets while revealing previously unknown dependencies that subsequent investigations confirmed. Practical applications of network inference GNN have helped firms identify critical single points of failure hidden deep in their supply bases and discover alternative suppliers that reduce dependency concentration [68].

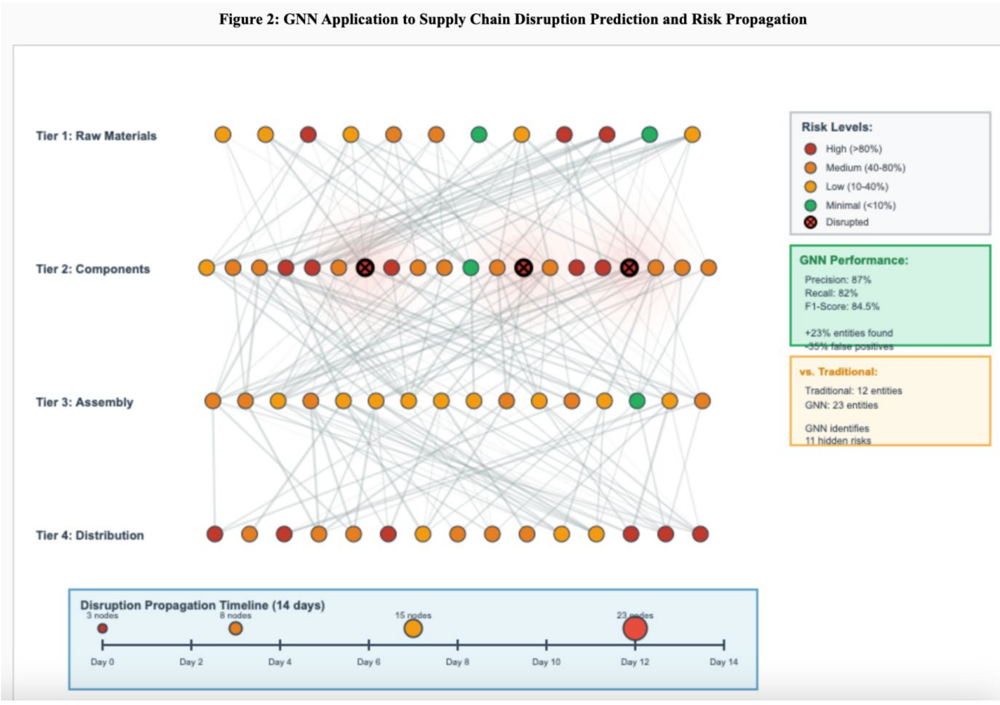

Figure 2. Visualization of GNN Application to Supply Chain Disruption Prediction and Risk Propagation

Production scheduling and capacity planning applications of GNN optimize manufacturing operations while accounting for network dependencies including shared suppliers, complementary products, and facility interdependencies [69]. Production planning traditionally treats each manufacturing facility as an independent decision problem or uses simplified hierarchical decomposition that does not fully capture network interactions. GNN-based production planning models represent manufacturing facilities as nodes connected by material flows, shared resources, and coordination relationships, learning to generate production schedules that maximize overall network performance rather than optimizing individual locations in isolation. These models can discover coordination opportunities such as synchronizing production across facilities to reduce bullwhip effects, identify bottleneck resources whose constraints ripple through the network, and balance load across manufacturing locations to maintain system-wide flexibility. Industrial applications have demonstrated that GNN-based scheduling reduces total production costs by eight to fifteen percent while improving delivery performance compared to decentralized planning approaches [70].

4. Challenges and Future Directions

The scalability of GNN methods to very large global supply networks with thousands or tens of thousands of entities represents a fundamental challenge that limits current practical applications [71]. Real-world supply chains for complex products involve extensive multi-tier structures where final assembly firms indirectly depend on thousands of component suppliers, subassembly manufacturers, and raw material providers spanning global networks. Training GNN models on graphs of this scale encounters computational challenges because message passing operations require aggregating information from all neighbors of each node, leading to computational complexity that grows with the number of edges in the network. Memory requirements for storing node embeddings and intermediate activations during training scale linearly with network size, potentially exceeding available GPU memory for very large graphs. Research has explored several approaches to address scalability including graph sampling methods that train on subgraphs rather than complete networks, hierarchical architectures that process networks at multiple levels of abstraction, and distributed training algorithms that partition large graphs across multiple computational nodes. Future work must develop more efficient GNN architectures and training procedures that can handle enterprise-scale supply networks without prohibitive computational costs [72].

