Article

Not peer-reviewed version

Enhancing SDN Performance: Machine Learning Integration with the POX Controller for Dynamic Routing and Congestion Management

Qutaiba Ibrahim * and Sohaib Awad

Posted Date: 7 April 2025

doi: 10.20944/preprints202504.0423.v1

Keywords: adaptive routing; congestion management; real-time network optimization; intelligent traffic management; machine learning in SDN

Article

Enhancing SDN Performance: Machine Learning Integration with the POX Controller for Dynamic Routing and Congestion Management

Qutaiba I. Ali [1,*] and Sohaib R. Awad [2]

[1] Computer and Information Engineering Department, Ninevah University, Mosul, Iraq
[2] Computer Engineering Department, College of Engineering, University of Mosul, Mosul, Iraq
* Correspondence: sohaib.awad@uoninevah.edu.iq

Abstract: Efficient network management by SDN controllers is challenging in dynamic and high-traffic environments. Traditional controllers like POX I2_learning rely on static algorithms, adaptability, and limiting scalability. AI solutions are crucial to achieving optimal performance in complex networks. This work improves the POX I2_learning controller towards optimizing its performance under dynamic and high-traffic networks and then incorporates machine learning on the same platform. The improvements include real-time congestion metrics, adaptive timeouts, and load balancing leading to improving scalability, stability, and congestion management. Also, an XG-Boost, a machine learning model, was incorporated to classify network states and improve routing decisions in real-time. The proposed method established above achieved a marked improvement in overall system performance and network control including a stable latency of 3.52 ms, zero packet loss, and a slight improvement in throughput to 9.56 Mbps. The lightweight XG-Boost model with a compact size of 140 KB is delivered for optimal realization of real-time SDN application to offer an effective and dynamic network adaptation. This resulted in an overall accuracy of 99.67% with a balanced measure of precision, recall, and F1 score at 99%. These experimental results outperform recent SDN approaches in adaptability and performance and show that the system is reliable and able to predict a proactive decision, as well as, optimize resource usage and make the proposed framework relevant to SDN application developments.

Keywords: adaptive routing; congestion management; real-time network optimization; intelligent traffic management; machine learning in SDN

I. Introduction

An SDN controller is an application at the center of the Software-Defined Network architecture and can be called the network brain. This means that it separates the control functions from the forwarding functions and makes it easy for centralized and dynamic management [1,2]. The switches and routers are managed with the help of southbound APIs and applications that use the network can interface through northbound APIs [3,4]. The current state SDN controllers' paradigm allows for controlling many protocols through central nodes and programmability, thus forming an important tool for dealing with complex networks [5,6]. However, traditional SDN controllers such as POX I2_learning don't include dynamic algorithms and measures of traffic/load for non-stationary scenarios. Such limitations can result in the wastage of resources, traffic jams, and limited network reliability and stability [7,8].

These challenges therefore call for improving the SDN controllers with real-time decision-making and adaptation capability [9]. That is why features such as actual link load, dynamic timeouts, and load sharing enhance a controller's capacity to manage variable real conditions [10,11]. Furthermore, incorporating machine learning provides an analysis of data with the culmination of predictive analytic results and real-time dynamic control over the networks [12–14]. This paper aims

 doi:10.20944/preprints202504.0423.v1

to propose such features in the POX I2_learning controller and integrate an XG-Boost machine-learning model for better routing and congestion control in the current SDN setting.

This paper is organized as follows: The subsequent section presents the literature review and focuses on recent developments in integrating SDN and machine learning. Section three focuses on the method, describing improvements in the POX I2_learning controller and the integration of the XG-Boost model. Section four provides a view of the proposed system results and the performance enhancement part. Section five will present and contrast the current studies and describe the implications of the research. In the last part of the paper, the authors present the main findings and suggest possible further developments.

