sustainability

Review

Graph Neural Networks for Routing Optimization: Challenges and Opportunities

Weiwei Jiang [1], Haoyu Han [1], Yang Zhang [1], Ji'an Wang [2], Miao He [3], Weixi Gu [4], Jianbin Mu [5,*] and Xirong Cheng [6,*]

1 School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China; jww@bupt.edu.cn (W.J.); 786898479@bupt.edu.cn (H.H.); zy2023140030@bupt.edu.cn (Y.Z.)
2 International School, Beijing University of Posts and Telecommunications, Beijing 100876, China; wang_ja@bupt.edu.cn
3 Yanqi Lake Beijing Institute of Mathematical Sciences and Applications, Beijing 101408, China; paris.muse7@gmail.com
4 China Academy of Industrial Internet, Beijing 100102, China; guweixi@china-aii.com
5 College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, China
6 School of Economics, Beijing Technology and Business University, Beijing 100048, China
* Correspondence: jianbinmu@zjut.edu.cn (J.M.); xirongcheng@btbu.edu.cn (X.C.)

Abstract: In this paper, we explore the emerging role of graph neural networks (GNNs) in optimizing routing for next-generation communication networks. Traditional routing protocols, such as OSPF or the Dijkstra algorithm, often fall short in handling the complexity, scalability, and dynamic nature of modern network environments, including unmanned aerial vehicle (UAV), satellite, and 5G networks. By leveraging their ability to model network topologies and learn from complex interdependencies between nodes and links, GNNs offer a promising solution for distributed and scalable routing optimization. This paper provides a comprehensive review of the latest research on GNN-based routing methods, categorizing them into supervised learning for network modeling, supervised learning for routing optimization, and reinforcement learning for dynamic routing tasks. We also present a detailed analysis of existing datasets, tools, and benchmarking practices. Key challenges related to scalability, real-world deployment, explainability, and security are discussed, alongside future research directions that involve federated learning, self-supervised learning, and online learning techniques to further enhance GNN applicability. This study serves as the first comprehensive survey of GNNs for routing optimization, aiming to inspire further research and practical applications in future communication networks.

Keywords: graph neural networks; routing optimization; distributed learning; supervised learning; reinforcement learning; dynamic networks; network topology; future networks

check for updates

Citation: Jiang, W.; Han, H.; Zhang, Y.; Wang, J.; He, M.; Gu, W.; Mu, J.; Cheng, X. Graph Neural Networks for Routing Optimization: Challenges and Opportunities. *Sustainability* **2024**, *16*, 9239. https://doi.org/10.3390/su16219239

Academic Editor: Jozsef Mezei

Received: 23 September 2024
Revised: 17 October 2024
Accepted: 22 October 2024
Published: 24 October 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Routing optimization can be expressed as a combinatorial optimization problem that selects packet forwarding paths with different optimization objectives and constraints under various scenarios [1]. Routing optimization has been considered in multiple disciplines, including operation research, transportation research, and computer science, for problems such as vehicle routing [2] and data packet routing [3]. Some classical optimization algorithms have been successfully applied in these domains, such as Dijkstra's algorithm for finding the shortest path in a graph [4]. Powered by these optimization algorithms, a large family of routing protocols has been designed and deployed in communication networks, e.g., the Open Shortest Path Protocol (OSPF) [5].

Routing optimization has been a pivotal research area in communication networks due to its significant application potential and the inherent challenges it poses [6]. In communication networks, it refers to the process of identifying the most efficient paths for data packets to traverse from a source to a destination within a network, based on inputs including network topology, traffic load, and other key variables. The primary objectives of routing optimization are to reduce latency, increase throughput, and ensure dependable communication while maximizing the efficiency of network resources. These optimizations are critical for enhancing network performance, scalability, and resilience across various applications, including the Internet, cloud computing, and large-scale enterprise systems. However, distinct challenges emerge across different network environments, especially in next-generation and specialized networks. For example, in unmanned aerial vehicle (UAV) networks, the primary obstacles include high mobility, frequent topology changes, energy constraints, and limited transmission range [7,8]. Similarly, satellite networks face difficulties stemming from satellite mobility, including dynamic constellation topologies and frequent inter-satellite link switching [9]. These challenges highlight the need for advanced routing solutions that can adapt to the unique demands of each network scenario.

Existing routing techniques fall short of meeting the demands of future networks, which are characterized by the need for extremely large bandwidth, ultra-low latency, deterministic delay, high reliability, and massive connectivity. To address these requirements, machine learning (ML) and artificial intelligence (AI) techniques have been introduced for routing optimization. These approaches can effectively learn optimized routing strategies by leveraging historical traffic patterns, allowing them to adapt to future conditions [1]. Compared to traditional routing algorithms and protocols, ML- and AI-based schemes have shown superior performance, as demonstrated in numerous studies [10,11]. ML paradigms can be more concisely and clearly related to the inputs that make up network topologies, traffic network matrices, routing schemes, and the output of routing optimization performance metrics. The testing of the trained models demonstrated that ML-based routing solutions significantly outperform traditional algorithms in terms of key performance metrics such as reduced latency and improved throughput, effectively adapting to dynamic network conditions and optimizing routing decisions. However, applying these ML-based routing methods in real-world network environments remains a challenge. This difficulty stems from the limitations in accurately modeling dynamic network topologies and the complexities involved in optimizing routing policies in real time, where adaptability and low-latency decision-making are crucial. These factors continue to pose significant barriers to the practical deployment of AI-driven routing solutions in modern communication systems.

As an extension of artificial neural networks on graph data, graph neural networks (GNNs) are seen as an opportunity to address the existing limitations and design new routing paradigms for future networks [12,13]. GNNs are a type of deep learning model designed to work with data that can be represented as graphs, such as social networks, biological networks, and recommendation systems. Unlike traditional neural networks that operate on grid-like data such as images, GNNs can directly process and learn from the complex relationships and structures inherent in graphs [14]. GNNs consist of layers of neural network units that aggregate and propagate information across the nodes and edges of a graph [15]. Through iterative message passing and aggregation steps, GNNs can capture both local and global patterns in the graph data, enabling tasks such as node classification, link prediction, and graph-level prediction [16]. These models have shown promising results in various domains where data are naturally represented as graphs, offering powerful tools for learning from interconnected data [17].

Traditional routing algorithms and protocols, such as Dijkstra's algorithm and OSPF, typically rely on predefined heuristics and static optimization strategies that excel in stable and predictable network environments. However, they struggle to adapt to the dynamic and complex nature of modern communication networks, where conditions can change rapidly due to factors like traffic fluctuations, mobility, and varying topology. In contrast, GNNs offer a more flexible and robust approach by leveraging their ability to learn from graph-structured data, capturing intricate relationships between nodes (e.g., network devices) and edges (e.g., network connections) [18]. GNNs excel in their capacity for generalization, allowing them to adapt to unseen topologies and conditions that were not part of the training data. This capability is particularly beneficial in next-generation communication networks, such as 5G and beyond, where high mobility and the need for real-time decision-making are paramount. Furthermore, GNNs can effectively model complex dependencies within the network, leading to improved routing decisions that optimize performance metrics such as latency, throughput, and energy efficiency. By employing iterative message-passing mechanisms, GNNs can continuously refine their predictions based on real-time data, making them a compelling choice for dynamic and scalable routing in the evolving landscape of communication networks [19].

Some surveys have focused on the routing algorithms and protocols in specific network scenarios. Routing algorithms for mobile ad hoc networks (MANETs) are classified into four categories in [20], namely, performance improvement, QoS-aware, energy-saving, and security-aware, with different optimization objectives. Both single-layer and multi-layer dynamic routing schemes in satellite networks are reviewed in [21], including software-defined networking (SDN)-based, QoS-based, and traffic-balancing dynamic routing. Considering the energy constraints in wireless sensor networks (WSNs), the selection of a cluster head has been an important factor when designing energy-efficient routing protocols in WSNs [22]. Both classical low-energy adaptive clustering hierarchy (LEACH) and bio-inspired protocols are reviewed in [23], with a focus on the criteria of cluster head selection.

The introduction of SDN has provided new opportunities for many applications in the networking domain, including routing [24]. SDN has a three-layer architecture: management, control, and data planes. By separating these planes, SDN empowers networking with strong control, programmability, and automation, adding new features to solve classical problems in traditional networks [25]. Some surveys have discussed routing algorithms and protocols for SDN [25,26]. Energy-efficient routing and load balancing in SDN is reviewed in [26] and a deep reinforcement learning (DRL)-based predictive and rate-adaptive energy-efficient routing scheme is proposed with guaranteed QoS. Three types of ML techniques for routing optimization in SDN, namely supervised learning, unsupervised learning, and reinforcement learning, are summarized and discussed in [25].

Several surveys have focused on ML-based techniques for network routing. ML-based intelligent routing algorithms are reviewed in [10], with the key concepts and applications, training and deployment strategies, and future development directions. Reinforcement learning (RL)-based routing protocols for vehicular ad hoc networks are reviewed in [11], with the introduction of their working procedure, advantages, disadvantages, and applications, and a qualitative comparison of their key features. ML-based routing optimization techniques for future communication networks have been reviewed in [1]. AI-enabled routing protocols for UAV networks are summarized and discussed in [7], including topology-predictive and self-adaptive learning-based routing algorithms. Q-learning-based position-aware routing protocols for flying ad hoc networks (FANETs) are reviewed in [8], with a focus on the relationship between Q-learning and routing when dealing with high-mobility and dynamic topology challenges. However, the discussion of GNN-based solutions is insufficient in the existing surveys of ML-based solutions.

There have also been surveys from the perspective of GNN-based methodologies. However, their focus was not on the specific field of network routing. Some existing surveys have focused on the application of graph-based methods in communications and networking domains [27]. The application of graph-based deep learning methods in three different scenarios is considered in [28], namely, wireless networks, wired networks, and software-defined networks, with a wide range of applications including routing, traffic prediction, resource allocation, and wireless link scheduling [29]. GNN is seen as a key enabler for the modeling, control and management of communication networks with strong generalization capabilities over traditional neural network solutions in [19]. Two example use cases, RouteNet [30] for performance evaluation in wired networks and

WCGCN [31] for radio resource management in wireless networks, are further implemented and discussed in [19] to demonstrate the superiority of GNN-based solutions. Another survey of combinatorial optimization on graphs is presented in [32], where the focus is on the machine learning structures used for solving combinatorial optimization problems on graphs, while these problems are from the telecommunications field. Routing is one of the networking applications discussed in [32].

Some surveys have discussed the application of graph-based deep learning to a specific network domain. The construction method for various wireless communication graphs is introduced in [33], with several GNN-based solutions for resource allocation, routing, and other applications in wireless networks. Graph-based solutions are discussed in [34] for resource allocation in integrated space and terrestrial communications, whereas routing optimization is not mentioned.

A comparison of this survey and other existing surveys is summarized in Table 1. Compared with existing surveys, this survey serves the specific purpose of providing an up-to-date literature review of GNN techniques for routing optimization. In existing surveys, GNN-based network routing solutions are not mentioned or the discussion of GNN-based solutions for routing optimization applications is not thorough. This paper is a comprehensive guide for newcomers with a full picture of existing studies. This paper is also insightful for experienced researchers with the collection of dataset and tool resources and the inspiring discussion of research challenges and opportunities.

The topic of GNN-based routing optimization, as explored in the paper, resonates with broader advancements in network management, artificial intelligence, and graph-based machine learning techniques [35,36]. Recent works on DRL for network optimization, for instance, focus on adaptive and real-time decision-making in areas such as load balancing, resource allocation, and traffic prediction. While DRL models excel at real-time routing decisions, GNNs provide a distinct advantage by capturing complex topological dependencies that are vital in networks with dynamic or irregular structures, such as UAV and satellite networks. Furthermore, the paper aligns with emerging trends in intent-based networking (IBN), where AI models predict and enforce network policies to meet high-level business or performance goals. GNNs, with their ability to generalize over unseen topologies, could further enhance the predictive capabilities of IBN by providing more robust and scalable solutions. Similarly, research on SDN and network function virtualization (NFV) intersects with GNN-based routing as both fields aim to make networks more programmable and efficient. These synergies suggest that GNN-based routing optimization could have broader applications in next-generation network architectures, making it a powerful tool not just for routing but for holistic network automation and intelligent infrastructure management.

Table 1. Comparison of this survey and existing surveys.

Article	Year	Summary	Shortcoming
[25]	2021	Supervised learning, unsupervised learning, and reinforcement learning techniques in SDN.	GNN-based solutions are not mentioned.
[11]	2021	RL-based routing protocols for vehicular ad hoc networks.	GNN-based solutions are not mentioned.
[1]	2021	ML-based routing optimization techniques for future networks.	The discussion for GNN-based solutions is not thorough.
[7]	2022	AI-enabled routing protocols for UAV networks.	GNN-based solutions are not mentioned.
[8]	2022	Q-learning-based position-aware routing protocols for FANETs.	GNN-based solutions are not mentioned.
[20]	2022	Routing algorithms for MANETs with performance improvement, QoS-aware, energy-saving, and security-aware categories.	GNN-based solutions are not mentioned.

Table 1. *Cont.*

Article	Year	Summary	Shortcoming
[23]	2022	Energy-efficient routing protocols for wireless sensor networks.	GNN-based solutions are not mentioned.
[21]	2022	Dynamic routing schemes in satellite networks.	GNN-based solutions are not mentioned.
[10]	2022	Machine learning-based intelligent routing algorithms.	The discussion for GNN-based solutions is not enough.
[34]	2022	Graph-based solutions for resource allocation in integrated space and terrestrial communications.	Routing optimization is not mentioned.
[19]	2022	A brief tutorial on GNNs and potential applications to communication networks, and two example use cases in wired and wireless networks.	The discussion for routing optimization is not enough.
[28]	2022	The application of graph-based deep learning methods in wireless, wired and software-defined networks.	The discussion for routing optimization is not thorough.
[37]	2023	Routing protocols in unmanned aerial vehicular networks.	GNN-based solutions are not mentioned.
[38]	2023	Routing protocols in vehicular adhoc networks.	GNN-based solutions are not mentioned.
[39]	2023	Reinforcement-learning-based routing algorithms in IoT.	GNN-based solutions are not mentioned.
[40]	2024	Machine learning solutions in IoT-based wireless sensor network routing.	GNN-based solutions are not mentioned.
[41]	2024	Routing algorithms in wireless sensor networks.	GNN-based solutions are not mentioned.
[42]	2024	Routing techniques for distributed cognitive radio networks.	GNN-based solutions are not mentioned.
[43]	2024	Routing and load-balancing mechanisms for software-defined vehicular networks.	GNN-based solutions are not mentioned.
This survey	2024	The application of graph-based deep learning methods for routing optimization in a wide range of communication and networking domains.	N/A

The process of selecting and evaluating relevant papers involved a comprehensive literature search using platforms like Google Scholar, employing keywords such as "Routing", "Network Modeling", "Graph Neural Network", and "Graph Convolutional Network". Additional criteria included the reputation of the journals or conferences, relevance to the study's objectives, and the quality of the papers, which was assessed through manual checks of methodologies and results. Specific metrics and measurements considered during the review included accuracy metrics such as mean relative error (MRE) and mean absolute percentage error (MAPE), as well as performance indicators related to latency, throughput, packet delivery ratio (PDR), and energy efficiency [44]. These metrics provide insight into the effectiveness of various GNN models in optimizing routing decisions under different network conditions. By systematically evaluating these studies against established criteria, the review aimed to present a clear overview of the current state of research in GNN-based routing optimization, highlighting both the advancements made and the challenges that remain. In summary, 36 studies are included in this survey, including 16 journal publications, 17 conference publications, and three preprint papers; the first relevant study was published in 2018.