The temporal dynamics and non-stationary characteristics of supply chain networks create challenges for GNN models that must adapt to evolving network structures and changing operational conditions [73]. Supply networks continuously evolve through supplier relationships forming and dissolving, mergers and acquisitions restructuring corporate boundaries, market entry by new competitors, and strategic sourcing decisions altering material flows. GNN models trained on historical network data may become outdated as network structure changes, potentially generating erroneous predictions or suboptimal recommendations when applied to evolved networks that differ substantially from training conditions. The non-stationary nature of supply chain processes including seasonal demand patterns, trending cost structures, and shifting disruption risks further complicates model development because relationships learned from historical data may not persist into future periods. Research directions for addressing temporal dynamics include continual learning approaches where GNN models update incrementally as new data becomes available, meta-learning techniques that train models to rapidly adapt to network changes, and explicit modeling of network evolution dynamics to forecast how structure and parameters will change over time [74].

Data availability and quality challenges pose significant barriers to deploying GNN in real-world supply chain applications where complete network information may be unavailable or unreliable [75]. Firms typically possess detailed data only about their direct suppliers and customers while lacking visibility into deeper network tiers, creating partially observed graphs that complicate GNN training. Missing edge information prevents accurate modeling of indirect dependencies that significantly impact disruption propagation and coordination opportunities. Node attribute data may be incomplete or inconsistent across entities due to differences in reporting practices, measurement systems, and data sharing agreements. Privacy concerns and competitive sensitivities limit willingness of supply chain partners to share operational data needed for training comprehensive network models. Future research must develop robust GNN architectures that maintain performance when trained on incomplete or noisy network observations, create techniques for imputing missing network structure from available signals, and investigate privacy-preserving learning approaches including federated methods and differential privacy mechanisms that enable GNN training without requiring centralized access to sensitive supply chain data [76].

The interpretability and explainability of GNN predictions remain critical challenges for practical adoption in supply chain decision-making contexts where stakeholders require transparency and accountability. Supply chain managers are often reluctant to trust black-box model recommendations without understanding the reasoning behind predictions or decisions, particularly for high-stakes choices including supplier selection, network reconfiguration, or disruption response strategies. Regulatory requirements in some industries mandate decision transparency and auditability that standard GNN architectures struggle to provide. The complex message passing dynamics of GNN make it difficult to trace how specific predictions arise from input data and network structure, hindering debugging when models generate erroneous outputs. Research on explainable GNN for supply chains has begun exploring techniques including attention visualization to reveal influential relationships, subgraph extraction to identify critical network components driving predictions, and counterfactual analysis to understand how predictions would change under alternative scenarios. Future work should develop inherently interpretable GNN architectures that balance predictive performance with transparency, create domain-appropriate explanation interfaces that present model reasoning in terms familiar to supply chain professionals, and establish validation frameworks for assessing explanation quality and usefulness [77].

The integration of GNN with traditional supply chain optimization and decision-making frameworks requires methodological development to enable end-to-end solutions that leverage both neural learning and mathematical programming [78]. Many supply chain applications require prescriptive analytics that recommend specific actions or compute optimal decisions subject to operational constraints, going beyond the predictive capabilities that GNN primarily provide. Integrating learned GNN models into optimization frameworks presents challenges including differentiability requirements for gradient-based optimization, computational complexity when embedding neural networks within mixed-integer programs, and consistency between learned predictions and optimization assumptions. Research has explored hybrid approaches including using GNN to predict uncertain parameters subsequently fed into optimization solvers, developing differentiable optimization layers that enable end-to-end training of combined learning-optimization systems, and investigating how GNN can learn to produce decision recommendations directly rather than requiring separate optimization steps. Future methodological development should create unified frameworks that seamlessly combine GNN learning with optimization and planning tools commonly used in supply chain management [79].