II. Literature Review

Network management strategies in Software Defined Networking (SDN) have been investigated systematically in the past with emulation on how to deal with scalability, flexibility, and utilization of networks to manage high rates of traffic fluctuation [5]. Many traditional SDN controllers such as POX I2_learning controller perform the task of centralized control and routing but suffer from static algorithms and fixed metrics that would not allow for flexibility in current network variability. Such restrictions have propelled research on improving the controller's interactions and sensitivity [3,7].

The following are some of the investigations that were launched to express real-time congestion measures for traffic supervision and management pursuance. For example, scholars have regarded different techniques to convey periodic reports from Open Flow switches such as the number of latency, packet loss, and bandwidth usage to control the controllers for optimum routing. However, applying conventional and prescriptive rules to solve congestion problems is insufficient for coping with highly dynamic situations [10,15].

To tackle these challenges, Machine Learning (ML) obtained rich research interest in the field of SDN [16,17]. In this case, XG-Boost ML is among the models that have been used to classify the states of networks, predict congestion, and the best routing options. As opposed to conventional methods, ML provides flexibility to allow controllers to understand previous conditions and apply that knowledge to current network conditions [18]. Research has revealed that incorporating ML into SDN controllers can notably boost scalability and performance, including when network traffic is heavy [19–26].

The recent work has integrated ML to SDN for load balancing, adaptive timeouts, path cost calculations, and efforts for congestion and unbalanced resource utilization. These approaches reveal that interesting but practical approaches to optimization, where enhancements to the traditional controller must be complemented with an ML-based method for decision-making, can simultaneously guarantee stability and dynamic adaptation. Based on this foundation, this study extends the newly advanced POX I2_learning controller to incorporate XG-Boost to achieve a highly reliable and scalable solution for the modern SDN context.

III. Methodology

A. Topology Overview

The topology aims to experiment with routing algorithms and congestion control using Software Defined Networking (SDN). To integrate it, it was run in Mininet over an Ubuntu-formed virtual machine under the POX I2_learning controller for network flow, and various performance metrics such as latency, packet loss, and throughput were collected. The configuration in the stage is illustrated in Figure 1. It is close to a real-time network environment created using different utilities, such as Traffic Control (tc), ping, and iperf that change packet loss, tune latency, and model certain traffic intensities. Optimizations like the adaptive timeout, load balancing, and path cost calculations increase scalability, stability, and congestion handling.

To enhance accurate tweaking of routing, the enhanced controller was fitted with an XG-Boost model that is trained with 3000 samples of latency, throughputs, and traffic load. This enables

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 7 April 2025 doi:10.20944/preprints202504.0423.v1

dynamic prediction of optimal paths thus increasing network reliability and performance. The network topology is a three-layered model, manifesting 20 hosts at the first tier and 5 switches in the second tier; thereby making it possible to implement enhanced routing and congestion control through the use of machine learning algorithms in dynamic networks.

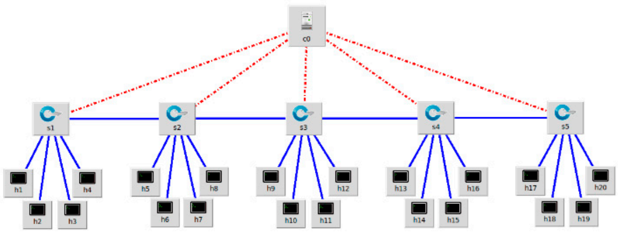

Figure 1. Main topology.

B. Enhancing the POX I2_Learning Controller

Though basic routing in SDN environments is manageable by the use of the POX I2_learning controller, there were several disadvantages as the network traffic intensity enhanced. Such limitations included congestion control, scalability, and optimality; consequently, performance degradation in dynamic and high-traffic networks [3,5]. Table 1 provides a clear description of all the problem areas that are explained in detail below alongside the congestion-aware controller's solutions.

