Traffic Engineering in Software Defined Networking(SDN) with emphasis on QOS

Karamveer Singh Ghai
Computer Science
University of the Fraser Valley
Abbotsford, Canada
Karamveerghai12@gmail.com

***Abstract*— Software-Defined Networking (SDN) has transformed network management by decoupling the control and data planes, enabling centralized and programmable network control. However, despite its advantages, traffic engineering in SDN encounters significant challenges, including congestion, Quality of Service (QoS), and scalability. This paper reviews four key research contributions from reputable sources that address these challenges, focusing on traffic monitoring, multi-class traffic management, QoS/QoE optimization, and AI-driven frameworks for congestion management. The findings reveal SDN's potential in achieving efficient traffic engineering while identifying unresolved challenges and proposing directions for future research.**

Keywords—Software-Defined Networking (SDN), traffic engineering, congestion management, QoS, QoE, dynamic routing, centralized control, AI-driven frameworks, service-oriented networks, hybrid network management.

I. Introduction

This Software-Defined Networking (SDN) represents a paradigm shift in network architecture, decoupling the control plane from the data plane to enable centralized and programmable network management. This innovation has transformed traditional static and hardware-centric networks into dynamic, scalable, and service-oriented infrastructures. Despite its transformative potential, SDN-based traffic engineering faces persistent challenges, including congestion management, Quality of Service (QoS) assurance, and scalability, particularly in complex and heterogeneous environments .The primary objective of traffic engineering in SDN is to optimize network resource utilization, improve traffic flow efficiency, and ensure user satisfaction by dynamically managing traffic loads. This paper examines these challenges through four significant research contributions, which propose solutions ranging from advanced traffic measurement techniques and adaptive routing algorithms to QoS/QoE optimization and AI-driven frameworks for predictive congestion control. This comprehensive exploration highlights the potential of SDN in addressing modern network demands while outlining unresolved challenges and suggesting future directions for research and innovation.

II. Traffic Management in Software Defined Networking: Measurement and Management

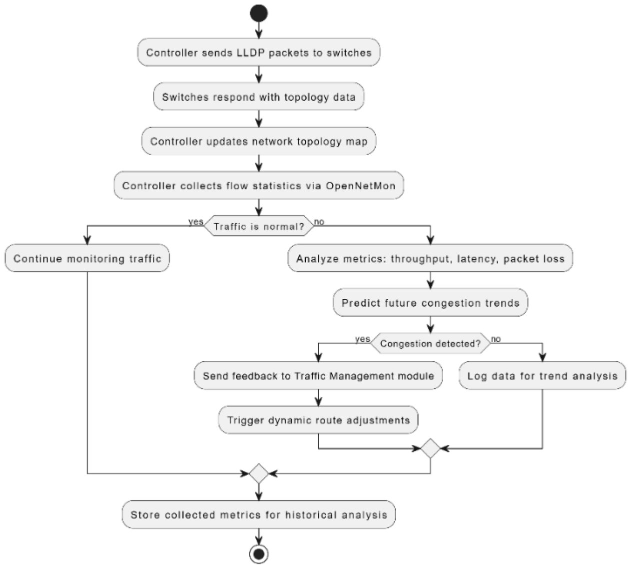
A. *Overview*

Software-Defined Networking (SDN) introduces centralized control and programmability, enabling significant advancements in traffic engineering compared to traditional networks. By decoupling the control and data planes, SDN provides a global network view, allowing real-time decision-making for optimized traffic flows. However, achieving effective traffic management in SDN requires addressing challenges such as congestion, load balancing, and Quality of Service (QoS) assurance. Shu *et al.* (2016) proposed an integrated framework combining traffic measurement and management to address these challenges and improve overall network performance [1].

B. *Traffic Measurement*

The Traffic measurement is critical for collecting real-time data about the network's state to enable informed decision-making. The process begins with topology discovery, where the Link Layer Discovery Protocol (LLDP) gathers information about network devices, their connections, and available bandwidth. This step provides the SDN controller with a complete global view of the network's physical structure. Performance metrics such as bandwidth utilization, latency, jitter, and packet loss are monitored continuously to ensure the network operates within acceptable thresholds.