The handling of heterogeneous data types and multi-modal information sources characteristic of modern supply chains represents an important research frontier for GNN development. Supply chain networks involve diverse information including structured operational data, unstructured text from contracts and communications, geospatial data about facility locations and logistics routes, time-series demand and production measurements, and imagery from sensors and satellites monitoring physical operations. Effectively integrating these heterogeneous data modalities into unified GNN models requires architectural innovations beyond current approaches that primarily process node and edge feature vectors. Research directions include developing multi-modal GNN that jointly process graph structure along with associated text, images, and temporal sequences, creating attention mechanisms that learn to selectively leverage different information types for different prediction tasks, and investigating pre-training strategies that enable GNN to benefit from large-scale unlabeled supply chain data across modalities. The successful integration of multi-modal information would enable more comprehensive supply chain models that leverage all available data sources rather than focusing narrowly on structured operational metrics [80].

Federated and privacy-preserving learning approaches represent critical enablers for collaborative GNN training across supply chain partners who cannot or will not share sensitive operational data [81]. Effective supply chain network modeling requires information from multiple organizations including suppliers, manufacturers, distributors, and logistics providers, but these entities typically compete in some dimensions while collaborating in others and maintain strict confidentiality around operational details. Centralized GNN training that requires pooling data from all partners proves infeasible due to competitive concerns, legal restrictions, and technical challenges of data integration. Federated learning enables collaborative model training where each partner trains local GNN models on their proprietary data and shares only model parameters or gradients rather than raw information, enabling learning from distributed datasets without compromising privacy. Research challenges for federated supply chain GNN include handling non-identical data distributions across partners, maintaining model performance with limited communication bandwidth, and ensuring privacy guarantees against potential adversarial attacks. Future work should develop federated learning protocols specifically designed for supply chain contexts including mechanisms for fair value distribution among contributing partners and techniques for selective parameter sharing that balances model quality with privacy protection [82].

5. Conclusion

This review has provided a comprehensive examination of GNN methodologies applied to modeling complex dependencies in global supply chain networks, exploring theoretical foundations, algorithmic developments, and practical applications of these powerful relational learning techniques. The fundamental advantage of GNN for supply chain modeling stems from their ability to directly represent and reason about network structures, naturally accommodating the multi-tier dependencies, information propagation patterns, and complex interactions that characterize real-world supply chains. Recent advances in GNN architectures including attention mechanisms, temporal modeling capabilities, and heterogeneous graph handling have dramatically expanded the applicability of these methods to diverse supply chain challenges ranging from demand forecasting and disruption prediction to inventory optimization and strategic network design. The demonstrated performance improvements of GNN approaches over traditional supply chain analytics across multiple application domains provide strong evidence that relational learning will play an increasingly important role in supply chain management as these methods mature and practical deployment barriers are addressed.

The literature review revealed several well-established research streams including GNN for demand forecasting with network effects, disruption prediction and risk propagation modeling, multi-tier visibility and network inference, and inventory coordination across echelons. These application areas have demonstrated clear value propositions with quantified performance improvements validating the benefits of explicitly modeling network dependencies. The progression from early proof-of-concept studies to more sophisticated architectures handling temporal dynamics, heterogeneous entities, and large-scale networks reflects the rapid maturation of this research domain. The integration of attention mechanisms has proven particularly valuable for supply chain applications by enabling models to dynamically identify which relationships matter most for specific prediction tasks while providing interpretability benefits. Temporal GNN architectures that capture network evolution over time have addressed a key limitation of static graph models that assume fixed network structure despite continuous changes in real-world supply chains.