In this study, the grouping of relevant literature on GNN-based routing optimization was primarily based on three criteria: the type of learning approach employed, the specific routing objectives addressed, and the characteristics of the network scenarios examined. The studies were categorized into three main groups: supervised learning for network modeling, supervised learning for routing optimization, and reinforcement learning for routing optimization. This classification allows for a structured understanding of how different GNN methodologies are applied across various contexts and objectives [45]. Besides these three main categories, semi-supervised network modeling and routing optimization tasks are also mentioned in the literature, with only a few studies. For example, a semi-supervised learning approach with graph convolutional networks is proposed to estimate communication delays between node pairs in large-scale communication networks [46,47]. Considering the limited number of relevant studies, semi-supervised learning is not listed as a primary category in this paper.

The main objective of this survey is to provide an overview of the promising aspects of GNN-based solutions for routing optimization, over traditional and ML techniques. Moreover, a systematic guideline for applying GNNs for routing optimization and a summary of learned lessons and research opportunities for future research can be found in this survey. The specific contributions of this survey are summarized as follows. The novelty of this survey is that it is the first survey of GNNs for routing optimization to the best of our knowledge. This survey answers the research question of whether GNN-based routing optimization is more effective than traditional solutions and the answer is yes.

- This survey provides an up-to-date literature review of GNN techniques for routing, which were performed by a diverse group of experts in a wide range of application domains.
- This survey presents an introduction to ML and GNN basics to help researchers who want to kick-start the relevant studies.
- This survey classifies the most recent works in the past four years (i.e., 2018–2022) within the scope of three main categories, namely, supervised learning for network modeling, supervised learning for routing optimization, and reinforcement learning for routing optimization.
- This survey analyzes the existing studies carefully, covering the proposed solution, GNN techniques involved, routing policy, and performance.
- This survey proposes a set of research challenges and opportunities for future research. As applying GNNs to routing problems appeared only a few years ago, it is still a relatively new field, there are many research opportunities in this research topic.

The remainder of this paper is organized as follows. Section 2 introduces the basic concepts of routing, machine learning and graph neural networks. Section 3 discusses relevant studies in the category of supervised learning for network modeling. Section 4 discusses relevant studies in the category of supervised learning for routing optimization. Section 5 discusses relevant studies in the reinforcement learning category of routing optimization. Section 6 presents the relevant academic resources, including open datasets and programming tools, which can be further employed by interested scholars. The challenges and opportunities for inspiring follow-up studies are summarized in Section 7. Finally, Section 8 concludes the paper.

The abbreviations of the terminologies used in this survey are summarized in Table 2.

Table 2. The abbreviations and the corresponding full names used in this survey.

Abbreviation	Full Name
A2C	Advantage Actor Critic
AI	Artificial Intelligence
BGP	Border Gateway Protocol
CCN	Content-Centric Network
DDPG	Deep Deterministic Policy Gradient
DGATR [48]	Deep Graph Attention Network Routing
DGL	Deep Graph Library
DGRL [49]	Deep Graph Reinforcement Learning
DL	Deep Learning

Table 2. *Cont.*

Abbreviation	Full Name
DQN	Deep Q-Network [50]
DRL	Deep Reinforcement Learning
ENERO [51]	EfficieNt rEal-time Routing Optimization
FANET	Flying Ad Hoc Network
FCT	Flow Completion Time
GAT	Graph Attention Network
GCN	Graph Convolutional Network
GG-NN	Gated Graph Neural Network [52]
GN	Graph Network
GNN	Graph Neural Network
GQNN	Graph-Query Neural Network [53]
IBN	Intent-Based Networking
IoT	Internet of Things
LEO	Low Earth Orbit
MANET	Mobile Ad Hoc Network
MAPE	Mean Absolute Percent Error
ML	Machine Learning
MPNN	Message Passing Neural Network
MPTCP	Multipath TCP
MRE	Mean Relative Error
OSPF	Open Shortest Path Protocol
PDR	Packet Delivery Ratio
PyG	PyTorch Geometric
QoS	Quality of Service
RIP	Routing Information Protocol
RL	Reinforcement Learning
RSA	Routing and Spectrum Assignment
SAGIN	Space-Air-Ground Integrated Network
SDN	Software Defined Networking
SDR	Software Defined Router
SLA	Service Level Agreement
S-LRR	Sequential Link-Reversal Routing
TCP	Transmission Control Protocol
UAV	Unmanned Aerial Vehicle
WAN	Wide Area Network
WSN	Wireless Sensor Network

2. Basics

2.1. Routing Basics

Routing is a process of transmitting data packets from a source node to a destination node. A node can be a server or user device in a communication network. Routing is widely considered and applied to the networking layers. This process involves two steps in general, namely, optimal path selection and data forwarding. The former is the key problem in designing a routing algorithm, and the latter is typically performed by a router based on the corresponding routing policy.

As shown in Figure 1, routing optimization methods have evolved significantly since their inception, starting with classical algorithms like Dijkstra's and Bellman–Ford in the 1950s and 1960s, which provided foundational techniques for finding the shortest paths in graphs. The 1980s introduced heuristic methods, such as the A* algorithm, which utilized heuristics to enhance search efficiency. The 1990s saw the rise of metaheuristic approaches, including Genetic Algorithms and Simulated Annealing, which employed evolutionary and probabilistic strategies to solve more complex routing problems. In the 2000s, techniques like Particle Swarm Optimization and Tabu Search emerged, inspired by natural phenomena and memory-based strategies. The 2010s marked the incorporation of machine learning, with neural networks and reinforcement learning being applied to optimize routing dynamically. Today, hybrid methods that combine various techniques including DNNs and DRLs are increasingly prevalent, addressing the growing complexity and variability of real-world routing challenges.

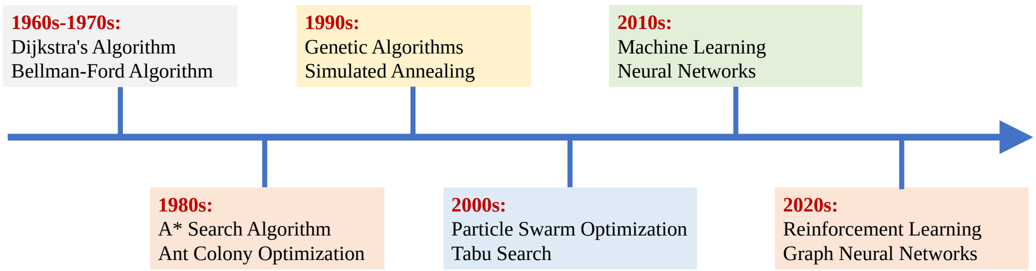

Figure 1. The timeline of routing optimization methods.

Existing routing optimization methods are interconnected through their foundational principles and strategies, often influencing one another to enhance performance and applicability. Classical algorithms like Dijkstra's and Bellman–Ford provide essential frameworks for shortest-path calculations, while heuristic methods like A* build on these foundations by incorporating heuristics to improve search efficiency in more complex scenarios. Metaheuristic techniques, such as Genetic Algorithms and Simulated Annealing, expand on these concepts by applying natural and probabilistic principles, allowing for effective solutions to difficult optimization problems. Additionally, methods like Tabu Search and Particle Swarm Optimization introduce memory and swarm intelligence concepts, respectively, showcasing diverse strategies that complement traditional approaches. In recent years, machine learning methods, including neural networks and reinforcement learning, have emerged, often integrating with existing algorithms to create adaptive and efficient routing solutions, highlighting a trend toward hybridization in optimization techniques.

Based on different criteria, existing routing solutions and protocols can be classified into different types, for example, central versus distributed, deterministic versus probabilistic, and static versus dynamic [7]. A comparison between centralized and distributed routing schemes is shown in Figure 2. The advantages of centralized routing include global optimization and control abilities, while the disadvantages include higher time and traffic overhead for real-time network status collection. The high decision delay for obtaining a global optimal routing solution as the network scale increases is both impractical and unacceptable. In distributed routing, each router makes its own decision based on its local information; thus, the decision-making time overhead is low, and some recent progress has been made to further improve the distributed routing efficiency, for example, multi-agent reinforcement learning algorithms. However, distributed routing has difficulty achieving a globally optimal solution.

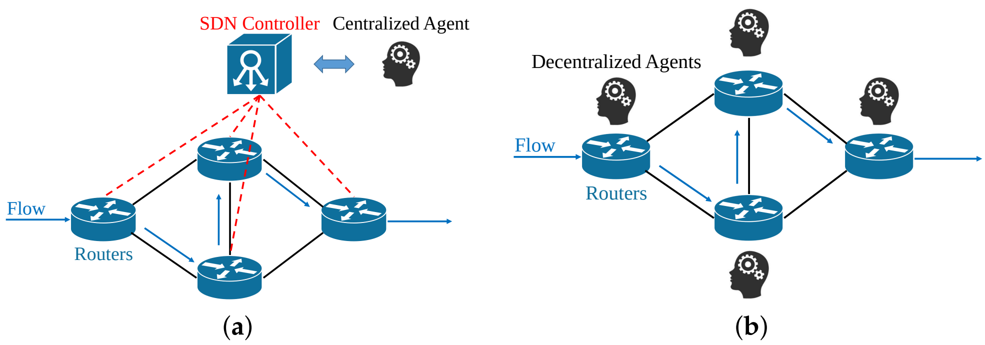

Figure 2. The comparison between centralized and distributed routing schemes [10]. (**a**) Centralized routing; (**b**) Distributed routing.

Some classical distributed routing protocols include distance vector routing protocols, such as Routing Information Protocol (RIP) and BGP, and link state routing protocols, such as OSPF. Link state routing is generally based on the shortest path first idea and consumes more resources (e.g., computer memory and computation units) than distance vector routing. However, it is free from the count-to-infinity problem and performs better when the network size is small. Although these classical routing protocols have already been applied in real-world networks, for example, the Internet, they are not perfect. An example of the suboptimal routing decision made by the OSPF is shown in Figure 3, in which the traditional shortest path-based routing algorithm assigns all traffic to the bottleneck link when the available bandwidth (e.g., 100 Mbps) of the selected path is much smaller than the service demand (e.g., 500 Mbps) and a smarter route with two paths and a traffic split exists.

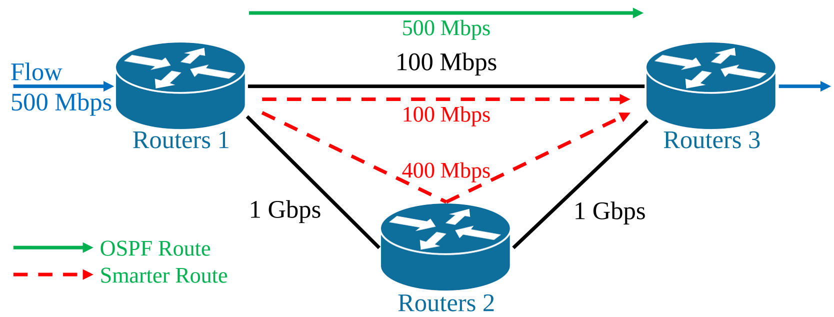

Figure 3. An example of the suboptimal routing decision made by OSPF [54].

More routing solutions emerge with the development of SDNs, including both centralized and distributed ones, facilitated by the ability to capture the real-time link status and perform global optimization. The introduction of software-defined routers (SDRs) allows complex routing operations to be integrated into programmable routers [1]. Currently, sophisticated intelligent routing schemes can be deployed in centralized controllers or distributed smart routers such as RouteFlow, which provides virtualized IP routing services over OpenFlow enabled hardware (http://cpqd.github.io/RouteFlow/, accessed on 22 September 2024).

Several metrics are widely adopted to evaluate and compare different routing schemes, including packet delivery ratio (PDR), transmission delay, throughput, and routing overhead [20].

- PDR is defined as the ratio of successfully transmitted packets between the source and destination nodes.
- Transmission delay is defined as the transfer time from the source to the destination.
- Throughput is defined as the multiplication of the packet size and the data packet number in a unit of time.
- Routing overhead is defined as the ratio of the control packet number, e.g., route discovery and maintenance packets, to the total transmitted data packet number.

2.2. Machine Learning Basics

Traditional routing algorithms are mainly based on the idea of finding the shortest path based on delay, distance, or hop count, without considering the real-time status of links. They are prone to problems including the local congestion of network links, low link utilization ratio, and waste of network resources. To overcome these problems, some manual routing policy optimization solutions have been proposed, for example, manual routing configuration optimization or switching among pre-defined routing configuration parameter sets for typical traffic scenarios. However, these solutions are neither effective nor efficient, and their routing performance deteriorates significantly in the worst case. ML techniques, especially deep learning (DL), are introduced to optimize the routing strategy automatically, which learns a desired routing configuration for future cases based on the knowledge of past conditions [1,55].

Another challenge for traditional routing algorithms is traffic bursts, which overwhelm physical links in a short time and cause traffic congestion, frequent packet losses and increased end-to-end delays. Traffic bursts arise for different reasons, such as protocol-side (e.g., different TCP window sizes caused by the congestion control mechanism) or application-side (e.g., different data sizes used for different frames in video streaming). Most of the existing routing protocols are traffic-free and lack adaptation ability under different traffic scenarios. Centralized routing algorithms, which are mainly based on linear programming methods, require a long computational time overhead, making it infeasible to handle traffic bursts. ML-based routing solutions can overcome this problem and adapt to traffic bursts quickly because the inference time of the trained ML models is negligible compared with that of traditional solutions. For example, the inference time of the trained ML models is about 10 s, and traditional solutions require more than 1000 s [56].

The third concern regarding traditional routing algorithms is the limitation of their best-effort optimization ability for finding the shortest path as the only objective, which cannot meet the differentiated QoS requirements of new network services, such as multimedia and remote cloud services. Nonetheless, more routing optimization objectives can be adopted in ML-based routing schemes, such as data packet reachability, algorithm scalability, latency, bandwidth, throughput, packet loss rate, and network stability.

Two specific ML-based approaches are widely used for routing optimization: supervised learning and reinforcement learning. In a supervised learning approach, input and output samples are collected and used to train a model so that the model can accurately complete a class of machine learning tasks that are mapped from input to output. The general framework for applying supervised learning to routing optimization is shown in Figure 4, in which the network topology and network state information are the inputs, and the routing decisions are the outputs.

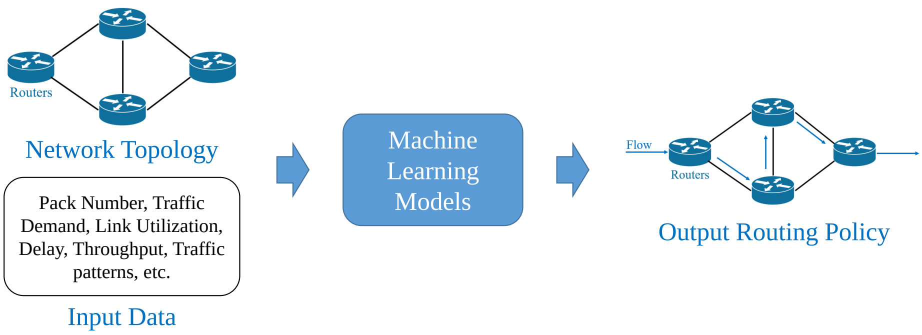

Figure 4. The general framework of applying supervised learning for routing optimization [10].