The modification of the POX I2_learning controller into the Congestion-Aware controller solved its main issues. The extensions included real-time congestion measurement, dynamic flow control, and interaction between routing and congestion control, and it was proven that with these changes the enhanced controller provided significant steps forward in terms of scalability, stability, and efficiency. These advancements lay the groundwork for the contemporary applications of SDN together with ongoing studies on joint routing and congestion control.

C. Integrating the Enhanced I2_Learning Controller with ML

The integration of machine learning with the enhanced I2_learning controller, shown in Figure 2, addresses the need for more adaptive and intelligent routing in dynamic, high-traffic environments. While the enhanced controller introduced congestion awareness and real-path optimization it still used static algorithms and predefined metrics for operation and could not adapt to conditions in the ways of the adaptive algorithm. The proposed XG-Boost model can help the controller identify congestion categories of the network state and suggest proper paths to take in real-time. This integration brings these congestion-aware techniques from static to dynamic ones, making them more proactive rather than reactive thus providing a more suitable paradigm shift for modern network optimization.

1). Dataset Aggregation and Composition

A dataset was generated using the Mininet topology with 20 hosts and 5 switches, simulated on an Ubuntu platform. Tools like iperf, ping, and Traffic Control (tc) replicated realistic conditions by varying traffic levels, latency, and packet loss. Metrics were categorized as directly extracted (e.g., latency, throughput, packet loss, bandwidth utilization) and derived (e.g., path cost, traffic load, routing efficiency, link quality), providing structured data for machine learning integration, as summarized in Table 2.

 doi:10.20944/preprints202504.0423.v1

Table 1. Issues and the proposed solutions for the I2_learning controller.

	Issue	Description	Solution
1	Lack of congestion awareness	The original controller couldn't monitor real-time link utilization, resulting in overloaded paths, higher latency, and uneven traffic distribution.	Introduced real-time link monitoring via open flow port statistics to guide congestion-aware routing decisions and avoid bottlenecks.
2	Static flow entries and inflexible timeouts	Flow entries had fixed or indefinite timeouts, causing inefficiencies and stale routes in dynamic environments.	Implemented adaptive timeouts, assigning shorter durations to high-utilization paths and longer ones to low-utilization paths.
3	No integration of routing and congestion control	Routing decisions ignored congestion metrics like latency and bandwidth utilization, leading to inefficient resource usage during high traffic.	Introduced joint routing and congestion control using **a** link cost function***.** This function prioritized paths that minimized delays and avoided congested links.
4	Reactive routing without optimization	Routes were chosen reactively and remained static, failing to adapt to changing traffic conditions.	Added periodic re-evaluation of active paths, enabling dynamic rerouting to less congested alternatives.
5	Limited scalability in high-density networks	The lack of congestion awareness limited scalability, resulting in traffic imbalances and reduced performance in larger networks.	Introduced dynamic load balancing**,** distributing traffic evenly across paths and improving adaptability in larger topologies.
6	High latency and poor bandwidth utilization	High-latency paths were often selected, leaving bandwidth underutilized and throughput inconsistent.	Optimized routing to prioritize lower-latency paths and avoid congested links, ensuring efficient bandwidth usage.

* Link cost = (latency weight × latency) + (bandwidth utilization weight × bandwidth utilization).