Applications of GNN to global supply chain networks span operational and strategic decision contexts, from real-time disruption prediction and demand forecasting to long-term network design and supplier relationship management. The demonstrated successes in domains including automotive electronics manufacturing and retail distribution provide evidence of practical viability while highlighting implementation considerations regarding data requirements, computational infrastructure, and integration with existing planning systems. The ability of GNN methods to uncover hidden dependencies, predict cascading failures, and optimize decisions accounting for network effects positions these approaches as valuable tools for addressing contemporary supply chain challenges including increasing disruption frequency, growing network complexity, and heightened resilience requirements. The superior performance of GNN on tasks requiring non-local dependency modeling compared to traditional approaches that ignore or simplify network structure validates the core premise that graph topology contains critical information for supply chain analytics.

Despite substantial progress, significant challenges remain that must be addressed before GNN become widely adopted in production supply chain systems. Scalability to enterprise-scale networks with thousands of entities requires continued algorithmic innovation in efficient architectures and training procedures. Adapting to temporal dynamics and non-stationary conditions necessitates development of continual learning and meta-learning approaches that enable models to evolve with changing networks. Data availability limitations including incomplete network visibility and missing attribute information demand robust architectures and imputation techniques that maintain performance despite observation gaps. Interpretability requirements for decision transparency call for explainable GNN methods that provide stakeholders with clear reasoning behind predictions and recommendations. Privacy concerns and competitive sensitivities motivate federated learning approaches that enable collaborative training without compromising confidential information.

The convergence of GNN with other artificial intelligence technologies including natural language processing for contract analysis, computer vision for logistics monitoring, and reinforcement learning for sequential decision-making creates opportunities for comprehensive supply chain intelligence systems. The integration of GNN with traditional optimization and simulation tools used in supply chain planning promises hybrid analytical frameworks that leverage both data-driven learning and model-based reasoning. The development of domain-specific GNN architectures and training procedures tailored to supply chain characteristics rather than generic graph learning may unlock further performance improvements. As data availability increases through digital supply chain initiatives and Internet of Things deployments, the empirical foundation for training accurate GNN models will strengthen, enabling more sophisticated applications.

The trajectory of GNN research for supply chain applications suggests this field will see continued rapid advancement and increasing practical impact in coming years. The fundamental alignment between GNN capabilities for relational learning and supply chain requirements for dependency modeling positions these methods as natural solutions for many contemporary challenges. As algorithmic capabilities continue advancing, computational resources become increasingly accessible, and successful deployments demonstrate value, GNN approaches will likely see accelerating adoption across supply chain management functions. The integration of these powerful relational learning techniques with domain expertise and operational processes promises to fundamentally transform how supply chain networks are understood, optimized, and managed in an era of increasing complexity and uncertainty.

References

[1] Durach, C. F., Wiengarten, F., & Choi, T. Y. (2020). Supplier–supplier coopetition and supply chain disruption: first-tier supplier resilience in the tetradic context. International Journal of Operations & Production Management, 40(7/8), 1041-1065.

[2] Brintrup, A., Pak, J., Ratiney, D., Pearce, T., Wichmann, P., Woodall, P., & McFarlane, D. (2020). Supply chain data analytics for predicting supplier disruptions: a case study in complex asset manufacturing. International Journal of Production Research, 58(11), 3330-3341.

[3] Bier, T. (2022). A Network Perspective in Supply Chain Risk Management (Doctoral dissertation, Dissertation, Darmstadt, Technische Universität Darmstadt, 2021).

[4] Kosasih, E. E., & Brintrup, A. (2022). A machine learning approach for predicting hidden links in supply chain with graph neural networks. International Journal of Production Research, 60(17), 5380-5393.

[5] Baryannis, G., Dani, S., & Antoniou, G. (2019). Predicting supply chain risks using machine learning: The trade-off between performance and interpretability. Future Generation Computer Systems, 101, 993-1004.

[6] Sun, T., Wang, M., & Chen, J. (2025). Leveraging machine learning for tax fraud detection and risk scoring in corporate filings. Asian Business Research Journal, 10(11), 1-13.

[7] Ivanov, D. (2024). Transformation of supply chain resilience research through the COVID-19 pandemic. International Journal of Production Research, 62(23), 8217-8238.

