Intelligent Routing Methods for Low-Earth Orbit Satellite Networks Based on Machine Learning: A Comprehensive Survey

Zhenyu Zhu[a], Zheheng Rao[a,*], Shitong Xiao[a], Ye Yao[a], Yanyan Xu[b] and Weizhi Meng[c]

[a] *School of Cyberspace, Hangzhou Dianzi University, Hangzhou 310018, China*
[b] *State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan 430072, China*
[c] *School of Computing and Communications, Lancaster University, United Kingdom*

ARTICLE INFO

Keywords:
Routing
Low Earth Orbit Satellite Network
Machine Learning
6G Network

ABSTRACT

With the continuous progress of modern communication technology and the emergence of the 6G concept, people's demand for high-quality and widely accessible data transmission is becoming increasingly intense. Low Earth Orbit (LEO) satellite networks show great attraction due to their characteristics of global coverage and low latency. Traditional terrestrial routing methods face significant challenges in adapting to LEO satellite networks due to challenges such as highly dynamic topologies, resource constraints, and insufficient multi-objective optimization capabilities. Therefore, developing routing methods suitable for LEO satellite application scenarios is crucial for further improving network transmission performance and is also one of the key technologies of future 6G. Compared with traditional algorithms, routing algorithms based on machine learning (ML) are more intelligent and begin to show obvious performance advantages, and are more suitable for 6G networks. However, in existing research work, there is a lack of comprehensive analysis content on integrating ML into LEO satellite network routing tasks. We comprehensively summarize the latest progress of intelligent routing algorithms based on ML in LEO satellite networks from four aspects: routing models, design challenges, training and deployment, and future research directions. The aim is to provide theoretical support for the design of artificial intelligence satellite communication systems and further promote the innovative development of satellite network optimization technologies.

1. Introduction

Currently, communication network have become essential infrastructure for daily life and production processes. Serving as the central hub of information and data transmission, these networks enable efficient connections, thereby facilitating global economic, social, and cultural activities. The growing global connectivity, coupled with the development of sixth-generation (6G) mobile communication standards, has driven the evolution of network technologies, towards higher bandwidth, wider coverage, and improved reliability [1]. Nevertheless, the current network infrastructure primarily depends on terrestrial equipment and optical fibers. Restricted by terrain, terrestrial base stations can only cover around 20% of the land area and about 7% of the Earth's surface, which is insufficient for achieving uninterrupted global coverage [2]. In this context, satellite networks serve as a valuable complement and alternative to traditional terrestrial communication networks, offering faster, more reliable data transmission and wider coverage. Satellite networks are classified into Geostationary Orbit (GEO) satellites, Middle Earth Orbit (MEO) satellites, and Low Earth Orbit (LEO) satellites [3]. Among them, LEO networks have attracted significant attention due to their advantages, including lower transmission delay, reduced propagation loss, lower construction costs, and broader global coverage capabilities. Therefore, LEO networks have become a focal point of interest for both industry and academia, due to their promising development prospects and significant research value [4].

In communication networks, routing protocols usually assist the source node to in finding a suitable path to the destination node by selecting an efficient next hop from intermediate nodes [5]. An effective routing scheme efficiently manages data packet transmission, ensuring quick and reliable delivery to enhance network performance and efficiency. Fig. 1 illustrates the routing process by using the LEO satellite network. The user uploads data to the LEO satellite network via a ground station. The LEO satellite network employs routing algorithms to select the optimal data transmission path and subsequently transmit the data to another ground station, which then delivers it to the target user. Traditional routing algorithms have been extensively developed and refined for terrestrial networks to accommodate diverse network environments and provide efficient routing services. However, directly applying these algorithms to satellite networks poses significant challenges due to the distinct characteristics of the satellite environment. Satellite routing involves high mobility, dynamic topology changes, constrained resources, and unbalanced network loads, which impose more stringent requirements on routing algorithms. Therefore, it is essential to develop customized routing schemes tailored to the unique characteristics of satellite networks. Such schemes can leverage the advantages of satellite communication to ensure reliable and efficient data transmission.

The integration of Space-Air-Ground networks with Artificial Intelligence (AI) is expected to be a vital aspect of

*Corresponding author.
zhuzhenyu@hdu.edu.cn (Zhenyu Zhu); raozheheng@hdu.edu.cn (Zheheng Rao); xiaoshitong@hdu.edu.cn (Shitong Xiao); yaoye@hdu.edu.cn (Ye Yao); xuyy@whu.edu.cn (Yanyan Xu); weizhi.meng@ieee.org (Weizhi Meng)

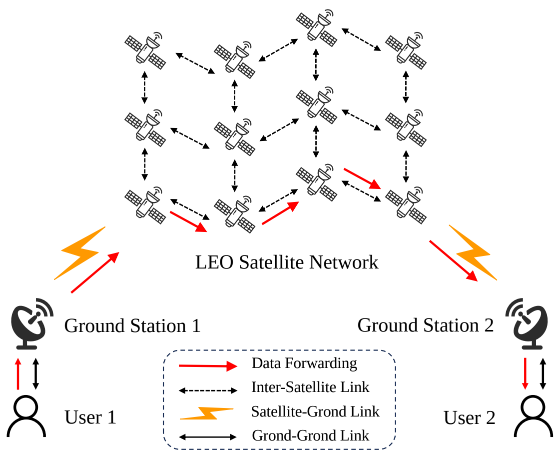

Fig. 1: Routing process using the LEO satellite network

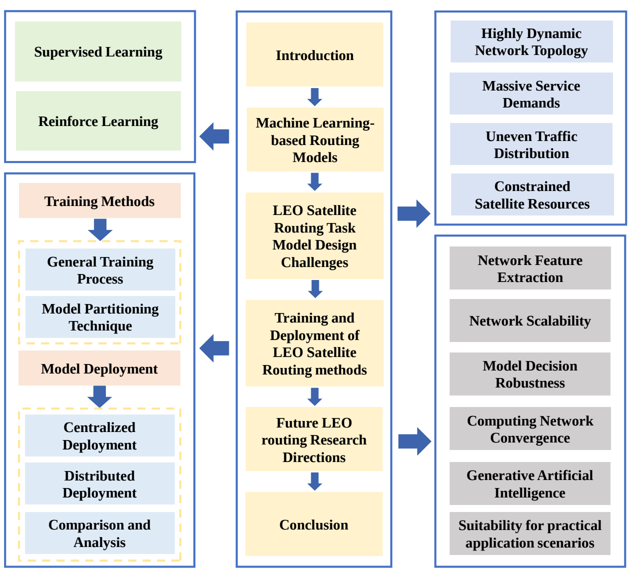

Fig. 2: Structure of the paper

future 6G technology. Currently, Machine learning (ML) is often characterized as a machine's capacity to replicate intelligent human behavior, which is the predominant approach to implementing AI [6]. Network routing is a fundamental function that ensures the quality of network services. Therefore, it is crucial to explore intelligent routing methods for LEO satellite networks that are based on ML. Intelligent routing algorithms demonstrate superior adaptability to the dynamics and complexity of satellite communication networks in comparison to traditional algorithms. These algorithms can dynamically adjust routing paths, enhance communication efficiency, and optimize system performance through real-time monitoring, data analysis, and route optimization. This achievement is attained by considering real-time data traffic, network topology, and user demand. Moreover, intelligent routing algorithms can dynamically modify routing strategies in real-time to adapt to changes in network status, various communication environments, and external interference factors, thereby ensuring communication quality and stability. The extensive implementation of intelligent routing algorithms opens up new possibilities for LEO satellite communication systems, offering crucial support for achieving more efficient and reliable satellite communications.

Comparison with state-of-the-art. Satellite networks are highly likely to be integrated into future 6G networks to support various high-quality communication needs. This emerging field has stimulated numerous studies to address the unique challenges and opportunities brought about by the development of satellite technology. However, despite these comprehensive reviews on satellite networks, there is currently no summary related to the application of ML-based intelligent routing algorithms in LEO satellite routing. For example, Shi et al. [7] reviewed state-of-the-art ML techniques tailored for 6G wireless networks, advocating for ML methods due to their superior performance, computational efficiency, scalability, and generalizability. Additionally, it addresses neural network design, theoretical tools, implementation issues, and future research directions to facilitate the practical application of ML models in 6G networks. Mahboob and Liu [8] discuss the important role of satellite networks in 6G networks and the unique challenges they face, proposing the use of AI to address issues such as latency, Doppler effects, frequent handovers, spectrum sharing, and resource allocation in satellite networks, summarizing existing research and exploring future research directions. Cao et al. [9] studied the dynamic routing problems in satellite networks, first introducing the architecture and development of satellite networks, then analyzing the latest single-layer and multi-layer dynamic routing schemes, summarizing their advantages, disadvantages, and applications, and finally discussing potential technologies and future research directions.

In contrast, this paper provides the first systematic review of the key technical challenges and solutions related to ML in LEO satellite routing, from the perspectives of ML methods, routing design challenges, and training deployment. It fills the gap in existing surveys regarding scene focus and technical depth.

Contributions. We have conducted an in-depth analysis of the AI-driven network routing solution. In the current research work, there is no comprehensive analysis content on integrating ML into the LEO satellite network routing task, so it cannot provide corresponding guidelines for researchers. To make up for this shortcoming, we have paid special attention to the research on how ML methods can automatically adjust configurations in the complex and variable LEO satellite scenario to improve network routing performance. We have also comprehensively analyzed the deficiencies of existing work and the promising research directions in the future. The purpose is to provide theoretical support for the design and optimization of AI satellite communication systems and thereby promote the innovation of satellite network optimization technologies. Table 1 shows the differences between our research work and existing work.

Table 1
Comparison of Existing Intelligent Routing Survey Papers

Reference	Shi et al. [7]	Mahboob and Liu [8]	Cao et al. [9]	Our work
Satellite Scenes		✓	✓	✓
LEO Satellite Routing Scenes				✓
Highly Dynamic Network Topology		✓	✓	✓
Massive Service Demands	✓		✓	✓
Uneven Traffic Distribution	✓		✓	✓
Constrained Satellite Resources		✓		✓
Deployment Mode				✓
Network Feature Extraction	✓			✓
Network Scalability		✓	✓	✓
Model Decision Robustness			✓	✓
Computing Network Convergence			✓	✓

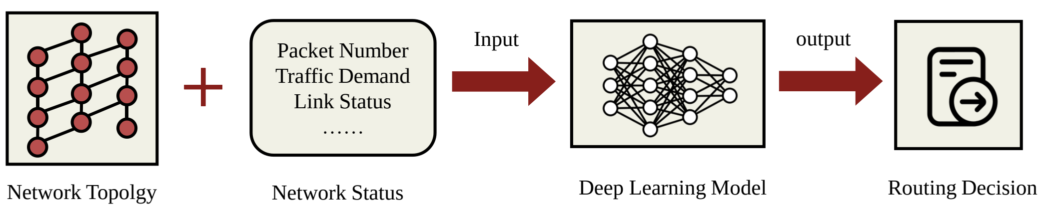

Fig. 3: Scheme of DL-based routing model

2. Machine Learning-based Routing Models

AI has brought enormous opportunities to various industries, prompting an increasing number of network researchers to start focusing on empowering network applications with AI technology. Among them, routing selection based on ML is one of the most representative research tasks. As shown in Table 2, existing routing methods based on ML can be roughly divided into two categories: algorithms based on Supervised Learning (SL) and algorithms based on Reinforcement Learning (RL). The technical principles and application scenarios of these algorithms will be discussed in detail in the following text.

2.1. Supervised Learning

SL is one of the ML methods that emerged first. Its core principle is to train model parameters using labeled input and output data to accurately fit the mapping relationship between input and output [25]. Among them, supervised Deep Learning (DL) methods are known for their strong capabilities in feature learning and representation learning. Since DL methods can effectively extract complex feature patterns from a large amount of previously accumulated empirical data, they are widely used in task scenarios such as network environment modeling, traffic load prediction, and congestion state detection. These scenarios directly or indirectly promote the intelligent decision-making process in network routing and are one of the most popular methods in current intelligent routing tasks. Fig. 3 shows the process of DL-based routing methods. The DL model can adaptively generate routing decisions according to the input network topology and network state information.

The prevalent DL model is the Deep Neural Network (DNN) [26], which processes input data by forwarding it from the input layer to the output layer. In this process, the data undergoes nonlinear transformations in multiple hidden layers to generate the final prediction results. To improve the accuracy and performance of the model, DNNs employ feedback propagation, also known as back-propagation. This technique entails back-propagating from the output layer to the hidden and input layers, using the error between the predicted result and the actual label, and then adjusting the network parameters to minimize prediction errors. The integration of feed-forward and feedback propagation mechanisms forms the training and inference processes of DNNs. For example, Barabas et al. [10] proposed a traffic prediction algorithm based on DNN was utilized and embedded within a routing management system to improve network performance. Likewise, Hardegen et al. [11] proposed a traffic classification and routing framework based on DNN, targeting traffic feature detection and routing decisions. Through continuous training and evaluation of DNNs using data streams collected from data centers, the system can predict vital characteristics of real traffic streams, such as throughput and duration, and classify these features to aid routing decisions. These intelligent routing algorithms highlight the potential of DNNs to revolutionize network routing technology, triggering further research and exploration into advanced intelligent routing algorithms.