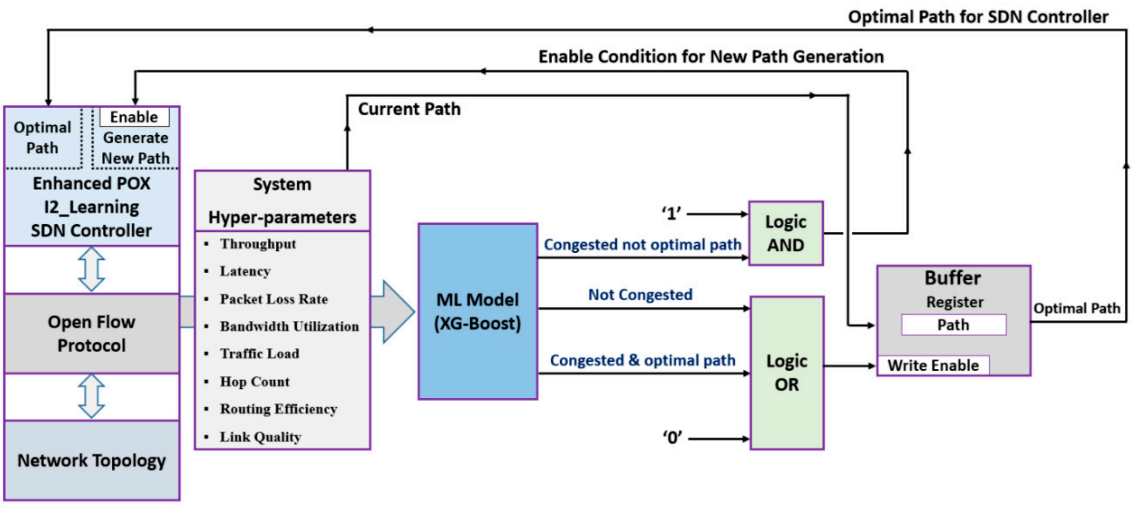

Figure 2. Proposed Integration of ML with the enhanced I2_learning POX controller.

 doi:10.20944/preprints202504.0423.v1

Table 2. Metrics Collection for ML Integration.

Category	Metric	Description
Directly extracted metrics	Latency	Measured as round-trip time (RTT) using ping.
	Throughput	Evaluated using iperf to measure data transfer rates.
	Packet loss	Calculated from dropped packets during ping tests.
	Bandwidth utilization	Calculated as a percentage of the link's maximum capacity.
Derived metrics	Path cost	Derived from latency and bandwidth utilization metrics.
	Traffic load	Monitored as the volume of packets traversing links over time.
	Routing efficiency	Calculated as the ratio of optimal paths used to all possible paths.
	Link quality	Determined based on packet error rates along paths.

The dataset, shown in Table 3, included key metrics like throughput, latency, and packet loss, collected under diverse conditions. It comprised 3,000 samples categorized into three classes: Class 0 (Not Congested) for smooth traffic, Class 1 (Congested but Not Optimal Path) for inefficient routing, and Class 2 (Congested with Optimal Path) for congestion on the best route. These balanced classes, detailed in Table 4, ensured comprehensive coverage for training and testing the machine learning model.

Table 3. Key of collected metrics.

Metrics	Min	Max	Mean	Std Dev
Throughput (Mbps)	10.06	999.54	508.24	287.56
Latency (ms)	1.05	99.93	55.45	27.62
Packet loss %	0.00006	9.99	5.52	2.81
Bandwidth utilization %	0.06	99.96	50.02	29.03
Traffic load (packets)	105	10,000	4,990.37	2,853.28
Hop count	1	10	5.59	2.81
Routing efficiency	0.00093	0.99	0.50	0.29
Link quality	0.00016	0.99	0.49	0.28

Table 4. Distribution of samples across these classes.

Class	Samples	Percentage (%)
Not Congested (Class 0)	1,772	59.09
Congested but Not Optimal (Class 1)	798	26.61
Congested with Optimal Path (Class 2)	429	14.30

2). Model Training and Evaluation

Cross-validation was done and concerning that, the data was divided into 80% training data and 20% for testing. The training set allowed the XG-Boost model to detect patterns in the network activity and define congestion states; features were scaled for equal scale and free of bias. The testing set assesses the model on real data outside the monitoring process to check how it fared in other conditions. Configured with 100 decision trees and a maximum depth of 5, the model balanced complexity and interpretability while avoiding overfitting. Learning rate between [0.01, 0.1, 0.2] helps the model to avoid oscillation while gradient boosting further enhances the precision, and enhances decision-making for the archive concerning the congestion states of the network.

