An Intelligent Q-Learning-Based Routing Model

[1]Ghazwan F. Yassin, [2]Ayad H. Abdulqader

[1,2]Computer Science Department, College of Computer Science and Mathematics, University of Mosul, Iraq
Authors E-mail: [1]ghazwan.21csp77@student.uomosul.edu.iq, [2]ayad_alezzi@uomosul.edu.iq

Abstract - **The rising speed of data transmission requires modern technology to meet its essential requirement of network communication efficiency through effective routing techniques. Routers act as the central elements of this functionality which demonstrates why their optimal performance needs attention. This work introduces an intelligent routing design that applies machine learning approaches for network performance optimization under dynamic network environments. The fluctuating environments decrease the effectiveness of traditional routing protocols including RIP, BGP and OSPF while causing their routing performance to deteriorate. The proposed model implements machine learning-based decision adjustment methods that apply current network information to dynamically reroute data. The routing system uses supervised and unsupervised learning approaches to predict network traffic congestions and choose the most suitable routes. Network performance optimization relies on the incorporation of latency, bandwidth, packet loss, congestion, jitter, reliability, energy efficiency in addition to cost parameters before training occurs using historical network information. Python-based development achieved enhanced network throughput in addition to faster operation with better adaptability and resource-efficient management of changing network conditions. Machine learning arrives as a transformative force for network routing through adaptive intelligent communication systems which surpass traditional protocols for modern networking requirements.**

Keywords: Intelligent Routing, Machine Learning, Network Performance, Dynamic Routing, OSPF, BGP, Latency Optimization.

I. INTRODUCTION

Functional efficiency and reliable routing for scalable communication in modern computer networks depends on dependable routing protocols. Among these protocols, two pivotal ones form the backbone of contemporary routing infrastructure: Open Shortest Path First (OSPF) and Border Gateway Protocol (BGP) [1]. Within intra-domain boundaries OSPF functions as a link-state routing protocol by automatically calculating routes that depend on link conditions within autonomous system (AS) boundaries. BGP operates as a path-vector protocol which executes inter-domain routing by route advertisement and path selection between multiple autonomous systems for establishing worldwide internet connections [2].

Despite their widespread adoption, both OSPF and BGP encounter inherent limitations. These routing protocols select subpar paths and show difficulty in adjusting to changing network situations. OSPF makes poor routing choices when empirical network conditions deviate from the link-costs assigned during protocol configuration [2]. The administrative policies through which BGP makes its route decisions replace essential real-time performance metrics thus leading to inadequate inter-domain path selection. Network protocols face challenges because the overhead created during the operations extends the convergence time needed for network changes [3].

Artificial intelligence systems at present show successful solutions to complicated networking decision problems. The recent developments in network optimization demonstrate reinforcement learning (RL) techniques as an efficient tool that exploits powerful capabilities [4][5]. The Q-Learning approach serves as a model-free RL algorithm that successfully addresses discrete state and action space problems through easy implementation and strong performance capabilities. Q-Learning's state-based action-value learning capability helps produce routing decisions that enhance both network protocol flexibility and operational velocity [6].

Researches employs Q-Learning models to enhance traditional routing protocols OSPF and BGP by addressing their existing constraints and maximizing routing performance. Q-Learning integration with OSPF enables the protocol to modify link costs dynamically by processing real-time network traffic patterns along with congestion level information [7]. The adaptive nature of OSPF routing improves route selection quality thus creating less overhead and better distribution of network traffic. In Q-Learning integrated with BGP the protocol evaluates multiple path properties together with latency parameters and bandwidth capacities and network reliability factors to determine optimal routing paths. With its altered route selection system BGP

addresses its single reference point static policy limitations that restrict decision capabilities[8].

The proposed system uses Q-Learning because its dynamic network adaptation capabilities lead to a more dependable routing solution. Through network feedback protocols generate timely responses which reduce service outages while optimizing resource distribution and accelerating convergence processes. The autonomous framework built by integrating Q-Learning with OSPF and BGP maintains optimal performance levels that meet modern communication requirements.

The proposed model implements Q-Learning to establish a new approach for autonomous routing optimization. The real-time evaluation of network conditions involving congestion together with latency and bandwidth and reliability allows the Q-Learning-based model to implement a more adaptable routing solution. The suggested system outshines conventional routing protocols because it can modify routes through real-time network feedback without being restricted by fixed parameters. The system provides superior performance with accelerated convergence combined with optimized resource distribution to support modern communication system requirements.

Network routing benefits from the progression achieved by integrating Q-Learning technology into its architecture. The adopted approach brings autonomous routing capabilities which creates opportunities for enhanced efficient global communication systems with better scalability and responsiveness.

II. RELATED WORKS

The research in [9] investigates how an optimized reinforcement learning-based adaptive network routing algorithm solves the problems faced by traditional shortest path routing in wireless networks. The application of shortest path routing in wired networks functions effectively yet produces congestion issues in wireless networks mainly because it selects the shortest path without considering ongoing traffic. Path selection through the proposed method relies on actual network traffic patterns to determine routes dynamically and minimize packet delivery time. The research performs computer simulations to analyze algorithm efficiency for periodic and non-periodic traffic patterns while delivery time functions as the main evaluation metric.