Deep Belief Network [27] (DBN) is an advancement over the conventional DNN. A DBN is structured as a multilayer neural network comprising multiple Restricted Boltzmann Machines (RBM). The DBN training process consists

Table 2
The summary of ML-based intelligent routing algorithms

Reference	ML Algorithm	Type	Deployment Mode	Key
[10]	DNN	SL	Centralized	Routing management through traffic prediction
[11]	DNN	SL	Centralized	Assisting routing decisions by categorizing predicted traffic characteristics
[12]	DBN	SL	Distributed	Inner routers and edge routers complete the routing process together
[13]	CNN	SL	Centralized	Routing management through traffic prediction
[14]	LSTM	SL	Centralized	Assisting routing decisions and resource allocation by learning timing information from network traffic data
[15]	GNN	SL	Centralized	Predicting routing by learning the complex relationship between topology, routing, and input flows
[16]	GNN+GRU	SL	Distributed	The router interfaces are also considered as topology nodes and computed routes according to the graph
[17]	Q-Learning	RL	Centralized	Selecting paths based on the current network state and updating the Q-table based on experience and rewards
[18]	Q-Learning	RL	Centralized	Adaptively learning the best routing policy based on multiple optimization objectives
[19]	Q-Learning	RL	Centralized	Utilizing the intelligence of RL and the global view provided by SDN to calculate routes
[20]	DQN	RL(DRL)	Centralized	Enabling efficient routing through two phases: an offline network construction phase and an online deep learning phase
[21]	DQN	RL(DRL)	Distributed	Combining offline and online policies is used to make globally optimal routing decisions
[22]	DDPG	RL(DRL)	Centralized	Utilizing TE-aware exploration and actor-critic-based prioritized experience replay, to enhance the optimization of the DRL framework
[23]	GNN+DQN	RL(DRL)	Centralized	MPNN was used for capturing the relationship between links and flows in the network topology
[24]	GAT+DQN	RL(DRL)	Distributed	Utilizing GAT to process network topology and aggregate network information and make decisions using multiple agents

of two main phases: first, the DBN model parameters are initialized using RBM. Then, these parameters are fine-tuned through gradient back-propagation. This divided training method allows DBN to learn higher-level feature representations from data more effectively. Mao et al. [12] proposed an intelligent routing scheme for backbone networks, utilizing DBN as the basis. In this scheme, routing nodes are categorized into two types: inner routers and edge routers, both actively involved in the training of the DBN model. During packet forwarding, the edge router that receives the packet computes the forwarding path using these intelligent routing models. The inner router subsequently forwards the packet based on the path determined by the edge router, thereby reducing the need for frequent information exchanges as commonly seen in traditional routing algorithms. The input data of each DBN model includes the number of packets received by each node within a specific cycle, while the output data indicates the next hop. Consequently, the edge router can depend on each DBN model to generate a complete routing path step-by-step.

Traditional DNN models are limited in their ability to handle highly sparse and complexly structured data, which impairs their ability to accurately capture correlations and patterns. In contrast, Convolutional Neural Networks [28] (CNNs) are more suitable for handling graphical data structures. By integrating convolutional and pooling layers, CNNs can extract local features effectively while preserving spatial information. This allows CNNs to more efficiently capture both local patterns and the global structure of network data, demonstrating superior adaptability and generalization capabilities. Modi and Swain [13] proposed an intelligent routing mechanism based on CNNs, with the goal of optimizing the calculation of routing paths. This model utilizes historical traffic data for traffic prediction, allowing the controller to make more intelligent path combinations and routing decisions. Consequently, the CNN-based routing mechanism greatly improves network throughput and reduces average delay and packet loss, as evidenced by experimental results. Compared to traditional routing algorithms, this CNN model provides significant enhancements in overall network performance.

In routing tasks, DNNs and CNNs may face challenges in effectively handling sequential data. Specifically, tasks such as extracting path information and predicting traffic based on historical data often require techniques capable of handling sequential data effectively. Recurrent Neural Networks (RNNs) [29] are often used in these tasks because they can capture long-term dependencies in sequences, improving the accuracy of routing predictions. Among various RNN variants, Long Short-Term Memory (LSTM) [30] units stand out for their exceptional performance. LSTM addresses the inherent limitation of RNNs in handling long-term dependencies by including a memory cell state in the hidden layers, preserving long-term information. An example of this is the intelligent routing algorithm proposed by Schuster et al. [14], which integrates LSTM with a tree routing protocol. This approach improves energy consumption and network performance by taking factors such as neighboring nodes and distances into account. Moreover, LSTMs decrease end-to-end delays in routing by effectively learning and comprehending timing information in network traffic data. As a result, they can predict future traffic trends based on historical data, enabling dynamic routing decisions and resource allocation in the network.

The rapid movement mode and complex inter-satellite environment of LEO satellites lead to a tendency for the satellite network topology to present a highly dynamic non-Euclidean data structure. Traditional neural networks often perform poorly when handling non-Euclidean data of this kind. In contrast, Graph Neural Networks (GNN) [31] provide a robust solution since they are designed specifically for handling graph-structured data by efficiently extracting topological information. GNNs excel at capturing relationships between nodes and the network topology, enabling the learning and representation of node features, and integrating these features with topological data to enhance the understanding and processing of routing data. Due to their parameter-sharing property and ability to utilize the inherent local connectivity in graph structures, GNNs maintain high parameter efficiency when handling large-scale routing tasks. Rusek et al. [15] proposed a predictive model based on GNN that can understand complex interactions among topology, routing paths, and incoming traffic. This model produces accurate estimates of average delay and jitter for source/destination pairs. The ability of GNN to learn and model graph-structured data enables the model to generalize across different network topologies, routing schemes, and varying traffic intensities. Additionally, Geyer and Carle [16] proposed a distributed intelligent routing algorithm, which combines GNN with a Gated Recurrent Unit (GRU) [32]. This innovative approach highlights the potential of GNNs in revolutionizing the handling of non-Euclidean and dynamic data in satellite network topologies, paving the way for more efficient and effective routing algorithms.

With the rapid increase in network size and data traffic, intelligent routing algorithms that utilize SL models show potential in improving routing performance and making more accurate and efficient routing decisions. However, these intelligent routing algorithms often face challenges related to limited scalability and lack of interpretability. To tackle these challenges, an encouraging direction for the future development of intelligent routing algorithms could involve integrating traditional route optimization techniques with SL models. This combination can leverage the advantages of DL models in handling large-scale data and identifying complex patterns, while also benefiting from the stability and interpretability of traditional routing algorithms, resulting in more efficient and dependable routing decisions.

As shown in Table 3, the comparison of SL models highlights that, in practical applications, the selection of an appropriate model should align with the task type and data characteristics. For instance, tasks requiring the processing of high-dimensional structured data, such as traffic prediction, may benefit from the DNN model. In contrast, for small-sample scenarios, such as distributed routing coordination, the DBN model, which enhances feature extraction capabilities through hierarchical pre-training, is better suited to meet these demands. Additionally, the CNN model excels in handling spatially correlated data, making it particularly suitable for tasks like path optimization and regional congestion detection. When addressing time-series data tasks, the LSTM model demonstrates superior performance due to its ability to model temporal dynamic variations. Meanwhile, the GNN model is ideal for routing selection in highly dynamic scenarios, especially when processing non-Euclidean data. Therefore, in practical implementations, the choice of model should be determined by specific task requirements, the structural characteristics of the data, and considerations of computational complexity.

2.2. Reinforce Learning

In RL, an agent learns to maximize long-term cumulative rewards in a dynamic environment by taking actions that interact with the environment. This process includes the agent perceiving the current state of the environment, selecting an appropriate action, receiving feedback in the form of a reward signal, and adjusting its strategy accordingly.

Q-learning [33] is a classic RL algorithm. It can solve dynamic decision-making problems in unknown environments by learning the optimal action-value function (Q). The essence of the LEO satellite routing task is to make reasonable routing choices under complex network topologies and dynamic network states, which belongs to the optimal decision-making problem in a dynamic environment. Therefore, this kind of method can also provide a feasible new solution for routing tasks[34]. In the initial stage, Q-learning was introduced to the packet routing task through the development of the Q-routing approach [17]. This method utilizes a scalar Q-value table to record the optimal path and corresponding values from each node to the target node, enabling nodes to update the table based on the current network state, experience, and rewards. Through iterative learning, Q-routing enables network nodes to progressively determine the optimal routing strategy. Building on this foundation, a least squares RL-based adaptive routing method was proposed [18]. Prominent features of this technique include its fast convergence, efficient data utilization, and resilience to initial settings, rendering it a robust solution for adaptive routing. Furthermore, subsequent research explored the integration of Software-defined Networking (SDN) with RL-based network routing algorithms [19]. By utilizing the SDN architecture, the RL model takes link state information as

Table 3
Comparison of supervised learning models

Model Type	Core Advantages	Application Scenarios	Computational Complexity	Literature Example
DNN	Strong feature learning ability, suitable for high-dimensional structured data	Flow prediction	High (requires multi-layer continuous training)	[10]
DBN	Layer-wise pre-training enhances feature extraction ability, suitable for small sample scenarios	Distributed routing coordination	Medium (stage-wise training reduces complexity)	[12]
CNN	Local feature extraction, suitable for spatially related data	Route composition optimization, holding zone detection	Medium (convolution kernels reduces computational cost)	[13]
LSTM	Time series model, captures long-term sequence state dynamics	Flow forecasting, dynamic route adjustment	High (model structure increases computational burden)	[14]
GNN	Graph construction model, directly processes non-Euclidean relationships	High-speed routing, cross-link relationships	High (requires large-scale computing resources)	[15]

an input. This approach leverages the intelligence of RL and the comprehensive perspective provided by SDN to pre-compute optimal routes on forwarding devices, thus improving the overall efficiency and effectiveness of network routing.

In recent years, researchers have been increasingly exploring the application of Deep Reinforcement Learning (DRL) techniques in intelligent routing designs, driven by the rapid advancements in deep learning technology. DRL combines the strengths of DL and RL, making it an attractive approach from an algorithmic perspective. In comparison to traditional RL, DRL methods possess the ability to learn more intricate strategies, effectively tackling routing optimization problems associated with larger state spaces, extensive decision spaces, and complex optimization objectives.

A classical approach in DRL is DQN [35], where DNNs are utilized to approximate the estimated Q-function instead of traditional Q-tables. By training DNNs, DQN can learn optimal routing policies for routing tasks. This method overcomes the limitations of traditional Q-value tables, particularly in scenarios involving large state spaces and continuous action spaces. Yao et al. [20] presented an efficient routing algorithm based on DQN, specifically designed to tackle the problem of energy wastage that may not constantly operate at full capacity. The DQN algorithm improves routing efficiency through a two-phase process: an offline network construction phase and an online deep learning phase. During training, the algorithm achieves significant energy savings and load balancing for controllers, showcasing comparable energy savings while noticeably reducing computation times compared to traditional solvers and heuristic algorithms. Furthermore, Su et al. [21] proposed an adaptive energy and delay-aware routing algorithm based on deep Q-networks. This approach combines offline and online strategies in the DQN algorithm to make globally optimal routing decisions. The algorithm chooses the node with the highest Q value as the transponder, taking into account the energy consumption and delay based on the energy and network state of the nodes during various communication phases. Additionally, an in-line policy approach can adapt to changes in network topology and generate new routing decisions if there is a failure in the current route. This method not only reduces energy consumption and ensures strict latency constraints but also prolongs network lifetime, thereby improving overall performance in terms of latency and energy efficiency.

In comparison to the value function-based DQN approach, the Deep Deterministic Policy Gradient (DDPG) algorithm [36], which is based on policy gradient, is better suited for managing continuous action spaces and meeting the requirement for deterministic policies. Unlike DQN, DDPG directly learns policy functions and can produce actions without estimating their values. This capability enables DDPG to effectively handle actions that require continuous adjustment, such as the continuous optimization of paths in network routing. By offering determined action choices, DDPG improves the rationality and stability of routing selection. Xu et al. [22] investigates the utilization of DRL in the in-domain Traffic Engineering (TE) problem and suggests a traffic engineering scheme that relies on DDPG. The algorithm maximizes a utility function to achieve route completion, utilizes DNNs to understand the network environment, and employs DDPG to make routing decisions accordingly. The study introduces two novel techniques, TE-aware exploration and actor-critic-based prioritized experience replay, to enhance the optimization of the DRL framework for TE problems. Moreover, the research suggests that applying DDPG directly to TE problems does not result in satisfactory performance and emphasizes the need for targeted improvements to DDPG to achieve the desired outcomes.