Supervised learning methods rely on many labeled data samples. In routing scenarios, it is difficult to obtain a large number of labeled samples within a short period. Fortunately, the reinforcement learning approach does not rely on a training process using historical data. The performance of an RL agent gradually improves in the process of continuous interaction with the environment, based on the reward as feedback to update the model parameters. The general framework for applying reinforcement learning for routing optimization [57] is shown in Figure 5. Reinforcement learning strategies further extend the capabilities of GNNs by enabling adaptive routing policies based on feedback from the environment [58]. In the context of routing optimization, RL algorithms operate on the principle of trial and error, where agents learn to make routing decisions by interacting with the network and receiving rewards or penalties based on the outcomes of their actions. For instance, techniques like Deep Q-Learning (DQN) and Proximal Policy Optimization (PPO) allow the model to learn optimal routing paths by continuously updating its policy based on real-time network performance metrics, such as end-to-end delay and throughput [59].

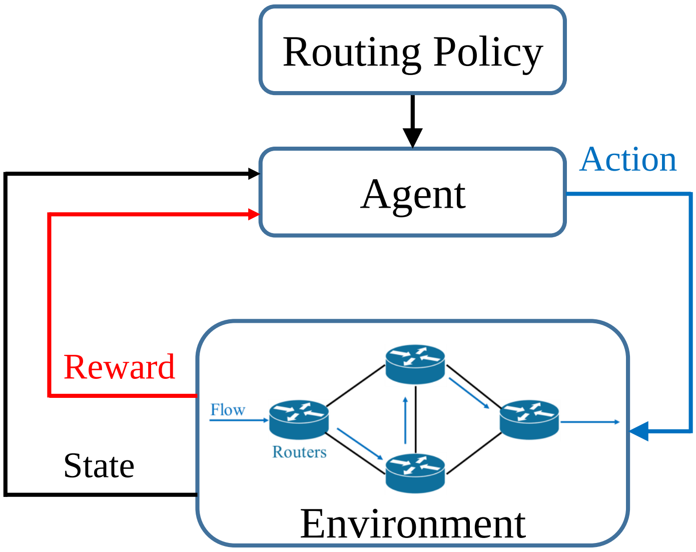

Figure 5. The general framework of applying reinforcement learning for routing optimization.

Each router or SDN controller is an agent with a different routing policy. The routing policy defines the routing optimization objective, for example, delay minimization or throughput maximization. The action is the specific operation to achieve the corresponding optimization objective, for example, the next hop selection probability or the traffic split ratio for multi-path routing, which is typically the output of a neural network model used by the agent. The state is the network status history data, e.g., the link utilization ratio in the past few rounds, which is usually the input of the neural network model. The reward is direct feedback from the network environment to the agent, which is usually a mapping function from different network metrics to a single value. The agent updates the model parameters based on the reward obtained after performing an action and interacting with the environment. RL-based routing optimization is a sequential decision process, and the action in the current round directly affects the state and action of the next round. Thus, the design of the cumulative reward function is essential for tuning an effective agent [60].

2.3. Graph Neural Network Basics

In intelligent routing schemes, local or global topology information of the underlying network is important for making routing decisions. However, owing to network topology complexity and dynamicity, traditional machine learning models defined in Euclidean space often have difficulty processing network topology information well. GNNs are novel neural network structures proposed in recent years that can effectively deal with the challenge of topological information extraction. The node and edge features are represented as vectors in the GNNs and updated in each round based on the topological dependencies and update function. GNNs have proven effective for various tasks, for example, network topology information extraction and link prediction, with good scalability and generalization performance. For routing problems, GNNs can learn the complex relationship between the network topology, traffic demand, and routing policy to generate accurate estimates of delay distribution and loss for the source/destination [61].

In this part, a brief introduction to representative GNNs is provided. More discussion on GNNs can be found in recent surveys [62–66]. GNNs [67] are based on the message-passing mechanism, which updates the state of a particular node with information from its neighbors. Subsequently, the convolution operation is introduced into graph convolutional networks (GCNs), and two families of GNNs are developed: spectral-based GCNs and spatial-based GCNs.

Spectral-based GCNs define the convolution operation in the spectral domain based on graph signal processing techniques and are successful with many well-known GCN variants, such as ChebNet [68], GCN [69], CayleyNet [70], and AGCN [71]. To provide the mathematical formulation of the GCN, the following notations are introduced. A graph is denoted as $G = (V, E)$, where V is the set of nodes and E is the set of edges. The network topology is denoted as the adjacency matrix $\mathbf{A}$. When there is an edge $e_{ij} \in E$ between node i and node j, and the element $A_{ij} = 1$. Otherwise, $A_{ij} = 0$. The degree matrix $\mathbf{D}$ measures the number of neighbors for each node, i.e., $D_{ii} = ||\mathcal{N}(v_i)||$, where $\mathcal{N}(v_i)$ is the neighbor node set of node v_i. The node feature matrix is $\mathbf{X} \in R^{N \times d}$, where d is the feature dimension and N is the node number. The Laplacian matrix is $\mathbf{L} = \mathbf{D} - \mathbf{A}$ and its normalized variant is $\tilde{\mathbf{L}} = \mathbf{I}_N - \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}$, where $\mathbf{I}_N$ is the identity matrix of size N. The graph convolution operation $*G$ in GCN is defined as follows:

$$\mathbf{X}_{*G} = \mathbf{W}(\tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}})\mathbf{X} \tag{1}$$

where $\mathbf{W}$ is the learnable model parameter, $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}_N$ and $\tilde{\mathbf{D}}_{ii} = \sum_j \tilde{\mathbf{A}}_{ij}$.

Spectral-based GCNs cannot handle directed graphs and have low scalability in most cases, with some exceptions [72,73]. Spatial-based GCNs are more flexible and more general. By defining the corresponding update, message passing and readout functions, different spatial-based GCN variants can be represented in a unified form of a message-passing neural network (MPNN) [74], for example, PATCHY-SAN [75] and DCNN [76].

The two stages run iteratively in the MPNN, namely, a message-passing phase and a readout phase. The message-passing phase is defined as follows:

$$\mathbf{m}_{v_i}^{(t)} = \sum_{v_j \in \mathcal{N}(v_i)} \mathcal{M}^{(t)}(\mathbf{X}_i^{(t-1)}, \mathbf{X}_j^{(t-1)}, \mathbf{e}_{ij}) \tag{2}$$

where $\mathbf{m}_{v_i}^{(t)}$ is the message aggregated from the neighbors of node v_i, $\mathcal{M}^{(t)}(\cdot)$ is the aggregation function in the tth iteration, $\mathbf{X}_i^{(t)}$ is the hidden state of node v_i in the tth iteration, and $\mathbf{e}_{ij}$ is the edge feature vector between node v_i and node v_j. The readout phase is defined as follows:

$$\mathbf{X}_i^{(t)} = \mathcal{U}^{(t)}(\mathbf{X}_i^{(t-1)}, \mathbf{m}_{v_i}^{(t)}) \tag{3}$$

where $\mathcal{U}^{(t)}(\cdot)$ is the readout function in the tth iteration.

In spectral-based GCNs, the filter depends on the Laplacian matrix, which is derived from the graph structure. Therefore, the model trained on a specific graph cannot be directly applied to other graph structures. To solve this problem, the graph attention network (GAT) [77] introduces an attention mechanism based on a graph convolutional network that enables the model to focus on the most relevant information. GAT has lower complexity and only focuses on adjacent nodes, without depending on the information from the entire graph. When applied to a new graph, GAT does not need to repeat the training model. The multi-head attention mechanism with K heads is leveraged in the propagation step in the GAT, which can be denoted as follows:

$$\mathbf{X}_i^{(t)} = ||_k \sigma(\sum_{j \in \mathcal{N}(v_i)} \alpha^k(\mathbf{X}_i^{(t-1)}, \mathbf{X}_j^{(t-1)})\mathbf{W}^{(t-1)}\mathbf{X}_j^{(t-1)}) \tag{4}$$

where $||$ is the concatenation operation, σ is the activation method, and $\alpha^k(\cdot)$ is the k-th attention mechanism. More GAT variants have been developed for more complex graphs, such as heterogeneous GAT [78] and dynamic GAT [79].

In routing optimization, different GNN architectures such as GCN, GAT, and GraphSAGE have been employed, each offering distinct advantages depending on the network conditions and tasks [80]. GCNs, which operate by applying spectral convolutions over graph structures, are effective for tasks where global network information needs to be captured, such as predicting delay or congestion across a broad topology. However, GCNs often struggle with scalability and handling dynamic, real-time updates in large networks [81]. GATs, which use attention mechanisms to focus on the most relevant neighbors during message passing, perform better in environments with high variability, such as mobile or UAV networks, where certain connections are more critical than others [82]. This ability to dynamically weigh edge importance allows GATs to excel in networks with frequent topology changes. GraphSAGE, which generates node embeddings by sampling and aggregating features from a node's neighborhood, is particularly advantageous for large-scale networks where computational efficiency is crucial. It performs well in tasks like traffic prediction and load balancing by leveraging inductive learning, enabling it to generalize effectively to unseen nodes or topologies without retraining [83]. Overall, the choice of GNN architecture depends on the specific routing task: GCNs are suited for static or semi-static conditions, GATs for highly dynamic networks, and GraphSAGE for large-scale, resource-constrained environments requiring real-time adaptability.

Scalability is a critical concern when applying GNNs to large-scale networks, particularly those with millions of nodes and edges. As network size increases, so do the challenges in terms of training time, computational resources, and inference efficiency. GNN models, such as GCNs, struggle with scalability because the graph convolution operations involve all neighbors of a node, which leads to an exponential growth in computations for deep networks [84]. This issue is compounded by the need to update node embeddings iteratively, which can make training prohibitively slow on large graphs. Moreover, memory consumption becomes a bottleneck, especially when storing feature vectors and adjacency matrices for large graphs. While architectures like GraphSAGE and GAT introduce techniques such as neighborhood sampling and attention mechanisms to mitigate computational complexity, they still face scalability limitations in real-world, high-throughput environments [85]. Inference time, especially for real-time applications like dynamic routing in large networks, can also be delayed, making it difficult to meet the stringent latency requirements of many network systems. To address these issues, efficient sampling methods, distributed training techniques, and hardware acceleration (e.g., using GPUs or specialized processors) are being explored [86], but GNN scalability remains a key challenge for their deployment in large-scale, real-time network optimization tasks.

Explainability in GNN-based routing models is a significant concern, especially in critical network operations where transparency is essential for trust and reliability. GNNs, like other deep learning models, function as black boxes, making it difficult for network administrators to interpret why specific routing decisions are made [87]. This opacity poses challenges in understanding how the model prioritizes certain paths, handles traffic bursts, or reacts to sudden topology changes. To address this, researchers are exploring techniques such as attention mechanisms in models like GAT, which offer some degree of interpretability by highlighting the most influential nodes and edges during decision-making [88]. However, even with these techniques, the intricate relationship between input features (e.g., topology, traffic patterns) and output routing policies can still be challenging to decipher. For network administrators to confidently deploy GNN-based systems, it is crucial to integrate post-hoc explainability tools such as SHAP or LIME, which can provide insight into how specific features influence routing decisions. Additionally, incorporating rule-based constraints or hybrid approaches, where GNN models suggest decisions while traditional algorithms validate them, can enhance transparency and ensure that critical operations, such as failover mechanisms or load balancing, remain interpretable and auditable in real-time.

The research on GNNs for routing optimization is highly relevant to the development and deployment of 5G and 6G networks, which are characterized by their need for ultra-low latency, high reliability, and massive connectivity. As these next-generation communication networks evolve, they face complex challenges such as dynamic topologies, variable traffic loads, and the integration of numerous devices, including IoT and edge computing systems. GNNs offer a promising solution by effectively modeling the intricate relationships between network nodes and edges, allowing for real-time adaptive routing that can respond to changing conditions. Additionally, GNNs can leverage historical traffic data to optimize routing policies, enhancing the overall performance of 5G and 6G networks. Their ability to generalize across unseen topologies and their capacity for continuous learning make GNNs particularly suitable for handling the demands of high-mobility environments, such as those encountered in vehicular networks or drone communications. Overall, the application of GNNs in routing optimization is poised to significantly enhance the performance, scalability, and resilience of future 5G and 6G networks, enabling them to meet the stringent requirements of emerging applications and services.

Lastly, understanding complex network topologies is essential for effectively applying GNNs and RL strategies [89]. These topologies can vary widely in structure, including dynamic environments where nodes frequently change their positions (such as UAV networks) or those characterized by highly interconnected nodes (like mesh networks) [90]. GNNs excel in these scenarios because they can naturally model the interdependencies between nodes and adapt their learning based on the unique relationships defined by the graph structure. This adaptability is particularly important for routing optimization in complex networks, where traditional algorithms may struggle to provide efficient solutions due to their inherent rigidity and reliance on predefined heuristics [91].

3. Supervised Learning for Network Modeling

3.1. Overview

In the traditional network modeling scheme, many assumptions are required to simplify the problem and achieve an efficient solution process using off-the-shelf mathematical tools under specific application scenarios. However, these ideal assumptions are rarely met in reality, making the follow-up optimized solutions invalid when deployed in real-world scenarios. As networks continue to grow in size and complexity, traditional network modeling approaches are becoming more cumbersome.

ML-based network modeling overcomes this problem by collecting real-world network data and training/updating ML models without making complex and unrealistic assumptions. When a model is trained well, it can be deployed in production with fast reasoning ability, usually of polynomial time complexity. Supervised learning algorithms can be applied to routing problems to improve the dynamic response capability of routing by predicting essential network status information, for example, the traffic matrix and link load.

In this section, we focus on the supervised learning approach for network modeling with GNNs, which is highly connected to routing optimization. The GNN model first predicts network performance under different routing configurations in a supervised learning approach. Then, the network administrator chooses the optimal routing configuration based on GNN predictions.

3.2. Literature Review

RouteNet [30,61,92,93] is the first GNN network model based on the MPNN, with the optimization goal of minimizing per-source/destination average delay and/or jitter, with the input of the network topology, traffic matrix, routing scheme, and output of the performance metrics. Based on the packet-level simulator OMNeT++, simulated datasets are generated using real-world computer network topologies and used to train the GNN model. The topologies used in the training and test stages are different for evaluating the model performance in the unseen cases. RouteNet can predict the delay distribution (mean delay and jitter) and loss accurately; even when information regarding the topologies, routing and traffic is unavailable in the training, it still achieves a worst-case mean relative error (MRE) of 15.4%. The predictions of RouteNet are further leveraged in a network planning use case to select optimal link placement. RouteNet (including its improved variants) and the simulated dataset are further used in the Graph Neural Networking Challenge 2020 as one of the open global challenges for ITU AI/ML in the 5G challenge [94].

Additional GNN-based network models are further developed following the basic ideas of RouteNet. A new GNN model is proposed to predict the per-path mean delay based on the input topology, routing configuration, queue scheduling policy, and traffic matrix [95]. Compared to RouteNet, the new GNN model adds support for different queuing policies and produces more accurate delay estimates in cases with complex queue scheduling configurations. To solve the problem that the generalization of RouteNet suffers greatly when predicting on super large graphs, the invariant features from the analytical queueing theory approach are extracted and fed into the GNN-based model QT-Routenet [96]. The traffic intensity and probability of being in state zero are calculated using the analytical baseline which is derived from queueing theory and used as new features. QT-RouteNet outperforms both RouteNet and the analytical baseline, which reduces the analytical baseline's 10.42 mean absolute percent error (MAPE) to 1.45 (1.27 with an ensemble). QT-RouteNet achieved first place in the GNNet Challenge 2021. RouteNet-Erlang [97] extends the input of RouteNet with multi-queue scheduling policies and outperforms all queueing theory baselines under several different traffic models with a worst-case delay prediction error of 6%.

