Journal of Wireless Networks and Communication Systems
Vol. 2, Issue. 2, Apr-Jun 2026, pp. 37~47
ISSN: 3107-6432, DOI: 10.32595/jwncs/v2i2.2026.31

A Cognitive SDN-Driven Adaptive Load Balancing Architecture for Next-Generation Wireless Edge Networks

Dr.Akhila K R[1], Anu Krishna[2]

[1]Lecturer, Cybersecurity and networks Unit, Department of Computing and Information Sciences, University of Technology and Applied Science, Shinas, Oman.
[2]Lecturer, Software Engineering and Data Technologies Unit, Department of Computing and Information Sciences, University of Technology and Applied Science, Shinas, Oman.

Article Info

Article History:

Received Apr 11, 2026
Revised May 10, 2026
Accepted Jun 12, 2026

Keywords:

Software-Defined Networking (SDN),
Intelligent Load Balancing,
Edge Computing Networks,
Network Performance Optimization,
Wireless Communication Systems.

ABSTRACT

Software-Defined Networking (SDN) has gained significant attention as a flexible solution for managing modern wireless edge networks. Despite its advantages, issues such as uneven traffic distribution and node congestion continue to affect overall network performance. In this work, an intelligent SDN-based adaptive load balancing framework is proposed to address these challenges. The approach continuously observes network conditions, including traffic load and available bandwidth, and makes real-time decisions to redirect data flows toward less congested paths. To evaluate its effectiveness, the proposed framework is implemented using Mininet-WiFi and compared with conventional techniques such as Round Robin and static load balancing. Performance is assessed using key metrics including throughput, latency, packet loss, and bandwidth utilization. The results indicate noticeable improvements, with higher throughput, reduced delay, and lower packet loss under varying traffic conditions. In particular, the system demonstrates better adaptability in handling dynamic network scenarios. Overall, the proposed framework provides a practical and efficient solution for improving resource utilization and Quality of Service in wireless edge environments, making it suitable for applications such as IoT systems, smart cities, and next-generation wireless networks.

Corresponding Author:

Dr.Akhila K R,
Lecturer, Cybersecurity and networks Unit, Department of Computing and Information Sciences,
University of Technology and Applied Science, Shinas.

1. INTRODUCTION

In recent years, there has been an explosive increase in the number of wireless communication technologies. This has resulted in the creation of vast amounts of new data processing and storage capabilities on various edge computing environments that are physically located closer to end-users than traditional data centers [1]. The introduction and development of billions of IoT devices, smart city infrastructure, autonomous systems, and real-time multimedia applications have all contributed to a tremendous increase in the volume of network traffic being generated at the edge of the network. Therefore, wireless edge networks need to deliver massive connectivity, low-latency communications, efficient use of resources, and reliable service. Unfortunately, due to the dynamic nature of wireless environments, wireless edge networks might present themselves with varying degrees of difficulty including; congestion and imbalanced distributions of bandwidth; packet loss; and increased latency in the transmission of data [2].

Distributed control mechanisms used in classic network architectures rely on routing and forwarding decisions made by involved devices [3]. Although these types of systems are very common, their lack of flexibility makes them unsuitable for operating in highly dynamic wireless edge environments. Static traffic management techniques often produce delayed responses to sudden changes in the state of the network, resulting in inefficient network resource use [4]. As a result, maintaining Quality of Service (QoS) is becoming increasingly challenging with respect to managing the volume of data produced by contemporary wireless applications over large geographic areas.

Software-Defined Networking (SDN) is transforming the way networks can be structured, which is beneficial because it solves many issues that traditional networking structures face. The two planes of networking have been separated; there is now a control plane, and a data plane, with a centralized way of controlling your network and programming it through this centralized network controller, which allows for a view of the entire network in a global sense so that intelligent choices can be made regarding how traffic will flow on your network. With this centric structure will enable more flexibility in networking, enable easier management of networks, and also allow for more efficient use of resources [5]. When SDN is used in wireless for the edge, there are many benefits, including dynamic routing and traffic engineering, virtualization, and continual or real-time monitoring of the network condition.