 doi:10.20944/preprints202504.0423.v1

3). Integration with the Enhanced Controller

The trained XG-Boost model was integrated into the enhanced I2_learning controller, which collected real-time network metrics for classification. Based on the results, the controller took action: The traffic of Class 0 (Not Congested) continued to use specified paths; in Class 1 (Congested but Not Optimal Path), traffic redirection occurred to other paths with less congestion; while traffic from Class 2 (Congested with Optimal Path) was balanced well. This integration allowed the proposed algorithms to respond in real-time to change conditions in order to offer the best path for routing and congestion control while also shedding light on the synergistic capabilities of multiple SDN controllers and machine learning in present modern networks.

IV. Experimental Results

A. I2_Learning vs. Enhanced I2_Learning Controller

Table 5 illustrates the enhancements made due to the enhanced I2_learning controller in this study over the existing one. This controller had less latency (0.523 ms) and average offered throughput (9.50 Mbps), but it did not scale well and has a high variability when there were many simultaneous flows. The congestion-aware and dynamically optimizing enhanced controller offered higher stability in latency (3.518ms), slightly improved throughput (9.56Mbps), no packet drops, and improved scalability. While it does increase the processing load slightly it does it uniformly providing for better overall performance in large and volatile networks while the original controller is better suited for static simple ones.

The enhanced I2_learning controller well responds to the weakness in the original controller, especially in dynamism and heavier traffic load situations, in terms of throughput amplitude, capability to accommodate more clients, and better utilization of bandwidth. However, it uses static algorithms to achieve these, and this must be increased again in highly dynamic environments. Implementing machine learning as an element of the system helps to include a data-based approach to the decision-making and on-time adjustments. The subsequent section provides an understanding of enhancing the network performance with an XG-Boost model and integrated predictive tools.

Table 5. I2_learning vs. enhanced I2_learning controller.

Metric	I2_learning	enhanced I2_learning
Min Latency (ms)	0.054	1.036
Avg Latency (ms)	0.523	3.518
Max Latency (ms)	6.300	7.624
Standard Deviation	1.314	1.046
Throughput (Mbps)	9.50	9.56
Data Transferred (MB)	11.6	11.9
Test Duration (sec)	10.260	10.424
Packet Loss (%)	0	0
Bandwidth Utilization	Moderate	High
Processing Overhead	Low	Moderate
Latency Stability	Moderate	High
Throughput Stability	Moderate	High
Network Scalability	Limited in high-density networks	High in dynamic networks

B. Machine Learning Integration

The integration of the proposed XG-Boost model with the enhanced controller offered high-level decision-making decisions of the network states and improved the routing mechanism. The following tables present the complexity of the model as well as the evaluation criteria used in the work. The setup for the XG-Boost model is presented in Table 6. The above parameters allow in-between model

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 7 April 2025 doi:10.20944/preprints202504.0423.v1

refinement to achieve the right combination of accuracy and computational efficiency. As shown in the table below, the indexes of the XG-Boost model indicate that the method has high accuracy and robustness.

Table 6. ML XG-Boost model complexity and performance.

Attribute	Value
Model Name	congestion_aware_model.pkl
Number of Trees	100
Max Tree Depth	5
Model Size in KB	140
Learnable Parameters	3,200
Accuracy %	99.67
Precision %	99
Recall %	99
F1 Score %	99.67

The confusion matrix, as shown in Figure 3, highlights the model's high accuracy across all classes, with minimal misclassification for the testing set which represents 20% (600 samples) of the total dataset. Table 7 illustrates the confusion matrix.

Table 7. Confusion matrix summary.