Based on a graph network (GN) [98], a GNN model is used to determine the flow completion time (FCT) in [99], which infers FCT statistics directly from the network topology and flow matrix in real-time. The estimation of unseen network states is then used for traffic optimization, including inflow routing, flow scheduling, and topology management, with a GNN-based optimizer that explores the optimal configuration based on both the network's state and the administrator's target. The proposed solution significantly reduces the flow completion time.

The multipath TCP (transmission control protocol) is considered in [100] for an SDN-based 5G network, and an MPNN-based GNN model is proposed to predict the expected throughput given the network topology and multipath routes. A topology explorer is responsive to maintaining the network topology information and three key features, namely bandwidth, delay, and packet loss rate of a link, as used to model the input. A routing generator is responsible for generating potential routing schemes using random or greedy algorithms based on the latest network topology and transmission demand. The GNN model is trained offline and deployed online to evaluate the performance of different routing schemes with the expected throughput as the metric. A decision maker chooses the optimal MPTCP scheme and sends the configurations to network devices, for example, routers and switches. The proposed approach is compared with a globally optimal baseline, which is computationally unrealistic in practice, and a traditional MPTCP full-mesh algorithm. The proposed approach outperforms the full-mesh algorithm and approaches the performance of the globally optimal solution in terms of throughput.

An intent-based networking (IBN) solution is proposed in [101] using RouteNet and LSTM models for optimized service path routing and computation resource prediction, respectively. Specifically, RouteNet is used for link utilization prediction. The proposed RouteNet-based IBN solution with end-to-end orchestration is successfully deployed for 8K and 4K video streaming services.

A GNN-based multipath routing scheme is proposed in [102] for SDN to achieve a balance between flow transmission granularity, reordering, and end-to-end transmission efficiency improvement. The proposed scheme includes a route planning module, state information collection module, delay prediction module and adaptive flow splitting scheme, in which the delay prediction module is based on the MPNN. The proposed approach outperforms other baselines including the shortest path, per-flow routing, per-packet routing, and original flowlet routing schemes, in terms of time overhead, end-to-end delay, flow completion time, and throughput.

An MPNN-based link delay model is proposed in [103], using the domain knowledge of network behaviors. The logic behind the proposed link delay model is that the end-to-end delay can be reflected by some typical network behaviors including jitter, packet loss, and throughput. Driven by this idea, an improved GNN model is proposed to aggregate messages in the modeling process. Both traditional Queuing model and MPNN-based RouteNet are used as baselines and the proposed approach outperforms Queuing model and RouteNet with an increased R^2 by 73% and 11%, respectively. The generalization ability of the proposed approach is further validated, which achieves a lower mean relative error under an unknown flow scheduling strategy.

The relevant studies with supervised learning for network modeling are summarized in Table 3.

Table 3. The summary of studies with supervised learning for network modeling.

Study	Scenario	Modeling Target	Proposed Solution	Performance
[30,61,92,93]	Computer network	Per-source/destination pair mean delay, jitter and packet loss ratio	RouteNet (MPNN-based)	RouteNet accurately predicts the delay distribution (mean delay and jitter) and loss even with topologies, routing and traffic unseen in the training (worst case MRE = 15.4%).
[95]	Computer network	Per-path mean delay	MPNN-based model	The model predicts the delay with an MRE of 3.88% in the unseen topology.
[96]	Computer network	Per-path mean delay	QT-Routenet (GCN and GAT-based)	The prediction MAPE is reduced to 1.45 (1.27 with an ensemble).
[97]	Computer network	Delay, jitter and packet loss ratio	RouteNet-Erlang (MPNN-based)	RouteNet-Erlang outperforms all queueing theory baselines under several different traffic models with a worst-case delay prediction error of 6%.
[99]	Datacenter network	Flow completion time	GN-based optimizer	The proposed solution can significantly reduce the flow completion time.
[100]	SDN-based 5G network	Expected throughput	MPNN-based GNN model	The proposed GNN model can predict the expected throughput of specific MPTCP connections with very low error.
[101]	5G network	Link utilization	RouteNet + LSTM	The proposed RouteNet-based IBN solution with end-to-end orchestration is successfully deployed for 8K and 4K video streaming services.
[102]	SDN	Delay	MPNN-based model	The proposed approach outperforms its baseline counterparts in terms of time overhead, end-to-end delay, flow completion time, and throughput.
[103]	SDN	Delay	MPNN-based model	The proposed approach outperforms Queuing model and RouteNet with an increased R^2 by 73% and 11%, respectively.

4. Supervised Learning for Routing Optimization

4.1. Overview

The general framework of applying GNNs in a supervised learning approach for routing optimization is shown in Figure 6. The GNN-based routing model is subsequently trained from offline supervised learning and then tested in the online deployment.

The supervised learning approach requires the collection of training data, which are often generated using traditional routing algorithms, for example, the shortest path and min-max routing. In this case, the performance of GNN-based solutions is upper-bounded by these traditional routing algorithms. However, this does not imply that GNNs have no benefits. In most cases, these traditional routing algorithms are deployed in a centralized mode, whereas some GNN-based solutions can be deployed in a distributed mode, which is more suitable for wireless networks. The fast online deployment and generalization abilities of unseen topologies are also the advantages of GNN-based solutions.

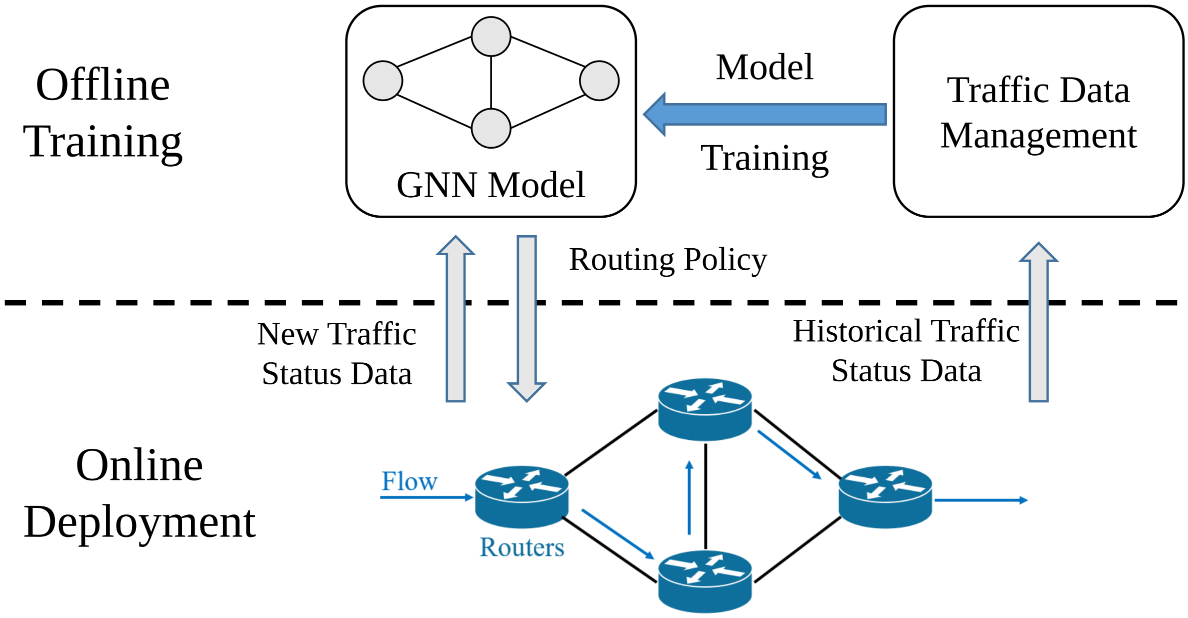

Figure 6. The general framework of applying GNNs in a supervised learning approach for routing optimization.

4.2. Literature Review

Early-stage studies validated the potential application of leveraging GNNs to learn the same routing policies in a distributed approach from heuristic centralized routing algorithms. The graph-query neural network (GQNN) [53] is the first study to apply GNNs to distributed routing protocols, in which the GNN model is used in each router to decide which output interface to use, given a destination router identifier. Temporal information propagation is modeled by a gated graph neural network (GG-NN) [52], and neighborhood interaction is captured by the edge attention mechanism. Path calculation results from two different routing strategies are employed here as learning objectives (to demonstrate the scalability of the proposed model), that is, shortest path routing based on Dijkstra's algorithm and min-max routing, which maximizes the minimum allocated bandwidth between all possible source-destination pairs in the network. The proposed GQNN achieves accuracies of 98% and 95% for the shortest path and min-max routing, respectively.

Subsequently, a similar study is conducted in [104], in which a GN-based model [98] is used to learn the routing route generated by the genetic algorithm. With bandwidth utilization maximization as the routing objective, the proposed model achieves 61.0% accuracy for predicting the routing table of the genetic algorithm, with a 150× faster prediction time.

A graph-aware deep learning-based intelligent routing strategy (GADL) [105] is proposed to predict the next forwarding node with a routing policy that minimizes the average end-to-end flow latency. The proposed graph-aware convolutional structure first extracts topological information from the graph, then processes the input data based on the extracted information, and finally applies a convolution to the processed data. GADL achieves an accuracy of 86.55% for predicting the next forwarding node and a lower average network latency than OSPF.

NGR [106] is the first deep-learning-based distributed routing system that aims to guarantee connectivity and avoid the self-loop problem in previous neural network-based routing solutions. A GNN [67] model is used to model spatial dependency and aggregate the feature vectors of neighboring nodes. A recurrent neural network is used to model the temporal dependency and update the feature vector for a node. Then, the feature vector and input packet ID are fed into a forwarding neural network. Finally, an S-LRR (sequential link-reversal routing) algorithm determines the forwarding port for the packet based on the value vector output from the forwarding neural network. The numerical experiments show that for shortest-path routing or load balancing, NGR achieves 100% routing reliability and gain performance close to the optimal solutions.

A GCN-based GLR is proposed in [107] to handle the highly dynamic topology and limited resource challenges in satellite networks. A high-order and low-order feature extractor and cross-process are well-designed to handle unseen topologies. Offline pre-training is introduced to reduce the on-board computation complexity. Instead of using the next node selection probability as the GNN output in previous studies, GLR uses the communication distance as the output, which is the hop count between the current and destination node. GLR outperforms the brute-force and shortest-path routing algorithms in terms of end-to-end transmission delay and packet drop rate. GLR is also more robust when facing link interruptions and has a lower routing computation cost.

The relevant studies with supervised learning for routing optimization are summarized in Table 4.

Table 4. Summary of studies with supervised learning for routing optimization.

Study	Scenario	Proposed Solution	Performance	Routing Policy	Deployment Mode
[53]	Computer Network	GQNN (GG-NN-based)	GQNN achieves accuracies of 98% and 95% for shortest path and min-max routing, respectively.	Shortest path or min-max fair routing	Distributed
[104]	Computer network	GN-based model	The proposed model achieves 61.0% accuracy for predicting the routing table of the genetic algorithm, with a 150× faster prediction time.	Bandwidth utilization maximization	Centralized
[105]	Computer Network	GADL (Graph-aware convolution-based)	GADL achieves an accuracy of 86.55% for predicting the next forwarding node and a lower average network latency than OSPF.	Latency minimization	Centralized
[106]	Computer network	NGR (GNN-based)	NGR achieves 100% routing reliability and gain performance close to the optimal solutions.	Shortest-path routing or load balancing	Distributed
[107]	Satellite network	GLR (GCN-based)	GLR outperforms brute-force and shortest path routing algorithms in terms of end-to-end transmission delay and packet drop rate.	Delay minimization	Centralized

5. Reinforcement Learning for Routing Optimization

5.1. Overview

Compared to supervised learning, reinforcement learning is more favorable for routing optimization because of its modest usage of memory and computational resources [11,108]. Many network nodes have limited energy and communication resources, for example, those in wireless sensor networks or satellite networks, which require a routing algorithm to achieve faster convergence to optimal decision-making.

By following the general RL scheme shown in Figure 5, the state space is defined using historical traffic status measurements, such as end-to-end delay, throughput, and energy efficiency, which are periodically collected by the agent from the network environment. In various studies, it can be formulated as a matrix or vector of different variables. The action space is defined as the routing policy performed by the agent. Examples of specific action choices include the split ratio for multipath routing and the weight matrix of individual links. GNN models are used within the agent to generate the routing decision. The reward function is defined as the optimization objective of the routing algorithms, for example, the different QoS parameters. Some examples of reward function design for routing include energy-saving optimization, end-to-end delay minimization, throughput maximization, and average flow completion time minimization. The reward function is important in RL-based routing solutions because it promotes solution convergence.

5.2. Literature Review

As a pioneering work on integrating DRL and GNN in network optimization, an MPNN-based deep Q-network (DQN) architecture is proposed in [109] to find the optimal routing configuration with a given traffic matrix and generalize over arbitrary network topologies. The GNN model is incorporated into the DRL agent to compute the Q value and the message-passing steps are briefly stated as follows.

1. The link features over its neighbors are combined for a single link with a fully connected neural network into messages.
2. The messages over its neighbors are aggregated for a single node with an element-wise sum.
3. The link hidden states are updated with the aggregated information with a recurrent neural network.
4. The resulting link states for T iterations are aggregated using an element-wise sum.
5. The q-value is the output of the readout function with a fully connected neural network.

The proposed architecture outperforms the state-of-the-art DRL algorithms without using GNNs on unseen network topologies.

Routing and spectrum assignment (RSA) in an elastic optical network are considered jointly in [110], in which a GCN is used to extract topology-related features and an RNN is used to extract path-related features. Then, the advantage actor critic (A2C) algorithm [111] is used to make the RSA decision, in which the reward function is designed as a binary value when a successful RSA decision (i.e., a successful provision of routing and spectrum assignment) generates a positive reward. The proposed approach achieves a lower blocking probability and exhibits good scalability, with different network topologies, bandwidth requirements, and traffic loads.

Another multi-task deep reinforcement learning framework is proposed in [112] for joint network slicing and routing in an SDN-based 6G network. A customized GCN is used to capture topological information from the graph-structure network status. The actor-critic algorithm is used in the DRL agent to minimize packet loss while simultaneously maximizing the link bandwidth utilization and the service level agreement (SLA) satisfaction ratio with different importance coefficients. Multiple metrics such as throughput, latency, and packet loss rate are used in the evaluation. The numerical experiments demonstrate that the GCN-based multi-task DRL outperforms other learning-based algorithms for joint network slicing and routing tasks and is robust to diverse network environments.

In AutoGNN [113], the MPNN is used to process traffic-related information collected by the SDN controller, and DRL is used to generate the action to be performed by the SDN controller. Numerical experiments demonstrate that AutoGNN improves the average end-to-end delay of the network by up to 19.7% and presents greater robustness against topology changes.

Based on the combination of the MPNN and the deep deterministic policy gradient (DDPG) algorithm, GRL-NET [114] obtains a lower transmission energy consumption and shows good generalization ability for unseen topologies. MPNN is used to model the graph structure in network topology and DDPG is used for routing under the possible failure of network nodes, which causes the network topology change.

GraphNET [115] is proposed to predict the optimal path in SDNs based on a combination of GNN and DQN. The GNN model is deployed in the SDN controller and selects the next node for each routing request received by the controller, in which the input is the hidden state matrix for all links, and the output is an expected reward vector for choosing each link. A deep Q-network is developed to train the GNN model with prioritized experience replay to maximize the expected reward and minimize the packet delay. Experiments on both small and large network topologies demonstrate that GraphNET outperforms q-routing without GNN and shortest-path routing algorithms in terms of packet delivery success ratio and average packet delay time. In addition, it is robust to changes in the network structure.