Load balancing continues as one of the most significant obstacles to overcoming via Software Defined Networking (SDN) technology on a wireless Edge Network despite the many innovations developed by SDN. Load balancing refers to the process in which network traffic is distributed among multiple nodes, links and/or servers in a manner that avoids overloading a single node and helps improve performance. In wireless networks with dynamic traffic environments, there will almost always be changes in the way users move around, how many users are using the system at any one time and how much bandwidth is available at any given moment. If the distribution of network traffic is uneven, this will result in some nodes being overloaded and others being underutilised. In addition to having an adverse effect on the throughput of the system, an unequal distribution of traffic can also result in an increase in latency and an overall decrease in the efficiency of the entire network.

SDN-enabled network researchers have proposed multiple strategies for load balancing [6]. Existing techniques include Round Robin Scheduling, Static Routing and Equal Cost Paths that provide basic mechanisms for the distribution of traffic among multiple links based on their load; however, these methods are typically static in nature and do not take into account current conditions in the network, which results in their inability to dynamically respond to changing traffic patterns or maintain an optimal level of service when faced with variations in the amount of congestion in the network. Furthermore, most existing methods of load balancing have been developed for use in wired networks and do not account for the unique characteristics of load balancing in wireless edge networks; for example, a wireless edge network has a dynamic topology, suffers from variations in the amount of available bandwidth between different nodes, and is impacted by mobility of users and devices.

In order to alleviate these issues, intelligent and adaptive load balancing mechanisms must be put in place. These mechanisms should monitor the current state of the network, identify the level of congestion, and dynamically change the path of traffic to lower-utilization paths. Adaptive decision-making in SDN architecture integrates efficient traffic management with good network performance [7,8]. Adaptive load balancing is thus able to improve the use of resources and Quality of Service in wireless edge environments by taking advantage of centralized control and real-time visibility of the network.

In response to these challenges, this research discusses the development of an Intelligent SDN-Based Adaptive Load-Balancing Framework within Wireless Edge Networks. The framework collects and evaluates different parameters from the network, including the load on individual nodes, available capacity (bandwidth) from wireless devices, and current traffic flow condition. Depending on what is learned from monitoring the above parameters, the SDN controller will redirect traffic over different paths to avoid congestion and to produce an efficient balance of the loads between the connected devices in real time, which is different from traditional methods that rely only on pre-existing routing protocols and therefore were not able to make use of all available resources.

This framework aims to enhance the performance of networks through reducing congestion and redistributing network traffic more optimally throughout wireless edge infrastructures; this will be done by implementing the proposed framework in a simulated SDN-enabled wireless environment using Mininet-WiFi and then comparing that the results against the traditional load balancing approaches. The performance analysis is conducted using several metrics including throughput, latency, packet loss and bandwidth utilization. The experimental results show that there is a significant improvement in the overall efficiency of the network with an increase in consistency of performance as traffic levels change.

The major contributions of this research are summarized as follows:

- The framework introduces a dynamic traffic management mechanism that continuously monitors network conditions and redistributes traffic in real time.
- It enables efficient handling of network overload by intelligently directing traffic toward underutilized nodes.
- The proposed system is implemented and evaluated using Mininet-WiFi simulation.
- Simulation results demonstrate significant improvements in network performance metrics.
- The framework achieves higher throughput compared to conventional load balancing techniques.
- It effectively reduces latency and minimizes packet loss.
- Improved bandwidth utilization is observed due to efficient traffic distribution.

This paper will continue as follows after this introduction: Section 2 gives an overview of existing SDN based load balancing systems and discusses the key papers relating to these methods. Section 3 will go into depth on the overall concept of the proposed system architecture and adaptive load balancing methods that we propose. Section 4 presents the methodology and result work. Finally we will conclude with future research ideas in Section 5.

2. LITERATURE REVIEW

[9] Proposed a new adaptive load balancing architecture using link-state prediction in the context of Software Defined Networks. This approach uses real-time monitoring of the network and predictive analysis to assess and estimate future link conditions and dynamically route traffic across several available paths. This architecture also increases the stability of the network and

reduces congestion by enabling the proactive identification of overloaded links. Experimental results indicate that both throughput and transmission efficiency were improved as compared to traditional routing techniques. However, this model mostly focused on assessing future link condition with no consideration given for other service-level functionality that could also affect the routing decision.