Given the growing recognition of the high efficiency of GNNs in managing topology, there has been increased research interest in integrating GNNs with DRL to tackle routing problems. As shown in Fig. 4, the fusion strategy of GNN and DRL primarily involves using GNN to process and learn graph-structured data to extract network topology information, which is then fed as input into the DRL algorithm for path selection and decision optimization. In this

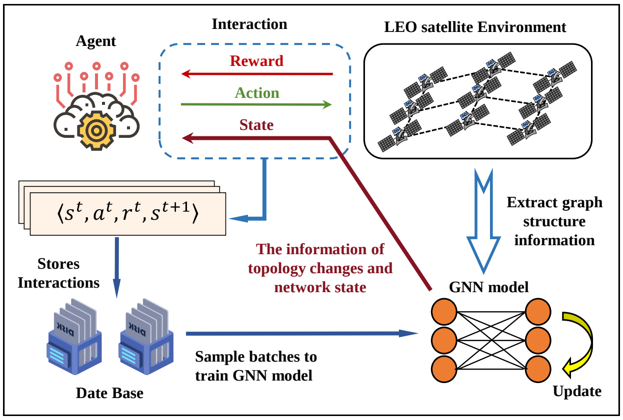

Fig. 4: Illustration of the structure of the hybrid GNN and DRL algorithm

process, GNN employs methods such as GCN to learn rich structural features from the relationships between nodes and edges, capturing complex topology changes and dynamic information. Subsequently, DRL algorithms, such as DQN or DDPG, leverage these topology features to make intelligent decisions, learning the optimal strategy through interaction with the environment. During the training process, DRL updates the policy based on the network state information extracted by GNN, optimizing overall performance. The fusion of these two approaches enables the model to make efficient decisions and plan in dynamic and highly complex environments, particularly in scenarios with high-dynamic topologies, such as satellite networks.

For example, Almasan et al. [23] proposed an architecture that combines GNN and DRL is proposed to optimize routing in communication networks. This framework, inspired by Message-Passing Neural Networks (MPNNs), designs a GNN model tailored to the routing optimization problem, capturing the relationship between links and traffic in network topology. By utilizing GNNs, this approach optimizes route paths efficiently and enhances its generative capacity on novel network topologies, making it applicable in various scenarios. To further leverage the graph structure of network topology, additional studies have utilized the Graph Attention Network (GAT) [37] in intelligent routing algorithms. GAT not only preserves global graph structure information by incorporating the attention mechanism but also enables weighted processing of neighboring nodes for each node, thus capturing essential inter-node relationships. Mai et al. [24] proposed a GAT-based intelligent routing algorithm, which utilizes a Multi-Agent Reinforcement Learning (MARL) model. This model regards each router as an independent agent, enabling it to make routing decisions based on local observations. By adopting this GAT-based approach, network topological structures are effectively managed, and network information is transferred to assist each router in extracting representations of its relationships with neighboring nodes. These relationship representations enable routers to make well-informed routing decisions from their local perspectives, thereby improving their ability to adapt to dynamic network requirements.

Despite the significant potential of GNN-based DRL approaches in addressing network topology, there are still several unresolved issues. Over-smoothing and over-compression present notable challenges among these issues [38]. Adversely, these issues can affect the performance and generative capacity of DRL models. Therefore, future research should prioritize resolving these challenges to enhance the performance and broaden the applicability of DRL approaches in routing scenarios.

3. LEO Satellite Routing Task Model Design Challenges

While LEO satellite networks bring new opportunities, they also face new challenges. As shown in Fig. 5, from a scenario perspective, the newly developed routing methods need to comprehensively consider the highly dynamic network topology, huge service demands, their unbalanced traffic distribution, and limited satellite resources.

3.1. Highly Dynamic Network Topology

Table 4 provides an overview of the intelligent routing algorithms that have been developed to address the dynamic nature of LEO satellite network topology.

On one hand, the rapid movement of LEO satellites brings about highly dynamic topology and link state changes. Continuous movement will cause the communication windows between satellites and ground stations to be continuously activated and closed, resulting in unstable links. In addition, when satellites pass through polar regions, intentional link interruptions occur, leading to frequent changes in the network topology between satellites. Therefore, a large number of frequent calculations must be performed to ensure that routers can adapt to these changes. These factors contribute to the complexity and time-consuming nature of router management in LEO satellite networks.

Liu and Wang [39] presented an intelligent routing algorithm specifically designed for LEO satellites to handle their high dynamics. This algorithm employs a dendritic neural network in conjunction with the Ant Colony Optimization (ACO) algorithm to make route choices and monitor the link status between satellites. During the link status awareness phase, the algorithm analyses visual constraints between satellites and periodically assesses the conditions of these links. In the neural network link quality awareness phase, the algorithm processes the link state information between satellite nodes using the dendritic neural network, generating estimated value matrices for route choices that satisfy various Quality of Service (QoS) criteria. By integrating these estimated value matrices with Dijkstra's algorithm, the intelligent routing algorithm calculates the initial path from the source node to the destination node. To handle the dynamic topology problem, the algorithm adjusts the calculated path in real-time by continuously monitoring the satellite network topology.

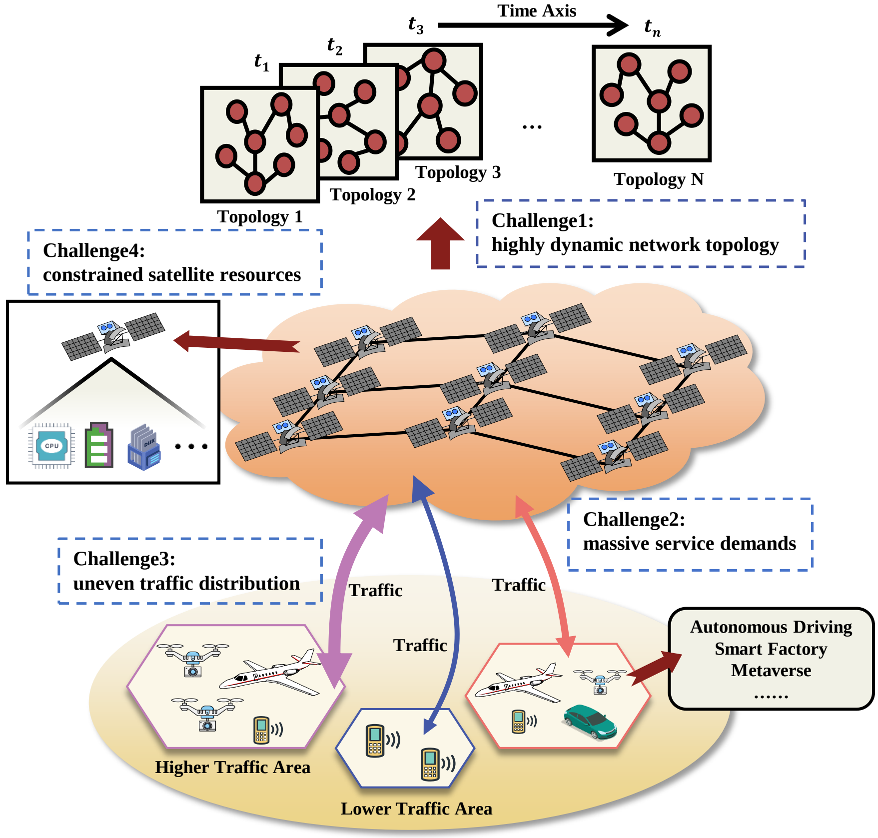

Fig. 5: LEO satellite routing task model design challenges

Table 4
The summary of solutions to the problem of high dynamic network topology

Reference	ML Algorithm	Deployment Mode	Routing Policy	Results
[39]	Dendritic Network	Distributed	Setting Link Weights	Achieved lower end-to-end path delay, delay jitter, and packet loss rate
[40]	Q-Learning	Distributed	Generating Next-Hop	Provide long-term Quality of Service optimization during the routing maintenance process
[41]	DGCN	Distributed	Generating Next-Hop	Using DGCN to capture the dynamic characteristics of LEO satellite networks and generate heuristic information
[42]	Q-Routing	Centralized	Generating Routing Path	Accelerate the convergence and retain the strong real-time performance of the Q-routing algorithm
[43]	Q-Learning	Distributed	Generating Next-Hop	Improved the problem of long convergence time and sometimes falls into local convergence

Huang et al. [40] employed a RL method to solve the routing problem in dynamic topologies. This method consists of three key modules: neighborhood discovery, router discovery, and router maintenance. The neighborhood discovery module periodically sends "hello" packets to neighboring satellites to exchange node information. Before a satellite enters orbit, the router discovery module is called to start the routing strategy model. In this process, the router maintenance module is responsible for making packet transmission decisions based on the initial Q-table and updating the above Q-table after changes in network topology and state information, thus achieving a balance between end-to-end delay and network traffic cost.

Wang et al. [41] further expanded the ability of LEO satellite networks to handle dynamic environments. In response to the challenges posed by 6G technologies, they proposed a scheduling mechanism based on cycle-specified queuing and forwarding. This mechanism resolves the strict time constraints of traditional methods through cyclic multi-queue scheduling, enabling packets to be transmitted within flexible time cycles and mitigating issues caused by link instability. At the same time, their proposed learning-based swarm intelligence method, which combines Dynamic Graph Convolutional Networks (DGCN) with Adaptive ACO, captures the dynamic characteristics of LEO satellite networks and generates heuristic information. This approach enhances the flexibility and adaptability of routing schedules, effectively addressing the challenges posed by changes in network topology.

On the other hand, the high dynamics of the LEO satellite network topology also poses a huge challenge to the training efficiency of ML algorithms. Intelligent models must achieve efficient convergence of routing algorithms within a limited time after topology changes.

To achieve the above goals, the current mainstream methods can be divided into two categories. The first category is to improve convergence performance by minimizing path calculation time as much as possible, such as introducing prior knowledge for pre-computation of routing. For example, Zheng et al. [42] designed an improved Q-routing algorithm assisted by the Dijkstra algorithm. At this time, through the simple calculation of the Dijkstra algorithm, pre-computation can be achieved before the data packet arrives. Although storing pre-computed routes consumes satellite storage space, it effectively reduces routing calculation time and thus speeds up the convergence of the Q-routing algorithm.

The second category of methods mainly reduces unnecessary calculation time by limiting the maximum number of hops of data transmission. Wang et al. [43] proposed a method that combines hop limit and dynamic greed rate. Initially, this method uses Q-learning to train agents so that they can dynamically adjust routing decisions according to real-time network states and requirements, thereby addressing the frequent changes in topology in LEO satellite networks. To alleviate the problem of excessive data packet transmission in the initial stage, this method introduces a strategy of limiting the number of transmission hops. When a data packet cannot reach the destination within the specified hop range, the path search attempt will be regarded as a failure and further exploration will be terminated, thereby speeding up algorithm convergence. On this basis, in order to avoid the algorithm falling into the local optimal trap, the dynamic greed rate strategy is incorporated into this work. At first, an extremely low greed rate can enable agents to fully understand the global network topology. As iterations progress, the greed rate gradually increases and iterates repeatedly on the previously explored paths to determine the optimal path.

Despite the relatively effective solutions proposed in the aforementioned studies to address the routing convergence problem in high-dynamic topologies, such as precomputed paths [42] and dynamic greedy rate adjustments [43], two core limitations remain. First, pre-computation relies on static historical data, which makes it difficult to adapt to sudden link failures (e.g., unexpected topology changes caused by space debris collisions). Second, online training of RL algorithms incurs significant overhead, potentially exacerbating the resource burden on the satellite. Future research should explore lightweight incremental learning frameworks, such as combining meta-learning to achieve rapid adaptation across scenarios, or utilizing federated learning to share local experiences among distributed nodes, reducing the frequency of global model updates. Moreover, Spatiotemporal Graph Neural Networks (STGNN) offer new perspectives for dynamic topology modeling, with their spatiotemporal attention mechanism further enhancing the accuracy of link state prediction.

3.2. Massive Service Demands

The evolving field of satellite network research constantly generates large-scale service requirements, with increasingly diverse demands for network transmission tailored to various services. For instance, bidirectional conversational services necessitate extremely low latency, whereas unidirectional data traffic services require high packet loss rates and substantial bandwidth [44]. Currently, researchers have proposed solutions leveraging the specific characteristics of LEO satellite networks. Nevertheless, as wireless communication technology advances, user requests continue to expand and diversify, leading to increased service traffic and a broader spectrum of services utilizing LEO satellite networks. Consequently, it becomes essential for LEO satellite routing research to ensure the fulfillment of diverse service requirements such as minimal delay, adequate bandwidth, and optimal throughput in response to user demands. Addressing the extensive service requirements of satellite networks demands the development of more flexible and intelligent routing algorithms capable of adapting to the increasingly varied service needs of users. Table 5 summarizes the intelligent routing algorithms designed to address the massive service requirements of LEO satellite networks.