The framework employs tools like OpenNetMon and FlowSense to collect these metrics efficiently. OpenNetMon measures throughput and packet loss, while FlowSense captures flow behavior without injecting additional probing traffic, enhancing system performance. These metrics inform dynamic routing and QoS adjustments, ensuring proactive traffic management [1].



Traffic Measurement Workflow: *Illustration of the SDN traffic measurement process, including topology discovery and performance metric collection.*

C. *Traffic Management*

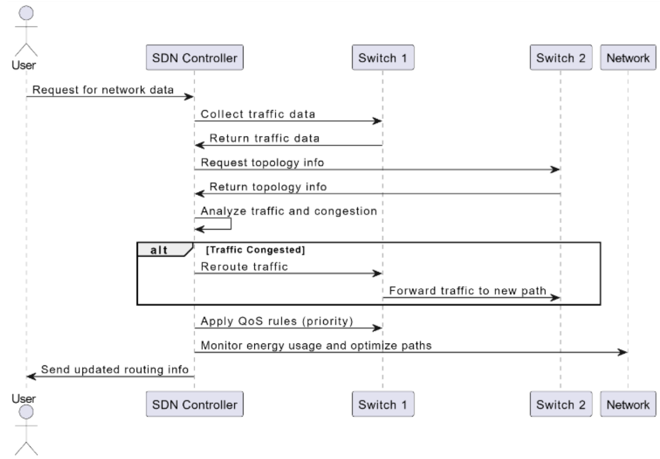
In the management phase, the collected data enables SDN to dynamically optimize traffic flows and maintain high performance.

1. **LoadBalancing**:
   Traffic is distributed evenly across available paths to prevent congestion. The SDN controller, using OpenFlow protocols, adjusts forwarding rules on switches in real-time to redirect traffic from overloaded paths to underutilized ones. For instance, the controller can split traffic across multiple paths based on current conditions, ensuring balanced utilization [1].

2. **QoS**:
   High-priority traffic, such as video calls or real-time applications, is prioritized over lower-priority flows. Differentiated Services Code Point (DSCP) markings in IP headers guide the network in routing traffic with minimal delay and adequate bandwidth allocation. This ensures optimal performance for latency-sensitive applications [1].

3. **CongestionControl**:
   By analyzing real-time metrics such as queue lengths and packet loss rates, the SDN controller can detect and mitigate congestion before it impacts network performance. When congestion is identified, the system dynamically reroutes traffic to less congested paths, ensuring seamless performance for real-time applications [1].

4. **EnergyEfficiency**:
   The framework incorporates energy-saving measures during low-demand periods. The SDN controller consolidates traffic onto fewer paths, allowing unused links to be deactivated, thereby reducing power consumption. This strategy ensures efficient energy utilization without compromising network performance [1].



Traffic Management Workflow: *Visualization of Traffic management.*

D. BENEFITS AND CHALLENGES

The framework offers several benefits, including real-time traffic optimization, enhanced QoS, and energy savings. However, it also presents challenges, such as reliance on OpenFlow protocols, which may limit scalability in hybrid environments where traditional and SDN-based networks coexist. Energy efficiency measures require careful tuning to avoid compromising performance during fluctuating traffic demand. Despite these challenges, the framework provides a robust foundation for traffic engineering in SDN-centric environments, addressing key challenges in modern network management [1].

III. TRAFFIC ENGINEERING AND QOS/QOE SUPPORTING TECHNIQUES FOR EMERGING SERVICE-ORIENTED SOFTWARE-DEFINED NETWORKS (SOSDN)

The integration of Software-Defined Networking (SDN) with Service-Oriented Architecture (SOA) has given rise to Service-Oriented Software-Defined Networks (SOSDN). Traditional networks struggled to ensure Quality of Service (QoS) due to decentralized control, with static engineering methods unable to adapt to real-time changes, leading to inefficiencies. Modern networks often fail to meet user satisfaction while enforcing QoS policies. The focus of SOSDN is to enhance Quality of Service (QoS) and Quality of Experience (QoE) through dynamic traffic engineering techniques that prioritize user satisfaction and optimize resource allocation.