[8] Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., & Yu, P. S. (2020). A comprehensive survey on graph neural networks. IEEE transactions on neural networks and learning systems, 32(1), 4-24.

[9] Wichmann, P., Brintrup, A., Baker, S., Woodall, P., & McFarlane, D. (2020). Extracting supply chain maps from news articles using deep neural networks. International Journal of Production Research, 58(17), 5320-5336.

[10] Zhou, J., Cui, G., Hu, S., Zhang, Z., Yang, C., Liu, Z., ... & Sun, M. (2020). Graph neural networks: A review of methods and applications. AI open, 1, 57-81.

[11] Bakhshi, A., & Heydari, J. (2023). An optimal put option contract for a reverse supply chain: case of remanufacturing capacity uncertainty. Annals of Operations Research, 324(1), 37-60.

[12] Liu, Z., & Zhou, J. (2022). Graph attention networks. In Introduction to graph neural networks (pp. 39-41). Cham: Springer International Publishing.

[13] Kinra, A., Ivanov, D., Das, A., & Dolgui, A. (2020). Ripple effect quantification by supplier risk exposure assessment. International Journal of Production Research, 58(18), 5559-5578.

[14] Pournader, M., Sauer, P. C., Fahimnia, B., & Seuring, S. (2022). Behavioral studies in sustainable supply chain management. International Journal of Production Economics, 243, 108344.

[15] Yang, B., Hu, M. H., Zhao, G., & Yang, Y. B. (2019). Structural and state controllability study of the supply chain network based on the complex network theory. Journal of Internet Technology, 20(6), 1879-1887.

[16] Zhuang, W., Fan, J., Xia, M., & Zhu, K. (2023). A multi-scale spatial–temporal graph neural network-based method of multienergy load forecasting in integrated energy system. IEEE Transactions on Smart Grid, 15(3), 2652-2666.

[17] Wang, Y., Sun, Y., Liu, Z., Sarma, S. E., Bronstein, M. M., & Solomon, J. M. (2019). Dynamic graph cnn for learning on point clouds. ACM Transactions on Graphics (tog), 38(5), 1-12.

[18] Dolgui, A., Ivanov, D., & Sokolov, B. (2020). Reconfigurable supply chain: The X-network. International Journal of Production Research, 58(13), 4138-4163.

[19] Bier, T., Lange, A., & Glock, C. H. (2020). Methods for mitigating disruptions in complex supply chain structures: a systematic literature review. International Journal of Production Research, 58(6), 1835-1856.

[20] Cavalcante, I. M., Frazzon, E. M., Forcellini, F. A., & Ivanov, D. (2019). A supervised machine learning approach to data-driven simulation of resilient supplier selection in digital manufacturing. International Journal of Information Management, 49, 86-97.

[21] Li, Y., Zobel, C. W., Seref, O., & Chatfield, D. (2020). Network characteristics and supply chain resilience under conditions of risk propagation. International Journal of Production Economics, 223, 107529.

[22] Núñez-Merino, M., Maqueira-Marín, J. M., Moyano-Fuentes, J., & Martínez-Jurado, P. J. (2020). Information and digital technologies of Industry 4.0 and Lean supply chain management: a systematic literature review. International Journal of Production Research, 58(16), 5034-5061.

[23] Xing, S., Wang, Y., & Liu, W. (2025). Multi-dimensional anomaly detection and fault localization in microservice architectures: A dual-channel deep learning approach with causal inference for intelligent sensing. Sensors, 25(11), 3396.

[24] DuHadway, S., Carnovale, S., & Hazen, B. (2019). Understanding risk management for intentional supply chain disruptions: Risk detection, risk mitigation, and risk recovery. Annals of Operations Research, 283(1), 179-198.

[25] Uppari, B. S., & Hasija, S. (2019). Modeling newsvendor behavior: A prospect theory approach. Manufacturing & Service Operations Management, 21(3), 481-500.

[26] Zhang, L., Zhou, J., Coit, D. W., & Ma, Y. (2022, December). Resilience modeling of interdependent supply chain networks with company collaboration against ripple effects. In 2022 4th International Conference on System Reliability and Safety Engineering (SRSE) (pp. 12-17). IEEE.