From the perspective of QoS, Wu et al. [45] proposed a supervised load balancing strategy. By defining QoS in four dimensions, this strategy can effectively distinguish the satellite traffic demands of different categories. The introduction of cooperation strategies can provide more accurate network status information for routing algorithms and other management strategies. On this basis, Software Defined Networking is introduced for centralized management of big data flows to ensure flexible and fine-grained QoS provision. For example, the data traffic involved in QoS type 1 has strict requirements for bandwidth, packet loss rate, latency, and jitter. It usually comes from government or military users, or high-value applications such as video conferences. In contrast, QoS type 4 involves traffic with the lowest priority, such as web browsing and email services. By classifying these different QoS types, this research can develop a more accurate link cost model, thereby promoting enhanced QoS guarantees.

Table 5
The summary of solutions to the problem of massive service demands

Reference	ML Algorithm	Deployment Mode	Routing Policy	Results
[45]	SVR	Centralized	Generating Routing Path	Achieves better QoS guarantee
[46]	DDQN	Distributed	Generating Next-Hop	Optimized the end-to-end delay, throughput, and packet loss rate
[47]	GNN+DQN	Centralized	Generating Routing Path	Optimized the end-to-end delay, throughput, and packet loss rate, and achieved a certain level of generalization
[48]	Actor-Critic	Centralized	Generating Routing Path	Improved average throughput and reduced average delay
[49], [50]	DQN	Distributed	Generating Routing Path	Fulfilled diverse service requirements
[51]	DDPG	Distributed	Generating Next-Hop	Leverage the processing capabilities onboard satellites to optimize the transmission delay in order to meet the QoS requirements of delay-sensitive applications
[52]	Double Q-Learning	Distributed	Generating Next-Hop	Can operate efficiently and securely in complex environments, increasing the delivery ratio and reducing the average delay and overhead ratio
[53]	D3QN	Distributed	Generating Next-Hop	Reduce packet loss rates when the system contains malicious nodes

In contrast, currently more studies rely on RL to address the QoS challenges in satellite networks. For example, Wang et al. [46] proposed a two-hop state-aware routing strategy based on DRL. Compared with the hop-by-hop sensing strategy, it has a larger receptive field and can alleviate some routing cycle problems. In addition, a more advanced Double Deep Q Network (DDQN) RL framework is introduced, which can effectively improve the decision-making ability of the routing model and thus ensure a more excellent QoS performance. Experimental results show that this strategy has achieved outstanding performance in indicators such as end-to-end delay, throughput, and packet loss rate. Wang et al. [47] combined a GNN with stronger network state perception ability and the DRL method to further optimize the feature extraction model of satellite network state and achieve the goal of maximizing long-term average throughput and minimizing average delay. Magadum et al. [48] proposed a DRL method based on a behavior-criticism network to guide network routing decisions. By designing specific state and action spaces and incorporating relevant QoS parameters such as packet loss rate, available bandwidth, and delay into the routing decision-making process, it can effectively improve the network congestion state and enhance service quality. Zhou et al. [49] proposed a hierarchical RL satellite routing method. By designing a hierarchical network architecture including LEO and MEO, and a regional division strategy, the convergence and decision-making efficiency of satellite routing can be effectively improved. On this basis, Mao et al. [50] further designed a multi-agent collaborative RL strategy to avoid resource usage conflicts caused by routing selection among different applications, effectively generate routing paths for different applications, and comprehensively enhance the overall performance of network services.

In contrast to the approaches mentioned above, another perspective is to leverage the processing capabilities of satellites to reduce data transmission volume, thereby meeting the QoS requirements of delay-sensitive applications. Building on this idea, He et al. [51] proposed an intelligent routing and resource allocation method based on DRL, which aims to reduce data transmission volume and improve routing efficiency by selecting the most suitable transmission path and utilizing onboard processing capabilities. Specifically, the proposed DQN-based Intelligent In-Orbit Routing algorithm selects routing paths with good channel conditions, sufficient energy, and low transmission load to minimize transmission delay while ensuring the rational allocation of energy and resources. Furthermore, the paper introduces a DDPG-based Intelligent Resource Allocation algorithm, which aims to achieve intelligent and continuous resource allocation. By considering the interaction between potential fields and resource states, the algorithm ensures that selected satellites can process data, further reducing transmission delays. Extensive simulation results demonstrate that these methods significantly reduce transmission delay and packet loss rate, effectively meeting the QoS requirements of satellite networks.

For users who are considering choosing satellite network services, security is undoubtedly a service requirement that they are extremely concerned about. Potential threats such as malicious nodes, attacks, link eavesdropping, and interference can seriously affect the security and reliability of satellite network data transmission. Therefore, formulating effective solutions to these challenges should also become a key link in the design of routing algorithms. Zhou et al. [52] proposed an adaptive routing strategy. By analyzing Q values, rewards, and discount coefficients, factors that can represent node and link states such as network congestion degree and node hop count are modeled, effectively enhancing the network's ability to respond to dynamic changes and attacks. Experimental results confirm that in a highly dynamic LEO satellite environment, this strategy can significantly improve the efficiency and security of routing tasks. In addition, Song et al. [53] deeply explored the common security problems in satellite networks and proposed a trusted and load-balanced routing scheme to deal with the security threats brought by malicious nodes and attacks. This scheme uses a multi-agent adversarial Double Deep Q-Network Learning algorithm (D3QN) to construct a fully distributed routing protocol, in which decisions are based on the trust value of nodes. The trust value is evaluated through the historical behavior and performance of nodes and can be used as an indicator of its credibility in the satellite network. Therefore, this scheme can effectively reduce the impact of malicious nodes and attacks, and at the same time enhance the network's adaptability to dynamic changes. On this basis, in order to further meet the individualized requirements of different satellite applications for service quality, this method introduces a variable delay constraint into the load minimization objective function, optimizing the load balancing performance and service quality. Experimental results show that the proposed routing scheme greatly reduces the link queue utilization rate and improves the system's processing ability for delay-sensitive services. In addition, it also greatly reduces the packet loss rate, especially in the case of malicious nodes, thus providing a reliable guarantee for the safe data transmission of satellite networks.

Despite existing approaches [49, 50] achieving diversified service demand routing through hierarchical RL and multi-agent collaboration, their limitation lies in the reliance on manually defined QoS labels (e.g., bandwidth, latency thresholds) for service classification, which makes it difficult to adapt to unknown or hybrid service scenarios (such as the coexistence of augmented reality and IoT data streams). In the future, unsupervised or semi-supervised learning techniques could be incorporated to automatically mine the latent features of service traffic and design multi-objective optimization frameworks to dynamically balance routing based on different service demands. Furthermore, routing algorithms considering security, such as trust-based methods [53], often introduce additional computational overhead, potentially affecting system real-time performance. To address this challenge, future work could explore the deployment of lightweight security mechanisms, such as combining homomorphic encryption and federated learning, to enhance both security and system efficiency while maintaining real-time capabilities.

3.3. Uneven Traffic Distribution

LEO satellite networks experience a severely imbalanced traffic distribution, particularly in high-latitude and densely populated regions. The primary cause of this phenomenon is the uneven distribution of satellites and ground gateway stations, which leads to the coexistence of overloaded areas and relatively idle ones. Moreover, due to the high-speed movement of satellites, these heavily loaded coverage areas can swiftly shift from one satellite to another. Such an unbalanced network load gives rise to network congestion and performance degradation, thereby exerting an adverse impact on communication quality and user experience. Consequently, it is essential to study and implement effective load balancing techniques to optimize the utilization of satellite network resources and enhance network performance and stability. Table 6 presents an overview of intelligent routing algorithms designed to address the issue of uneven service distribution in LEO satellite networks.

On one hand, researchers concentrate on extracting more accurate traffic patterns from the perspective of feature modeling. For instance, Kato et al. [54] have achieved network load balancing by leveraging deep learning technology, especially in satellite networks handling multi-source and multi-destination data transmissions. In such networks, the selection of efficient path combinations is of utmost importance for maximizing network performance and preventing congestion. This study utilizes CNN to model, select, and dynamically adjust path combinations based on real-time traffic patterns. This approach enables the network to adapt flexibly to diverse transmission requirements, effectively avoiding congestion and improving the overall performance of the network. In response to the limitations of traditional neural networks in modeling network topology data, He et al. [55] proposed a novel method that combines the GNN structure with the DDPG approach. This method, known as "message-passing DRL", allows GNN to more precisely perceive the network load state through information exchange during the message passing process. As a result, this method can make more effective use of network environment information to achieve load balancing and performance optimization of network traffic.

On the other hand, more intelligent traffic scheduling models have been further developed. Dong et al. [56] proposed a load balancing routing algorithm based on DQN. This algorithm models the satellite traffic load scheduling process as a Markov decision process and utilizes the DQN strategy to realize the dynamic adjustment of traffic load scheduling. Experimental results demonstrate that this algorithm can reduce congestion occurrence rate, successfully achieve network load balancing, and thus obtain better routing performance. Huang et al. [57] introduced an intelligent multi-path traffic scheduling method based on DDPG, aiming to enhance autonomous and efficient communication in LEO satellite networks. This method employs enhanced pheromones to describe the changing state of network traffic, which facilitates the timely identification of traffic overload

Table 6
The summary of solutions to the problem of uneven traffic distribution

Reference	ML Algorithm	Deployment Mode	Routing Policy	Results
[54]	CNN	Centralized	Generating Routing Path	Achieved satellite traffic balance
[55]	GNN+DDPG	Centralized	Setting Link Weights	Obtains better performance by improving the overall load balancing performance, reducing the end-to-end delay, and improving the utility of the network
[56]	DQN	Distributed	Generating Next-Hop	Reduces queue utilization and congestion to achieve load balancing
[57]	DDPG	Distributed	Generating Next-Hop	Adaptive Traffic Scheduling and Reduced Network Load
[58]	Q-Learning	Distributed	Generating Next-Hop	Offloads the scheduling tasks into the accessing IoT nodes, realizes fast congestion control, and improves QoS
[59]	Q-Routing	Centralized	Generating Routing Path	Effectively optimizes average latency, packet arrival rate, and network load balancing in dynamic environments
[60]	GAT	Centralized	Generating Routing Path	Achieves low computational complexity in dynamic environments with frequent service requests, while maintaining high service acceptance and load fairness

paths in dynamic network topologies. On this basis, the advanced DDPG framework is utilized to achieve better traffic load scheduling optimization compared to DQN, ensuring the system's adaptability to the complex LEO network environment and thereby improving the overall communication efficiency and performance.

It is noteworthy that in LEO satellite networks, high latency and high interruption probability of data transmission are inevitable. To alleviate these problems, some studies have proposed the concept of Delay Tolerant Networks (DTN). However, DTN requires sufficient storage resources to ensure the store-and-forward process, and the storage capacity of satellite networks is limited, making it prone to congestion and service quality degradation due to traffic overload. To address this issue, Wang et al. [58] proposed a fast congestion control algorithm called "Finite Greedy". This algorithm integrates the use of multi-hop packet-switched links into DTN opportunistic routing through an incomplete information game mechanism and RL. By migrating the scheduling task to each satellite node, the Finite Greedy algorithm achieves fast congestion control, thereby enhancing network efficiency and stability.

Apart from the uneven traffic distribution caused by the distribution of users, the rapidly moving LEO satellites can also disrupt the balance of traffic load, leading to additional deterioration of network performance. Therefore, the decision-making model needs to converge quickly to adapt to these dynamic network requirements. Considering the continuous satellite movement and the evolving user requirements, the demands for network transmission also frequently change. Therefore, the routing decision model needs to rapidly converge to align with these dynamic network needs.

Ding et al. [59] proposed an enhanced learning satellite routing algorithm aiming for rapid convergence. This method conceptualizes each satellite node as an intelligent entity that can jointly perceive and quickly understand the changes in network link states in a distributed architecture and adaptively adjust its forwarding strategy through local updates to accommodate the dynamic changes in link configurations. Experimental results show that this algorithm significantly reduces the average latency, improves the packet arrival rate, and achieves effective network load balancing, demonstrating its excellent performance.

In the face of the challenges posed by dynamic network changes and frequent service requests, it is crucial to ensure service continuity and reasonable resource allocation while maintaining low computational complexity. To address the issues of load imbalance and service interruptions caused by the dynamic variations in satellite orbits, propagation delays, and wireless environments in LEO networks, He et al. [60] designed a hierarchical RL approach based on GAT. Regarding the load balancing problem, the paper takes into account the dynamic characteristics of LEO satellite networks, such as satellite orbit changes and frequent service requests. The authors employ a greedy mechanism for embedding Virtual Network Functions (VNFs) and propose a Tabu search algorithm to optimize the migration of VNFs, thereby ensuring service continuity and equitable load distribution. The GAT-based low-complexity RL method proposed in this paper effectively achieves low computational complexity in dynamic environments with frequent service requests, while maintaining high service acceptance and load fairness.