A. *Adaptive Service Prioritization*

The goal of adaptive service prioritization is to dynamically adjust service priorities in real-time, ensuring that critical services like video conferencing are prioritized during congestion to maintain consistent performance and user satisfaction. Differentiated Services Code Point (DSCP) marks packets, categorizing them into high-priority real-time traffic and low-priority bulk data transfers. During congestion, high-priority traffic is given more bandwidth, while lower-priority traffic is delayed or rerouted. This ensures that critical services, such as video streaming, maintain optimal performance. For example, in a network where video streaming and FTP services share bandwidth, the controller allocates more bandwidth to the video stream and delays file transfer to maintain QoS [1], [2],[3].

B. Multi- Criteria Routing Optimization

In SOSDN, multi-criteria routing optimization selects the best path for traffic by evaluating various QoS metrics, such as packet loss, jitter, delay, and available bandwidth, rather than relying on a single metric like the shortest path in traditional networks. The method uses a cost function that combines these metrics into a single integral function to evaluate all potential paths. Optimization considers the importance of each criterion, ensuring the optimal path is chosen for each data flow [3].

Mathematical Model for Routing Optimization:

$$F(x) = \sum_{i=1}^{n} \alpha_i \cdot f_i(x)$$

The optimization is done using an additive criterion, where multiple QoS metrics are combined into a single function. For example, considering two paths:

- **Path 1**: Latency = 10ms, Bandwidth = 50Mbps, Packet Loss = 0.5%
- **Path 2**: Latency = 15ms, Bandwidth = 100Mbps, Packet Loss = 0.2%

For Path 1: Fpath1 = 1· (10) +1· (1/50) +1· (0.5) =10+0.02+0.5=10.52

For Path 2: Fpath 2=1· (15) +1· (1/100) +1· (0.2) =15+0.01+0.2=15.21

Path 1 is selected as it has the lower cost function, ensuring optimal routing.

C. Optimal Server Selection Using Multi-Criteria Optimization

Criterion, i	Weighting factor	Measured criterion values for different server options		
		Server 1	Server 2	Server 3
CPU loading, %	0.3	60	70	50
RAM loading, %	0.2	50	30	60
Average request processing time, ms	0.2	100	150	400
Probability of losing requests, %	0.2	0.5	0.8	0.3
Network interface loading, %	0.1	80	70	60

Table 1: Server Data: *Initial Data Determining the Optimal Service Server Based on Monitoring the state of their operation [3].*

Multi-criteria optimization is used to dynamically choose the best server for data flow based on metrics such as CPU load, RAM load, request processing time, probability of request loss, and network load. If a server exceeds its capacity, the traffic is rerouted to a more suitable server. The same mathematical approach is applied to determine the most efficient server based on these criteria, ensuring better resource utilization and preventing server overloads [3].

Example Calculation for Server Selection using Table 1:

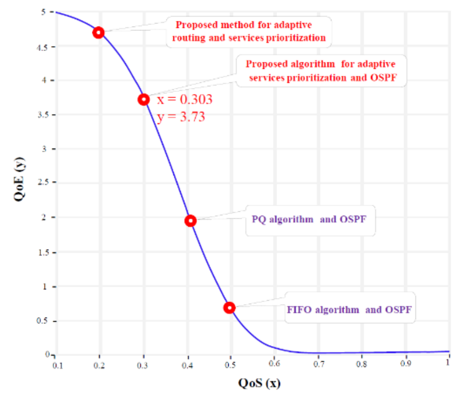
Server 1: F(x)=0.3· (50/60) +0.2· (30/50) +0.2· (100/100) +0.2· (0.3/0.3) +0.1· (60/80) =0.845

Server 2: F(x)=0.3· (50/70) +0.2· (30/30) +0.2· (100/150) +0.2· (0.3/0.8) +0.1· (60/70) =0.707

Server 3: F(x)=0.3· (50/50) +0.2· (30/60) +0.2· (100/400) +0.2· (0.3/0.3) +0.1· (60/60) =0.75

Server 1 is the optimal choice because it has the highest score [3].