[10] Proposed an Adaptive Symmetrical Load Balancing (ASLB) method for Software-Defined WiFi networks. The system used OpenFlow-capable access points and applied metrics like packet quantity, packet category, and latency for traffic distribution. An Analytical Hierarchy Process (AHP) was utilized to enhance flow transitions between controllers. The findings revealed increased fairness, better traffic allocation, and minimized controller overload. Nonetheless, the framework primarily focused on WiFi settings and did not provide support for diverse wireless edge networks.

[11] Presented a thorough overview of the current state of research on QoS-aware load balancing approaches for SDN-enabled IoT environments. The authors showed how using a centralized SDN controller can help with managing traffic flow, providing better resource usage, and providing better quality of service to users. This paper discusses multiple different load balancing methods based upon metrics such as throughput, latency, packet loss and network scalability. The authors concluded that load-balancing systems that are both intelligent and adaptive to the current environment outperform static load balancing systems by a significant margin in dynamic IoT environments. Despite these findings, the authors point out the necessity of creating more effective real-time optimization models that can support dynamic traffic patterns.

The dynamic weighted round-robin (DWRR) load balancing mechanism with distributed SDN controllers is presented by [12] to improve scalability and traffic distribution. Weight adjustments are done based upon the condition of whole network, effectively reducing controller overload and creating better response time through higher throughput than what traditional methods provide. Ryu and Mininet were used for implementation of the proposed method and results were significantly superior when compared to traditional methods. Improved performance was achieved with both but there is no consideration of multi-metric path optimization outside of the controller level for the wireless edge networks.

3. METHODOLOGY

3.1 System Architecture

Centralized control and global visibility of your network are provided by the software-defined networking (SDN) paradigm. Traditional distributed networks do not consider the network as a whole, so they make local routing decisions based solely on local conditions. This leads to poorly balanced loads in the network. The controller in the SDN-based architecture uses the state of the entire network, which is maintained by the controller, for its decision-making processes. The centralized architecture of an SDN is particularly appropriate for edge networks because the topologies, traffic demands, and channel conditions of edge networks change frequently. An illustration of the SDN-based architecture proposed by this thesis can be found in Figure 1.