Existing solutions have made some progress in addressing the issue of uneven traffic distribution in LEO satellite networks, but certain limitations remain. For instance, [54] employs deep learning techniques to extract network traffic patterns and utilizes CNNs for path selection and dynamic adjustment. While these methods can flexibly accommodate multi-source, multi-destination data transmission, they typically rely on static network topologies and historical traffic data, which poses challenges in responding to the dynamically changing network environment (such as frequent positional shifts between satellites). Moreover, existing solutions often focus on traffic load balancing but may not account for the rapid fluctuations in user demands and network conditions, leading to suboptimal convergence times for decision-making models. Future optimization could involve the development of more adaptive algorithms that integrate multiple intelligent techniques with real-time satellite and traffic data to enhance load balancing, reduce latency, and improve system stability. These methods should be designed with scalability and low computational complexity in mind to efficiently handle the high dynamics and frequent service requests inherent in LEO satellite networks.

3.4. Constrained Satellite Resources

Properly resolving the issue of limited satellite resources is of utmost significance for ensuring the reliable operation and realizing the sustainable development of satellite communication systems. At the current level of hardware development, satellite resources, encompassing aspects such as energy, communication resources, computing resources, and storage capacity, all face numerous limitations, which have a severe impact on the performance and reliability of satellite communication systems. Considering that satellites mainly rely on solar panels to provide energy during operation, the instability of energy supply (due to multiple factors such as weather conditions, time, and the satellite's location) poses a challenge to the reliability of satellite communication. There is an urgent need to optimize energy management strategies to extend the service life of satellites and ensure the stable operation of the entire system. In addition, satellites are also greatly constrained in terms of communication resources, computing resources and storage capacity. These limiting factors seriously hamper the efficiency and performance of satellite communication systems, especially in situations where a large amount of data needs to be processed, and this hindrance is particularly evident. Therefore, when designing and operating satellite routing algorithms, effective strategies and technical means should be combined to overcome these resource limitations and thereby enhance the performance and reliability of the system. Table 7 provides a brief overview of intelligent routing algorithms developed to address the resource-constrained problem in LEO satellite networks.

Solution to energy supply and usage issues: Shi et al. [61] designed a routing scheme based on RL and set the remaining energy and bandwidth utilization as routing constraints. However, this method only takes remaining energy as one of the constraints and does not give priority to reducing the overall energy consumption of satellites. Under this premise, Lyu et al. [62] proposed a dynamic routing strategy. By combining the Lagrange multiplier method with RL, they achieved the minimization of packet transmission delay under the constraint of overall energy consumption. Y H et al. [63] proposed an event-driven DRL method, which expands the granularity of performance optimization to event-driven and further improves the energy usage rate of satellite systems.

Solution for resource optimization (communication, computing and storage). Qiu et al. [64] used DRL methods to jointly manage and allocate network, cache and computing resources between satellites and ground users, aiming to solve the problem of resource optimization. On this basis, Wang et al. [65] integrated GNN into the satellite resource management strategy, effectively improving the modeling ability of complex heterogeneous resources. While ensuring a further increase in satellite resource utilization, better QoS performance is achieved.

Joint optimization solution for resources and energy. Although the aforementioned resource optimization methods have made certain progress in resource utilization and network performance, they have not fully addressed the impact of energy expenditure on satellite systems. To overcome this limitation, Liu et al. [66] proposed a high-energy-efficiency routing protocol based on DQN. This protocol takes into account the energy level and aging rate of satellite batteries while optimizing resources, thereby achieving effective resource management over a longer time interval. In addition, Wang et al. [67] also proposed a dynamic laser interstellar link scheduling algorithm. This algorithm combines multi-agent DRL technology with compressed sensing technology to reduce the energy expenditure in communication, storage, and computing processes.

Existing resource optimization schemes, such as those presented in references [66, 67], have improved resource utilization through model compression and energy consumption constraints. However, these designs are often limited to a single resource dimension (e.g., energy or computation) and fail to fully account for the coupled effects of multi-resource joint optimization (e.g., the trade-off between communication energy consumption and computational load). Future research should focus on developing cross-domain joint optimization models that comprehensively consider the interdependencies between multiple resources, particularly in satellite networks with limited resources and diverse demands. A promising research direction is to use game theory to model the inter-satellite resource competition. Through game-theoretic models, the impact of different resource allocation strategies on the entire system can be effectively analyzed, leading to the identification of optimal resource configuration solutions.

4. Training and Deployment of LEO Satellite Routing methods

As LEO satellite intelligent routing algorithms attract increasing attention, how to effectively train and deploy

Table 7
The summary of solutions to the problem of constrained satellite resources

Reference	ML Algorithm	Deployment Mode	Routing Policy	Results
[61]	Q-Learning	Centralized	Generating Routing Path	Achieve the lowest delay under the premise of ensuring the remaining energy and bandwidth utilization
[62]	A3C	Distributed	Generating Next-Hop	Minimizes average packet latency while maximizing energy efficiency and reducing packet loss rates
[63]	DDPG	Centralized	Traffic Prediction	Long-term optimization of energy performance while satisfying end-to-end latency constraints for each data stream
[64]	DQN	Centralized	Optimization resources allocation	Enabling management and coordination of network, cache, and computing resources
[65]	GNN+DQN	Centralized	Generating Routing Path	maximize the utilization of network resources while guaranteeing the requirement of transmission delay
[66]	DQN	Distributed	Generating Next-Hop	Ability to effectively manage energy consumption over the long term
[67]	DQN	Distributed	Link Scheduling	Significantly reduce energy consumption and communication delays

them has become a crucial link in subsequent practical applications.

4.1. Training Methods

4.1.1. General Training Process

This section describes the conventional training process of ML-based intelligent routing algorithms, which generally follows a structured pipeline comprising three core phases: data preparation, model optimization, and operational deployment. The initial stage involves systematic collection and preprocessing of multi-dimensional historical network data, including but not limited to temporal link state metrics, traffic distribution patterns, and topological evolution characteristics. Through rigorous data cleansing, temporal synchronization, and spatial normalization procedures, this phase establishes a reliable foundation for subsequent algorithmic development by ensuring data integrity and temporal-spatial consistency.

In the model optimization phase, SL and RL paradigms demonstrate distinct methodological approaches. SL models rely on labeled historical datasets containing input features and corresponding optimal routing decisions, performing offline optimization by minimizing prediction errors through loss functions. This approach leverages explicit supervision signals to adjust model parameters via backpropagation, with regularization techniques like Dropout preventing overfitting to historical patterns. In contrast, RL frameworks adopt an environment-driven optimization strategy, where policy networks progressively develop routing strategies through iterative interactions with simulated satellite network environments. This paradigm emphasizes long-term reward maximization through carefully designed exploration-exploitation mechanisms, with stabilization techniques like prioritized experience replay and soft target network updates being commonly adopted to enhance training convergence.

The validation process necessitates comprehensive evaluation across diverse simulated scenarios to assess operational robustness. Test cases typically incorporate extreme network conditions such as polar region occlusion events, intermittent inter-satellite link failures, and burst traffic patterns. Notably, SL validation primarily focuses on static performance metrics, while RL evaluation extends to dynamic adaptation measurements under evolving network states.

Finally, the model is deployed. After training, the model is deployed to the actual satellite network. During deployment, it may be necessary to fine-tune the model to adapt to the dynamic changes in the real-world environment. Additionally, once deployed, the model should continuously monitor network performance and perform online learning and updates based on new network state data to maintain its effectiveness and adaptability.

4.1.2. Model Partitioning Technique

The training of LEO intelligent routing algorithm models is highly resource-intensive, and conducting this training directly within the satellite network can lead to inefficiencies due to the network's limited resources. To address this issue, the model partitioning technique offers a viable solution. Decomposing the model into different components allows for the delegation of complex training tasks to be handled by ground center clouds. The remaining parts of the training tasks are executed within the LEO satellite system. This approach not only improves the efficiency of resource utilization but also reduces response delays.

Model parallelism and data parallelism are two prevalent model partitioning techniques employed to enhance computational efficiency and resource utilization in distributed systems. Parallel methods enable the system to fully exploit all distributed computing resources, thereby expediting analysis and computation. In model parallelism [68], the model is divided into multiple sub-models, each of which is assigned to a different computational node. Each node is responsible for processing one part of the model, and to maintain global consistency, information exchange and synchronization among nodes are necessary during training. Conversely, data parallelism [69] entails replicating the same model across multiple computational nodes and distributing different batches of data to each node for training purposes. Additionally, Huang et al. [70] introduced a pipelined parallelism approach, ideal for handling a series of interdependent tasks. In this approach, tasks are decomposed into a sequence of consecutive computational stages, where each stage roughly requires an equal execution time. These stages are then connected sequentially, forming a unified pipeline. As a task progresses through the pipeline, its stages are executed in parallel across different processing units. Upon the completion of a stage, the resulting data is passed to the next processing unit, thereby maintaining a constant flow of processing.

With the continuous expansion of satellite networks, the complexity of training intelligent routing algorithms is gradually increasing. Therefore, the use of model segmentation techniques is expected to play an increasingly significant role in the development of intelligent routing algorithms for LEO satellites. In response to these escalating challenges, future research should not only account for the actual computational demands associated with training routing models but also adopt more flexible and efficient training methods. Such adaptive approaches are essential to meet the growing data processing requirements inherent in expanding satellite networks.

4.2. Model Deployment

4.2.1. Centralized Deployment

In a centralized deployment, a centralized controller is responsible for obtaining the global state and making routing decisions. All compute, storage, and network resources are consolidated in a single location or data center and are managed and maintained by this centralized controller. This architectural approach simplifies the complexity of network management and maintenance, thereby enhancing the controllability and manageability of the system. As shown in Fig. 6(a), by collecting and analyzing global state information, the centralized controller can optimize the utilization of network resources, achieving effective load balancing of traffic and enabling fine-grained routing decisions. Moreover, centralized deployment facilitates the unified management and implementation of security policies, thereby ensuring network security and reliability. The algorithms mentioned in this section are summarized in Table 8.

SDN is a flexible network architecture that facilitates centralized deployment of intelligent routing algorithms, offering numerous theoretical possibilities. SDN employs a layered approach, decoupling the data plane from the control plane of the network. This separation is achieved through network function virtualization, enabling the network to transcend hardware limitations and to be customized according to business needs for efficient orchestration and resource reconfiguration. As a result, SDN realizes fast and flexible network organization. In the data plane, satellites only need to perform straightforward data forwarding and hardware configuration functions, thereby reducing their processing load. Conversely, the control plane is capable of obtaining a global view of the network and continuously monitoring the status of network devices. This capability allows for fine-grained network management, improved routing decisions, and QoS assurance. Consequently, the entire network benefits from enhanced management, efficient resource allocation, and optimized performance. Furthermore, many SDN controllers incorporate modular design principles, allowing for the addition or removal of specific modules according to architectural designs and business requirements, thereby ensuring excellent system scalability [80].

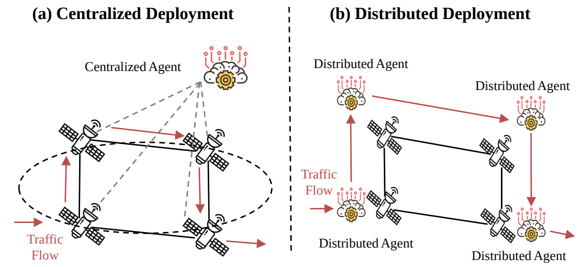

Fig. 6: The model deployment of LEO satellite routing methods

Bao et al. [81] introduced traditional SDN techniques into satellite networks, proposing the concept of Software-Defined Satellite Network (SDSN). In the SDSN framework, GEO satellites function as the control plane, MEO and LEO satellites serve as the data plane, and ground stations act as the management plane. In this configuration, the control plane is responsible for receiving commands from the management plane, transmitting them to the data plane, monitoring the real-time status of the satellite network, and sending feedback information back to the management plane. Despite its advantages, the inherently dynamic nature of satellite networks presents significant challenges to the SDSN control architecture. To address these challenges, integrating ML methods has been proposed as a solution for smarter and more adaptive network management and routing optimization. ML techniques can analyze and learn from network traffic, topology, and performance data, allowing the SDSN to adapt to changing network environments and requirements. Consequently, combining ML methods with SDSN architectures offers promising prospects for the future. Tu et al. [71] explores the integration of SDN and DRL to tackle dynamic network topology and link traffic sensing issues in satellite networks. Specifically, this approach employs the DDPG algorithm for route optimization. The DDPG algorithm enables routing decisions based on

Table 8
The summary of routing algorithms of different deployment models

Reference	Deployment Mode	ML Algorithm	Characteristics
[71]	Centralized	DDPG	Explores the integration of SDN and DRL to tackle dynamic network topology and link traffic sensing issues in satellite networks
[72]	Centralized	DQN	Model a general double-layer satellite network into the SDSN structure to manage the network in a centralized paradigm
[73]	Centralized	Chebyshev neural network	Utilizes neural network training to discern the transmission patterns of data streams, which in turn predicts routing paths
[74]	Centralized	CNN	Presented an integration of SDN, CNN, and fuzzy logic to propose an innovative multi-task routing algorithm that relies on fuzzy CNNs
[75]	Distributed	RL	proposed a distributed angular routing algorithm in time-varying dynamic LEO satellite constellation networks
[76]	Distributed	MARL	Proposed a distributed routing algorithm based on MARL, mobile and node agents work together to realize a distributed routing decision process
[77]	Distributed	DQN	Introduced spatial location information as a foundation for routing decisions
[78]	Distributed	DDQN	Proposes a decentralized flow-centric DRL approach, shifting the focus from routing individual packets to entire traffic flows
[79]	Hybrid	GNN+DQN	Decouple the complexity of cross-domain routing decisions employing hierarchical control

real-time link states, effectively addressing the challenges presented by dynamic changes in satellite network topology and link states. To enhance the ability to perceive contextual relationships between data streams, the method incorporates a neural network design that includes LSTM and fully connected layers. Experimental results demonstrate that this approach significantly improves end-to-end throughput and reduces delay. Furthermore, it exhibits superior performance in adapting to continuously changing data flows and link states. Wei et al. [72] proposed an intelligent and reliable routing framework based on centralized DRL, which further extends the routing optimization methods under the SDSN architecture. This framework aims to enhance the efficiency and reliability of packet transmission in LEO satellite networks, particularly in highly dynamic and unstable space environments. By employing centralized management, the approach fully leverages the advantages of the SDSN architecture, enabling dynamic adjustment of routing decisions, adaptation to network topology changes, and ensuring high reliability even in the case of frequent link disruptions.

