Information, Computing and Intelligent Systems, 2025, No. 7, 4 – 22

UDC 004.725.5 https://doi.org/10.20535/2786-8729.7.2025.334049

INTELLIGENT TRAFFIC MANAGEMENT METHOD IN SOFTWARE-DEFINED NETWORKS BASED ON BEHAVIOURAL CLASSIFICATION AND ADAPTIVE PRIORITY SERVICE

Dmytro Oboznyi*
https://orcid.org/0000-0003-0108-4587

Yurii Kulakov
https://orcid.org/0000-0002-8981-5649

National Technical University of Ukraine
“Igor Sikorsky Kyiv Polytechnic Institute”, Kyiv, Ukraine

*Corresponding author: dobozniy@gmail.com

The growing complexity of modern enterprise network environments demands sophisticated traffic management solutions that can provide quality of service (QoS) guarantees for encrypted and heterogeneous flows. Existing traffic management approaches face significant challenges when dealing with encrypted protocols and diverse application requirements, resulting in performance degradation for critical services and inefficient resource utilization. This paper addresses the problem of intelligent traffic management in software-defined networks through behavioral classification and adaptive priority service mechanisms.

The study examines the development and implementation of an integrated traffic management method that combines behavioral deep packet inspection, class-based queuing, and weighted random early detection algorithms. The research investigates how behavioral flow characteristics remain observable in encrypted traffic environments and how these patterns can be leveraged for effective QoS provisioning. The proposed method utilizes packet timing patterns, connection behaviors, and flow statistics to classify traffic without relying on payload inspection or predefined port assignments.

Experimental validation through discrete-event simulation demonstrates significant performance improvements compared to traditional first-in-first-out mechanisms. The behavioral classification component achieves over 95% classification accuracy. The experimental results demonstrate up to 97.5% improvement in latency performance and 0% packet loss for high-priority traffic.

Integrating behavioral traffic recognition with adaptive queue management within a programmable network framework provides an effective and innovative approach to maintaining stable service quality in encrypted, multi-service environments. The proposed method is compatible with existing software-defined network controllers and can be deployed without modification of application protocols or infrastructure components.

Keywords: software-defined networks, intelligent traffic management, behavioral classification, adaptive priority service, deep packet inspection, class-based queuing, weighted random early detection, quality of service.

1. Introduction

The digital transformation of modern enterprises has fundamentally altered network traffic patterns and quality requirements, creating unprecedented challenges for network infrastructure management. Organizations increasingly depend on real-time communication applications including voice over IP (VoIP) systems that require sub-100 ms latency for acceptable call quality, high-definition multiparty video conferencing platforms supporting 4K resolution streams with adaptive bitrate algorithms, and interactive collaborative tools such as remote desktop sessions with screen sharing capabilities, cloud-based integrated development environments with real-time code synchronization, and distributed database consoles requiring immediate query response times. These mission-critical workloads exhibit stringent Quality of Service (QoS) requirements, being particularly sensitive to network delay variations that can cause audio dropouts or video artifacts, jitter fluctuations that result in choppy media playback, and packet loss events exceeding 0.1.

The proliferation of Internet of Things (IoT) devices, mobile workforce solutions, and cloud migration initiatives has exponentially increased the heterogeneity of network traffic patterns. Modern enterprise networks must simultaneously support traditional bulk data transfers, real-time sensor telemetry from thousands of IoT endpoints, video surveillance streams requiring guaranteed bandwidth allocation, and interactive applications with microsecond-level timing requirements. This traffic diversity creates complex interdependencies where the performance of critical applications can be unexpectedly impacted by seemingly unrelated background processes such as software updates, file synchronization, or automated backup operations.

Simultaneously, the contemporary network landscape presents unprecedented challenges for traditional traffic management approaches. The widespread adoption of end-to-end encryption protocols, including Transport Layer Security (TLS) 1.3 and Quick UDP Internet Connections (QUIC), has rendered conventional Deep Packet Inspection (DPI) techniques largely ineffective for application identification. The proliferation of Content Delivery Networks (CDNs) and edge computing infrastructures has further complicated traffic classification, as the same application may utilize multiple domains, ports, and transport protocols depending on content type, geographic location, and load balancing decisions. Consequently, classical traffic identification methods based on well-known port numbers, domain name patterns, or payload signatures have become increasingly unreliable in modern encrypted network environments.