D. Simulation Results



Qos/Qoe Graph: *Graph Dynamics of QoE growth after proposed solutions [3].*

The above graph shows that the QoE scores significantly improve when the adaptive routing and traffic prioritization methods are applied. For example, when IPTV users were given priority due to their specific QoE requirements, the network could effectively reallocate resources, reducing packet loss and improving user satisfaction, raising the QoE score from 1.9 to 4.7.

E. Benefits and Challenges

SOSDN provides a user-centric approach that integrates adaptive prioritization and multi-criteria routing, dynamically adjusting network resources to meet both QoS and QoE requirements. However, the complexity of managing multiple criteria and the need for real-time adjustments may introduce computational challenges. The integration of SDN with service-oriented networks requires sophisticated coordination between network and server domains, which can be resource-intensive

IV. A Computationally Intelligent Framework for Traffic Engineering and Congestion Management in Software-Defined Networks (SDN)

This paper presents an advanced framework for traffic engineering (TE) and congestion management in Software-Defined Networks (SDN). The study introduces two key components to improve network performance under dynamic traffic conditions: the Multiplicative Gated Recurrent Neural Network (mGRNN) for traffic prediction and Congestion-

Aware Hunter Prey Optimization (CA-HPO) for dynamic traffic routing. Both methods aim to optimize Quality of Service (QoS) by reducing congestion, balancing network load, and enhancing network adaptability.

A. *mGRNN for Traffic Prediction*

The paper introduces a novel mGRNN model that enhances traffic prediction by combining the strengths of the Multiplicative Recurrent Neural Network (mRNN) and Gated Recurrent Unit (GRU) architectures. The mGRNN is designed to handle long-term dependencies in traffic data, including periodic fluctuations and sudden changes. This hybrid model improves the network's ability to predict future traffic patterns, which is crucial for effective congestion management in SDNs [4].

B. *CA-HPO for Congestion-Aware Routing*

The paper presents CA-HPO, a meta-heuristic optimization algorithm inspired by the behaviors of predatory animals. This algorithm dynamically determines the optimal routing paths by considering real-time congestion and network conditions. The CA-HPO model is designed to minimize congestion and balance network traffic efficiently, ensuring high QoS and reduced network delays [4].

C. *Network Traffic Management in SDNs*

The framework addresses the challenges posed by the growing complexity of network traffic, especially in SDN environments. By using centralized control and dynamic resource management, the framework optimizes network performance and enhances traffic flow efficiency. The combination of mGRNN for prediction and CA-HPO for routing ensures the efficient handling of congestion and network load [4].

D. *Experimental Setup and Results*

Models	MAE	MSE	RMSE	RRMSE
Decision tree	0.00792	0.039437	0.07708	0.065079
KNN	0.00453	0.030577	0.03946	0.0852192
SVM	0.00482	0.061588	0.058821	0.072883
ANN	0.08065	0.01002	0.12840	0.05929
RNN	0.08766	0.01468	0.12623	0.06296
Proposed MG-RNN model	0.0026	0.00905	0.029	0.0127

Table 2: Model Performance test in MATLAB: *Performance of proposed and existing methods [4].*

The experiments were conducted in MATLAB (R2021a), and the results were compared with existing models such as Genetic Algorithms (GA) and Particle Swarm Optimization (PSO). Performance metrics like MAE, MSE, RMSE, and RRMSE were used for evaluation. The Table 2 demonstrates that the mGRNN algorithms outperform traditional models, providing better predictions and more effective congestion management in SDNs [4].

E. *Future Work*

Future research will focus on further scaling the models, applying them to larger SDN networks, and extending the framework to handle more complex scenarios, such as multi-path routing, multiple traffic classes, and differentiated QoS levels [2].

V. A Computationally Intelligent Framework for Traffic Engineering and Congestion Management in Software-Defined Networks (SDN)

This paper presents a novel approach to Traffic Engineering (TE) in wide-area networks (WANs) called Asynchronous Multi-Class Traffic Management (AMTM). The primary objective of AMTM is to enhance network utility, reduce end-to-end (E2E) delay, and improve scalability while managing diverse traffic flows in a WAN. The authors compare AMTM with other approaches, such as centralized and hierarchical schemes based on resource preallocation[2].