GDDR [116] is proposed to minimize link congestion in intradomain traffic engineering, based on a combination of GNN and DRL. Given the network topology in graph format and the traffic demand matrix as inputs, GDDR predicts a routing strategy with proximal policy optimization (PPO) as the DRL algorithm. The reward is defined as the maximum link utilization ratio between the DRL-based routing and the optimal routing solved using a linear solver named Google OR-Tools. GDDR achieves a lower maximum link utilization ratio than the multilayer perceptron-based baseline and shortest path routing, with the ability to generalize unseen network topologies during training.

Based on the MPNN and DQN, GRouting [117] is proposed to select the optimal routing path between two satellites in the fast-growing low Earth orbit (LEO) satellite network, with the routing objective of maximizing the utilization of satellite network resources while guaranteeing the requirement of transmission delay. The reward function is designed as the gain from successful transmission minus the delay penalty. The experiments show that GRouting outperforms four baseline algorithms, including the shortest path, random path, request balance, and DQN without GNN, in terms of throughput and shows better generalization for time-varying topologies.

Based on GAT and DQN, deep graph attention network routing (DGATR) [48] is a multi-agent routing framework that minimizes end-to-end delay. GAT is introduced for the first time to perform routing by leveraging the local information at each router with the graph attention mechanism. Both RL-based and non-RL-based algorithms are used as baselines in the experiments, including the shortest path, tabular Q-learning, a hybrid method of tabular Q-learning and policy gradient method, DQN routing without GNNs, and optimal global routing. DGATR outperforms other RL-based algorithms without GNNs in terms of packet transmission delay and affordable load. DGATR also shows better adaptability and can sustain a higher network load than other algorithms, except for the optimal global routing scheme. Three different learning paradigms are proposed and compared: centralized, federated, and cooperated learning [118]. The experiments show that while the three learning paradigms achieve indistinguishable maximum affordable loads, the cooperated learning approach achieves a lower delay and better stability because the variation in the parameter update is relatively small when parameter aggregation is restricted in cooperated learning.

A multi-task framework for joint routing and scheduling optimization in a time-sensitive network is proposed in [119] based on the combination of GCN and deep Q-learning. The GCN is leveraged to capture spatial dependency, and a priority experience replay is employed to accelerate the GCN training process. In the experiments, the proposed approach achieves good convergence and a lower end-to-end delay than baselines, including DQN-based, Q-learning-based, and shortest-path routing schemes.

A deep graph reinforcement learning (DGRL)-based framework is proposed in [49] for intelligent traffic control in a software-defined wireless sensor network. DGRL is based on the actor-critic framework, in which a GCN model is used in the actor network to extract and aggregate the state features of the current node and all its neighbors. Each wireless node optimizes its own transmission path and creates the best hop policy in an online training approach. The reward function is well-designed for multiple objectives, including a shorter forwarding time, shorter forwarding path, and lower buffer occupation. The proposed DGRL outperforms the OSPF and DRL baselines without GNNs in simulation experiments based on the OMNet++ platform, in terms of packet transmission delay, PDR, and network congestion probability.

An efficient real-time routing optimization (ENERO) solution is proposed in [51] for routing optimization in wide area networks (WANs) using a two-stage process. In the first stage, a GNN-based DRL agent is used to generate the initial routing solution. In the second stage, a local search algorithm is used to improve the solution without adding computational overhead to the optimization process. The proposed ENERO operates in real-world dynamic network topologies in 4.5 s on average for topologies up to 100 edges and outperforms the shortest available path heuristic baseline in terms of the link utilization ratio.

Although the GNN-based DRL framework has been proven effective in the above studies, as discussed, DRL scales poorly with the problem size and complexity, which is further considered in [120]. To solve this challenge, evolutionary strategies are introduced for the first time in the training process of DRL agents and help to speed up the training time by 128 and 6 times for two network topologies, namely, NSFNET and GEANT2, respectively.

With the joint minimization of end-to-end delay and packet loss rate as the routing objective, a routing optimization policy based on GCN and DDPG is proposed in [121], in which the GCN is responsible for network topology structure information extraction and DDPG is responsible for making routing decisions. The proposed approach outperforms OSPF algorithm, DRL-TE strategy, and DDPG routing algorithm in terms of average end-to-end delay and packet loss rate.

To enhance the robustness against topology changes, GAPPO is proposed for network routing in [122], which combines GAT and PPO. The state, action and reward in DRL are carefully designed and the experiments demonstrate a superior robustness performance of the proposed approach against different link failures, which outperforms benchmark algorithms with a lower packet loss ratio and a lower end-to-end delay.

The Actor-Critic architecture is further explored in [123] for network routing and GCN is used to update the link weight of the whole network. The proposed scheme outperforms baselines in terms of network average end-to-end delay, packet loss rate, and throughput.

Knowledge-defined networking is leveraged to handle the dynamic characteristics of the network topology in [124], in which message-passing deep reinforcement learning (MPDRL) is proposed for routing optimization. The message-passing mechanism in MPNN is used to extract exploitable knowledge and DRL is used to make routing decisions. Experiments show that MPDRL achieves the load balance of network traffic and improves network performance, compared with baseline methods.

To deal with the network topology changes, a GNN-based multi-agent DRL routing scheme is proposed in [125], in which the deployed multi-agent approach is capable of routing in dynamic network conditions without retraining. The proposed distributed approach is compared with traditional routing baselines as well as multi-agent learning without GNN, and experiment results show that the proposed achieves fewer flow set collisions.

The relevant studies with reinforcement learning for routing optimization are summarized in Table 5.

Table 5. Summary of studies with reinforcement learning for routing optimization.

Study	Scenario	Proposed Solution	Performance	Routing Policy	Deployment Mode
[109]	SDN-based optical transport network	MPNN + DQN	The proposed architecture outperforms the state-of-the-art DRL algorithms on unseen network topologies.	Traffic volume routed through the network maximization	Centralized
[112]	SDN-based 6G Network	GCN + Actor-Critic	The GCN-based multi-task DRL outperforms other learning-based algorithms for joint network slicing and routing tasks and is robust to diverse network environments.	link bandwidth utilization maximization, packet loss minimization, and SLA satisfaction ratio maximization	Centralized
[113]	Computer network	AutoGNN (MPNN + DRL)	AutoGNN improves the average end-to-end delay of the network by up to 19.7% and presents higher robustness against topology changes.	Delay minimization	Centralized
[114]	Wireless sensor network	GRL-NET (MPNN + DDPG)	GRL-NET obtains a lower transmission energy consumption and shows a good generalization ability on unseen topologies.	Transmission energy consumption minimization	Centralized
[110]	Elastic optical network	GCN + RNN + A2C	The proposed approach achieves a lower blocking probability and a better generalization ability.	Service blocking probability minimization	Centralized
[115]	Software-defined network	GraphNET (GNN + DQN)	GraphNET outperforms q-routing without GNN and shortest path routing algorithms in terms of packet delivery success ratio and average packet delay time and is robust to network structure changes.	Delay minimization	Centralized
[116]	Computer network	GDDR (DNN + PPO)	GDDR achieves a lower maximum link utilization ratio than the multilayer perceptron-based baseline and shortest path routing.	Link congestion minimization	Centralized
[117]	LEO satellite network	GRouting (MPNN + DQN)	GRouting outperforms four baseline algorithms in terms of throughput.	Throughput maximization with delay guarantee	Centralized
[48]	Computer network	DGATR (GAT + DQN)	DGATR outperforms other RL-based algorithms without GNNs in terms of packet transmission delay and affordable load.	Delay minimization	Centralized, federated, and cooperated

Table 5. *Cont.*

Study	Scenario	Proposed Solution	Performance	Routing Policy	Deployment Mode
[119]	5G network	GCN + Deep Q-learning	The proposed approach achieves a lower end-to-end delay than baselines.	Delay minimization	Centralized
[49]	Software-defined wireless sensor network	DGRL (GCN + Actor-Critic Network)	DGRL can effectively reduce packet transmission delay, increase PDR, and reduce the probability of network congestion.	Delay minimization, PDR maximization, and congestion minimization	Distributed
[51]	Wide area network	GNN + DRL	ENERO operates in real-world dynamic network topologies in 4.5 s on average for topologies up to 100 edges and outperforms the shortest available path heuristic baseline in terms of link utilization ratio.	Link utilization maximization	Centralized
[120]	SDN-based optical transport network	GNN + PPO	The introduction of evolutionary strategies helps to speed up the training time by 128 and 6 times for two network topologies, namely, NSFNET and GEANT2, respectively.	Traffic demand allocation maximization	Centralized
[121]	SDN Network	GCN + DDPG	The proposed strategy outperforms OSPF algorithm, DRL-TE strategy, and DDPG routing algorithm in terms of average end-to-end delay and packet loss rate.	Delay minimization and packet loss rate minimization	Centralized
[122]	Computer networks	GAPPO (GAT + PPO)	GAPPO outperforms benchmark algorithms with a lower packet loss ratio and a lower end-to-end delay.	Delay minimization	Centralized
[123]	SDN network	DGL-Routing (GCN + Actor-Critic)	The proposed scheme outperforms baselines in terms of network average end-to-end delay, packet loss rate, and throughput.	Delay minimization	Centralized
[124]	Computer Networks	MPDRL (MPNN + DRL)	MPDRL achieves the load balance of network traffic and improves network performance.	Network load balance	Centralized
[125]	Computer Networks	GCN + Multi-agent DRL	The proposed method achieves a better performance in terms of various QoS metrics	Flow set collision minimization	Distributed

6. Datasets and Tools

6.1. Overview

In GNN-based routing optimization studies, datasets typically consist of traffic matrices and network topology information from real-world networks such as Abilene, GÉANT, and CERNET, which are publicly available. Each traffic matrix element reflects the amount of data transmitted between node pairs at regular intervals (e.g., every 5 to 15 min). These datasets are critical for training GNN models to predict network performance under varying conditions, including different routing configurations, traffic demands, and topologies. To adapt these datasets for different network types, such as satellite or UAV networks, synthetic or simulation-generated datasets are often used, incorporating domain-specific constraints like mobility, energy limitations, and dynamic topology changes. The characteristics of these datasets, particularly their scale, traffic patterns, and topology complexity, significantly impact GNN performance. High-quality datasets with diverse topologies and traffic conditions help improve the model's generalization ability, allowing GNNs to perform well in unseen or time-varying network environments. Conversely, limited or homogeneous datasets may restrict the model's capability to handle real-world variability, leading to suboptimal routing predictions in complex networks.

Benchmarking practices in GNN-based routing optimization studies typically involve comparisons with traditional routing algorithms such as Dijkstra's shortest path algorithm, OSPF, and other heuristic-based or ML-driven techniques. These comparisons assess the ability of GNN models to outperform conventional approaches in key performance metrics like latency, throughput, and energy efficiency. Studies often use simulated or real-world datasets to evaluate how well GNNs handle diverse network scenarios, including unseen topologies or dynamic environments. Latency minimization, throughput maximization, and energy efficiency—especially in resource-constrained networks like UAVs or satellite systems—are critical performance indicators. GNN-based methods, due to their ability to model complex topological relationships and adapt to changing network states, frequently show superior performance in terms of reducing end-to-end latency and increasing throughput compared to traditional algorithms. However, the scalability and real-time application of GNNs remain challenges, and their performance heavily depends on the quality and variety of the training data, with some studies demonstrating only marginal improvements over conventional methods in highly controlled environments.

6.2. Datasets

To validate and compare different routing algorithms, real-world traffic matrices have been used in the literature [126–128], for example, the datasets collected in the Abilene, GÉANT, and CERNET networks [129], which are publicly available (https://github.com/jwwthu/DL4Internet/tree/main/TrafficMatrixPrediction/OD_pair, accessed on 22 September 2024). Each element in a traffic matrix measures the traffic from one node to another. Datasets are collected at different frequencies. The GÉANT traffic matrix is sampled every 15 min, and the Abilene and CERNET datasets are sampled every five minutes. More real-world traffic matrices can be obtained using SNDlib [130] (http://sndlib.zib.de/home.action, accessed on 22 September 2024).

Although real-world traffic matrices have been widely used in the literature, they have some limitations when used for network modeling and routing optimization [131]. First, most existing real-world traffic data are collected in a network topology with a small size, for example, less than 100 nodes, which is not sufficient for evaluating the scalability of new routing algorithms. Second, most existing real-world network traffic matrices were collected many years ago. For example, the Abilene dataset was collected in 2004, the GÉANT dataset was collected in 2005, and the CERNET dataset was collected in 2013. However, the Internet has changed significantly since the time when these traffic matrices were collected. These real-world network traffic matrices cannot reflect the latest situations. In addition, it is time and money-consuming to collect traffic measurements in our current complex networks, which makes it difficult to keep the data up-to-date. On the other hand, massive probe information may also affect the normal operation of the Internet. Third, extreme cases, such as network faults and bursty traffic, are rare and difficult to collect in practice. However, these corner cases are essential for evaluating the resilience and recovery capacity of the routing algorithms.

To overcome the limitations of real-world traffic measurements, synthetic network traffic matrices are created and leveraged in the evaluation process for routing algorithms based on a given network topology and traffic demand generators.

Most existing synthetic network traffic matrices are based on real-world network topologies, which can be found on the website of The Internet Topology Zoo (http://www.topology-zoo.org/, accessed on 22 September 2024). For example, synthetic network traffic matrices are created on three real-world network typologies, namely EBONE (Europe), Sprintlink (US), and Tiscali (Europe), all of which are all publicly available [132] (https://github.com/yanghu-bit/FlexEntry/tree/main/stage1/data, accessed on 22 September 2024). If the existing network topologies are still not large enough, some graph generators are available for generating massive-scale realistic networks with more than 10,000 nodes and redundancy features, e.g., YARGG (Yet Another Realistic Graph Generator) (https://github.com/JroLuttringer/YARGG, accessed on 22 September 2024). The summary of both real-world and synthetic traffic matrices is listed in Table 6.

Traffic demand generators are used to create synthetic traffic values following predefined distributions, for example, gravity and bimodal distributions. Traffic demand generators can be implemented as generic software or general-purpose programming languages such as MATLAB and Python. For example, TMGEN (https://github.com/progwriter/TMgen, accessed on 22 September 2024) is a Python tool used for generating traffic matrices.

Table 6. Summary of network traffic matrices.

Topology	Type	Node	Link	Traffic Matrices
Abilene	Real	12	30	48,096
CERNET	Real	14	32	9999
GÉANT	Real	23	74	10,769
Nobel-Germany	Real	17	52	288
Germany50	Real	50	176	288
EBONE (Europe)	Synthetic	23	76	100
Sprintlink (US)	Synthetic	44	166	100
Tiscali (Europe)	Synthetic	49	172	100

With the popularity of applying GNNs to routing problems, open global competitions with relevant research topics have been held to attract a broad audience from both academia and industry. For example, the "Graph Neural Networking Challenge 2020" (https://www.itu.int/en/ITU-T/AI/challenge/2020/Pages/default.aspx, accessed on 22 September 2024) attracted more than 1300 participants from 60+ countries [94], with the purpose of predicting the per-path mean delay given a network snapshot based on the network topology, traffic matrix and routing configuration. RouteNet [61] is provided as the baseline and more neural network-based solutions are proposed by the participants. A simulated dataset is generated from a packet-level simulator called OMNeT++ [133] to evaluate different solutions, which can also be used in future research.

6.3. Tools

In this part, some widely adopted network simulators and software libraries are summarized to help readers implement network simulations and prototype new models. These tools are summarized in Table 7.