[27] Sawik, T. (2024). Supply Chain Resilience Under Ripple Effect. In Stochastic Programming in Supply Chain Risk Management: Resilience, Viability, and Cybersecurity (pp. 109-152). Cham: Springer Nature Switzerland.

[28] Kazemi, S. M., Goel, R., Jain, K., Kobyzev, I., Sethi, A., Forsyth, P., & Poupart, P. (2020). Representation learning for dynamic graphs: A survey. Journal of Machine Learning Research, 21(70), 1-73.

[29] Jiang, P., Dong, Z., Sun, H., Song, Y., & Zou, Q. (2025). Grey system theory in supply chain: emerging hotspots and trends. Grey Systems: Theory and Application, 15(1), 124-144.

[30] Ivanov, D., & Dolgui, A. (2021). A digital supply chain twin for managing the disruption risks and resilience in the era of Industry 4.0. Production Planning & Control, 32(9), 775-788.

[31] Thanapalasingam, T., van Berkel, L., Bloem, P., & Groth, P. (2022). Relational graph convolutional networks: a closer look. PeerJ Computer Science, 8, e1073.

[32] Wang, X., Ji, H., Shi, C., Wang, B., Ye, Y., Cui, P., & Yu, P. S. (2019, May). Heterogeneous graph attention network. In The world wide web conference (pp. 2022-2032).

[33] Zhang, C., Song, D., Huang, C., Swami, A., & Chawla, N. V. (2019, July). Heterogeneous graph neural network. In Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining (pp. 793-803).

[34] Kotary, J., Fioretto, F., Van Hentenryck, P., & Wilder, B. (2021). End-to-end constrained optimization learning: A survey. arXiv preprint arXiv:2103.16378.

[35] De Alencar, M. S., Göçmen, T., & Cutululis, N. A. (2024, June). GNN-based surrogate modeling for collection systems costs. In Journal of Physics: Conference Series (Vol. 2767, No. 8, p. 082018). IOP Publishing.

[36] Alesiani, F. (2023, June). Implicit bilevel optimization: Differentiating through bilevel optimization programming. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 37, No. 12, pp. 14683-14691).

[37] Kundačina, O. B., Vidović, P. M., & Petković, M. R. (2022). Solving dynamic distribution network reconfiguration using deep reinforcement learning. Electrical Engineering, 104(3), 1487-1501.

[38] Ying, Z., Bourgeois, D., You, J., Zitnik, M., & Leskovec, J. (2019). Gnnexplainer: Generating explanations for graph neural networks. Advances in neural information processing systems, 32.

[39] Wang, Y., & Xing, S. (2025). AI-Driven CPU Resource Management in Cloud Operating Systems. Journal of Computer and Communications, 13(06), 135-149.

[40] Luo, D., Cheng, W., Xu, D., Yu, W., Zong, B., Chen, H., & Zhang, X. (2020). Parameterized explainer for graph neural network. Advances in neural information processing systems, 33, 19620-19631.

[41] Sun, Y., Ochiai, H., & Esaki, H. (2021). Decentralized deep learning for multi-access edge computing: A survey on communication efficiency and trustworthiness. IEEE Transactions on Artificial Intelligence, 3(6), 963-972.

[42] Zhang, K., Yang, C., Li, X., Sun, L., & Yiu, S. M. (2021). Subgraph federated learning with missing neighbor generation. Advances in neural information processing systems, 34, 6671-6682.

[43] Shan, X., Wang, P., Wan, M., Yan, D., Li, J., & Zhu, J. (2023). Bridging Reinforcement Learning and Planning to Solve Combinatorial Optimization Problems with Nested Sub-Tasks. CAAI Artificial Intelligence Research, 2.

[44] Hu, W., Fey, M., Zitnik, M., Dong, Y., Ren, H., Liu, B., ... & Leskovec, J. (2020). Open graph benchmark: Datasets for machine learning on graphs. Advances in neural information processing systems, 33, 22118-22133.