A. *Overview of the AMTM Approach*

1. **Asynchronous TE Paradigm:** AMTM employs an asynchronous model where local traffic control is executed at the network edge by service brokers. Link prices are updated periodically by a central TE server to avoid conflicts between service brokers. This asynchronous approach is particularly useful for large-scale networks, where real-time synchronization is impractical.

2. **Pricing Strategy Based on Virtual Queues**: AMTM uses a dynamic pricing strategy based on virtual queues at intermediate network nodes. Link prices reflect network states, guiding the flow of traffic to optimize overall performance. The pricing strategy helps manage congestion and improves the efficient use of available bandwidth[2].

B. *Key Simulation Results*

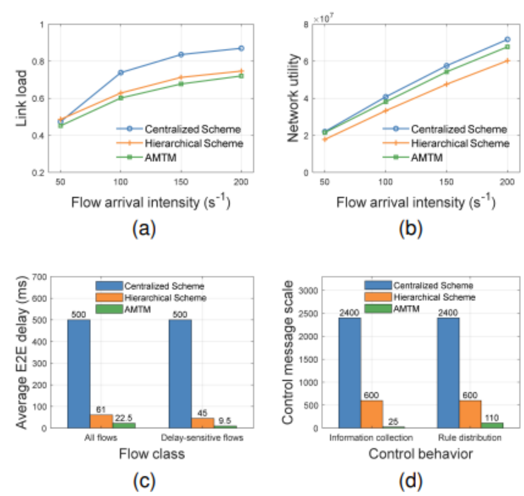


Chart 3 Simulation results: *(a) Link load. (b) Network utility. (c) Average E2E delay, excluding propagation delay. (d) Control message scale[2].*

- **Link Load:** The paper compares link load across different traffic management schemes. Link load, representing average bandwidth utilization, increases with higher traffic arrival intensity but slows as link saturation occurs under heavy traffic. As shown in chart 3(a) ,AMTM effectively balances link utilization without overwhelming the network[2].

- **Network Utility:** Network utility is an aggregate performance measure considering both traffic demands and available resources. Chart 3(B) shows AMTM achieves 12–20% higher utility than the hierarchical scheme, and it performs very close to the maximum achievable utility in a centralized scheme, with only a 2–7% performance gap[2].

- **E2E Delay:** AMTM consistently achieves the lowest E2E delay, with delays for delay-sensitive flows kept under 10 ms as mentioned in chart3(c). This demonstrates AMTM's effectiveness in reducing latency, which is crucial for real-time applications like multimedia streaming or VoIP[2].

- **Scalability:** The paper assesses scalability by evaluating the number of control messages generated during network operation. The centralized scheme generates control messages proportional to the number of flows, leading to bottlenecks in large networks. The hierarchical scheme scales better, but AMTM exhibits the lowest control overhead as illustrated in chart 3(d), scaling efficiently even as the number of flows increases. This makes AMTM ideal for large-scale networks with numerous flows[2].

C. *Benefits and Challenges*

AMTM offers several benefits, including enhanced network utility by optimizing resource allocation, reduced end-to-end (E2E) delay, and improved scalability for large networks. Its asynchronous control reduces synchronization overhead and ensures efficient traffic management, especially for delay-sensitive applications. However, challenges arise in coordinating local service brokers with the central server and potential inefficiencies in the dynamic pricing strategy during rapid traffic changes. The decentralized approach can complicate optimization, and the method requires significant computational resources, especially for large-scale networks with diverse traffic demands.

VI. INTERPRETATION OF ALL 4 PAPERS

Paper Reference	Proposed Solutions	Strengths	Weaknesses	Applicability
Shu et al. (2016)	Traffic Measurement and Traffic Management	- Real-time traffic data collection - Dynamic route adjustments - Centralized SDN control	- Heavy dependency on OpenFlow - Limited scope for energy efficiency in mixed networks	Effective in SDN-centric environments
Wu et al. (2024)	AMTM for Traffic Management	- Asynchronous model reduces latency - Decentralized control for scalability	- Assumes availability of ideal edge resources - Complexity in implementation	Ideal for WANs and large-scale network setups
Beshley et al. (2024)	SOSDN with Adaptive Prioritization and Multi-Criteria Routing	Dynamic QoS prioritization - Integration with service-oriented networks	- High computational cost - Requires precise QoE parameter tuning	Service-centric networks needing high QoE
Prasanth & Uma (2024)	AI-driven mGRNN and CA-HPO	- Accurate traffic prediction - Congestion-aware routing using bio-inspired algorithms	- High computation power required - Limited multi-path adaptability	Best in highly dynamic traffic scenarios