The research in [10] examines how reinforcement learning (RL) techniques solve optimization challenges in distributed systems through network routing implementations. The development of sixty network protocols using reinforcement learning methods has permitted route

optimization in communication networks that fulfill multiple quality of service objectives. A detailed review of these protocols examines their evolution through different network requirements and characteristics at various stages. The analysis develops criteria for evaluating different routing protocols built on reinforcement learning approaches.

Research [11] puts forward a reinforcement learning (RL) framework which targets urban intersection signal control to resolve present issues stemming from subpar signal administration. The method employs neural networks to approximate Q-functions which solve the complex traffic signal control challenge involving extensive state-action spaces of binary values. Real-time traffic data serving as state information includes positional data along with directional data and speed data. Various signal phases comprise the available actions for the system. Through SUMO traffic simulation testing the research demonstrates that RL-based traffic control outperforms traditional approaches by minimizing queue formation and reducing wait times while providing adaptive and generalized solutions.

The research in [12] examines how machine learning systems help operate 5G wireless networks to handle the numerous devices connected to Internet of Things (IoT). The paper shows how machine learning permits wireless networks to self-optimize their configurations while learning from past experiences to deliver diverse service demands without extensive human supervision. The research explores standard machine learning approaches together with their implementation across spectrum sensing and channel estimation and device clustering along with behavior prediction and adaptive routing systems. The research demonstrates deep reinforcement learning's effectiveness for network resource management through simulation results while proposing its use for intelligent resource control.

Researchers in [13] have created a machine learning hybrid system to enhance freeway traffic speed estimation capabilities while resolving existing data quality limitations in monitoring systems. The approach builds a training variable from the second-order traffic flow model which blends PeMS system data alongside iPeMS system information delivered by Utah Department of Transportation. Traffic speed estimation relies on three machine learning models: Random Forest (RF) and Extreme Gradient Boosting (XGBoost) and Artificial Neural Network (ANN). The hybrid approach delivers better time-series traffic modeling capabilities compared to pure machine learning procedures which were validated against PeMS data as the benchmark.

A routing solution called DQN-EC addresses network management in distributed file systems applying erasure

coding techniques according to [14]. The deep reinforcement learning framework enables erasure coding by implementing an adaptive system which can respond to network disruptions including node and switch failures. Through network information collection the SDN controller enhances route optimization by processing essential data about block sizes and node statuses. The experimental evaluation of DQN-EC on a fat-tree network topology showed it provided quick block transmission at 2.5 times the rate and enhanced network throughput by 0.4 times when compared with traditional OSPF routing despite lowering both bottleneck bandwidth usage and link transmission expenses.

The research in [15] present an SDN routing optimization approach which employs deep reinforcement learning (DRL) to optimize network traffic management under quality of service standards. The agent employs deep reinforcement learning to monitor network switch traffic loads and end-to-end network performance metrics during which it determines weight configurations that minimize delay and packet loss for end-to-end paths. The SDN controller performs path definition and switch flow rule installation using link weight values generated by the controller. An M/M/1/K queue-based network model serves for offline learning until convergence to solve long learning times stemming from topology adjustments. The simulation outcomes reveal that the proposed method produces better outcomes than standard hop-count routing and traffic demand-based RL algorithms while remaining effective across multiple network configurations.

A traffic signal control (TSC) system that utilizes off-policy deep reinforcement learning (deep RL) agents with background removal residual networks was developed by researchers in [16] to optimize road intersection traffic flow. Images from traffic intersections enable the system to calculate optimal signal controls that minimize vehicle delays at intersections. A trained deep RL agent generates suboptimal signaling choices for traffic control based on real-time traffic data after completing sufficient training. Deep RL technology performs better than traditional TSC systems in various intersection environments by adapting quickly to changing traffic patterns and yielding major improvements in traffic control together with reduced congestion.

The Critical Flow Rerouting with Weight-Reinforcement Learning (CFRW-RL) algorithm developed in [17] builds upon CFR-RL by enhancing network traffic scheduling operations in large-scale enterprise environments. The Critical Flow Rerouting with Weight-Reinforcement Learning (CFRW-RL) network algorithm redirects lower-weight data flows first through its data classification system which protects high-priority networks from outages. The CFRW-RL protocol reduces business disruptions effectively while maintaining equivalent computational complexity and delivers improved user experience through major packet reordering reduction.

Table (1) summarizes different studies which use machine learning methods to tackle routing and network optimization problems. The table summarizes research investigations through displaying their applied ML methods in combination with their main working areas along with their significant outcomes. The research includes deployments across wireless network routing together with traffic signal control followed by 5G network management and distributed system erasure coding. The evaluation of these studies reveals how reinforcement learning (RL) alongside hybrid ML methods changes routing systems and network optimization systems across various fields.

III. BACKGROUND

3.1 Routing Protocols: OSPF and BGP

The ability for computer networks to function relies on routing protocols which build their underlying communication structures. OSPF serves as an Interior Gateway Protocol (IGP) that dominates intra-domain routing applications [18]. The protocol uses link-state algorithms for its operation that makes routers distribute network-wide connection status reports. Routing devices obtain complete network map details through this procedure to compute shortest paths by running Dijkstra's algorithm. OSPF areas enhance scalability through reduced network traffic however their path convergence speed remains slow and they select suboptimal routes during congestion [19].