The limited storage resources in satellite networks constrain the storage of SDN streaming tables, necessitating efficient storage solutions. This challenge has prompted researchers to investigate various approaches. Liang et al. [73] addresses the escalation of Ternary Content Addressable Memory (TCAM) space consumption by flow tables in SDSN. The study highlights that the increasing complexity of flow table entries and the associated lookup and matching processes result in diminished routing and forwarding efficiency, thereby failing to meet the diverse requirements of modern applications. To tackle this issue, they propose a Chebyshev neural network-based intelligent routing architecture for SDSNs. This architecture utilizes neural network training to discern the transmission patterns of data streams, which in turn predicts routing paths. By replacing traditional flow tables with a neural network, the approach not only conserves TCAM storage space but also enhances the routing and forwarding efficiency of data streams.

Wang et al. [74] presented an integration of SDN, CNN, and fuzzy logic to propose an innovative multi-task routing algorithm that relies on fuzzy CNNs. Considering the dynamic nature of satellite networks, this approach utilizes the GEO satellite and ground computing center as a unified SDN control plane. The responsibilities of the GEO satellite involve collecting load data in each cycle to create a multidimensional matrix, while the ground computing center captures historical traffic data using the GEO controller for training and updating the CNN model. Subsequently, the GEO satellite employs the trained CNN model for route planning and transmits the data stream to the LEO satellite. Moreover, recognizing that the CNN's decision-making may sometimes conflict with user requirements, this method incorporates fuzzy inference to evaluate the task requirements. This integration improves the CNN's output efficiency and guarantees the assignment of optimal paths.

4.2.2. *Distributed Deployment*

Compared to the centralized deployment method, the distributed deployment approach offers superior parallelism, reliability, and scalability. As shown in Fig. 6(b), in a centralized deployment, all routing decisions and management tasks are handled by a central controller. This centralized design poses risks such as a single point of failure and makes it challenging to support large-scale networks. In contrast, distributed routing protocols distribute the decision-making tasks across multiple nodes within the network. Each node can independently execute routing algorithms, ensuring that the failure of a single node does not compromise the overall network operation, thus enhancing system reliability and robustness. The distributive nature of these routing protocols allows nodes to communicate and collaborate effectively, enabling faster responses to changes within the network. By executing routing decision tasks in parallel, distributed protocols can swiftly adjust routing paths, thereby improving both the adaptability and performance of the network. Additionally, the distributed deployment method provides excellent scalability. As the network expands, new routing nodes can be flexibly incorporated to accommodate the growing network size and adapt to changing environmental conditions.

In time-varying dynamic satellite networks, global centralized computation is nearly impossible to implement. As a result, an increasing number of studies use distributed operations when designing satellite routing algorithms. Simultaneously, angle computation has become an essential factor in the design process. Park et al. [75] proposed a routing algorithm based on distributed angular computation, employing a Markov Decision Process (MDP) for discrete-time sequential decision-making in time-varying LEO satellite networks. Through this approach, the authors introduced the MDP-based Distributed Angular Routing algorithm to achieve seamless routing in LEO satellite networks. This algorithm calculates the angular differences between the source and destination and pursues geometrically straight-line routing in orbital coordinates, ultimately improving data transmission efficiency. The distributed routing algorithm enables efficient data transfer through distributed decision-making while avoiding the challenges posed by centralized computation. By utilizing real satellite data, such as Two-Line Elements (TLEs), in realistic environments, experimental results show that the proposed algorithm outperforms others in routing success rate, reward convergence, and throughput, further demonstrating the effectiveness of the distributed angular routing algorithm in LEO satellite networks.

Many researchers in ML-based LEO intelligent routing algorithms have opted for MARL to achieve a distributed deployment approach. This approach allows each agent to gather local observations and make hop-by-hop routing decisions based on those observations. The complete routing process is accomplished through the collective effort of all agents, enabling the distributed deployment of routing algorithms. Gao et al. [76] proposed a distributed routing algorithm based on MARL. In this algorithm, mobile and node agents work together to realize a distributed routing decision process. First, mobile agents are designed to traverse the entire satellite network, where forward agents are responsible for path exploration and data transmission from the source node to the target node, simulating the process of packet hop-by-hop movement, collecting network path and traffic information, as well as selecting the next hop node. The forward agent generates the reverse agent after reaching the target node and moves along the path opposite to that of the forward agent; the reverse agent, on the other hand, updates the routing information of the intermediate nodes, interacts with the cache, compares the information and updates the routing information in the process of returning from the target node to the source node, in order to realize the dynamic updating of the paths and the maintenance of the network state. Secondly, node agents are generated by each satellite node and are responsible for managing the local routing information and the information in the cache. When the reverse agents return to the intermediate nodes, the node agents interact with them. The node agents compare the information in the cache with the information carried by the reverse agents, select the more efficient of them, and update the local routing information. This process is carried out in a distributed manner, where each node agent handles its own task independently, with no global scheduling or central node. Through the distributed collaboration of mobile and node agents, the algorithm realizes the routing management of satellite networks, which can effectively adapt to changes of network topology and fluctuations of traffic.

However, traditional distributed deployment methods often heavily depend on local information and communication between neighboring nodes, leading to limited routing decisions. To overcome this limitation, Xu et al. [77] introduced spatial location information as a foundation for routing decisions, enhancing traditional distributed deployment methods. Spatial location information enables intelligence to have a more precise understanding of the relative positions of satellites, thereby improving the effectiveness of routing path selection. By incorporating the spatial arrangement of satellites, unnecessary long routing paths are avoided, resulting in enhanced data transmission efficiency. Implemented through MARL, each satellite serves as an agent that only needs to perceive the spatial location and queue states of its neighboring nodes within one hop, effectively reducing the high communication overhead and information collection delays associated with global information collection.

Furthermore, Liu et al. [78] proposed a decentralized flow-centric DRL approach, shifting the focus from traditional packet-level routing to flow-level routing. Similar to other distributed approaches, this method enables decision-making by treating each satellite as an independent agent. However, it differs by locally defining traffic flows, which reduces computational and communication overhead, allowing each satellite to perform routing decisions solely for its local traffic without relying on global information. Through this distributed architecture, the approach effectively avoids the high communication costs associated with centralized methods, enabling each satellite to independently route traffic based on locally defined flow information. This not only enhances the flexibility and adaptability of routing decisions but also ensures high throughput and low latency in dynamic network environments, thereby achieving efficient flow management in low-Earth orbit satellite networks.

4.2.3. *Comparison and Analysis*

There are significant differences between distributed and centralized AI routing algorithms in terms of architecture design, performance, and applicable scenarios. This section compares the two from various dimensions, such as decision-making criteria, scalability, robustness, and communication overhead, regarding to the high dynamics and resource constraints of LEO satellite networks (as shown in Table 9), and analyzes their adaptability in practical applications.

In LEO satellite networks, the design differences between distributed and centralized AI routing algorithms directly impact their performance and suitability for various scenarios. Centralized algorithms rely on global network state information for routing decisions, typically managed by a ground control center or high-altitude satellites (such as GEO or MEO). This architecture can generate globally optimal paths based on complete topology and traffic information, avoiding local suboptimal issues. Additionally, it is efficient in model training and parameter updating, making it particularly suitable for offline training of large-scale DRL models. However, its core drawback is the risk of a single point of failure. If the central node fails, it may lead to the collapse of routing across the entire network, which poses a significant threat to the high-reliability requirements of LEO satellite networks. Furthermore, frequent synchronization of global state information must be completed via the satellite-ground link, which may cause decision delays in bandwidth-limited and high-latency environments. Additionally, the limited computational resources of satellite platforms make it difficult for centralized architectures to support the scalability requirements of large-scale constellations (such as Starlink). As a result, it is better suited for relatively static topologies in short-term scenarios (such as equatorial region coverage windows) or lightweight models for online inference tasks (e.g., traffic prediction).

In contrast, distributed algorithms enable autonomous decision-making by satellites through multi-agent collaboration, where each satellite relies on local information (e.g., neighboring node states and link quality) to generate routing strategies. This architecture inherently provides high robustness. Even if a local link is interrupted or a satellite fails, the network can maintain connectivity through dynamic path adjustments, making it especially suited for high-dynamic topologies (such as frequent link switches in polar regions). At the same time, the distributed architecture reduces the need for global data exchanges between satellite-ground or inter-satellite communications, significantly lowering communication overhead, and it supports the scalability requirements of constellations with thousands of satellites. However, its limitations lie in the incompleteness of local information, which may lead to path redundancy or load imbalance. Additionally, training multiple agents requires addressing complex issues such as policy consistency and credit allocation, increasing the difficulty of algorithm design and resource distribution. For instance, under resource-constrained satellite platforms, distributed models need to store and update parameters at each node, which imposes higher requirements on storage and computing capabilities, potentially limiting the real-time deployment of complex models such as GNNs.

To balance the advantages of both architectures, some studies have proposed hierarchical hybrid solutions. As an example, Li et al. [79] presented a hierarchical DRL routing method that incorporates a reward mechanism based on pheromones. This mechanism aims to improve collaboration during the training phase of distributed intelligence. The proposed method utilizes a hybrid approach that combines centralized and distributed strategies. It accomplishes this by dividing the network into subdomains, with each subdomain having both upper and lower-level agents. The upper-layer agents are responsible for making cross-domain routing decisions. They utilize pheromones at connection points to facilitate collaborative routing among neighboring upper-layer agents. In contrast, the lower layer agents concentrate on optimizing intra-domain routing. They achieve this by employing GNNs to exchange messages and distribute network state information, enabling appropriate intra-domain routing decisions. This hybrid approach, which combines centralized and distributed methods, facilitates the acquisition of global network information, leading to more comprehensive and accurate routing decisions. Additionally, it significantly reduces network load and management burden, while also providing scalability. Overall, the choice of algorithm should be based on the specific requirements of the scenario: centralized algorithms are suitable for short-term static topologies and lightweight models, while distributed algorithms are better suited for high-dynamic environments and large-scale networks. Hybrid architectures offer a compromise for complex scenarios, but their engineering implementation still requires further exploration of cross-layer coordination mechanisms and standardized interface designs.

5. Future LEO routing Research Directions

5.1. Network Feature Extraction

With the continuous exploration of the potential of GNNs in network feature extraction, a series of intelligent routing algorithms based on these networks have emerged. Despite the advancements achieved through these studies, GNNs continue to confront challenges such as over-smoothing and over-compression, particularly in the context of large-scale LEO satellite networks. These issues hinder the routing model's capacity to accurately capture and model the satellite topology, thereby diminishing the decision-making performance of the model. To address

Table 9
Comparison of the two deployment models

Comparison Dimension	Centralized Routing Algorithm	Distributed AI Routing Algorithm
Decision Dependence	Global network state	Local node state
Robustness	Low (depends on central node)	High (node self-tolerance)
Communication Range	High (requires full network communication)	Low (depends on local communication)
Computational Load	Focused on the central node	Distributed to each satellite node
Scalability	Limited (suitable for small-scale networks)	Strong (supports ultra-large model constellation)
Typical Technologies	DQN, DDPG	MARL, distributed GNN
Application Scenarios	Quiet or short-term stability, lightweight models	High dynamic extension, large-scale networks

these challenges, methods such as multi-scale GCN, self-attention mechanisms, and sparse graph techniques can be employed to enhance feature extraction efficiency and model performance. Therefore, enhancing the structure of GNNs with these methods to better accommodate satellite routing tasks may represent a pivotal direction for future research.