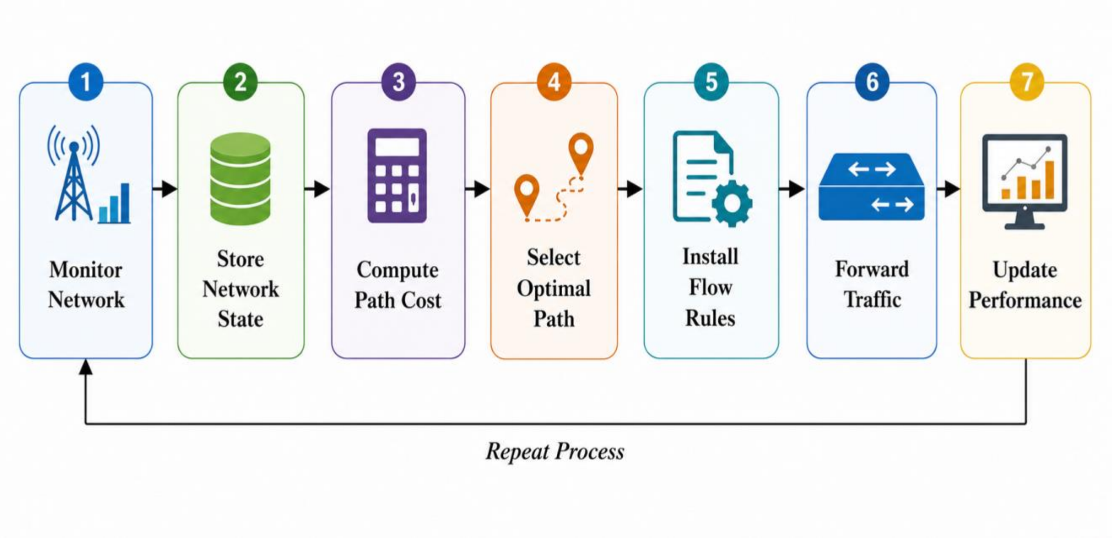

Figure 1. Proposed System Framework

3.2 Network State Modeling

In order to achieve effective load balancing, accurate mathematical modeling of network conditions is necessary. This model consists of a graph $G = (V, E)$ where V represents all of the wireless nodes and E represents all of the communication links. Each node and link will have associated dynamic parameters including load, bandwidth, delay and packet loss.

The load on a node is characterized as the relationship between incoming traffic and its highest processing capability:

$$\boldsymbol{L_i} = \frac{T_i}{C_i^{max}} \qquad (1)$$

This normalization guarantees that node load is expressed within a limited range of $0 \leq L_i \leq 1$, facilitating equitable comparison among diverse nodes.

Similarly, the available bandwidth of a link is modeled as:

$$B_i = B_i^{max} - B_i^{used} \qquad (2)$$

This formulation reflects real-time link usage and allows for the detection of congested links as available bandwidth drops.

3.3 Multi-Metric Cost Function Formulation

Conventional techniques of load balancing use one or few metrics alone to optimize load balancing. However, this is inadequate for dynamically changing wireless environments. Thus, in response to this problem, we propose a multi-objective optimization model that takes into account many different performance metrics together.

The overall expense of a path p is characterized as a weighted aggregation of normalized metrics:

$$C_p = \alpha \hat{L}_p + \beta \hat{D}_p + \gamma \hat{P}_p + \delta \hat{B}_p \qquad (3)$$

In this context, every term signifies a standardized performance measure. The normalization procedure guarantees that all parameters lack dimensions and can be compared.

$$\hat{L}_p = \frac{L_p}{max_{p\in \mathrm{q}} L_p}, \hat{D}_p = \frac{D_p}{max_{p\in \mathrm{q}} D_p} \tag{4}$$

$$\hat{P}_p = \frac{P_p}{max_{p\in \mathrm{q}} P_p}, \hat{B}_p = \frac{B_{max}}{B_p} \tag{5}$$

In this formulation, increased load, delay, and packet loss raise the cost, while greater bandwidth alleviates congestion and is modeled inversely. The coefficients of weighting meet:

$$\alpha + \beta + \gamma + \delta = 1 \tag{6}$$

This constraint guarantees that all parameters have equal importance and provides the model with the flexibility to modify its parameters in accordance with network requirements. As an example, applications that are heavily dependent on latency may have a relative weight assigned to delay.

3.4 Optimal Path Selection Strategy

The load balancing problem is formulated as a minimization problem, where the objective is to select the path with minimum cost:

$$p^* = \arg\min_{p\in \mathrm{q}} C_p \tag{7}$$

With this form of optimization, all traffic will be routed through low congestion/default delay/efficient bandwidth paths. Dynamic routing methods differ to static routing methods in the sense that they continuously adapt to dynamic network conditions by updating cost values.

3.5 Dynamic Traffic Adaptation

To ensure peak performance in changing conditions, the SDN controller regularly revises routing choices according to newly detected network statuses. The decision to forward at time t is represented as

$$F(t) = p^*(t) \tag{8}$$

The adaptation of the system is reflected in its time dependent formulations where the controller will recalculate and update flow rules (based upon changes in network conditions) as to assure continuous load balancing and to prevent performance degradation.

3.6 Theoretical Advantages of the Proposed Model

There are a number of theoretical benefits from using the proposed methodology as opposed to more typical methodologies. First, the use of normalized multi-metric optimization provides for an equal distribution of weight among metrics, thus avoiding any potential for bias in favor of a single metric. Second, centralized SDN architecture promotes global optimization as compared to local optimization in making routing decisions. Third, the adaptive nature of the methodology promotes an effective response to dynamic network conditions due to the fact that the methodology is built on data and analytics rather than fixed configurations. Finally, the formulation of the problem as a minimization problem provides for the convergence toward optimal routing decisions (as defined by minimization), rather than optimal routing decisions through other means such as iteration.