Table 1: Summary for previous ML-based routing models

Research Study	ML Technique Used	Focus Area	Key Result
[6] Wireless Network Routing Optimization	Reinforcement Learning (RL)	Optimized Routing in Wireless Networks	Improved routing performance by considering real-time traffic patterns.
[7] RL-based Network Routing Protocols Review	Reinforcement Learning (RL)	Review of RL-based Routing Protocols	Development of 60 RL-based protocols for diverse QoS requirements.
[8] RL Framework for Traffic Signal Control	Reinforcement Learning (RL) with Neural Networks	Traffic Signal Control in Urban Intersections	Better queue management and reduced wait times at intersections.
[9] ML for 5G and IoT Network Management	Reinforcement Learning (RL)	5G IoT Network Management and Resource Allocation	Self-optimizing 5G networks with minimal human intervention.

[10] Hybrid ML for Traffic Speed Estimation	Hybrid ML (RF, XGBoost, ANN)	Traffic Speed Estimation on Freeways	Improved traffic speed modeling over traditional methods.
[11] DQN-EC in Distributed File Systems	Deep Reinforcement Learning (DRL)	Erasure Coding and Network Disruption Management	Faster transmission and better throughput using DRL in network management.
[12] SDN Routing Optimization with DRL	Deep Reinforcement Learning (DRL)	SDN Routing Optimization for QoS	Enhanced path selection and minimized delays and packet loss.
[13] Deep RL for Traffic Signal Control	Off-policy Deep RL	Traffic Signal Control with Deep RL Agents	Better traffic flow control by adapting to changing conditions.
[14] Critical Flow Rerouting with Weight-RL	Reinforcement Learning (RL)	Network Traffic Scheduling and Flow Rerouting	Improved traffic scheduling with reduced packet reordering.

As the primary inter-domain routing protocol BGP enables worldwide network connectivity between autonomous systems (ASes) to operate the global internet. The path-vector routing capability of BGP operates through network route distribution alongside the spread of AS-path attributes combined with next-hop and local preference settings[20]. The path-vector routing system built into BGP creates opportunities for network operators who can use policy-based routing to direct traffic as needed. BGP's operation faces major limitations because it requires manual policy configuration while facing vulnerabilities that affect its routing and requires fixing routing misconfigurations. Internet stability suffers alongside performance degradation during route changes because BGP requires substantial time to converge [21].

3.2 Challenges in Traditional Routing Protocols

Despite their widespread adoption, traditional routing protocols like OSPF and BGP face several limitations [22][2]:

1. Suboptimal Path Selection: Dynamically changing traffic patterns expose OSPF's route selection weakness due to its static link metric calculation mechanism. BGP policy-based routing systems implement administrative preferences instead of using performance indicators such as bandwidth or latency for route selection.
2. Slow Convergence: Network adjustments force both protocols to adapt which creates short-term routing loops that result in packet loss while reducing performance levels.
3. Lack of Adaptability: Traditional protocols struggle to keep up with modern network requirements because they lack adaptive capabilities when networks need to manage changing traffic patterns together with strict QoS quality standards.

3.3 Machine Learning in Networking

Machine learning technology presents potential solutions which overcome the operational restrictions of traditional routing protocols. Using reinforcement learning methods machine learning enables systems to find best actions by enabling them to interact with their environments. The application of reinforcement learning to network routing enables automatic decision adaptations which respond to both network environment transitions and traffic pattern variations [23].

The networking applications benefit from Q-Learning because it operates effectively in situations that have discrete state-action spaces. A Q-Learning agent determines optimal actions across time by maintaining expected utility data within its Q-table to accumulate maximal reward. Q-Learning demonstrates successful implementation in traffic engineering and congestion control together with resource allocation problems thereby proving its capabilities to enhance network protocol efficiency and performance [24].

3.4 Enhancing OSPF and BGP with Q-Learning

The integration of Q-Learning with OSPF and BGP demonstrates promise to solve their core operational restrictions. Q-Learning integrated within OSPF systems enables real-time network status analysis to dynamically adjust link metrics for improved route computations while preventing network congestion [25].

Q-Learning enables BGP to identify the most suitable paths after analyzing metrics that include latency as well as bandwidth alongside reliability and policy limitations. The updated systems use network state observations alongside traffic pattern analysis to enhance performance and create adaptive and fast converging solutions for next-generation network infrastructure [26].

When Q-Learning interacts with standard routing networks such as OSPF it could indicate that reinforcement learning alters the current protocol system. This research introduces a standalone Q-Learning based routing model instead of connecting it together with OSPF and other such protocols.

The real-time analysis of network conditions through Q-Learning allows dynamic routing decision adjustments to match changing conditions thereby improving the adaptability of network routing. The networking system achieves improved performance levels as the model uses intelligent selection of routes which bypasses traditional protocols' static limitations.

IV. METHODOLOGY

The research methodology involves implementing routing protocols for 30-node networks by studying the random Erdős–Rényi model statistically. The researchers chose this random model to simulate internet structure by creating a node connection probability at 0.4 among randomly chosen edges.

The proposed Q-Learning-based routing model operates independently to optimize network performance dynamically. By leveraging reinforcement learning principles, the model continuously refines its path selection based on evolving network conditions

The network tested its ability to connect all node pairs by implementing the is connected function from the network library. The network system executed multiple regeneration cycles to obtain complete network connection. The network established connections that linked nodes with attributes showing latency (1-10ms) and bandwidth (1-10Mbps) and additional attributes including congestion (0-1) and cost (1-5 units), reliability (0.8-1), jitter (1-5ms), packet loss rate (0-1) and energy efficiency (0.7-1). The program used random number generation within range boundaries prior to saving edge data properties to CSV files for future analysis purposes.