Actual / Predicted	Not Congested	Congested Not Optimal	Congested Optimal
Not Congested	354	0	0
Congested Not Optimal	1	159	0
Congested Optimal	1	0	85

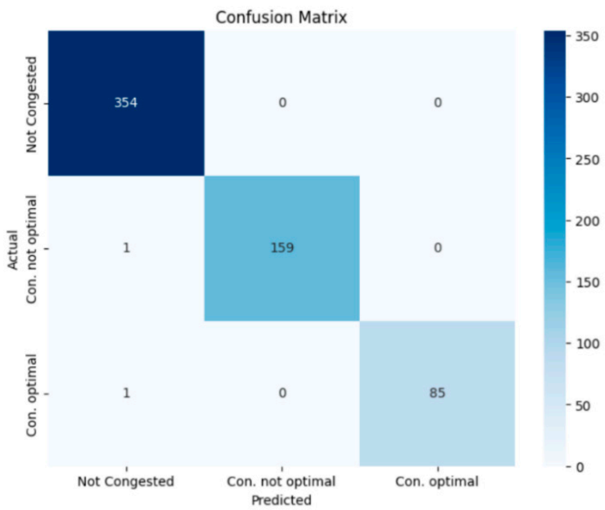

Figure 3. Testing set generated XG-Boost model CM.

The integration of the XG-Boost model with the developed l2_learning controller significantly improves real-time classification of network states and routing optimization, achieving an accuracy of 99.67% with balanced precision, recall, and F1 score of 99%. The resulting XG-Boost model is relatively fast and lightweight with 100 boost-rounds, a max depth equal to 5, 140 KB size, and around 3200 learnable parameters. Such attributes make the model less computational, appropriate for real-time applications in SDN, and therefore ideal for dynamic network management, this shows how the model can accurately indicate congestion levels and where the correct path should be to avoid crowded areas. The training on a dataset of features like throughput, latency, packet loss, and bandwidth utilization, the model effectively learns and generalizes diverse network conditions, with minimal misclassifications as shown in the results confusion matrix. Beyond reacting to current conditions, the integration enables the controller to predict and mitigate congestion through dynamic rerouting and load balancing, ensuring efficient resource utilization and network stability. This advancement positions the controller as a robust solution for modern SDN environments.

V. Discussion

The integration of machine learning into SDN has been extensively studied to address challenges such as congestion management, scalability, and dynamic traffic optimization. In this section, the proposed enhancements to the POX I2_learning controller are compared to the recent works on the following aspects: methodology, machine learning techniques, and performance outcomes. A comparison has been made as depicted below in Table 8 in terms of the strengths that are proportional to this study which utilizes an XG-Boost model for dynamic routing and congestion control in real time.

The comparison highlights the strengths of this work in integrating an XG-Boost model with the POX I2_learning controller, enabling accurate classification of network states and optimized routing decisions. Unlike [27], who describe optimization algorithms in a congestion-aware category, the proposed work shows flexibility in real-time with quantifiable latency, packet loss, and improved throughput rates. Although [28] offers a general overview of using machine learning implementation, this proposed method presents a particular application that was shown with validated results. Compared to [29], which focuses on predictive traffic management, this work integrates prediction with reactive congestion control while achieving reliability and scalability.

Integrating XG-Boost allows for lightweight functionality and high accuracy (99.67%), making this work a robust and efficient solution for modern SDN challenges. These comparisons amplify the pointers regarding the effectiveness of the proposed improvements to offset the drawbacks of the conventional SDN controllers and also promote research in machine learning for enhancing the SDN functionality.

Table 8. Comparison of proposed work with recent SDN studies.

Feature / Study	Proposed work	Prasanth and Uma [27]	Yassin and Ali [28]	Xu [29]
Controller enhancement	Enhanced POX I2_learning with XG-Boost integration	Congestion-aware framework with optimization algorithms	Review of various ML techniques for congestion control	Traffic prediction and congestion control using ML
Machine learning technique	XG-Boost	Gated recurrent neural network	Reinforcement learning (actor-critic algorithm)	Classification and prediction models
Performance metrics	Stable latency (3.518 ms), zero packet loss, throughput (9.56 Mbps)	Normalized throughput, reduced packet loss	Efficiency in congestion management	Improved service quality and user experience
Adaptability to dynamic traffic	High adaptability with real-time optimization	Focus on congestion-aware traffic management	Emphasis on learning-based traffic handling	Dynamic traffic management with predictive capabilities
Scalability	Improved scalability in high-traffic environments	Supports high-density networks	Discusses scalability for ML in SDN	Ensures scalability for dynamic and complex conditions