3.7 Performance Evaluation Model

To quantitatively evaluate the effectiveness of the proposed framework, standard performance metrics are defined mathematically. The throughput of the network is given by:

$$\text{Throughput} = \frac{\sum_{k=1}^{N} s_k}{T} \qquad (9)$$

The average end-to-end delay is expressed as:

$$D_{avg} = \frac{1}{N}\sum_{k=1}^{N}(t_k^{receive} - t_k^{send}) \qquad (10)$$

The packet loss ratio is defined as:

$$PLR = \frac{N_{sent} - N_{received}}{N_{sent}} \qquad (11)$$

Bandwidth utilization is computed as:

$$U = \frac{B_{Used}}{B_{max}} \qquad (12)$$

These formulations provide a rigorous basis for evaluating system performance and comparing it with conventional load balancing techniques.

4. RESULTS AND DISCUSSION

Using the Mininet-WiFi simulation environment, we evaluated how well our proposed intelligent SDN-based adaptive load-balanced framework can perform in a variety of different conditions. In addition to this, we created an environment where the traffic patterns we used were dynamic in order to test how adaptive and efficient they would be against traditional load-balancing methods like round robin and static routing.

4.1 Throughput Analysis

Compared to conventional routing algorithms, the new framework delivers greater overall throughput because it is designed to utilize less congested links by dynamically selecting from available links in the network. Further, by constantly collecting/monitoring current conditions of the network and redistributing traffic accordingly, the framework can minimize the number of lost packets and provide better performance for packets transmitted within the network. The data in Figure 2 confirm the increase in overall throughput that results from using this novel approach, particularly during peak usage times when the conventional methods often exhibit extensive amounts of congestion and/or poor load balancing.

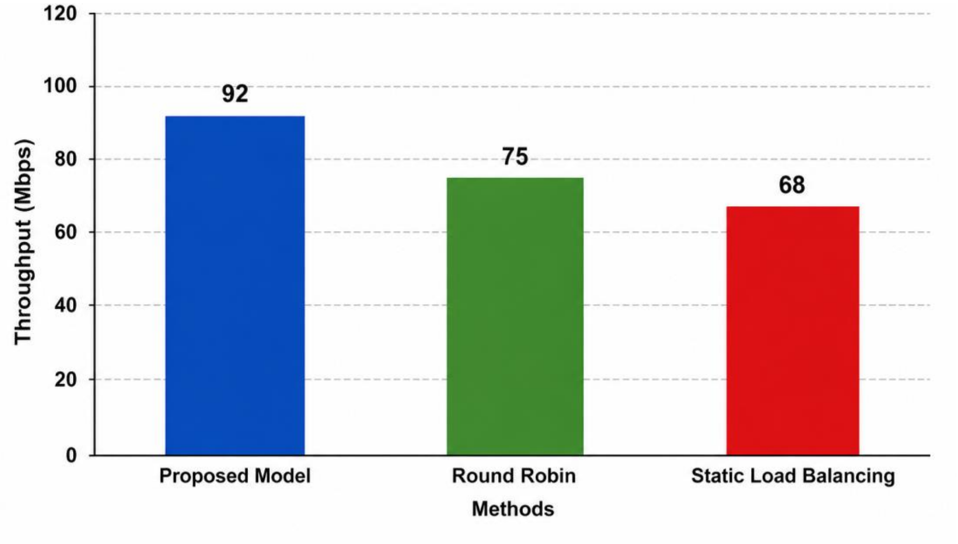

Figure 2. Throughput Comparison

4.2 Latency Analysis

The proposed model's end-to-end delay is greatly decreased compared to the previous static models because of their SDN controller; it chooses the best available route between each source and destination. The proposed system does not overload nodes by using its SDN components to evaluate current real-time delay and congestion conditions; as such, the amount of time it takes to send data through a node is also decreased. The graph in Figure 3 indicates that the overall end-to-end delay remains low when a network is loaded heavily; the adaptive framework shows continued low latencies when the network is busy.