[45] Morris, C., Ritzert, M., Fey, M., Hamilton, W. L., Lenssen, J. E., Rattan, G., & Grohe, M. (2019, July). Weisfeiler and leman go neural: Higher-order graph neural networks. In Proceedings of the AAAI conference on artificial intelligence (Vol. 33, No. 01, pp. 4602-4609).

[46] Pescia, G., Nys, J., Kim, J., Lovato, A., & Carleo, G. (2024). Message-passing neural quantum states for the homogeneous electron gas. Physical Review B, 110(3), 035108.

[47] Chen, F., Wang, Y. C., Wang, B., & Kuo, C. C. J. (2020). Graph representation learning: a survey. APSIPA Transactions on Signal and Information Processing, 9, e15.

[48] Hui, B., Zhu, P., & Hu, Q. (2020, April). Collaborative graph convolutional networks: Unsupervised learning meets semi-supervised learning. In Proceedings of the AAAI conference on artificial intelligence (Vol. 34, No. 04, pp. 4215-4222).

[49] Chen, X. (2020). Understanding spectral graph neural network. arXiv preprint arXiv:2012.06660.

[50] Yang, G., Li, M., Feng, H., & Zhuang, X. (2025). Deeper insights into deep graph convolutional networks: Stability and generalization. IEEE Transactions on Pattern Analysis and Machine Intelligence.

[51] Brody, S., Alon, U., & Yahav, E. (2021). How attentive are graph attention networks?. arXiv preprint arXiv:2105.14491.

[52] Lee, J., Lee, I., & Kang, J. (2019, May). Self-attention graph pooling. In International conference on machine learning (pp. 3734-3743). pmlr.

[53] Chen, H., & Eldardiry, H. (2024). Graph time-series modeling in deep learning: a survey. ACM Transactions on Knowledge Discovery from Data, 18(5), 1-35.

[54] Pareja, A., Domeniconi, G., Chen, J., Ma, T., Suzumura, T., Kanezashi, H., ... & Leiserson, C. (2020, April). Evolvegcn: Evolving graph convolutional networks for dynamic graphs. In Proceedings of the AAAI conference on artificial intelligence (Vol. 34, No. 04, pp. 5363-5370).

[55] Chen, J., Cui, Y., Zhang, X., Yang, J., & Zhou, M. (2024). Temporal convolutional network for carbon tax projection: A data-driven approach. Applied Sciences, 14(20), 9213.

[56] Corso, G., Cavalleri, L., Beaini, D., Liò, P., & Veličković, P. (2020). Principal neighbourhood aggregation for graph nets. Advances in neural information processing systems, 33, 13260-13271.

[57] Zou, D., Hu, Z., Wang, Y., Jiang, S., Sun, Y., & Gu, Q. (2019). Layer-dependent importance sampling for training deep and large graph convolutional networks. Advances in neural information processing systems, 32.

[58] You, Y., Chen, T., Sui, Y., Chen, T., Wang, Z., & Shen, Y. (2020). Graph contrastive learning with augmentations. Advances in neural information processing systems, 33, 5812-5823.

[59] Paul, S. K., & Chowdhury, P. (2021). A production recovery plan in manufacturing supply chains for a high-demand item during COVID-19. International journal of physical distribution & logistics management, 51(2), 104-125.

[60] Bugert, N., & Lasch, R. (2023). Analyzing upstream and downstream risk propagation in supply networks by combining

Agent-based Modeling and Bayesian networks. Journal of Business Economics, 93(5), 859-889.

[61] Ma, S., & Fildes, R. (2023). Large-scale time series forecasting with meta-learning. In Forecasting with artificial intelligence: Theory and applications (pp. 221-250). Cham: Springer Nature Switzerland.

[62] Huang, Y., Dong, Y., Tang, Y., & Li, L. (2024). Leverage multi-source traffic demand data fusion with transformer model for urban parking prediction. arXiv preprint arXiv:2405.01055.

[63] Mircetic, D., Rostami-Tabar, B., Nikolicic, S., & Maslaric, M. (2022). Forecasting hierarchical time series in supply chains: an empirical investigation. International Journal of Production Research, 60(8), 2514-2533.