5.2. Network Scalability

The long-term expansion of the LEO satellite network requires careful consideration of scalability to accommodate future growth. This involves addressing challenges such as large-scale satellite deployment, network congestion, and dynamic changes in network topology. Distributed routing control strategies have emerged as a promising solution due to their flexibility and fault tolerance. These architectures can better manage scaling and frequent topology changes, ensuring network stability and reliability.

To tackle scalability challenges, emerging technologies like network slicing, SDN, and ML should be integrated into routing strategies. These technologies can optimize resource allocation and enhance routing decisions. Additionally, distributed routing should be designed to quickly adapt to satellite failures and topology changes, ensuring continuous network operation even in the event of node failure or network disruption.

Therefore, developing flexible, fault-tolerant distributed routing systems that can incorporate these technologies is crucial for the sustainable growth of LEO satellite networks and remains a key area for future research.

5.3. Model Decision Robustness

The topology of the satellite network undergoes periodic dynamic changes due to factors such as atmospheric disturbances and satellite orbital adjustments. These shifts, combined with the intrinsic properties of inter-satellite links—such as short link establishment times and dynamic distance variations—complicate network stability. Additionally, future satellite networks will face increasingly complex challenges, including natural factors like optoelectronic interference and energy constraints, as well as human-induced threats such as military strikes. High maintenance costs and long repair cycles further exacerbate the persistence of satellite failures, making network resilience a critical concern.

A key aspect of robustness lies in topology adaptation speed, which measures the time required for a routing algorithm to regain stable performance after a topology change. Faster adaptation ensures timely decision-making and minimizes disruptions. Another crucial metric is malicious node tolerance, which quantifies the percentage of adversarial nodes the network can withstand while maintaining acceptable communication performance. Higher tolerance enhances network security and reliability, ensuring stable operations even under adversarial conditions.

To improve these robustness aspects, advanced strategies such as adversarial training and attention-based models can enhance the resilience of routing decisions against deceptive attacks. Additionally, self-healing routing protocols, inspired by biological immune systems, enable networks to recover from node failures through dynamic topology reconstruction and multipath redundancy. By integrating these techniques, future LEO satellite networks can achieve greater robustness, ensuring efficient, adaptive, and secure routing in the face of dynamic changes and complex threats.

5.4. Computing Network Convergence

Computing network convergence refers to the integration of communication networks and computing networks to enable unified management and collaborative operation of communication and computing resources. This advancement allows communication devices and computing devices to function synergistically, optimizing the integration of computing resources and communication channels for more efficient data processing and transmission. The development of LEO satellite technology has significantly enhanced computing capabilities, offering increased resources and broader application scenarios for computing network convergence. LEO satellites can serve as communication relay stations while also undertaking portions of computing tasks, thus supporting both communication and computation in a unified manner.

To address the challenges posed by limited on-board computational resources, future research should focus on lightweight models, such as micro-GNNs generated through knowledge distillation, combined with hardware accelerators like FPGA deployments, to enable millisecond-level response times for routing decisions. Moreover, cross-layer optimization and task offloading, such as offloading remote sensing data processing tasks from satellites, should be considered. Designing multi-objective optimization models to balance communication latency and computational energy consumption will be critical for enhancing the efficiency of satellite networks. Therefore, incorporating on-planet computing power into LEO satellite routing algorithms and optimizing the synergy between communication and computation are key areas for future research.

5.5. Generative Artificial Intelligence

Generative Artificial Intelligence (GenAI) is a type of AI technology that can generate new data based on existing data or models. Unlike traditional AI systems, GenAI learns the distribution and structure of data to create entirely new content, including text, images, audio, synthetic time-series data, and more. Currently, an increasing number of studies are applying GenAI to the field of network optimization, but research related to LEO intelligent routing algorithms remains limited. Through self-learning and reasoning, GenAI can generate more flexible, intelligent, and high-quality routing strategies in the dynamically changing LEO network environment. As the number of satellites increases and technologies mature, GenAI is expected to become one of the key technologies in future LEO satellite intelligent routing algorithms, providing more efficient and reliable solutions for global satellite communications and internet connectivity.

Moreover, the full potential of GenAI has yet to be fully explored. One promising direction is the dynamic generation of routing protocols. Leveraging GenAI's sequence generation capabilities, such as those of Diffusion models, adaptive routing rules can be generated in real-time for handling scenarios like sudden traffic surges or topology changes, replacing traditional predefined protocols. This ability to dynamically create routing strategies in response to rapidly changing network conditions could significantly enhance the flexibility and responsiveness of LEO satellite networks.

5.6. Suitability for Practical Application Scenarios

While ML-based intelligent routing algorithms have demonstrated significant theoretical advantages, their practical deployment must align closely with the diverse requirements of real-world LEO satellite network applications. Future research should focus on evaluating the adaptability of these algorithms in scenarios such as satellite internet services, disaster emergency communications, and global IoT coverage. For instance, in satellite internet services, urban areas often demand high throughput, whereas remote regions prioritize link stability. Dynamic routing strategies should be designed to balance load distribution and enhance fault tolerance. In disaster emergency communication scenarios, damage to ground infrastructure may disrupt satellite-ground communication links, leading to network topology instability. To address this, RL techniques can be integrated to enable rapid rerouting and efficient disaster recovery. Moreover, practical deployment faces constraints such as limited onboard computing resources and the latency sensitivity of satellite-ground coordination. To mitigate these challenges, engineering techniques—including model compression, edge computing, and inter-satellite cooperative computing—should be leveraged to reduce algorithmic complexity. Additionally, establishing high-fidelity simulation platforms and onboard prototype testing environments will be essential for validating algorithm robustness in complex conditions, such as Doppler shift and channel fading. Furthermore, optimizing the compatibility of cross-domain heterogeneous networks (e.g., interconnecting with terrestrial 5G/6G networks and high-altitude platforms) and developing standardized evaluation metrics will be critical for bridging the gap between theoretical validation and real-world engineering deployment. Through scenario-driven technological iterations and rigorous validation, ML-based intelligent routing algorithms are expected to provide efficient and reliable foundational support for global communication, emergency response, and IoT applications.

6. Conclusion

Developing routing methods tailored for LEO satellite application scenarios is of crucial significance for further enhancing network transmission performance and constitutes one of the key technologies in future 6G. Compared with traditional algorithms, routing algorithms based on ML are more intelligent and start to exhibit obvious performance advantages, making them more suitable for 6G networks. However, in existing research works, there is a dearth of comprehensive analysis content on integrating ML into LEO satellite network routing tasks. We comprehensively summarize the latest progress of intelligent routing algorithms based on ML in LEO satellite networks from four aspects: routing models, design challenges, training and deployment, and future research directions. The objective is to provide theoretical support for the design of artificial intelligence satellite communication systems and further promote the innovative development of satellite network optimization technologies.

Declaration of Competing Interest

The authors declared that they have no conflicts of interest to this work.

CRediT authorship contribution statement

Zhenyu Zhu: Methodology, Conceptualization, Investigation, Validation, Writing - original draft, Writing - review & editing. **Zheheng Rao**: Methodology, Conceptualization, Validation, Writing - review & editing, Project administration, Funding acquisition, Supervision. **Shitong Xiao**: Conceptualization, Writing - review & editing. **Ye Yao**: Conceptualization, Writing - review & editing, Project administration, Supervision. **Yanyan Xu**: Writing - review & editing, Project administration, Funding acquisition. **Weizhi Meng**:

Conceptualization, Writing - review & editing, Project administration, Supervision.

Acknowledgments

This work is supported by National Key Research and Development Program of China (No.2022YFB3902804, 2022YFB3903404); Funded By Open Research Fund Program of LIESMARS (Grant No.24I05); Hubei Province Major Science and Technology Innovation Program(No.2024BAA011); National Natural Science Foundation of China (No. 42271431, 42271425), and the Zhejiang Provincial Natural Science Foundation of China under Award (No. LY24F020017).

References

[1] H. J. Hadi, Y. Cao, S. Li, L. Xu, Y. Hu, M. Li, Real-time fusion multi-tier DNN-based collaborative idps with complementary features for secure UAV-enabled 6G networks, Expert Systems with Applications 252 (2024) 124215.

[2] S. Chen, S. Sun, S. Kang, System integration of terrestrial mobile communication and satellite communication—the trends, challenges and key technologies in B5G and 6G, China communications 17 (12) (2020) 156–171.

[3] T. Tetik, G. S. Daş, B. Birgoren, A multi-objective perspective to satellite design and reliability optimization, Expert Systems with Applications 246 (2024) 123178.

[4] J. K. Pedersen, M. Bøchman, W. Meng, Security analysis in satellite communication based on geostationary orbit, in: 2022 19th Annual International Conference on Privacy, Security & Trust (PST), IEEE, 2022, pp. 1–10.

[5] X. Yang, J. Yan, D. Wang, Y. Xu, G. Hua, Woad3qn-rp: An intelligent routing protocol in wireless sensor networks—a swarm intelligence and deep reinforcement learning based approach, Expert Systems with Applications 246 (2024) 123089.

[6] A. A. Amin, M. S. Iqbal, M. H. Shahbaz, Development of intelligent fault-tolerant control systems with machine leaprning, deep learning, and transfer learning algorithms: A review, Expert Systems with Applications (2023) 121956.

[7] Y. Shi, L. Lian, Y. Shi, Z. Wang, Y. Zhou, L. Fu, L. Bai, J. Zhang, W. Zhang, Machine learning for large-scale optimization in 6G wireless networks, IEEE Communications Surveys & Tutorials (2023).

[8] S. Mahboob, L. Liu, Revolutionizing future connectivity: A contemporary survey on ai-empowered satellite-based non-terrestrial networks in 6G, IEEE Communications Surveys & Tutorials (2024).

[9] X. Cao, Y. Li, X. Xiong, J. Wang, Dynamic routings in satellite networks: An overview, Sensors 22 (12) (2022) 4552.

[10] M. Barabas, G. Boanea, V. Dobrota, Multipath routing management using neural networks-based traffic prediction, in: The Third International Conference on Emerging Network Intelligence, 2011, pp. 118–124.

[11] C. Hardegen, B. Pfülb, S. Rieger, A. Gepperth, Predicting network flow characteristics using deep learning and real-world network traffic, IEEE Transactions on Network and Service Management 17 (4) (2020) 2662–2676.

[12] B. Mao, Z. M. Fadlullah, F. Tang, N. Kato, O. Akashi, T. Inoue, K. Mizutani, Routing or computing? the paradigm shift towards intelligent computer network packet transmission based on deep learning, IEEE Transactions on Computers 66 (11) (2017) 1946–1960.

[13] T. M. Modi, P. Swain, Intelligent routing using convolutional neural network in software-defined data center network, The Journal of Supercomputing 78 (11) (2022) 13373–13392.

[14] A. Sumathi, A. Javadpour, P. Pinto, A. K. Sangaiah, W. Zhang, S. Mahmoodi Khaniabadi, Newtr: a multipath routing for next hop destination in internet of things with artificial recurrent neural network (RNN), International Journal of Machine Learning and Cybernetics 13 (10) (2022) 2869–2889.

[15] K. Rusek, J. Suárez-Varela, A. Mestres, P. Barlet-Ros, A. Cabellos-Aparicio, Unveiling the potential of graph neural networks for network modeling and optimization in SDN, in: Proceedings of the 2019 ACM Symposium on SDN Research, 2019, pp. 140–151.

[16] F. Geyer, G. Carle, 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, 2018, pp. 40–45.

[17] J. Boyan, M. Littman, Packet routing in dynamically changing networks: A reinforcement learning approach, Advances in neural information processing systems 6 (1993).

[18] P. Wang, T. Wang, Adaptive routing for sensor networks using reinforcement learning, in: The Sixth IEEE International Conference on Computer and Information Technology (CIT'06), IEEE, 2006, pp. 219–219.

[19] D. M. Casas-Velasco, O. M. C. Rendon, N. L. da Fonseca, Intelligent routing based on reinforcement learning for software-defined networking, IEEE Transactions on Network and Service Management 18 (1) (2020) 870–881.

[20] Z. Yao, Y. Wang, X. Qiu, DQN-based energy-efficient routing algorithm in software-defined data centers, International Journal of Distributed Sensor Networks 16 (6) (2020) 1550147720935775.

[21] Y. Su, R. Fan, X. Fu, Z. Jin, Dqelr: An adaptive deep q-network-based energy-and latency-aware routing protocol design for underwater acoustic sensor networks, IEEE Access 7 (2019) 9091–9104.

[22] Z. Xu, J. Tang, J. Meng, W. Zhang, Y. Wang, C. H. Liu, D. Yang, Experience-driven networking: A deep reinforcement learning based approach, in: IEEE INFOCOM 2018-IEEE conference on computer communications, IEEE, 2018, pp. 1871–1879.