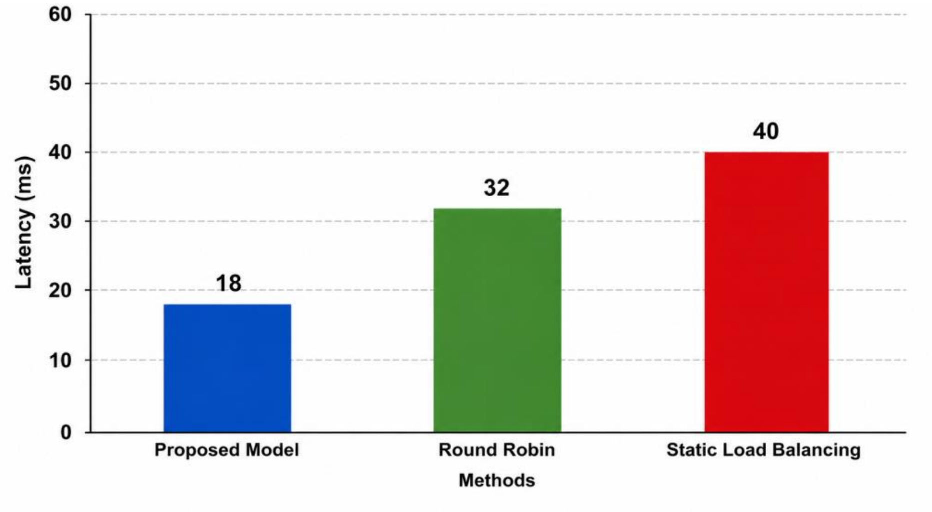

Figure 3. Latency Comparison

4.3 Packet Loss Evaluation

In the suggested framework, packet loss is reduced through good efficiency in terms of traffic distribution, and proactive measures in regard to congestion avoidance. The framework also uses packet loss as part of a cost function so that data is routed across stable/reliable paths. The experimental results, as shown in Figure 4, show that the packet loss ratio for the proposed method is less than that for conventional methods, particularly under the conditions of a dynamic and/or high density networks.

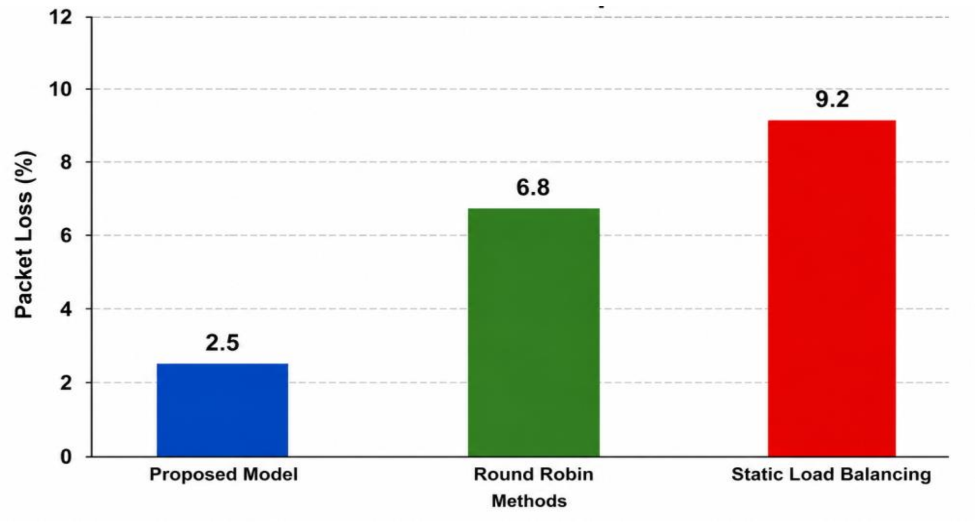

Figure 4. Packet Loss Comparison

4.4 Bandwidth Utilization

In the suggested framework, packet loss is reduced through good efficiency in terms of traffic distribution, and proactive measures in regard to congestion avoidance. The framework also uses packet loss as part of a cost function so that data is routed across stable/reliable paths. The experimental results, as shown in Figure 4, show that the packet loss ratio for the proposed method is less than that for conventional methods, particularly under the conditions of a dynamic and/or high density networks.