The network path determination used Dijkstra's algorithm to establish minimum-paths between pair-wise nodes while employing latency measurements as weight values. The implementation represents the OSPF (Open Shortest Path First) routing protocol through its operation of selecting paths for minimal latency.

The reference model provided by this algorithm served as a basis for assessing multiple alternative algorithms. The Q-Learning algorithm selects the best path based on latency, bandwidth and packet loss whereas OSPF depends exclusively on latency when picking its shortest path.

The Q-Learning algorithm serves as a reinforcement learning method dedicated to packet path selection in network protocols. Each transition between two nodes within the network received a value through the Q-table representation. Training selects paths through the network according to each transition's Q value that represents the "quality" measure for node connections.

A high Q value selects the associated path for transmission. The system calculates new values for each node transition through the received rewards from the network.

The rewards given to the system incorporate measurements of latency alongside bandwidth and packet loss

which generate higher rewards for low-latency and high-bandwidth connections. The system aims to enhance the Q-table's performance by using repeated learning experiences to reach better decision quality with time.

An exploration-exploitation strategy was used to determine the paths: During the first stage agents examined several paths at random to discover system features before utilizing this knowledge in the second stage to determine the best path.

A dataset recording information about all paths between connected nodes was generated to assess the performance of the implemented algorithms. The analysis calculated total latency alongside edge minimum bandwidth and end-to-end packet loss rate by multiplying individual edge packet losses for each path. This data was split into two sets: a training set and a test set.

Q-Learning received training through the training set to boost path selection performance and the test set served to assess path selection abilities of various algorithms. A post-training evaluation of OSPF and Q-Learning path selection examined latency and bandwidth together with packet loss metrics. The application stored resulting data in a CSV format for future analysis purposes.

The Q-table gained updates across various training sessions totaling about 20,000 runs to improve learning and decision-making ability. The algorithms executed performance tests between every active node pair after finishing their training successfully. The evaluation of OSPF and Q-Learning algorithms centered on calculating metrics consisting of path-specified total latency and minimum bandwidth and packet loss rates. The chosen metrics helped identify which algorithm performed better when facing complex network conditions.

The proposed Q-Learning-based routing model learns optimal paths by continuously evaluating network conditions. It updates its Q-table using real-time traffic metrics, ensuring adaptive and intelligent decision-making

The study presented its findings with tabular displays of important metrics between OSPF and Q-Learning and visual diagrams of network infrastructures. Visual representations detail the way nodes select their paths in the network infrastructure as they show the working performance of each algorithm. The research's methodology flow appears in Figure (1).

4.1 Python Thonny Environment

Thonny operates as an integrated development environment (IDE) developed to facilitate Python learning for

beginner programmers. Through its well-designed user interface Thonny provides educational tooling that supports novice programmers learning Python. The native execution model features of Thonny allow users to walk through code execution step-by-step while they track variable changes and watch interpreter operations in real time. Users receive automatic syntax highlighting along with a simple Python shell in addition to the IDE's built-in debugger features. Python beginners should consider Thonny because its straightforward design makes it their preferred choice.

Figure 1: Overall Methodology

4.2 raining Parameters of the Machine Learning Model

A set of key parameters in Table (1) controls how Q-Learning algorithm learns and makes path selections in the network through its training parameters.

Table 1: Training Parameters of the Model

Parameter	Description	Value
Alpha	Learning Rate	0.1
Gamma	Discount Factor	0.9
Epsilon	Exploration Rate	1
min_epsilon	Minimum Value of Epsilon	0.01
epsilon_decay	Rate of Decrease of Epsilon Over Time	0.995
Episodes	Number of Training Episodes During Training	20,000

q_table	Q-Table for Storing Learning Values for All States and Actions	Matrix of zeros with size (number of nodes × number of nodes)

The learning rate alpha determines the speed of value updates and the gamma discount parameter balances present and future reward calculation. As training progresses the epsilon_decay schedule turns down exploration rates while increasing exploitation because the exploration rate controls how often the model explores instead of using its previous knowledge. Each model training session runs for a specific amount determined by "episodes" as the q_table accumulates all learned values needed for future decision making.