Discrete-event packet-level network simulators are widely used in the literature for networking research with the purpose of simulating various real-world networks and evaluating different routing algorithms. Popular packet-level network simulators include ns-3, QualNet and OMNeT++ [133]. With the development of the SDN concept, more network simulators have been developed to facilitate integration with SDN protocols and tools, such as Mininet. Some network simulators have been developed with embedded reinforcement learning, making it convenient to implement RL-based routing algorithms, for example, PRISMA [134], which is a packet routing simulator for developing multi-agent reinforcement learning algorithms for the distributed routing problem.

Table 7. The collection of relevant tools.

Tool	Type	Link (Accessed on 22 September 2024)
ns-3	Network simulator	https://www.nsnam.org/
QualNet	Network simulator	https://www.ncs-in.com/product/qualnet-network-simulator-software/
OMNeT++	Network simulator	http://omnetpp.org/
Mininet	Network simulator	http://mininet.org/
PRISMA	Packet routing simulator	https://github.com/raparicioPardo/PRISMA
scikit-learn	ML software library	https://scikit-learn.org/
TensorFlow	DL software library	https://www.tensorflow.org/
PyTorch	DL software library	https://pytorch.org/
DGL	GNN software library	https://www.dgl.ai/
PyG	GNN software library	https://pytorch-geometric.readthedocs.io/en/latest/
Spektral	GNN software library	https://graphneural.network/
IGNNITION	GNN software library	https://github.com/BNN-UPC/ignnition

Some software libraries are available for the implementation of ML and DL algorithms, most of which are based on Python, for example, scikit-learn, TensorFlow, and PyTorch. They have been widely used for data preprocessing, model implementation and performance evaluation [135,136]. Based on general DL libraries (e.g., TensorFlow and PyTorch), some GNN software libraries have been developed for the rapid development and training of GNN models, e.g., Deep Graph Library (DGL), PyTorch Geometric (PyG) and Spektral. IGNNITION [137] is another framework designed for fast prototyping of GNNs in communication networks and has been used in [19] to implement two example use cases: RouteNet [30] for performance evaluation in wired networks and WCGCN [31] for radio resource management in wireless networks.

Finally, a collection of open-source routing algorithms is summarized in Table 8, which can be implemented as baselines in follow-up studies.

Table 8. The collection of open-source routing algorithms.

Study	Implemented Algorithm(s)	Link (Accessed on 22 September 2024)
-	30+ routing algorithms, e.g., Dijkstra and Bellman–Ford	https://github.com/AmoVanB/eces-routing
[138]	An application-aware segment routing algorithm	https://github.com/vanvantong/rl-sr
[139]	A DRL-based routing algorithm	https://github.com/danielaCasasv/DRSIR_DRL_routing_approach_for_SDN
[51]	A GNN+DRL-based routing algorithm	https://github.com/BNN-UPC/ENERO
[59]	A DRL-based routing algorithm	https://github.com/GuetYe/experiment-code
[132]	A RL-based routing algorithm	https://github.com/yanghu-bit/FlexEntry

7. Challenges and Opportunities

7.1. Challenges

Based on the literature review in the above sections, some research challenges have been identified in existing studies.

The first challenge is the scarcity of GNN-based benchmarks for routing optimization. For GNN-based network modeling, RouteNet can be considered as a benchmark and has been used in several follow-up studies. However, a widely accepted GNN benchmark is still lacking for the remaining two research paradigms. Designed with different routing policies, as shown in Tables 4 and 5, it is unfair to compare different GNN-based solutions. For most surveyed studies, the baselines are still traditional routing algorithms, for example, shortest path routing or OSPF. A GNN-based routing benchmark is essential for developing more effective GNN-based methods, particularly those with open data and source codes for replication.

The second research challenge is the training efficiency of the DRL model, which is considered in [120] with the introduction of evolutionary strategies. This challenge occurs when the computational complexity increases exponentially with network growth. With an increase in features, the dimension of the action space in the DRL solutions becomes larger, which makes the convergence of the DRL model slow or difficult. The DRL agent requires a long training process before achieving acceptable performance, which is unrealistic for applications in real-world networks, for example, in real elastic optical networks [110].

The third research challenge is the implementation of GNN-based solutions in practice. Most of the surveyed studies are based on network simulations, without being validated in real-world networks. Centralized routing based on the SDN assumption accounts for the majority, with only a few exceptions [49,53,106]. Centralized routing relies on the entire network status information to update the hidden state, which could be unavailable in practice and has a poor generalization ability for time-varying network topologies. Communication overhead is also introduced in the processes of network status collection and parameter updates, which may cause failure to guarantee QoS for time-critical applications.

Deploying GNNs for routing in real-world networks faces several challenges, including integration with existing infrastructure, hardware limitations, and security concerns. Many current network systems rely on well-established protocols like OSPF or MPLS, which are deeply embedded in the hardware and software of routers and switches. Integrating GNN-based models requires compatibility with these legacy systems, which can be complex and costly. Furthermore, the computational demands of GNNs, especially in large-scale networks, present significant hardware challenges. Network devices may lack the necessary processing power and memory to run GNN inference in real-time, making it necessary to offload computations to centralized or edge servers, which can introduce latency and reliability issues. Security is another concern, as GNNs, like all machine learning models, can be vulnerable to adversarial attacks where small perturbations in input data (such as traffic patterns or topology information) lead to incorrect routing decisions. While GNNs can potentially improve security by learning robust patterns and detecting anomalies in network behavior, they can also exacerbate risks if not adequately protected or if their decision-making processes are opaque to network administrators. Addressing these issues requires the development of lightweight, secure GNN implementations, improved explainability, and careful integration with existing routing protocols to ensure seamless and secure deployment.

7.2. Opportunities

Several research opportunities are discussed here, from various perspectives of exploration of the possibility of varied GNN models, in combination with novel techniques and extensions to wider applications.

7.2.1. Exploration of Novel GNN Architectures

One research opportunity is the exploration of additional GNN structures for routing applications to enhance learning capabilities in complex scenarios. GNNs have grown into large families of variants. However, most of the surveyed studies use only early-stage GNNs, such as GNN [67] and MPNN [74]. The application of GCN and GAT models, which are extremely successful in other domains, is rarely seen in the existing studies for routing optimization, which leaves a large research gap for exploration. Other specific research ideas include advancements in model interpretability, scalability to handle larger networks, and integration with real-time data for dynamic routing.

7.2.2. Combination with Emerging Techniques

Another research opportunity is the combination with other emerging techniques for providing real-time routing decisions, for example, quantum computing, which has been considered for vehicle routing problems [140]. Another example is the content-centric network (CCN), which entirely changes the current end-to-end communication transmission mechanism of the traditional TCP/IP network architecture. The content is separated from the IP addresses and users can directly access the data by name. Existing routing schemes have been inapplicable in this next-generation network structure, and new routing solutions are required. Furthermore, routing and efficient content caching can be jointly considered in a content-centric network with potential GNN-based solutions.

Future research directions for enhancing GNN performance in dynamic and distributed network environments point toward the integration of emerging techniques like federated learning, self-supervised learning, and online learning. Federated learning offers a promising avenue by enabling decentralized training of GNN models across multiple network nodes without sharing raw data, thus addressing privacy concerns and reducing communication overhead in large, distributed networks like satellite or IoT systems. This approach could allow models to learn collaboratively while preserving the autonomy of individual nodes. Self-supervised learning presents another opportunity by allowing GNNs to learn useful network representations from unlabeled data, which are abundant in network environments. By leveraging tasks like link prediction or graph reconstruction, GNNs can gain a better understanding of network structures without relying on extensive labeled data, which are often hard to acquire in real-time network scenarios. Additionally, online learning methods could significantly improve GNN adaptability in dynamic networks where topologies change frequently. These methods allow GNNs to update their models incrementally as new data arrive, facilitating quicker responses to evolving network conditions. Together, these techniques could enhance the robustness, scalability, and efficiency of GNN-based routing, making them more practical for real-world deployments in complex, large-scale networks.

7.2.3. Extension to Emerging Scenarios and Applications

The third research opportunity is the extension to emerging scenarios and applications, particularly those rarely considered in existing studies. For example, the space-air-ground integrated network (SAGIN) is a promising scenario driven by large-scale LEO constellations and software-defined satellites. In SAGINs, a heterogeneous architecture with both terrestrial and non-terrestrial networks requires a higher intelligent routing protocol to find an optimal path in terms of delay and energy cost, when the UAV and satellite nodes are energy sensitive [141]. Another example is Metaverse [142], which is based on real-time, multi-media, and remote cloud services and has not been considered in the literature with GNN-based routing optimization.

8. Conclusions

In conclusion, this paper presents a comprehensive overview of the application of GNNs for routing optimization in modern communication networks, emphasizing their advantages over traditional routing algorithms. By systematically categorizing existing

GNN-based approaches, we have highlighted their effectiveness in adapting to dynamic network conditions while improving key performance metrics such as latency and throughput. Notably, our review showcases numerical indicators of model accuracy, with several studies achieving mean relative errors as low as 1.27% and significantly enhancing routing decisions compared to conventional methods. Additionally, we provide a curated collection of data resources and tools for practitioners in the field, addressing a critical gap in the literature. This contribution not only underscores the importance of GNNs in evolving network architectures but also offers valuable insights and practical resources for researchers and engineers seeking to implement these advanced techniques. By addressing the challenges and opportunities inherent in this domain, we hope to inspire further exploration and innovation in GNN-based routing solutions, ultimately contributing to the development of more efficient and resilient communication networks.

Author Contributions: Conceptualization, W.J., H.H., Y.Z., J.W., M.H. and W.G.; methodology, W.J., H.H., Y.Z., J.W., M.H. and W.G.; software, W.J., H.H., Y.Z., J.W., M.H. and W.G.; validation, W.J., H.H., Y.Z., J.W., M.H. and W.G.; formal analysis, W.J., H.H., Y.Z., J.W., M.H. and W.G.; investigation, W.J., H.H., Y.Z., J.W., M.H. and W.G.; resources, W.J., H.H., Y.Z., J.W., M.H. and W.G.; data curation, W.J., H.H., Y.Z., J.W., M.H. and W.G.; writing—original draft preparation, W.J., H.H., Y.Z., J.W., M.H. and W.G.; writing—review and editing, W.J., H.H., Y.Z., J.M. and X.C.; visualization, W.J., H.H., Y.Z., J.W., M.H. and W.G.; supervision, J.M. and X.C.; project administration, J.M. and X.C.; funding acquisition, W.J. and J.M. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Social Science Foundation Youth Program under Grant 24CTJ011; in part by the National Natural Science Foundation of China under Grant 62401070; in part by the Fundamental Research Funds for the Central Universities under Grant 2023RC16; in part by Zhejiang Provincial Natural Science Foundation of China under Grant No. LQ24F030023.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Dai, B.; Cao, Y.; Wu, Z.; Dai, Z.; Yao, R.; Xu, Y. Routing optimization meets Machine Intelligence: A perspective for the future network. *Neurocomputing* **2021**, *459*, 44–58. [CrossRef]
2. Bertsimas, D.; Jaillet, P.; Martin, S. Online vehicle routing: The edge of optimization in large-scale applications. *Oper. Res.* **2019**, *67*, 143–162. [CrossRef]
3. Liu, D.; Zhang, J.; Cui, J.; Ng, S.X.; Maunder, R.G.; Hanzo, L. Deep-learning-aided packet routing in aeronautical Ad Hoc networks relying on real flight data: From single-objective to near-pareto multiobjective optimization. *IEEE Internet Things J.* **2021**, *9*, 4598–4614. [CrossRef]
4. Lewis, R. Algorithms for finding shortest paths in networks with vertex transfer penalties. *Algorithms* **2020**, *13*, 269. [CrossRef]
5. Prasad, P.R.; Shankar, S. Efficient Performance Analysis of Energy Aware on Demand Routing Protocol in Mobile Ad-Hoc Network. *Eng. Rep.* **2020**, *2*, e12116. [CrossRef]
6. Lakew, D.S.; Sa'ad, U.; Dao, N.N.; Na, W.; Cho, S. Routing in flying ad hoc networks: A comprehensive survey. *IEEE Commun. Surv. Tutorials* **2020**, *22*, 1071–1120. [CrossRef]
7. Rovira-Sugranes, A.; Razi, A.; Afghah, F.; Chakareski, J. A review of AI-enabled routing protocols for UAV networks: Trends, challenges, and future outlook. *Ad Hoc Netw.* **2022**, *130*, 102790. [CrossRef]
8. Alam, M.M.; Moh, S. Survey on Q-Learning-Based Position-Aware Routing Protocols in Flying Ad Hoc Networks. *Electronics* **2022**, *11*, 1099. [CrossRef]
9. Jiang, W. Software defined satellite networks: A survey. *Digit. Commun. Netw.* **2023**, *9*, 1243–1264. [CrossRef]
10. Yang, S.; Tan, C.; Madsen, D.Ø.; Xiang, H.; Li, Y.; Khan, I.; Choi, B.J. Comparative Analysis of Routing Schemes Based on Machine Learning. *Mob. Inf. Syst.* **2022**, *2022*, 4560072. [CrossRef]
11. Nazib, R.A.; Moh, S. Reinforcement learning-based routing protocols for vehicular ad hoc networks: A comparative survey. *IEEE Access* **2021**, *9*, 27552–27587. [CrossRef]
12. Jiang, W. Applications of deep learning in stock market prediction: Recent progress. *Expert Syst. Appl.* **2021**, *184*, 115537. [CrossRef]