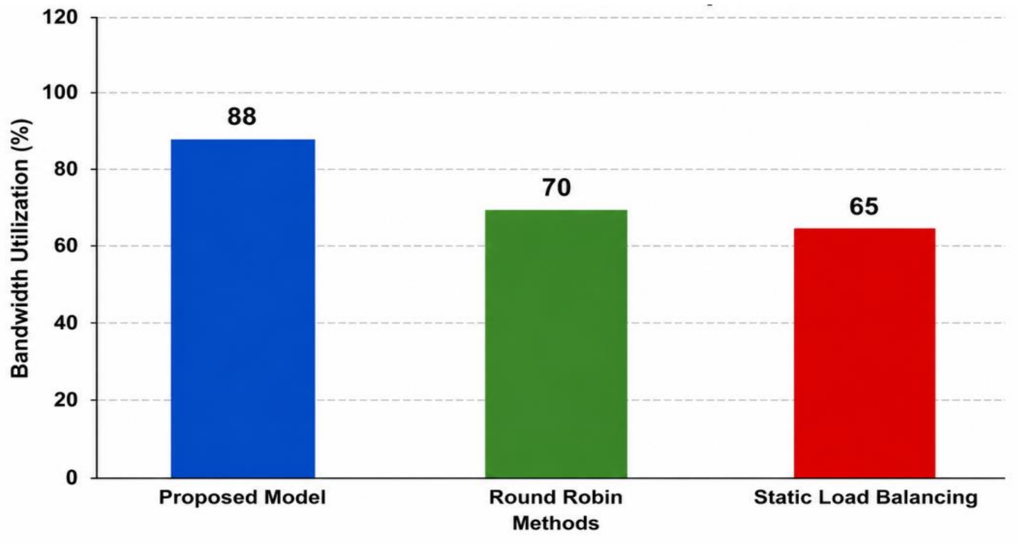

Figure 5. Bandwidth Utilization Comparison

4.5 Comparative Analysis

In this paper, we compare and evaluate the new model against previous load balancing methods found in literature and Table 1 shows how the new concept is superior to traditional techniques across all measureable performance factors. The traditional method of Round Robin (traffic evenly distributed regardless of how the network is doing) will not appropriately distribute traffic during dynamic conditions. Another example is static routing that does not adapt. In contrast to these techniques, the proposed solution uses real-time monitoring and multi-metric optimization of routing to improve overall performance.

Table 1. Overall Performance Comparison

Metric	Proposed Model	Round Robin	Static Load Balancing
Throughput (Mbps)	92	75	68
Latency (ms)	18	32	40
Packet Loss (%)	2.5	6.8	9.2
Bandwidth Utilization (%)	88	70	65

In general, the suggested framework shows substantial enhancements in throughput, latency, packet loss, and bandwidth usage, establishing it as a strong and scalable option for future wireless edge networks.

4.6 Discussion

The performance improvements we see validate how effectively an adaptive load balancing mechanism is being integrated with a software-defined network (SDN). Having a centralized controller in the SDN provides global visibility of the network and allows for informed routing decisions to be made. Combining this, with an addition of a multi-metric cost function improves the decision making process for both diversified metrics in a single time period.

The system is designed to adapt to changes in the network and can therefore react quickly to fluctuations in conditions. This makes the system applicable for various real-world scenarios including the Internet of Things (IoT), smart cities and edge computing. Due to ongoing monitoring and decisions being made by the server, there is a small amount of processing lift being added by these processes compared to traditional methods. Nevertheless, the greater efficiency and performance achieved through the additional workload will far exceed any processing delay by the added work for the server.