V. IMPLEMENTATION AND RESULTS

5.1 OSPF 25 Node Experiment

A randomly constructed network topology in Figure (2) establishes connections between 25 nodes. The network structure remains complicated because nodes follow a disorganized connection pattern that extends data transfer routes. Network operation reliability and flexibility remain high as multiple alternative paths link nodes which ensure system performance while minimizing disruptions caused by node or link failures.
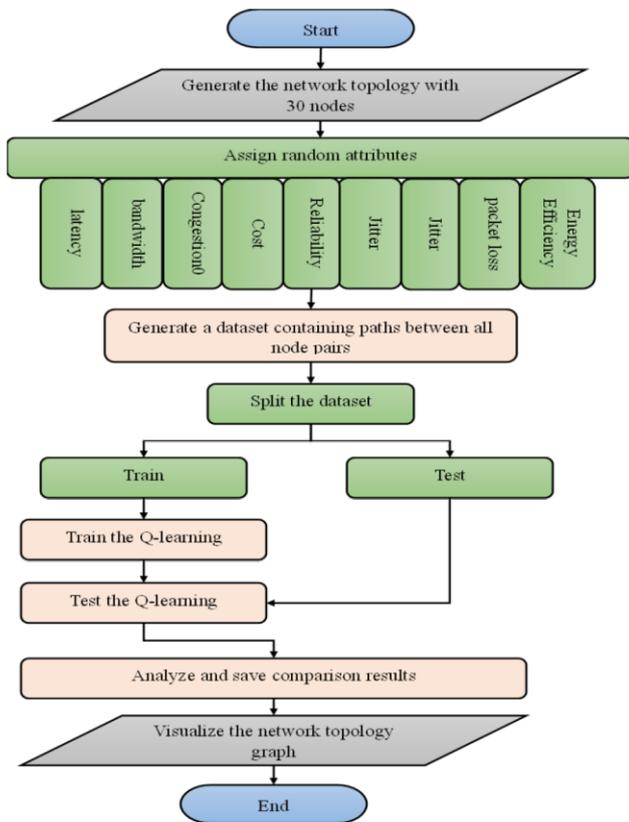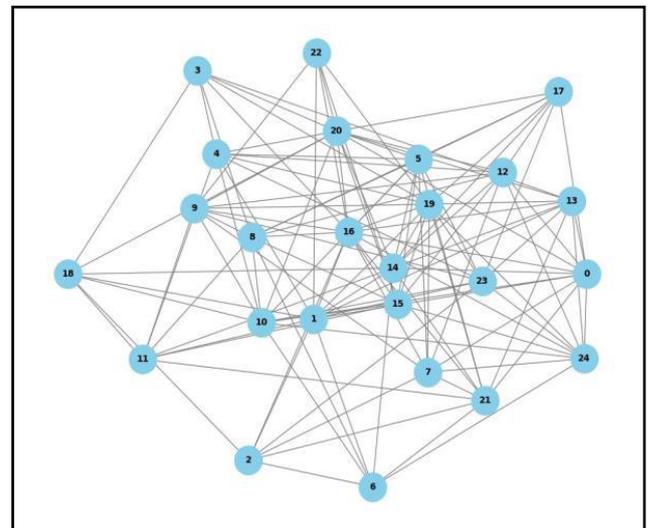

Figure 2: Network Topology with 25 Nodes

This topology system optimizes routing protocol performance evaluation by measuring both response times and bandwidth enhancements and congestion control metrics. The model uses computational methods to build network simulations which replicate the structure of extensive network systems used in practical applications.

During experiments on the 25-node topology the OSPF protocol delivered superior response time performance with an average duration of 4.20 milliseconds that surpassed the 5.50 milliseconds duration of Q-Learning. The Q-Learning algorithm showed better bandwidth performance than OSPF by choosing higher capacity paths which reached 5.36 Mbps compared to OSPF's 4.44 Mbps maximum. Through its adaptive network characteristics Q-Learning proved better at minimizing packet loss to 0.62 while also lowering congestion to 4.33 than the alternative 0.71 packet loss and 5.54 congestion.

OSPF demonstrated better stability and reliability in this evaluation with 0.83 reliability versus 0.82 and reduced costs at 10.20 versus 9.39 making it ideal for applications needing reliable performance. Q-Learning delivered superior energy efficiency to OSPF through its operation at 2.42 units while OSPF required 2.63 units, making Q-Learning ideal for applications focused on energy sustainability. This is summarized in Table (2).

Table 2: Comparison of OSPF and Q-Learning Performance in a 25-Node Network Topology

Criterion	OSPF	Q-Learning	Result
Response Time (ms)	4.2	5.5	OSPF has lower response time, suitable for time-sensitive apps.
Bandwidth (Mbps)	4.44	5.36	Q-Learning offers higher bandwidth, ideal for data-heavy apps.
Packet Loss	0.71	0.62	Q-Learning reduces packet loss, improving reliability.
Congestion (Units)	5.54	4.33	Q-Learning chooses less congested paths, reducing delays.
Jitter (ms)	5.03	4.72	Q-Learning provides less jitter, improving stability.
Reliability	0.83	0.82	OSPF is slightly more reliable, better for stable links.
Energy Efficiency (Units)	2.63	2.42	Q-Learning is more energy-efficient, suitable for resource-limited networks.
Total Cost	10.2	9.39	Q-Learning has lower total cost, reflecting better routing efficiency.

5.2 OSPF 50 Node Experiment

The Figure (3) illustrates a network topology which uses 50 nodes linked densely to achieve redundancy through backup connection paths. The network allows efficient analysis of OSPF and Q-Learning routing protocols through performance-based assessments that examine response times and congestion control operations and packet drop prevention along with energy management capabilities.

The framework delivers simulation capabilities that meet contemporary requirements of smart city networks and data centers thus generating innovative approaches for extensive network performance enhancement.
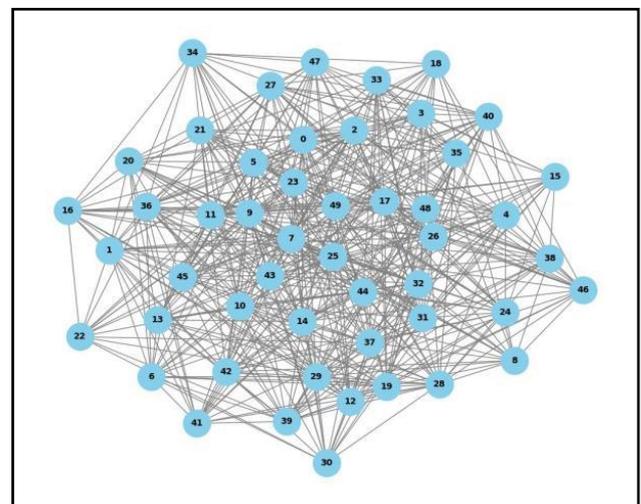
Figure 3: Topology of the eighth network (50 nodes)

The evaluation of OSPF routing protocols alongside Q-Learning deep learning methods shows their performance characteristics in network environments containing 50 nodes. The evaluation combines response time metrics with bandwidth measurements alongside packet loss statistics for analyzing network behavior alongside congestion and jitter effects and evaluates energy efficiency and total cost performance.