13. Jiang, W.; Luo, J. Graph neural network for traffic forecasting: A survey. *Expert Syst. Appl.* **2022**, *207*, 117921. [CrossRef]
14. Yang, B.; Wang, X.; Xing, Y.; Cheng, C.; Jiang, W.; Feng, Q. Modality Fusion Vision Transformer for Hyperspectral and LiDAR Data Collaborative Classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2024**, *17*, 17052–17065. [CrossRef]
15. Zhang, Y.; Xu, S.; Zhang, L.; Jiang, W.; Alam, S.; Xue, D. Short-term multi-step-ahead sector-based traffic flow prediction based on the attention-enhanced graph convolutional LSTM network (AGC-LSTM). In *Neural Computing and Applications*; Springer: Berlin/Heidelberg, Germany, 2024; pp. 1–20.
16. Wu, J.P.; Qiu, G.Q.; Wu, C.M.; Jiang, W.W.; Jin, J.H. Federated learning for network attack detection using attention-based graph neural networks. *Sci. Rep.* **2024**, *14*, 19088. [CrossRef]
17. Chen, Y.; He, J.; Jiang, W.; Zhang, Y.; Huang, S.; Feng, Z. Toward Collaborative and Channel-Robust Automatic Modulation Classification for OFDM Signals. *IEEE Wirel. Commun. Lett.* **2024**, *early access*. [CrossRef]
18. Lu, Y.; Wang, W.; Bai, R.; Zhou, S.; Garg, L.; Bashir, A.K.; Jiang, W.; Hu, X. Hyper-relational interaction modeling in multi-modal trajectory prediction for intelligent connected vehicles in smart cites. *Inf. Fusion* **2024**, *114*, 102682. [CrossRef]
19. Suárez-Varela, J.; Almasan, P.; Ferriol-Galmés, M.; Rusek, K.; Geyer, F.; Cheng, X.; Shi, X.; Xiao, S.; Scarselli, F.; Cabellos-Aparicio, A.; et al. Graph Neural Networks for Communication Networks: Context, Use Cases and Opportunities. *IEEE Netw.* **2022**, *37*, 146–153. [CrossRef]
20. Quy, V.K.; Nam, V.H.; Linh, D.M.; Ngoc, L.A. Routing Algorithms for MANET-IoT Networks: A Comprehensive Survey. *Wirel. Pers. Commun.* **2022**, *125*, 3501–3525. [CrossRef]
21. Cao, X.; Li, Y.; Xiong, X.; Wang, J. Dynamic Routings in Satellite Networks: An Overview. *Sensors* **2022**, *22*, 4552. [CrossRef]
22. Yan, J.; Qi, B. CARA: A Congestion-Aware Routing Algorithm for Wireless Sensor Networks. *Algorithms* **2021**, *14*, 199. [CrossRef]
23. Behera, T.M.; Samal, U.C.; Mohapatra, S.K.; Khan, M.S.; Appasani, B.; Bizon, N.; Thounthong, P. Energy-Efficient Routing Protocols for Wireless Sensor Networks: Architectures, Strategies, and Performance. *Electronics* **2022**, *11*, 2282. [CrossRef]
24. Jiang, W.; Han, H.; He, M.; Gu, W. ML-based pre-deployment SDN performance prediction with neural network boosting regression. *Expert Syst. Appl.* **2024**, *241*, 122774. [CrossRef]
25. Amin, R.; Rojas, E.; Aqdus, A.; Ramzan, S.; Casillas-Perez, D.; Arco, J.M. A survey on machine learning techniques for routing optimization in SDN. *IEEE Access* **2021**, *9*, 104582–104611. [CrossRef]
26. Etengu, R.; Tan, S.C.; Kwang, L.C.; Abbou, F.M.; Chuah, T.C. AI-assisted framework for green-routing and load balancing in hybrid software-defined networking: Proposal, challenges and future perspective. *IEEE Access* **2020**, *8*, 166384–166441. [CrossRef]
27. Jiang, W.; Han, H.; He, M.; Gu, W. When game theory meets satellite communication networks: A survey. *Comput. Commun.* **2024**, *217*, 208–229. [CrossRef]
28. Jiang, W. Graph-based Deep Learning for Communication Networks: A Survey. *Comput. Commun.* **2022**, *185*, 40–54. [CrossRef]
29. Jiang, W.; Zhang, Y.; Han, H.; Huang, Z.; Li, Q.; Mu, J. Mobile Traffic Prediction in Consumer Applications: A Multimodal Deep Learning Approach. *IEEE Trans. Consum. Electron.* **2024**, *70*, 3425–3435. [CrossRef]
30. Rusek, K.; Suárez-Varela, J.; Almasan, P.; Barlet-Ros, P.; Cabellos-Aparicio, A. RouteNet: Leveraging Graph Neural Networks for network modeling and optimization in SDN. *IEEE J. Sel. Areas Commun.* **2020**, *38*, 2260–2270. [CrossRef]
31. Shen, Y.; Shi, Y.; Zhang, J.; Letaief, K.B. Graph neural networks for scalable radio resource management: Architecture design and theoretical analysis. *IEEE J. Sel. Areas Commun.* **2020**, *39*, 101–115. [CrossRef]
32. Vesselinova, N.; Steinert, R.; Perez-Ramirez, D.F.; Boman, M. Learning combinatorial optimization on graphs: A survey with applications to networking. *IEEE Access* **2020**, *8*, 120388–120416. [CrossRef]
33. He, S.; Xiong, S.; Ou, Y.; Zhang, J.; Wang, J.; Huang, Y.; Zhang, Y. An overview on the application of graph neural networks in wireless networks. *IEEE Open J. Commun. Soc.* **2021**, *2*, 2547–2565. [CrossRef]
34. Ivanov, A.; Tonchev, K.; Poulkov, V.; Manolova, A.; Neshov, N.N. Graph-Based Resource Allocation for Integrated Space and Terrestrial Communications. *Sensors* **2022**, *22*, 5778. [CrossRef] [PubMed]
35. Heidari, A.; Navimipour, N.J.; Zeadally, S.; Chamola, V. Everything you wanted to know about ChatGPT: Components, capabilities, applications, and opportunities. *Internet Technol. Lett.* **2024**, e530, *early view*. [CrossRef]
36. Amiri, Z.; Heidari, A.; Navimipour, N.J. Comprehensive Survey of Artificial Intelligence Techniques and Strategies for Climate Change Mitigation. *Energy* **2024**, *308*, 132827. [CrossRef]
37. Mansoor, N.; Hossain, M.I.; Rozario, A.; Zareei, M.; Arreola, A.R. A fresh look at routing protocols in unmanned aerial vehicular networks: A survey. *IEEE Access* **2023**, *11*, 66289–66308. [CrossRef]
38. Sohail, M.; Latif, Z.; Javed, S.; Biswas, S.; Ajmal, S.; Iqbal, U.; Raza, M.; Khan, A.U. Routing protocols in vehicular adhoc networks (vanets): A comprehensive survey. *Internet Things* **2023**, *23*, 100837. [CrossRef]
39. Musaddiq, A.; Olsson, T.; Ahlgren, F. Reinforcement-Learning-Based Routing and Resource Management for Internet of Things Environments: Theoretical Perspective and Challenges. *Sensors* **2023**, *23*, 8263. [CrossRef]
40. Priyadarshi, R. Exploring machine learning solutions for overcoming challenges in IoT-based wireless sensor network routing: A comprehensive review. *Wirel. Netw.* **2024**, *30*, 2647–2673. [CrossRef]
41. Priyadarshi, R. Energy-efficient routing in wireless sensor networks: A meta-heuristic and artificial intelligence-based approach: A comprehensive review. *Arch. Comput. Methods Eng.* **2024**, *31*, 2109–2137. [CrossRef]
42. Priyadarshi, R.; Kumar, R.R.; Ying, Z. Techniques employed in distributed cognitive radio networks: A survey on routing intelligence. In *Multimedia Tools and Applications*; Springer: Berlin/Heidelberg, Germany, 2024; pp. 1–52.

43. Malakar, M.; Mahapatro, J.; Ghosh, T. A survey on routing and load-balancing mechanisms in software-defined vehicular networks. *Wirel. Netw.* **2024**, *30*, 3181–3197. [CrossRef]
44. Alekseeva, D.; Stepanov, N.; Veprev, A.; Sharapova, A.; Lohan, E.S.; Ometov, A. Comparison of machine learning techniques applied to traffic prediction of real wireless network. *IEEE Access* **2021**, *9*, 159495–159514. [CrossRef]
45. Waikhom, L.; Patgiri, R. A survey of graph neural networks in various learning paradigms: Methods, applications, and challenges. *Artif. Intell. Rev.* **2023**, *56*, 6295–6364. [CrossRef]
46. Suzuki, T.; Yasuda, Y.; Nakamura, R.; Ohsaki, H. On estimating communication delays using graph convolutional networks with semi-supervised learning. In Proceedings of the 2020 International Conference on Information Networking (ICOIN), Barcelona, Spain, 7–10 January 2020; pp. 481–486.
47. Suzuki, T.; Ohsaki, H. On Inferring Communication Delays Using Semi-Supervised Learning. In Proceedings of the 2022 International Conference on Information Networking (ICOIN), Jeju-si, Republic of Korea, 12–15 January 2022; pp. 260–265.
48. Mai, X.; Fu, Q.; Chen, Y. Packet routing with graph attention multi-agent reinforcement learning. In Proceedings of the 2021 IEEE Global Communications Conference (GLOBECOM), Madrid, Spain, 7–11 December 2021; pp. 1–6.
49. Huang, R.; Guan, W.; Zhai, G.; He, J.; Chu, X. Deep Graph Reinforcement Learning Based Intelligent Traffic Routing Control for Software-Defined Wireless Sensor Networks. *Appl. Sci.* **2022**, *12*, 1951. [CrossRef]
50. Mnih, V.; Kavukcuoglu, K.; Silver, D.; Graves, A.; Antonoglou, I.; Wierstra, D.; Riedmiller, M. Playing atari with deep reinforcement learning. *arXiv* **2013**, arXiv:1312.5602.
51. Almasan, P.; Xiao, S.; Cheng, X.; Shi, X.; Barlet-Ros, P.; Cabellos-Aparicio, A. ENERO: Efficient real-time WAN routing optimization with Deep Reinforcement Learning. *Comput. Netw.* **2022**, *214*, 109166. [CrossRef]
52. Li, Y.; Tarlow, D.; Brockschmidt, M.; Zemel, R. Gated graph sequence neural networks. In Proceedings of the International Conference on Learning Representations (ICLR '16), San Juan, Puerto Rico, 2–4 May 2016.
53. Geyer, F.; Carle, G. Learning and generating distributed routing protocols using graph-based deep learning. In Proceedings of the 2018 Workshop on Big Data Analytics and Machine Learning for Data Communication Networks, Munich, Germany, 20 August 2018; pp. 40–45.
54. Liu, C.; Xu, M.; Geng, N.; Zhang, X. A survey on machine learning based routing algorithms. *J. Comput. Res. Dev.* **2020**, *57*, 671–687.
55. Pampapathi, B.; Guptha, N.; Hema, M. Towards an effective deep learning-based intrusion detection system in the internet of things. *Telemat. Inform. Rep.* **2022**, *7*, 100009.
56. Almasan, P.; Suárez-Varela, J.; Wu, B.; Xiao, S.; Barlet-Ros, P.; Cabellos-Aparicio, A. Towards real-time routing optimization with deep reinforcement learning: Open challenges. In Proceedings of the 2021 IEEE 22nd International Conference on High Performance Switching and Routing (HPSR), Paris, France, 7–10 June 2021; pp. 1–6.
57. Lan, J.; Yu, C.; Hu, Y.; Li, Z. A SDN routing optimization mechanism based on deep reinforcement learning. *J. Electron. Inf. Technol.* **2019**, *41*, 2669–2674.
58. Munikoti, S.; Agarwal, D.; Das, L.; Halappanavar, M.; Natarajan, B. Challenges and opportunities in deep reinforcement learning with graph neural networks: A comprehensive review of algorithms and applications. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**, *early access*. [CrossRef]
59. Huang, L.; Ye, M.; Xue, X.; Wang, Y.; Qiu, H.; Deng, X. Intelligent routing method based on Dueling DQN reinforcement learning and network traffic state prediction in SDN. *Wirel. Netw.* **2024**, *30*, 4507–4525. [CrossRef]
60. Xu, Z.; Tang, J.; Meng, J.; Zhang, W.; Wang, Y.; Liu, C.H.; Yang, D. Experience-driven networking: A deep reinforcement learning based approach. In Proceedings of the IEEE INFOCOM 2018—IEEE Conference on Computer Communications, Honolulu, HI, USA, 16–19 April 2018; pp. 1871–1879.
61. Rusek, K.; Suárez-Varela, J.; Mestres, A.; Barlet-Ros, P.; Cabellos-Aparicio, A. Unveiling the potential of Graph Neural Networks for network modeling and optimization in SDN. In Proceedings of the 2019 ACM Symposium on SDN Research, San Jose, CA, USA, 3–4 April 2019; pp. 140–151.
62. Wu, Z.; Pan, S.; Chen, F.; Long, G.; Zhang, C.; Philip, S.Y. A comprehensive survey on graph neural networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *32*, 4–24. [CrossRef]
63. Zhou, J.; Cui, G.; Hu, S.; Zhang, Z.; Yang, C.; Liu, Z.; Wang, L.; Li, C.; Sun, M. Graph neural networks: A review of methods and applications. *AI Open* **2020**, *1*, 57–81. [CrossRef]
64. Zhang, Z.; Cui, P.; Zhu, W. Deep learning on graphs: A survey. *IEEE Trans. Knowl. Data Eng.* **2020**, *34*, 249–270. [CrossRef]
65. Xia, F.; Sun, K.; Yu, S.; Aziz, A.; Wan, L.; Pan, S.; Liu, H. Graph learning: A survey. *IEEE Trans. Artif. Intell.* **2021**, *2*, 109–127. [CrossRef]
66. Ruiz, L.; Gama, F.; Ribeiro, A. Graph neural networks: Architectures, stability, and transferability. *Proc. IEEE* **2021**, *109*, 660–682. [CrossRef]
67. Scarselli, F.; Gori, M.; Tsoi, A.C.; Hagenbuchner, M.; Monfardini, G. The graph neural network model. *IEEE Trans. Neural Netw.* **2008**, *20*, 61–80. [CrossRef]
68. Defferrard, M.; Bresson, X.; Vandergheynst, P. Convolutional neural networks on graphs with fast localized spectral filtering. In Proceedings of the 30th International Conference on Neural Information Processing Systems, Changsha, China, 20–23 November 2016; pp. 3844–3852.

69. Kipf, T.N.; Welling, M. Semi-supervised classification with graph convolutional networks. In Proceedings of the International Conference on Learning Representations (ICLR '17), Toulon, France, 24–26 April 2017.
70. Levie, R.; Monti, F.; Bresson, X.; Bronstein, M.M. Cayleynets: Graph convolutional neural networks with complex rational spectral filters. *IEEE Trans. Signal Process.* **2018**, *67*, 97–109. [CrossRef]
71. Park, C.; Park, J.; Park, S. AGCN: Attention-based graph convolutional networks for drug-drug interaction extraction. *Expert Syst. Appl.* **2020**, *159*, 113538. [CrossRef]
72. Ma, Y.; Hao, J.; Yang, Y.; Li, H.; Jin, J.; Chen, G. Spectral-based graph convolutional network for directed graphs. *arXiv* **2019**, arXiv:1907.08990.
73. Tong, Z.; Liang, Y.; Sun, C.; Rosenblum, D.S.; Lim, A. Directed graph convolutional network. *arXiv* **2020**, arXiv:2004.13970.
74. Gilmer, J.; Schoenholz, S.S.; Riley, P.F.; Vinyals, O.; Dahl, G.E. Neural message passing for quantum chemistry. In Proceedings of the International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; pp. 1263–1272.
75. Niepert, M.; Ahmed, M.; Kutzkov, K. Learning convolutional neural networks for graphs. In Proceedings of the International Conference on Machine Learning, New York, NY, USA, 22–22 June 2016; pp. 2014–2023.
76. Atwood, J.; Towsley, D. Diffusion-convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2016**, *29*, 2001–2009.
77. Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; Bengio, Y. Graph Attention Networks. In Proceedings of the International Conference on Learning Representations, Vancouver, BC, Canada, 30 April–3 May 2018.
78. Wang, X.; Ji, H.; Shi, C.; Wang, B.; Ye, Y.; Cui, P.; Yu, P.S. Heterogeneous graph attention network. In Proceedings of the World Wide Web Conference, San Francisco, CA, USA, 13–17 May 2019; pp. 2022–2032.
79. Yang, S.; Li, G.; Yu, Y. Dynamic graph attention for referring expression comprehension. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 4644–4653.
80. Khemani, B.; Patil, S.; Kotecha, K.; Tanwar, S. A review of graph neural networks: Concepts, architectures, techniques, challenges, datasets, applications, and future directions. *J. Big Data* **2024**, *11*, 18. [CrossRef]
81. Yang, S.; Zhuang, L.; Zhang, J.; Lan, J.; Li, B. A Multi-Policy Deep Reinforcement Learning Approach for Multi-Objective Joint Routing and Scheduling in Deterministic Networks. *IEEE Internet Things J.* **2024**, *11*, 17402–17418. [CrossRef]
82. Marwani, M.; Kaddoum, G. Scalable Spatial and Geometric Learning Approach for Joint Power Control and Channel Allocation. *IEEE Trans. Wirel. Commun.* **2024**, *early access*. [CrossRef]
83. Shi, Y.; Wang, W.; Zhu, X.; Zhu, H. Low Earth Orbit Satellite Network Routing Algorithm Based on Graph Neural Networks and Deep Q-Network. *Appl. Sci.* **2024**, *14*, 3840. [CrossRef]
84. Ding, M.; Guo, Y.; Huang, Z.; Lin, B.; Luo, H. GROM: A generalized routing optimization method with graph neural network and deep reinforcement learning. *J. Netw. Comput. Appl.* **2024**, *229*, 103927. [CrossRef]
85. Xu, J.; Wang, Y.; Zhang, B.; Ma, J. A Graph reinforcement learning based SDN routing path selection for optimizing long-term revenue. *Future Gener. Comput. Syst.* **2024**, *150*, 412–423. [CrossRef]
86. Han, C.; Xiong, W.; Yu, R. Deep Reinforcement Learning-Based Multipath Routing for LEO Megaconstellation Networks. *Electronics* **2024**, *13*, 3054. [CrossRef]
87. Agarwal, C.; Queen, O.; Lakkaraju, H.; Zitnik, M. Evaluating explainability for graph neural networks. *Sci. Data* **2023**, *10*, 144. [CrossRef]
88. Vrahatis, A.G.; Lazaros, K.; Kotsiantis, S. Graph Attention Networks: A Comprehensive Review of Methods and Applications. *Future Internet* **2024**, *16*, 318. [CrossRef]
89. Nie, M.; Chen, D.; Wang, D. Reinforcement learning on graphs: A survey. *IEEE Trans. Emerg. Top. Comput. Intell.* **2023**, *7*, 1065–1082. [CrossRef]
90. Gao, H.; Yu, X.; Sui, Y.; Shao, F.; Sun, R. Topological graph convolutional network based on complex network characteristics. *IEEE Access* **2022**, *10*, 64465–64472. [CrossRef]
91. Chung, D.; Sohn, I. Neural network optimization based on complex network theory: A survey. *Mathematics* **2023**, *11*, 321. [CrossRef]
92. Badia-Sampera, A.; Suárez-Varela, J.; Almasan, P.; Rusek, K.; Barlet-Ros, P.; Cabellos-Aparicio, A. Towards more realistic network models based on graph neural networks. In Proceedings of the 15th International Conference on emerging Networking EXperiments and Technologies, Orlando, FL, USA, 9–12 December 2019; pp. 14–16.
93. Suárez-Varela, J.; Carol-Bosch, S.; Rusek, K.; Almasan, P.; Arias, M.; Barlet-Ros, P.; Cabellos-Aparicio, A. Challenging the generalization capabilities of graph neural networks for network modeling. In Proceedings of the ACM SIGCOMM 2019 Conference Posters and Demos, Beijing, China, 19–23 August 2019; pp. 114–115.
94. Suárez-Varela, J.; Ferriol-Galmés, M.; López, A.; Almasan, P.; Bernárdez, G.; Pujol-Perich, D.; Rusek, K.; Bonniot, L.; Neumann, C.; Schnitzler, F.; et al. The graph neural networking challenge: A worldwide competition for education in AI/ML for networks. *ACM SIGCOMM Comput. Commun. Rev.* **2021**, *51*, 9–16. [CrossRef]
95. Ferriol-Galmés, M.; Suárez-Varela, J.; Barlet-Ros, P.; Cabellos-Aparicio, A. Applying graph-based deep learning to realistic network scenarios. *arXiv* **2020**, arXiv:2010.06686.
96. Afonso, B.K.d.A.; Berton, L. QT-Routenet: Improved GNN generalization to larger 5G networks by fine-tuning predictions from queueing theory. *ITU J. Future Evol. Technol.* **2022**, *3*. [CrossRef]