VI. Conclusion

The enhanced POX I2_learning controller which is integrated with machine learning effectively addressed its limitations in dynamic and high-traffic environments. Features such as real-time congestion metrics, adaptive timeouts, and load balancing significantly improve network stability, scalability, and congestion management. Acquiring an XG-Boost ML model contributed to making networks more flexible for the classification of their states and improving routing policies. The performance improvements are a stable latency of 3.52 ms, zero packet loss, and a slight boost in data rate to 9.56 Mbps compared to the basic controller. The optimal configuration of the AI model is 100 trees with a maximum depth of 5, the model is compact at 140 KB requiring 3200 trainable

 doi:10.20944/preprints202504.0423.v1

parameters, and it is well suited for real-time applications. The integrated XG-Boost ML model established accuracy at 99.67%, equally supported by precision, recall, and F1 scores of 99%, thus providing reliable, robust classification and adaptive anticipative decisions. The proposed enhancements outperform recent SDN approaches by integrating XG-Boost, enabling real-time optimization and achieving significant performance improvements in scalability, latency, and resource utilization. Future work may try to explore deep learning models, expand the datasets, and discuss the probable real-world implementation to assist improve the parameters related to adaptability and scalability.

References

1. N. Vuppalapati and T. G. Venkatesh, "Modeling & analysis of software defined networks under non-stationary conditions," Peer-to-Peer Networking and Applications, vol. 14, no. 3, pp. 1174–1189, 2021.
2. Q. I. Ali and J. K. Jalal, "Practical design of solar-powered IEEE 802.11 backhaul wireless repeater," in *Proc. 6th Int. Conf. on Multimedia, Computer Graphics and Broadcasting*, 2014.
3. F. Bannour, S. Souihi, and A. Mellouk, "Distributed SDN control: Survey, taxonomy, and challenges," IEEE Communications Surveys & Tutorials, vol. 20, no. 1, pp. 333–354, 2017.
4. Q. I. Ali, "Green communication infrastructure for vehicular ad hoc network (VANET)," Journal of Electrical Engineering, vol. 16, no. 2, p. 10, 2016.
5. P. B. Bautista, J. Comellas, and L. Urquiza-Aguiar, "Evaluating Scalability, Resiliency, and Load Balancing in Software-Defined Networking," Engineering Proceedings, vol. 47, no. 1, p. 16, 2023.
6. H. M. Mohammed and Q. I. Ali, "E-proctoring systems: A review on designing techniques, features, and abilities against threats and attacks," Quantum Journal of Engineering, Science, and Technology, vol. 3, no. 2, pp. 14-30, 2022.
7. A. Rastogi and A. Bais, "Comparative analysis of software defined networking (SDN) controllers—In terms of traffic handling capabilities," in 2016 19th International Multi-Topic Conference (INMIC), 2016, pp. 1–6.
8. Q. I. Ali, "Security Issues of Solar Energy Harvesting Road Side Unit (RSU)," Iraqi Journal for Electrical & Electronic Engineering, vol. 11, no. 1, 2015.
9. A. Sharma, V. Balasubramanian, and J. Kamruzzaman, "A novel dynamic software-defined networking approach to neutralize traffic burst," Computers, vol. 12, no. 7, p. 131, 2023.
10. M. He, et al., "How flexible is dynamic SDN control plane?," in 2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), 2017, pp. 978–983.
11. Q. I. Ali, "Design and implementation of an embedded intrusion detection system for wireless applications," IET Information Security, vol. 6, no. 3, pp. 171-182, 2012.
12. A. Sharma and H. Babbar, "Machine Learning-based Threat Detection for DDoS Prevention in SDN-Controlled IoT Networks," in 2024 5th International Conference for Emerging Technology (INCET), 2024, pp. 1–7.
13. S. Q. I. Ali, "Realization of a robust fog-based green VANET infrastructure," *IEEE Systems Journal*, vol. 17, no. 2, pp. 2465-2476, 2022.
14. R. H. Serag, et al., "Machine-Learning-Based Traffic Classification in Software-Defined Networks," Electronics, vol. 13, no. 6, p. 1108, 2024.
15. C. N. Shivayogimath and N. V. Uma Reddy, "Modification of l2 learning switch code for firewall functionality in pox controller," in Silicon Photonics & High Performance Computing: Proceedings of CSI 2015, Springer Singapore, 2018, pp. 367–372.
16. R. Amin, et al., "A survey on machine learning techniques for routing optimization in SDN," IEEE Access, vol. 9, pp. 104582–104611, 2021.
17. Z. Liu, "ML-based SDN performance prediction," Applied and Computational Engineering, vol. 29, pp. 57–67, 2023.
18. N. Bilal, S. Askar, and K. Muheden, "Challenges and Outcomes of Combining Machine Learning with Software-Defined Networking for Network Security and Management Purpose: A Review," Indonesian Journal of Computer Science, vol. 13, no. 2, pp. 73–85, 2024.