The evaluation shows that OSPF outperforms other protocols by delivering faster responses alongside improved reliability and reduced packet loss costs which enables time-sensitive applications.

Q-Learning shows better performance through efficient bandwidth utilization while reducing network congestion and jitter and improving energy efficiency for busy network environments. By evaluating these methods against each other we can determine which protocol best fulfills the needs of our network infrastructure.

Table 3: Comparison of OSPF Protocol and Q-Learning Algorithm Performance in 50-node Network Topology

Criterion	OSPF	Q-Learning	Result
Response Time (ms)	4.38	8.63	OSPF provides lower response time, making it suitable for time-sensitive applications.
Bandwidth (Mbps)	4.29	5.05	Q-Learning achieves higher bandwidth, enhancing data transfer speed and efficiency.
Packet Loss	0.66	0.71	OSPF reduces packet loss, improving network stability.
Congestion (Units)	4.77	2.97	Q-Learning selects less congested paths, improving performance in busy networks.
Jitter (ms)	4.82	3.97	Q-Learning provides lower jitter, enhancing transmission stability.
Reliability	0.79	0.77	OSPF shows slightly higher reliability, ensuring stable performance.
Energy Efficiency (Units)	2.55	2.31	Q-Learning is more energy-efficient.
Total Cost	9.75	10.82	OSPF offers a lower total cost.

5.3 BGP Protocol Model

Figure (4) depicts a connected system with randomly positioned and interconnected nodes that maintain full connectivity among all nodes. The network displays more than one link between adjacent nodes which creates alternative routing paths.
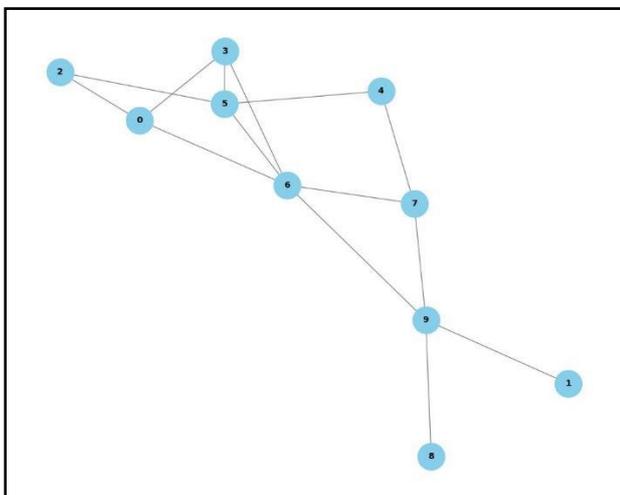
Figure 4: First model of a network topology consisting of 10 nodes

The Erdős–Rényi model generates a network which replicates traditional Internet topology through its asymmetric distribution of nodes and link-creation procedure between all node pairs according to a fixed probability value. Real-world network simulations are possible through this topology which enables researchers to test routing protocols and analyze performance across diverse conditions.

The Q-Learning protocol delivers superior results than BGP in various performance measurements according to Table (4). Q-Learning demonstrated faster response (8.42 ms) than BGP (12.84 ms) while reducing packet loss to 0.59 and congestion/jitter levels to 3.41/3.21 compared to BGP's 0.76/4.93/5.67. Q-Learning demonstrated enhanced resource efficiency through its lower overall cost (7.89) than BGP (11.34). The network utilizing BGP demonstrated better bandwidth performance by delivering 6.12 Mbps compared to Q-Learning's 4.89 Mbps.

Table 4: Performance comparison of BGP and Q-Learning protocols in a 10-node network

Metric	BGP	Q-Learning
Response Time (ms)	12.84	8.42
Bandwidth (Mbps)	6.12	4.89
Packet Loss	0.76	0.59
Congestion (Units)	4.93	3.41
Jitter (ms)	5.67	3.21
Cost	11.34	7.89

5.4 Comparison between OSPF and Q-Learning

The performance analysis of OSPF and Q-Learning methods (presented in Table 5) demonstrates several distinct operational characteristics. The results show that OSPF excels at multiple tasks but Q-Learning demonstrates exceptional capabilities in reducing congestion and packet loss.

Table 5: Comparison between OSPF and Q-Learning

Metric	OSPF	Q-Learning	Improvement for Q-Learning (%)
Response Time (ms)	4.35	6.76	-55.50%
Bandwidth (Mbps)	4.72	5.76	21.92%
Packet Loss	0.63	0.62	1.72%
Congestion (Units)	4.23	3.05	27.90%
Jitter (ms)	4.45	4.26	4.19%
Cost	8.57	8.93	-4.17%

- Response Time (ms): The results show OSPF required 4.35 milliseconds to respond while Q-Learning needed 6.76 milliseconds which demonstrates a 55.5% better

response time for OSPF. Because minimizing latency stands as a critical factor for time-sensitive applications OSPF proves itself more efficient in reducing response times.