5. CONCLUSION

An Adaptive Load Balancing Framework based on Intelligent Software Defined Networking for Wireless Edge Networks is presented in this paper to improve Network Performance in Dynamic Environments. The Adaptive Load Balancing Framework achieves this by providing centralized control and real-time monitoring of the Network's Conditions. This allows the Framework to effectively manage and distribute traffic over all available paths; thereby optimally distributing the Network's Load, decreasing congestion and improving the overall performance of the Network. Additionally, the use of a Multi-Metric Cost Function within the framework enables the Central Controller to make Routing Decisions based on Key Parameters associated with each of the available paths i.e. Load, Delay, Packet Loss and Available Bandwidth. Simulation Results demonstrated the superiority of the proposed approach over Traditional Methods with respect to Throughput, Latency, Packet Loss and Bandwidth Utilization. The Adaptive Load Balancing Framework is, therefore, capable of continuously optimizing and being responsive to changes in the Network by supporting new applications (such as Internet of Things, Smart Cities and Edge Computing) that are emerging in modern times. The minor computational overhead introduced by the Adaptive Load Balancing Framework is extremely insignificant when compared to the Performance Gains achieved with the Adaptive Load Balancing Framework relative to Traditional Methods; highlighting the Adaptive Load Balancing Framework's potential as a Scalable, Efficient Solution for Next Generation Wireless Networks.

REFERENCES

[1] Kreutz, D., Ramos, F. M., Verissimo, P. E., Rothenberg, C. E., Azodolmolky, S., & Uhlig, S. (2014). Software-defined networking: A comprehensive survey. *Proceedings of the IEEE*, *103*(1), 14-76.

[2] Bera, S., Misra, S., & Vasilakos, A. V. (2017). Software-defined networking for internet of things: A survey. *IEEE Internet of Things Journal*, *4*(6), 1994-2008.

[3] Chen, J., Wang, Y., Huang, X., Xie, X., Zhang, H., & Lu, X. (2022). ALBLP: Adaptive Load-Balancing Architecture Based on Link-State Prediction in Software-Defined Networking. *Wireless Communications and Mobile Computing*, *2022*(1), 8354150.

[4] Manzoor, S., Mazhar, F., Binaris, A., Hassan, M. U., Rasab, F., & Mohamed, H. G. (2023). An adaptive symmetrical load balancing scheme for next generation wireless networks. *Symmetry*, *15*(7), 1316.

[5] Shona, M., & Sharma, R. (2025). Design and Deployment of a Dynamic Weighted Round-Robin SDN Load Balancing Mechanism with Distributed Controllers. *Engineering, Technology & Applied Science Research*, *15*(6), 30260-30266.

[6] Abbas, N., Zhang, Y., Taherkordi, A., & Skeie, T. (2017). Mobile edge computing: A survey. *IEEE Internet of Things Journal*, *5*(1), 450-465.

[7] Pribadi, A., & Kaburuan, R. (2026). IoT-Based Load Management System with Edge Computing For Real-Time Decision Making. *Journal of Wireless Networks and Communication Systems*.

[8] Surender Kumar Yallagoud. (2022). A Study on Enhancement of System Security in Openflow Structure Utilizing Software Defined Networking, *IIRJET*, *2*(4). 66-70. https://doi.org/10.32595/iirjet.org/v2i4.2017.38

[9] Chen, J., Wang, Y., Huang, X., Xie, X., Zhang, H., & Lu, X. (2022). ALBLP: Adaptive Load-Balancing Architecture Based on Link-State Prediction in Software-Defined Networking. *Wireless Communications and Mobile Computing*, *2022*(1), 8354150.

[10] Manzoor, S., Mazhar, F., Binaris, A., Hassan, M. U., Rasab, F., & Mohamed, H. G. (2023). An adaptive symmetrical load balancing scheme for next generation wireless networks. *Symmetry*, *15*(7), 1316.

[11] Rostami, M., & Goli-Bidgoli, S. (2024). An overview of QoS-aware load balancing techniques in SDN-based IoT networks. *Journal of cloud computing*, *13*(1), 89.

[12] Shona, M., & Sharma, R. (2025). Design and Deployment of a Dynamic Weighted Round-Robin SDN Load Balancing Mechanism with Distributed Controllers. *Engineering, Technology & Applied Science Research*, *15*(6), 30260-30266.