Traditional QoS mechanisms, including First-in-First-out (FIFO) queuing, weighted fair queuing (WFQ), and conventional Random Early Detection (RED), face significant limitations when deployed in heterogeneous, encrypted traffic scenarios. These approaches often underperform in mixed traffic environments where queue buffers are finite and buffer pressure fluctuates dynamically due to varying application demands and network conditions. The fundamental challenge lies in the inability of these traditional mechanisms to adapt intelligently to the behavioral characteristics of encrypted flows while maintaining fairness and preventing starvation of lower-priority traffic classes.

Software-Defined Networking (SDN) architectures offer a promising paradigm shift by providing centralized policy management and comprehensive network telemetry capabilities combined with distributed enforcement mechanisms at the data plane level. This separation of control and data planes enables sophisticated, class-aware traffic control policies without requiring modifications to existing application protocols or end-user software. The programmable nature of SDN switches, combined with standardized southbound interfaces such as OpenFlow, creates opportunities for implementing advanced traffic management algorithms that can adapt dynamically to changing network conditions and traffic patterns.

Considering the aforementioned aspects, the increasing dominance of encrypted, heterogeneous traffic in enterprise environments requires a fundamentally new approach to maintaining QoS. Existing queuing and congestion avoidance techniques cannot adequately handle dynamically shifting traffic loads and encrypted flow characteristics without compromising either fairness or real-time application performance. Therefore, developing an integrated SDN-based framework that combines behavioral DPI, Class-Based Queuing (CBQ), and Weighted Random Early Detection (WRED) becomes essential. Such a framework must enable simulation-driven evaluation of network performance under varying load conditions, buffer capacities, and service priorities, providing measurable metrics such as delay, packet loss, and throughput for different traffic classes. Modeling the system through a Discrete-Event Simulation (DES) environment, supported by analytical validation using queueing theory (e.g., M/M/1/K), allows researchers to capture the dynamic, event-driven nature of modern networks. The resulting prototype model serves as a foundation for designing adaptive, controller-agnostic QoS mechanisms that ensure predictable service delivery, efficient resource utilisation, and sustained performance in encrypted, software-defined enterprise infrastructures.

2. Literature review and problem statement

The evolution of network traffic management in software-defined environments has been marked by significant advances in programmable data plane technologies and sophisticated control plane algorithms. However, existing approaches exhibit fundamental limitations when addressing the complex challenges of QoS provisioning in modern encrypted, heterogeneous network environments.

Recent developments in programmable data plane technologies, particularly those based on the Programming Protocol-Independent Packet Processors (P4) language, have demonstrated substantial improvements in network path utilization and load distribution capabilities. The research [1] present comprehensive load balancing mechanisms that leverage P4's flexibility to implement sophisticated forwarding policies. While these approaches excel at optimizing network-wide resource utilization, they fundamentally lack mechanisms for guaranteeing class-specific QoS at the individual queue level. The P4-based solutions focus primarily on path selection and traffic distribution rather than addressing the critical challenges of buffer management and service differentiation within individual network devices.

The limitations of current programmable approaches become particularly evident when considering the temporal dynamics of QoS requirements. Modern applications exhibit highly variable traffic patterns with burst characteristics that can overwhelm traditional load balancing mechanisms. Furthermore, the stateless nature of most P4 implementations prevents the retention of flow-level behavioral information necessary for intelligent traffic classification in encrypted environments.

The integration of software-defined networking principles with IoT and edge computing environments has spawned innovative approaches to distributed traffic management. The study [2] propose clustering-based mechanisms that significantly reduce signaling overhead in Software-Defined Wireless Sensor Networks (SDWSN). These approaches demonstrate the feasibility of hierarchical control structures that can scale to large numbers of heterogeneous devices while maintaining centralized policy enforcement.

However, the existing SDWSN and IoT-focused solutions exhibit a critical limitation in their treatment of traffic heterogeneity. While clustering mechanisms effectively reduce control plane overhead, the per-class behavior of different traffic types under congestion conditions remains largely unexplored. The absence of sophisticated queue management mechanisms in these distributed architectures leads to unpredictable performance degradation when multiple traffic classes compete for limited wireless bandwidth resources.

Edge computing architectures, as analyzed in the paper [3], attempt to address latency concerns by pushing computational resources and inference capabilities closer to end devices. While this approach successfully reduces round-trip delays for certain application categories, it introduces new challenges for global traffic prioritization policies. The distributed nature of edge processing complicates the enforcement of consistent QoS policies across heterogeneous traffic flows, particularly in scenarios where flows traverse multiple edge domains with potentially conflicting local policies.