- Bandwidth (Mbps): Q-Learning demonstrated better bandwidth performance than OSPF by reaching 5.76 Mbps at 21.9% greater speed. Q-Learning demonstrates greater ability to handle large data transfers which makes it a better choice for environments that require high throughput performance.

- Packet Loss: The implementation of Q-Learning resulted in a 1.72% improvement for packet reliability when compared to OSPF as Q-Learning achieved a packet loss of 0.62 compared to OSPF's 0.63.

- Congestion (Units): The congestion measurement at 4.23 units for OSPF decreased to 3.05 units with Q-Learning which produced a 27.9% improvement for Q-Learning. Q-Learning demonstrates better network path selection capabilities by avoiding congested routes.

- Jitter (ms): OSPF exhibited a slight deviation in jitter statistics at 4.45 ms while Q-Learning measured 4.26 ms. Q-Learning demonstrates a 4.19% enhancement in performance compared to conventional methods even though this improvement is considered small for improving data transmission stability.

- Cost: The operational expense of Q-Learning protocols increased by 4.17% to reach 8.93 compared to OSPF's 8.57. Q-Learning demonstrates improved operational characteristics but its implementation leads to marginally elevated operating expenses.

5.5 Comparison between BGP and Q-Learning

The performance analysis in Table 6 reveals that Q-Learning surpasses BGP by demonstrating superior results throughout multiple measurement points.

Table 6: Comparison between BGP and Q-Learning

Metric	BGP	Q-Learning	Improvement for Q-Learning (%)
Response Time (ms)	12.68	8.78	30.70%
Bandwidth (Mbps)	6.01	4.87	-23.40%
Packet Loss	0.78	0.62	20.50%
Congestion (Units)	5.22	3.78	27.60%
Jitter (ms)	6.17	4.73	23.40%
Cost	13.92	10.82	22.30%

- Response Time (ms): Absolute response time measurements indicate BGP required 12.68 milliseconds but Q-Learning required only 8.78 milliseconds thereby

improving Q-Learning performance by 30.7 percent. The data shows Q-Learning effectively minimizes latency to ensure quick responses essential for fast-response applications.

- Bandwidth (Mbps): BGP demonstrates superior bandwidth performance by delivering 6.01 Mbps compared to Q-Learning's 4.87 Mbps bandwidth. The performance of Q-Learning decreased by 23.4% during testing which demonstrates that BGP technology provides more suitable solutions for bandwidth-intensive applications.

- Packet Loss: The comparison reveals that BGP maintained a packet loss rate of 0.78 but Q-Learning achieved a rate of 0.62 representing a 20.5% improvement in reliability and data transmission quality.

- Congestion (Units): Q-Learning showed superior network congestion management by reducing congestion units from 5.22 to 3.78 which represented a 27.6% improvement over BGP.

- Jitter (ms): A significant decrease in jitter occurred when using Q-Learning instead of BGP because the Q-Learning values were 4.73 ms while BGP readings reached 6.17 ms leading to performance enhancements of 23.4%. Q-Learning demonstrates better management of network dynamic conditions because of its superior performance.

- Cost: Q-Learning demonstrated superior resource allocation by reducing costs by 22.3% compared to BGP which reported 13.92 in costs resulting in a lower figure of 10.82 for Q-Learning.

VI. CONCLUSIONS

An intelligent Q-Learning-based routing model provides a strong alternative solution to classical routing protocols. The model adjusts its network operations in real-time to maximize bandwidth usage while minimizing congestion and packet loss which leads to substantial performance benefits. The Q-Learning model yields improved bandwidth usage of 21.92% combined with 27.90% reduced congestion and packet loss dropped by 1.72%. The performance of this model surpasses BGP by reducing response time by 30.70% and lowering jitter by 4.19% which makes it ideal for time-sensitive data-intensive applications. The Q-Learning model offers a routing system with adaptive capability and flexibility that handles the weaknesses of OSPF's response time strengths and BGP's bandwidth access. The upcoming research direction for Q-Learning model scalability improvement aims to optimize its functionality when used in networks composed of thousands of nodes. The model needs further improvement through deep reinforcement learning (DQN) implementation for optimizing decision-making alongside deployment tests in different network conditions to gauge operational effectiveness.

REFERENCES

[1] A. F. M. Z. Abadin, S. Sarker, M. S. Hosain, M. M. Ahmed, and A. Imtiaz, "A Comprehensive Study and Analysis of Different Routing Protocols for Enterprise LAN," *... J. Sci. ...*, no. December, pp. 20–28, 2021.

[2] K. Shahid, S. N. Ahmad, and S. T. H. Rizvi, "Optimizing Network Performance: A Comparative Analysis of EIGRP, OSPF, and BGP in IPv6-Based Load-Sharing and Link-Failover Systems," *Futur. Internet*, vol. 16, no. 9, 2024.

[3] R. N. Devikar, D. V. Patil, and V. Chandraprakash, "Study of BGP convergence time," *Int. J. Electr. Comput. Eng.*, vol. 6, no. 1, pp. 413–420, 2016.

[4] A. Srinivasan, "Reinforcement Learning: Advancements, Limitations, and Real-world Applications," *Interational J. Sci. Res. Eng. Manag.*, vol. 07, no. 08, 2023.