Table 4 Comparison of papers: *Comparing all 4 papers in accordance with strengths, weaknesses and applicability.*

A. *Optimal Solutions:*

- Shu et al. (2016): Highly effective in SDN-heavy environments, ensuring efficiency and adaptability through centralized control and dynamic traffic management.
- Wu et al. (2024): The AMTM model excels in wide-area networks with asynchronous management, ideal for real-time applications and large-scale scalability.

B. *Partially Optimal:*

- Beshley et al. (2024): The SOSDN approach offers dynamic QoS prioritization but is complex and computationally demanding, suited for resource-rich, service-centric architectures.
- Prasanth & Uma (2024): AI-driven methods are effective in congestion management but require significant computational resources, limiting their practicality in certain environments.

VII. CONCLUSION

This paper explores various innovative techniques in traffic engineering for Software-Defined Networks (SDNs), focusing on congestion management, Quality of Service (QoS), and Quality of Experience (QoE). Four key papers were analyzed: Shu et al. (2016) emphasized centralized traffic measurement and dynamic management in SDN, while Wu et al. (2024) introduced the AMTM model for scalable, asynchronous traffic management. Prasanth & Uma (2024) combined AI methods with congestion-aware routing. While these approaches present promising solutions, each faces challenges in scalability, computational resources, and integration complexity. Among the four papers, Beshley et al. (2024) focuses on integrating SDN with Service-Oriented Architecture (SOA) for dynamic QoS and QoE support. This paper employs adaptive prioritization, multi-criteria routing, and real-time adjustments to ensure high QoS. Their approach dynamically adjusts traffic flows, offering strong QoS guarantees, especially in service-centric environments. Centralized SDN models work best in controlled environments, while more advanced, decentralized models like AMTM and AI-driven methods show great promise for real-time traffic management at larger scales. However, the computational load and complexity of dynamic solutions need further refinement for practical deployment in diverse network settings.

VIII. FUTURE DIRECTIONS

Future research should explore the development of hybrid SDN architectures that combine centralized and decentralized management to strike a balance between scalability and control. Furthermore, the integration of machine learning and artificial intelligence for predictive traffic routing could significantly enhance the adaptability of networks to dynamic traffic patterns. Multi-criteria optimization techniques for routing, focusing on real-time QoS/QoE metrics, should be further refined to reduce computational overhead and improve their effectiveness in large-scale, heterogeneous networks. Finally, the application of these advanced models to IoT networks and edge computing scenarios presents a promising avenue for future exploration, addressing both scalability and resource limitations in these emerging environments.

REFERENCES

[1] Z. Shu, J. Wan, J. Lin, S. Wang, D. Li, S. Rho, and C. Yang, “Traffic engineering in software-defined networking: Measurement and management,” IEEE Access, vol. 4, pp. 3246–3256, 2016. doi: 10.1109/ACCESS.2016.2582748.

[2] H. Wu, J. Yan, and L. Kuang, “Asynchronous multi-class traffic management in wide area networks,” IEEE Transactions on Network and Service Management, vol. 21, no. 2, pp. 1750–1763, 2024. doi: 10.1109/TNSM.2024.3354793.

[3] M. Beshley, N. Kryvinska, H. Beshley, O. Panchenko, and M. Medvetskyi, “Traffic engineering and QoS/QoE supporting techniques for emerging service-oriented software-defined network,” Journal of Communications and Networks, vol. 26, no. 1, pp. 99–114, 2024. doi: 10.23919/JCN.2023.000065.

[4] L. L. Prasanth and E. Uma, “A computationally intelligent framework for traffic engineering and congestion management in software-defined network (SDN),” Journal of Wireless Communications and Networking, vol. 63, 2024. doi: 10.1186/s13638-024-02392-2.