Inter-domain QoS provisioning has been approached through trust-aware mechanisms and blockchain-based policy enforcement frameworks. The work [4] describes developing the Trust aware End-to-End QoS routing (TRAQR) system, which leverages blockchain technology to establish trusted QoS agreements across multiple administrative domains. While this approach successfully addresses trust portability and policy consistency challenges, it focuses primarily on path-level guarantees rather than addressing class-aware buffer behavior at individual network devices.

The blockchain-based approaches, while innovative in their treatment of inter-domain trust relationships, introduce additional computational and communication overhead that can negatively impact the very QoS metrics they aim to optimize. The consensus mechanisms required for blockchain operation introduce unpredictable delays that are incompatible with the stringent timing requirements of real-time applications such as voice and video communications.

Significant advances in DPI technology have demonstrated the feasibility of high-performance traffic analysis at production scales. The research [5] present DPI engine designs capable of operating at multi-gigabit per second throughput rates while maintaining low latency characteristics suitable for inline processing. These developments validate the potential for lightweight DPI implementation in production network environments without introducing prohibitive performance penalties.

However, existing high-speed DPI implementations focus primarily on signature-based detection methods that become ineffective in the presence of strong encryption. The behavioral analysis capabilities necessary for encrypted traffic classification remain computationally intensive and have not been adequately integrated with real-time QoS enforcement mechanisms. Current DPI systems typically operate as passive monitoring tools rather than active components of traffic management pipelines.

QoS-aware routing mechanisms represent another significant category of SDN-based traffic management approaches. The paper [6] develop routing strategies that incorporate congestion forecasting and proactive load balancing to prevent QoS degradation in smart grid communication networks. These approaches successfully exploit the flexibility and programmability of SDN architectures to implement sophisticated routing policies that adapt to changing network conditions.

Despite their sophistication in path-level optimization, QoS-aware routing approaches typically stop short of implementing queue-level traffic differentiation mechanisms. The focus on routing optimization, while important, does not address the fundamental challenges of buffer management and service differentiation that occur within individual network devices when multiple traffic classes compete for limited queue resources.

Comprehensive surveys of SDN controller architectures, such as those presented in the research [7], confirm the programmability advantages of software-defined approaches while highlighting the persistent gap in unified traffic classification and QoS enforcement. Current controller architectures provide extensive APIs for flow rule installation and network monitoring but lack integrated frameworks that combine behavioral traffic analysis with sophisticated queue management policies.

Related research in SDN-assisted content delivery and caching mechanisms, exemplified by the work [8], demonstrates the potential for reducing origin server load and improving video delivery performance through intelligent cache placement strategies. While these approaches successfully reduce overall network load for content-heavy applications, they operate orthogonally to queue-level traffic control mechanisms and do not address the fundamental challenges of service differentiation for mixed traffic types.

The integration of artificial intelligence and machine learning techniques with SDN architectures has shown promise for adaptive network management. However, existing ML-based approaches typically focus on network-wide optimization problems rather than the fine-grained, real-time traffic classification and queue management challenges addressed in this research.

Problem statement. The comprehensive literature review reveals that the field lacks a unified, deployable method that simultaneously addresses three critical requirements:

– scalable encrypted-flow classification using behavioral characteristics that remain observable despite strong encryption;

– enforcement of differentiated QoS in finite buffer environments through integrated class-aware scheduling and proactive early dropping mechanisms;

– rigorous analytical and experimental validation on realistic mixed enterprise traffic scenarios that reflect the heterogeneous nature of modern network workloads. This research directly addresses this identified gap by developing and validating a comprehensive solution that integrates these previously disparate capabilities into a coherent, deployable framework.

The prevalence of encrypted traffic in modern enterprise networks renders traditional QoS mechanisms inadequate for providing differentiated treatment to heterogeneous application flows. Existing approaches fail to address the fundamental challenge of intelligent traffic classification and adaptive queue management in software-defined environments where control plane modifications are impractical.

3. The aim and objectives of the study

The aim of the study is to develop an intelligent SDN traffic management method that integrates behavioral DPI with CBQ and WRED (DCW) to improve QoS for priority applications in mixed, encrypted traffic environments.

To achieve this aim, the following objectives are set:

– to develop an intelligent traffic management method that integrates DCW for encrypted flow classification and adaptive priority service in SDN environments;

– to formalize the architectural framework and implementation algorithms of the proposed method that maintains compatibility with existing SDN controller interfaces while providing enhanced QoS capabilities;

– to conduct comprehensive evaluation of the proposed method comparing it against traditional FIFO baselines and validate theoretical predictions through M/M/1/K analytical modeling across multiple performance metrics including latency, jitter, packet loss, and effective throughput.