[5] M. Y. Kashmola and N. R. H. Al-Kadow, "Improving AntNet algorithm in routing protocols to find the optimal path according to time," *2013 Int. Conf. Electr. Commun. Comput. Power, Control Eng. ICECCPCE 2013*, pp. 76–81, 2014.

[6] M. N. A. Al-Hamadani, M. A. Fadhel, L. Alzubaidi, and H. Balazs, "Reinforcement Learning Algorithms and Applications in Healthcare and Robotics: A Comprehensive and Systematic Review," *Sensors*, vol. 24, no. 8, 2024.

[7] M. I. Alsaydia, et al., "Analysis and Performance Evaluation of OSPF and RIP Routing Protocols Using QualNet," *Int. J. Comput. Netw. Technol.*, vol. 4, no. 3, pp. 125–132, 2016.

[8] M. A. Al-Jawaherry, A. A. Abdulmajeed, and T. M. Tawfeeq, "Developing a Heuristic Algorithm to Solve Uncertainty Problem of Resource Allocation in a Software Project Scheduling," *Iraqi J. Sci.*, vol. 63, no. 5, pp. 2211–2229, 2022.

[9] R. Desai and B. P. Patil, "Adaptive routing based on predictive reinforcement learning," *Int. J. Comput. Appl.*, vol. 40, no. 2, pp. 73–81, 2018.

[10] Z. Mammeri, "Reinforcement learning based routing in networks: Review and classification of approaches," *IEEE Access*, vol. 7, no. June, pp. 55916–55950, 2019.

[11] M. Guo, P. Wang, C. Y. Chan, and S. Askary, "A Reinforcement Learning Approach for Intelligent Traffic Signal Control at Urban Intersections," *2019 IEEE Intell. Transp. Syst. Conf. ITSC 2019*, no. June, pp. 4242–4247, 2019.

[12] H. Yang, X. Xie, and M. Kadoch, "Machine Learning Techniques and A Case Study for Intelligent Wireless Networks," *IEEE Netw.*, vol. 34, no. 3, pp. 208–215, 2020.

[13] Z. Zhang, Y. Yuan, and X. Yang, "A Hybrid Machine Learning Approach for Freeway Traffic Speed Estimation," *Transp. Res. Rec.*, vol. 2674, no. 10, pp. 68–78, 2020.

[14] D. J. Shin and J. J. Kim, "Deep reinforcement learning-based network routing technology for data recovery in exa-scale cloud distributed clustering systems," *Appl. Sci.*, vol. 11, no. 18, 2021.

[15] G. Kim, Y. Kim, and H. Lim, "Deep Reinforcement Learning-Based Routing on Software-Defined Networks," *IEEE Access*, vol. 10, pp. 18121–18133, 2022.

[16] K. F. Chu, A. Y. S. Lam, and V. O. K. Li, "Traffic Signal Control Using End-to-End Off-Policy Deep Reinforcement Learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 7, pp. 7184–7195, 2022.

[17] H. Cheng, Y. Luo, L. Zhang, and Z. Liao, "A Reinforcement Learning-Based Traffic Engineering Algorithm for Enterprise Network Backbone Links," *Electron.*, vol. 13, no. 8, 2024.

[18] A. yaseen Hamoo and S. A. Muhamed, "Designing a Monitoring Tool for Mosul University Network to Detect Worm Infected Computers," *Alrafidian J.*, vol. 2, pp. 43–60, 2013.

[19] J. Ferdaus and R. Salihi, "Routing: internet routing protocols and algorithms," *no. Sept.*, no. September, pp. 0–19, 2015.

[20] O. A. Hazim and J. M. Abdul-Jabbar, "An Adaptive Triple-Path Mechanism for Congestion Control in Computer Networks," *AL-Rafdain Eng. J.*, vol. 22, no. 4, pp. 44–56, 2014.

[21] S. Sendra, P. A. Fernández, M. A. Quilez, and J. Lloret, "Study and Performance of Interior Gateway IP routing Protocols," *Netw. Protoc. Algorithms*, vol. 2, no. 4, 2011.

[22] D. Mitra, S. Sarkar, and D. Hati, "A Comparative Study of Routing Protocols," *Int. Res. J. Adv. Eng. Sci.*, no. January 2021, 2016.

[23] Y. Jiang and J. Zhu, "Environment-Aware Adaptive Reinforcement Learning-Based," pp. 1–31, 2024.

[24] V. S. R. Kosuru and A. K. Venkitaraman, "Developing a Deep Q-Learning and Neural Network Framework for Trajectory Planning," *Eur. J. Eng. Technol. Res.*, vol. 7, no. 6, pp. 148–157, 2022.

[25] J. Yang, K. Zhu, Y. Ran, W. Cai, and E. Yang, "Joint Admission Control and Routing Via Approximate Dynamic Programming for Streaming Video Over Software-Defined Networking," *IEEE Trans. Multimed.*, vol. 19, no. 3, pp. 619–631, 2017.

[26] C. L. Duenas Santos, A. M. Mezher, J. P. Astudillo León, J. Cardenas Barrera, E. Castillo Guerra, and J. Meng, "Q-RPL: Q-Learning-Based Routing Protocol for Advanced Metering Infrastructure in Smart Grids,"

Sensors, vol. 24, no. 15, pp. 1–29, 2024.

****