 doi:10.20944/preprints202504.0423.v1

19. C. Gonzalez and S. M. Charfadine, "SDN Controllers and ML-Based Anomaly Detection in Embedded Systems: A Comparative Analysis," in 2023 10th International Conference on Wireless Networks and Mobile Communications (WINCOM), 2023, pp. 56–62.
20. Q. Ibrahim, "Enhanced power management scheme for embedded road side units," *IET Computers & Digital Techniques*, vol. 10, no. 4, pp. 174-185, 2016.
21. Q. I. Ali, "Performance evaluation of WLAN internet sharing using DCF & PCF modes," *Int. Arab. J. e Technol.*, vol. 1, no. 1, pp. 38-45, 2009.
22. Q. I. Ali, "Design, implementation & optimization of an energy harvesting system for VANETs' road side units (RSU)," *IET Intelligent Transport Systems*, vol. 8, no. 3, pp. 298-307, 2014.
23. Q. I. Ali, "An efficient simulation methodology of networked industrial devices," in *Proc. 5th Int. Multi-Conference on Systems, Signals and Devices*, 2008, pp. 1-6.
24. Q. I. Ali, "Securing solar energy-harvesting road-side unit using an embedded cooperative-hybrid intrusion detection system," *IET Information Security*, vol. 10, no. 6, pp. 386-402, 2016.
25. Q. Ibrahim, "Design & Implementation of High-Speed Network Devices Using SRL16 Reconfigurable Content Addressable Memory (RCAM)," *Int. Arab. J. e Technol.*, vol. 2, no. 2, pp. 72-81, 2011.
26. M. H. Alhabib and Q. I. Ali, "Internet of autonomous vehicles communication infrastructure: a short review," *Diagnostyka*, vol. 24, 2023.
27. L. L. Prasanth and E. Uma, "A computationally intelligent framework for traffic engineering and congestion management in software-defined network (SDN)," EURASIP Journal on Wireless Communications and Networking, vol. 2024, no. 1, p. 63, 2024.
28. T. Yassin and O. Ali, "Using Machine Learning to Control Congestion in SDN: A Review," in International Conference on Emerging Trends and Applications in Artificial Intelligence, Cham, Switzerland: Springer Nature, 2023, pp. 45–52.
29. Y. Xu, "Machine Learning Based Traffic Prediction and Congestion Control Algorithms in Software Defined Networks," in 2024 International Conference on Interactive Intelligent Systems and Techniques (IIST), IEEE, 2024, pp. 12–18.