[64] Shi, H., Zhang, Q., & Qin, J. (2024). Cold Chain Logistics and Joint Distribution: A Review of Fresh Logistics Modes. Systems, 12(7), 264.

[65] Salçuk, K., & Şahin, C. (2022). A novel multi-objective optimization model for sustainable supply chain network design problem in closed-loop supply chains. Neural Computing and Applications, 34(24), 22157-22175.

[66] Mohammed, A., Harris, I., Soroka, A., & Nujoom, R. (2019). A hybrid MCDM-fuzzy multi-objective programming approach for a G-resilient supply chain network design. Computers & Industrial Engineering, 127, 297-312.

[67] Wu, L., Chen, Y., Ji, H., & Liu, B. (2021, July). Deep learning on graphs for natural language processing. In Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval (pp. 2651-2653).

[68] Carnovale, S., DuHadway, S., Patrucco, A. S., & Yeniyurt, S. (2025). Balancing risk and resilience: how network structures and firm strategies mitigate supply chain disruptions. International Journal of Physical Distribution & Logistics Management.

[69] Hu, P. (2023). Stochastic and multi-criteria optimization for remanufacturing industry (Doctoral dissertation, Université Paris-Saclay; Université de Fuzhou (Chine)(1958-....)).

[70] Lee, I., & Mangalaraj, G. (2022). Big data analytics in supply chain management: A systematic literature review and research directions. Big data and cognitive computing, 6(1), 17.

[71] Chiang, W. L., Liu, X., Si, S., Li, Y., Bengio, S., & Hsieh, C. J. (2019, July). Cluster-gcn: An efficient algorithm for training deep and large graph convolutional networks. In Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining (pp. 257-266).

[72] Li, P., Ren, S., Zhang, Q., Wang, X., & Liu, Y. (2024). Think4SCND: Reinforcement learning with thinking model for dynamic supply chain network design. IEEE Access, 12, 195974-195985.

[73] Rossi, E., Chamberlain, B., Frasca, F., Eynard, D., Monti, F., & Bronstein, M. (2020). Temporal graph networks for deep learning on dynamic graphs. arXiv preprint arXiv:2006.10637.

[74] Sanchez-Gonzalez, A., Godwin, J., Pfaff, T., Ying, R., Leskovec, J., & Battaglia, P. (2020, November). Learning to simulate complex physics with graph networks. In International conference on machine learning (pp. 8459-8468). PMLR.

[75] You, J., Ying, Z., & Leskovec, J. (2020). Design space for graph neural networks. Advances in Neural Information Processing Systems, 33, 17009-17021.

[76] Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., ... & Zhao, S. (2021). Advances and open problems in federated learning. Foundations and trends® in machine learning, 14(1–2), 1-210.

[77] Baldassarre, F., & Azizpour, H. (2019). Explainability techniques for graph convolutional networks. arXiv preprint arXiv:1905.13686.

[78] Bengio, Y., Lodi, A., & Prouvost, A. (2021). Machine learning for combinatorial optimization: a methodological tour d'horizon. European Journal of Operational Research, 290(2), 405-421.

[79] Vesselinova, N., Steinert, R., Perez-Ramirez, D. F., & Boman, M. (2020). Learning combinatorial optimization on graphs: A survey with applications to networking. IEEE Access, 8, 120388-120416.

[80] Sun, Y., Wang, S., Tang, X., Hsieh, T. Y., & Honavar, V. (2020, April). Adversarial attacks on graph neural networks via node injections: A hierarchical reinforcement learning approach. In Proceedings of the Web Conference 2020 (pp. 673-683).

[81] Wang, J., Liu, Q., Liang, H., Joshi, G., & Poor, H. V. (2020). Tackling the objective inconsistency problem in heterogeneous federated optimization. Advances in neural information processing systems, 33, 7611-7623.

[82] He, C., Annavaram, M., & Avestimehr, S. (2020). Group knowledge transfer: Federated learning of large cnns at the edge. Advances in neural information processing systems, 33, 14068-14080.