97. Ferriol-Galmés, M.; Rusek, K.; Suárez-Varela, J.; Xiao, S.; Shi, X.; Cheng, X.; Wu, B.; Barlet-Ros, P.; Cabellos-Aparicio, A. Routenet-erlang: A graph neural network for network performance evaluation. In Proceedings of the IEEE INFOCOM 2022—IEEE Conference on Computer Communications, London, UK, 2–5 May 2022; pp. 2018–2027.
98. Battaglia, P.W.; Hamrick, J.B.; Bapst, V.; Sanchez-Gonzalez, A.; Zambaldi, V.; Malinowski, M.; Tacchetti, A.; Raposo, D.; Santoro, A.; Faulkner, R.; et al. Relational inductive biases, deep learning, and graph networks. *arXiv* **2018**, arXiv:1806.01261.
99. Li, J.; Sun, P.; Hu, Y. Traffic modeling and optimization in datacenters with graph neural network. *Comput. Netw.* **2020**, *181*, 107528. [CrossRef]
100. Zhu, T.; Chen, X.; Chen, L.; Wang, W.; Wei, G. Gclr: Gnn-based cross layer optimization for multipath tcp by routing. *IEEE Access* **2020**, *8*, 17060–17070. [CrossRef]
101. Khan, T.A.; Abbas, K.; Rivera, J.J.D.; Muhammad, A.; Song, W.C. Applying RouteNet and LSTM to Achieve Network Automation: An Intent-based Networking Approach. In Proceedings of the 2021 22nd Asia-Pacific Network Operations and Management Symposium (APNOMS), Tainan, Taiwan, 8–10 September 2021; pp. 254–257.
102. Yan, B.; Liu, Q.; Shen, J.; Liang, D. Flowlet-level multipath routing based on graph neural network in OpenFlow-based SDN. *Future Gener. Comput. Syst.* **2022**, *134*, 140–153. [CrossRef]
103. Zhu, Y.; Liu, W.; Ling, S.; Luo, J. Network modeling based on GNN and network behaviors. In Proceedings of the 2022 7th International Conference on Computer and Communication Systems (ICCCS), Wuhan, China, 22–25 April 2022; pp. 554–559.
104. Sawada, K.; Kotani, D.; Okabe, Y. Network routing optimization based on machine learning using graph networks robust against topology change. In Proceedings of the 2020 International Conference on Information Networking (ICOIN), Barcelona, Spain, 7–10 January 2020; pp. 608–615.
105. Zhuang, Z.; Wang, J.; Qi, Q.; Sun, H.; Liao, J. Toward greater intelligence in route planning: A graph-aware deep learning approach. *IEEE Syst. J.* **2019**, *14*, 1658–1669. [CrossRef]
106. Xiao, S.; Mao, H.; Wu, B.; Liu, W.; Li, F. Neural packet routing. In Proceedings of the Workshop on Network Meets AI & ML, Virtual Event, 10–14 August 2020; pp. 28–34.
107. Liu, M.; Li, J.; Lu, H. Routing in small satellite networks: A GNN-based learning approach. *arXiv* **2021**, arXiv:2108.08523.
108. Tan, J.; Guan, W. Resource allocation of fog radio access network based on deep reinforcement learning. *Eng. Rep.* **2022**, *4*, e12497. [CrossRef]
109. Almasan, P.; Suárez-Varela, J.; Badia-Sampera, A.; Rusek, K.; Barlet-Ros, P.; Cabellos-Aparicio, A. Deep reinforcement learning meets graph neural networks: Exploring a routing optimization use case. *arXiv* **2019**, arXiv:1910.07421. [CrossRef]
110. Xu, L.; Huang, Y.C.; Xue, Y.; Hu, X. Deep Reinforcement Learning-based Routing and Spectrum Assignment of EONs by Exploiting GCN and RNN for Feature Extraction. *J. Light. Technol.* **2022**, *40*, 4945–4955. [CrossRef]
111. Mnih, V.; Badia, A.P.; Mirza, M.; Graves, A.; Lillicrap, T.; Harley, T.; Silver, D.; Kavukcuoglu, K. Asynchronous methods for deep reinforcement learning. In Proceedings of the International Conference on Machine Learning, New York, NY, USA, 20–22 June 2016; pp. 1928–1937.
112. Dong, T.; Zhuang, Z.; Qi, Q.; Wang, J.; Sun, H.; Yu, F.R.; Sun, T.; Zhou, C.; Liao, J. Intelligent joint network slicing and routing via GCN-powered multi-task deep reinforcement learning. *IEEE Trans. Cogn. Commun. Netw.* **2021**, *8*, 1269–1286. [CrossRef]
113. Chen, B.; Zhu, D.; Wang, Y.; Zhang, P. An Approach to Combine the Power of Deep Reinforcement Learning with a Graph Neural Network for Routing Optimization. *Electronics* **2022**, *11*, 368. [CrossRef]
114. Xu, X.; Lu, Y.; Fu, Q. Applying Graph Neural Network in Deep Reinforcement Learning to Optimize Wireless Network Routing. In Proceedings of the 2021 9th International Conference on Advanced Cloud and Big Data (CBD), Xi'an, China, 26–27 March 2022; pp. 218–223.
115. Swaminathan, A.; Chaba, M.; Sharma, D.K.; Ghosh, U. GraphNET: Graph Neural Networks for routing optimization in Software Defined Networks. *Comput. Commun.* **2021**, *178*, 169–182. [CrossRef]
116. Hope, O.; Yoneki, E. GDDR: GNN-based Data-Driven Routing. In Proceedings of the 2021 IEEE 41st International Conference on Distributed Computing Systems (ICDCS), Washington, DC, USA, 7–10 July 2021; pp. 517–527.
117. Wang, H.; Ran, Y.; Zhao, L.; Wang, J.; Luo, J.; Zhang, T. GRouting: Dynamic Routing for LEO Satellite Networks with Graph-based Deep Reinforcement Learning. In Proceedings of the 2021 4th International Conference on Hot Information-Centric Networking (HotICN), Nanjing, China, 25–27 November 2021; pp. 123–128.
118. Jiang, W.; Han, H.; Zhang, Y.; Mu, J. Federated split learning for sequential data in satellite–terrestrial integrated networks. *Inf. Fusion* **2024**, *103*, 102141. [CrossRef]
119. Yang, L.; Wei, Y.; Yu, F.R.; Han, Z. Joint Routing and Scheduling Optimization in Time-Sensitive Networks Using Graph Convolutional Network-based Deep Reinforcement Learning. *IEEE Internet Things J.* **2022**, *9*, 23981–23994. [CrossRef]
120. Güemes-Palau, C.; Almasan, P.; Xiao, S.; Cheng, X.; Shi, X.; Barlet-Ros, P.; Cabellos-Aparicio, A. Accelerating Deep Reinforcement Learning for Digital Twin Network Optimization with Evolutionary Strategies. In Proceedings of the NOMS 2022-2022 IEEE/IFIP Network Operations and Management Symposium, Budapest, Hungary, 25–29 April 2022; pp. 1–5.
121. Guo, Y.; Wu, Q.; She, H. A Routing Optimization Policy Using Graph Convolution Deep Reinforcement Learning. In Proceedings of the 2023 IEEE/CIC International Conference on Communications in China (ICCC), Dalian, China, 10–12 August 2023; pp. 1–6.
122. Li, X.; Xiao, Y.; Liu, S.; Lu, X.; Liu, F.; Zhou, W.; Liu, J. GAPPO-A Graph Attention Reinforcement Learning based Robust Routing Algorithm. In Proceedings of the 2023 IEEE 34th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), Toronto, ON, Canada, 5–8 September 2023; pp. 1–7.

123. Sun, H.; Wu, Q.; She, H.; Guo, Y.; Cao, H. DGL-Routing: One Routing Optimization Model Based on Deep Graph Learning. In Proceedings of the 2023 IEEE International Conference on Communications Workshops (ICC Workshops), Rome, Italy, 28 May–1 June 2023; pp. 891–896.
124. He, Q.; Wang, Y.; Wang, X.; Xu, W.; Li, F.; Yang, K.; Ma, L. Routing optimization with deep reinforcement learning in knowledge defined networking. *IEEE Trans. Mob. Comput.* **2023**, *23*, 1444–1455. [CrossRef]
125. Bhavanasi, S.S.; Pappone, L.; Esposito, F. Dealing with Changes: Resilient Routing via Graph Neural Networks and Multi-Agent Deep Reinforcement Learning. *IEEE Trans. Netw. Serv. Manag.* **2023**, *20*, 2283–2294. [CrossRef]
126. Jiang, W.; Zhang, L. Geospatial data to images: A deep-learning framework for traffic forecasting. *Tsinghua Sci. Technol.* **2018**, *24*, 52–64. [CrossRef]
127. Jiang, W. Internet traffic prediction with deep neural networks. *Internet Technol. Lett.* **2022**, *5*, e314. [CrossRef]
128. Jiang, W. Cellular traffic prediction with machine learning: A survey. *Expert Syst. Appl.* **2022**, *201*, 117163. [CrossRef]
129. Jiang, W. Internet traffic matrix prediction with convolutional LSTM neural network. *Internet Technol. Lett.* **2022**, *5*, e322. [CrossRef]
130. Orlowski, S.; Pióro, M.; Tomaszewski, A.; Wessäly, R. SNDlib 1.0–Survivable Network Design Library. In Proceedings of the 3rd International Network Optimization Conference (INOC 2007), Spa, Belgium, 22–25 April 2007.
131. Xu, Z.; Tang, N.; Xu, C.; Cheng, X. Data science: Connotation, methods, technologies, and development. *Data Sci. Manag.* **2021**, *1*, 32–37. [CrossRef]
132. Ye, M.; Hu, Y.; Zhang, J.; Guo, Z.; Chao, H.J. Mitigating Routing Update Overhead for Traffic Engineering by Combining Destination-Based Routing With Reinforcement Learning. *IEEE J. Sel. Areas Commun.* **2022**, *40*, 2662–2677. [CrossRef]
133. Varga, A.; Hornig, R. An overview of the OMNeT++ simulation environment. In Proceedings of the 1st International Conference on Simulation Tools and Techniques for Communications, Networks and Systems & Workshops, Marseille, France, 3–7 March 2008; pp. 1–10.
134. Alliche, R.A.; Barros, T.D.S.; Aparicio-Pardo, R.; Sassatelli, L. PRISMA: A Packet Routing Simulator for Multi-Agent Reinforcement Learning. In Proceedings of the 2022 IFIP Networking Conference (IFIP Networking), Catania, Italy, 13–16 June 2022; pp. 1–6.
135. Pugliese, R.; Regondi, S.; Marini, R. Machine learning-based approach: Global trends, research directions, and regulatory standpoints. *Data Sci. Manag.* **2021**, *4*, 19–29. [CrossRef]
136. Schröder, T.; Schulz, M. Monitoring machine learning models: A categorization of challenges and methods. *Data Sci. Manag.* **2022**, *5*, 105–116. [CrossRef]
137. Pujol-Perich, D.; Suárez-Varela, J.; Ferriol, M.; Xiao, S.; Wu, B.; Cabellos-Aparicio, A.; Barlet-Ros, P. IGNNITION: Bridging the gap between graph neural networks and networking systems. *IEEE Netw.* **2021**, *35*, 171–177. [CrossRef]
138. Tong, V.; Souihi, S.; Tran, H.A.; Mellouk, A. SDN-Based Application-Aware Segment Routing for Large-Scale Network. *IEEE Syst. J.* **2021**, *16*, 4401–4410. [CrossRef]
139. Casas-Velasco, D.M.; Rendon, O.M.C.; da Fonseca, N.L. DRSIR: A Deep Reinforcement Learning Approach for Routing in Software-Defined Networking. *IEEE Trans. Netw. Serv. Manag.* **2021**, *19*, 4807–4820. [CrossRef]
140. Osaba, E.; Villar-Rodriguez, E.; Oregi, I. A Systematic Literature Review of Quantum Computing for Routing Problems. *IEEE Access* **2022**, *10*, 55805–55817. [CrossRef]
141. Huang, Y.; Yang, D.; Feng, B.; Tian, A.; Dong, P.; Yu, S.; Zhang, H. A GNN-Enabled Multipath Routing Algorithm for Spatial-Temporal Varying LEO Satellite Networks. *IEEE Trans. Veh. Technol.* **2023**, *73*, 5454–5468. [CrossRef]
142. Mystakidis, S. Metaverse. *Encyclopedia* **2022**, *2*, 486–497. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.