[23] P. Almasan, J. Suárez-Varela, K. Rusek, P. Barlet-Ros, A. Cabellos-Aparicio, Deep reinforcement learning meets graph neural networks: Exploring a routing optimization use case, Computer Communications 196 (2022) 184–194.

[24] X. Mai, Q. Fu, Y. Chen, Packet routing with graph attention multi-agent reinforcement learning, in: 2021 IEEE Global Communications Conference (GLOBECOM), IEEE, 2021, pp. 1–6.

[25] S. J. Russell, P. Norvig, Artificial intelligence: a modern approach, Pearson, 2016.

[26] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, nature 521 (7553) (2015) 436–444.

[27] G. E. Hinton, S. Osindero, Y.-W. Teh, A fast learning algorithm for deep belief nets, Neural computation 18 (7) (2006) 1527–1554.

[28] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, G. Wang, J. Cai, et al., Recent advances in convolutional neural networks, Pattern recognition 77 (2018) 354–377.

[29] M. Schuster, K. K. Paliwal, Bidirectional recurrent neural networks, IEEE transactions on Signal Processing 45 (11) (1997) 2673–2681.

[30] A. Graves, A. Graves, Long short-term memory, Supervised sequence labelling with recurrent neural networks (2012) 37–45.

[31] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, S. Y. Philip, A comprehensive survey on graph neural networks, IEEE transactions on neural networks and learning systems 32 (1) (2020) 4–24.

[32] J. Chung, C. Gulcehre, K. Cho, Y. Bengio, Empirical evaluation of gated recurrent neural networks on sequence modeling, arXiv preprint arXiv:1412.3555 (2014).

[33] C. J. Watkins, P. Dayan, Q-learning, Machine learning 8 (1992) 279–292.

[34] L. Chen, B. Hu, Z.-H. Guan, L. Zhao, X. Shen, Multiagent meta-reinforcement learning for adaptive multipath routing optimization, IEEE Transactions on Neural Networks and Learning Systems 33 (10) (2021) 5374–5386.

[35] V. Mnih, Playing atari with deep reinforcement learning, arXiv preprint arXiv:1312.5602 (2013).

[36] T. Lillicrap, Continuous control with deep reinforcement learning, arXiv preprint arXiv:1509.02971 (2015).

[37] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, Y. Bengio, Graph attention networks, arXiv preprint arXiv:1710.10903 (2017).
[38] B. Gutteridge, X. Dong, M. M. Bronstein, F. Di Giovanni, Drew: Dynamically rewired message passing with delay, in: International Conference on Machine Learning, PMLR, 2023, pp. 12252–12267.
[39] Y. LIU, L.-n. WANG, LEO satellite intelligent-sensing routing algorithm based on a dendrite network, Chinese Journal of Engineering 45 (3) (2023) 465–474.
[40] Y. Huang, W. Shufan, K. Zeyu, M. Zhongcheng, H. Huang, W. Xiaofeng, A. J. Tang, X. Cheng, Reinforcement learning based dynamic distributed routing scheme for mega LEO satellite networks, Chinese Journal of Aeronautics 36 (2) (2023) 284–291.
[41] Z. Wang, H. Yao, T. Mai, Z. Li, C. L. P. Chen, Learning-driven swarm intelligence: Enabling deterministic flows scheduling in LEO satellite networks, IEEE Transactions on Mobile Computing (2024) 1–17.
[42] F. Zheng, C. Wang, Z. Zhou, Z. Pi, D. Huang, LEO laser microwave hybrid inter-satellite routing strategy based on modified Q-routing algorithm, EURASIP Journal on Wireless Communications and Networking 2022 (1) (2022) 36.
[43] X. Wang, Z. Dai, Z. Xu, LEO satellite network routing algorithm based on reinforcement learning, in: 2021 IEEE 4th International Conference on Electronics Technology (ICET), IEEE, 2021, pp. 1105–1109.
[44] P.-D. Matasaru, L. Scripcariu, F. Diaconu, An up-to-date overview on qos for satellite IP networks, in: 2017 International Symposium on Signals, Circuits and Systems (ISSCS), IEEE, 2017, pp. 1–4.
[45] S. Wu, L. Yang, J. Guo, Q. Chen, X. Liu, C. Fan, Intelligent quality of service routing in software-defined satellite networking, IEEE Access 7 (2019) 155281–155298.
[46] C. Wang, H. Wang, W. Wang, A two-hops state-aware routing strategy based on deep reinforcement learning for LEO satellite networks, Electronics 8 (9) (2019) 920.
[47] H. Wang, Y. Ran, L. Zhao, J. Wang, J. Luo, T. Zhang, Grouting: dynamic routing for LEO satellite networks with graph-based deep reinforcement learning, in: 2021 4th International Conference on Hot Information-Centric Networking (HotICN), IEEE, 2021, pp. 123–128.
[48] A. A. Magadum, A. Ranjan, D. Narayan, Deepqosr: A deep reinforcement learning based qos-aware routing for software defined data center networks, in: 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT), IEEE, 2021, pp. 1–7.
[49] X. Zhou, Y. Weng, B. Mao, J. Liu, N. Kato, Intelligent multi-objective routing for future ultra-dense LEO satellite networks, IEEE Wireless Communications (2024).
[50] B. Mao, X. Zhou, J. Liu, N. Kato, On an intelligent hierarchical routing strategy for ultra-dense free space optical low earth orbit satellite networks, IEEE Journal on Selected Areas in Communications (2024).
[51] H. He, W. Yuan, Y. Hou, S. Chen, X. Jiang, R. Zhu, J. Yang, Onboard processing-aided transmission delay minimization for LEO satellite networks, IEEE Transactions on Communications 72 (11) (2024) 7047–7062.
[52] J. Zhou, X. Gong, L. Sun, Y. Xie, X. Yan, Adaptive routing strategy based on improved double q-learning for satellite internet of things, Security and Communication Networks 2021 (1) (2021) 5530023.
[53] J. Song, Y. Ju, L. Liu, Q. Pei, C. Wu, M. A. Jan, S. Mumtaz, Trustworthy and load-balancing routing scheme for satellite services with multi-agent DRL, in: IEEE INFOCOM 2023-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), IEEE, 2023, pp. 1–6.
[54] N. Kato, Z. M. Fadlullah, F. Tang, B. Mao, S. Tani, A. Okamura, J. Liu, Optimizing space-air-ground integrated networks by artificial intelligence, IEEE Wireless Communications 26 (4) (2019) 140–147.
[55] Q. He, Y. Wang, X. Wang, W. Xu, F. Li, K. Yang, L. Ma, Routing optimization with deep reinforcement learning in knowledge defined networking, IEEE Transactions on Mobile Computing 23 (2) (2023) 1444–1455.
[56] F. Dong, J. Song, Y. Zhang, Y. Wang, T. Huang, DRL-based load-balancing routing scheme for 6G space–air–ground integrated networks, Remote Sensing 15 (11) (2023) 2801.
[57] Y. Huang, X. Jiang, S. Chen, F. Yang, J. Yang, Pheromone incentivized intelligent multipath traffic scheduling approach for LEO satellite networks, IEEE Transactions on Wireless Communications 21 (8) (2022) 5889–5902.
[58] Z. Wang, J. Zhang, X. Zhang, W. Wang, Reinforcement learning based congestion control in satellite internet of things, in: 2019 11th International Conference on Wireless Communications and Signal Processing (WCSP), IEEE, 2019, pp. 1–6.
[59] Z. Ding, H. Liu, F. Tian, Z. Yang, N. Wang, Fast-convergence reinforcement learning for routing in LEO satellite networks, Sensors 23 (11) (2023) 5180.
[60] J. He, N. Cheng, Z. Yin, H. Zhou, C. Zhou, K. Aldubaikhy, A. Alqasir, X. Shen, Load-aware network resource orchestration in LEO satellite network: A GAT-based approach, IEEE Internet of Things Journal 11 (9) (2024) 15969–15984.
[61] X. Shi, P. Ren, Q. Du, Reinforcement learning routing in space-air-ground integrated networks, in: 2021 13th International Conference on Wireless Communications and Signal Processing (WCSP), IEEE, 2021, pp. 1–6.
[62] Y. Lyu, H. Hu, R. Fan, Z. Liu, J. An, S. Mao, Dynamic routing for integrated satellite-terrestrial networks: A constrained multi-agent reinforcement learning approach, IEEE Journal on Selected Areas in Communications (2024).
[63] Y.-H. Hsu, J.-I. Lee, F.-M. Xu, A deep reinforcement learning based routing scheme for LEO satellite networks in 6G, in: 2023 IEEE Wireless Communications and Networking Conference (WCNC), IEEE, 2023, pp. 1–6.
[64] C. Qiu, H. Yao, F. R. Yu, F. Xu, C. Zhao, Deep q-learning aided networking, caching, and computing resources allocation in software-defined satellite-terrestrial networks, IEEE Transactions on Vehicular Technology 68 (6) (2019) 5871–5883.
[65] H. Wang, Y. Ran, L. Zhao, J. Wang, J. Luo, T. Zhang, Grouting: dynamic routing for LEO satellite networks with graph-based deep reinforcement learning, in: 2021 4th International Conference on Hot Information-Centric Networking (HotICN), IEEE, 2021, pp. 123–128.
[66] J. Liu, B. Zhao, Q. Xin, J. Su, W. Ou, DRL-ER: An intelligent energy-aware routing protocol with guaranteed delay bounds in satellite mega-constellations, IEEE Transactions on Network Science and Engineering 8 (4) (2020) 2872–2884.
[67] G. Wang, F. Yang, J. Song, Z. Han, Optimization for dynamic laser inter-satellite link scheduling with routing: A multi-agent deep reinforcement learning approach, IEEE Transactions on Communications (2023).
[68] J. Dean, G. Corrado, R. Monga, K. Chen, M. Devin, M. Mao, M. Ranzato, A. Senior, P. Tucker, K. Yang, et al., Large scale distributed deep networks, Advances in neural information processing systems 25 (2012).
[69] A. Krizhevsky, One weird trick for parallelizing convolutional neural networks, arXiv preprint arXiv:1404.5997 (2014).
[70] G. Huang, Z. Liu, L. Van Der Maaten, K. Q. Weinberger, Densely connected convolutional networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 4700–4708.
[71] Z. Tu, H. Zhou, K. Li, G. Li, Q. Shen, A routing optimization method for software-defined SGIN based on deep reinforcement learning, in: 2019 IEEE Globecom Workshops (GC Wkshps), IEEE, 2019, pp. 1–6.
[72] W. Wei, L. Fu, H. Gu, X. Lu, L. Liu, S. Mumtaz, M. Guizani, Iris: Toward intelligent reliable routing for software-defined satellite networks, IEEE Transactions on Communications 73 (1) (2025) 454–468.
[73] J. Liang, W. Sun, N. Xiao, W. Chen, Z. Guo, Intelligent routing strategy for software-defined satellite network based on Chebyshev neural

network, Journal of National University of Defense Technology 42 (5) (2020) 23–30.

[74] F. Wang, D. Jiang, Z. Wang, Z. Lv, S. Mumtaz, Fuzzy-CNN based multi-task routing for integrated satellite-terrestrial networks, IEEE Transactions on Vehicular Technology 71 (2) (2021) 1913–1926.

[75] S. Park, G. Seon Kim, S. Jung, J. Kim, Markov decision policies for distributed angular routing in LEO mobile satellite constellation networks, IEEE Internet of Things Journal 11 (23) (2024) 38744–38754.

[76] Z. Gao, Q. Guo, Z. Na, Novel optimized routing algorithm for LEO satellite ip networks, Journal of Systems Engineering and Electronics 22 (6) (2011) 917–925.

[77] G. Xu, Y. Zhao, Y. Ran, R. Zhao, J. Luo, Spatial location aided fully-distributed dynamic routing for large-scale LEO satellite networks, IEEE Communications Letters 26 (12) (2022) 3034–3038.

[78] H. Liu, J. Lai, J. Zhu, L. Gan, Z. Chang, Enabling high-throughput routing for LEO satellite broadband networks: A flow-centric deep reinforcement learning approach, IEEE Internet of Things Journal 11 (17) (2024) 28705–28720.

[79] Y. Li, Q. Zhang, H. Yao, R. Gao, X. Xin, F. R. Yu, Stigmergy and hierarchical learning for routing optimization in multi-domain collaborative satellite networks, IEEE Journal on Selected Areas in Communications (2024).

[80] C. Li, Y. Zhang, Z. Cui, Y. Zhang, J. Liu, Z. Yu, P. Zhang, An overview of low earth orbit satellite routing algorithms, 2023 International Wireless Communications and Mobile Computing (IWCMC) (2023) 866–870.

[81] J. Bao, B. Zhao, W. Yu, Z. Feng, C. Wu, Z. Gong, Opensan: A software-defined satellite network architecture, ACM SIGCOMM Computer Communication Review 44 (4) (2014) 347–348.