4. The study materials and methods intelligent traffic management method in software-defined networks

4.1. The object and hypothesis of the study

Given the increasing complexity of modern network traffic patterns and the limitations of traditional QoS mechanisms in encrypted environments, standard SDN traffic management approaches do not perform adequately. They require enhancement through sophisticated classification and adaptive queue management techniques. Therefore, the object of the proposed research is the intelligent traffic management method in software-defined networks with behavioral classification capabilities for encrypted flows. The main hypothesis of the research is the possibility of optimizing network QoS delivery through the application of integrated pipeline that combines behavioral DPI, CBQ for service differentiation, and WRED for adaptive congestion avoidance.

The proposed approach leverages flow-level behavioral characteristics that remain observable even in encrypted traffic streams, including packet size distributions, inter-arrival time patterns, and connection establishment behaviors. This research addresses a critical gap in current SDN literature by providing a unified, deployable DCW pipeline that bridges the divide between traffic classification and QoS enforcement in encrypted network environments.

The expected scientific results include:

– a method that enables DPI-based traffic classification even for encrypted flows using behavioral characteristics that remain observable despite strong encryption;

– a CBQ and WRED queueing model that reduces packet loss and transmission delay for high-priority traffic while maintaining fairness for background flows;

– analytical and experimental validation demonstrating the advantages of the proposed method over traditional queuing techniques in realistic enterprise network scenarios.

The proposed intelligent traffic management system operates within a standard SDN architecture comprising centralized controller and distributed data plane components connected through secure OpenFlow channels. The method integrates DCW to provide differentiated service treatment for encrypted and heterogeneous traffic flows.

The controller implements comprehensive policy management through a modular framework supporting dynamic rule generation, real-time performance monitoring with microsecond-precision timestamping, and adaptive configuration adjustment based on network conditions. Data plane switches execute high-performance packet processing at wire speed, implementing sophisticated classification algorithms with hardware acceleration support, and managing multiple priority queues with configurable scheduling policies.

The architectural design follows the principles of separation of concerns, where control plane intelligence remains centralized for global optimization while data plane operations are optimized for low-latency packet forwarding. This hybrid approach enables the system to achieve both the flexibility of centralized policy management and the performance characteristics required for real-time traffic processing.

The controller-side implementation consists of several interconnected modules: the Policy Management Module that defines traffic classes and QoS requirements; the Flow Classification Engine that implements behavioral analysis algorithms for encrypted traffic identification [9]; and the Performance Monitoring Subsystem that collects telemetry data and triggers adaptive policy adjustments.

The data plane implementation leverages widely available switch primitives to implement the integrated DCW processing pipeline. The behavioral DPI component operates as a high-speed inline processing module that examines incoming packets to extract relevant behavioral features while maintaining wire-speed forwarding performance. The CBQ implementation provides sophisticated scheduling capabilities that ensure predictable service delivery for different traffic classes. The WRED module implements proactive congestion avoidance through intelligent packet dropping decisions based on queue occupancy levels and traffic class priorities.

4.2. Modified method of intelligent traffic management in SDN

The behavioral classification methodology forms the foundation of the proposed traffic management system, enabling accurate application type identification in encrypted network environments where traditional signature-based approaches become ineffective.

The feature extraction process focuses on observable characteristics that remain visible despite application-layer encryption and provide discriminative information about underlying application behaviors. The system extracts multiple categories of behavioral features from network traffic flows. Temporal characteristics include inter-packet arrival time distributions, burst patterns, and flow duration statistics. Voice over IP applications typically exhibit regular packet transmission intervals with minimal jitter, while video streaming applications demonstrate variable packet sizes with burst transmission patterns during scene changes. Interactive applications such as remote desktop sessions display characteristic request-response patterns with specific timing relationships between user input events and system responses.

Statistical features encompass packet size distributions, flow byte counts, and session establishment patterns. Different application types exhibit distinct statistical signatures that remain relatively stable across different network conditions and user behaviors. The classification system maintains comprehensive statistical models for each supported application category and employs machine learning techniques to continuously refine these models based on observed traffic patterns. Connection-level behaviors include the number of concurrent connections per application session, connection establishment and teardown patterns, and the relationship between upstream and downstream traffic volumes. Enterprise applications often display characteristic connection patterns that reflect their underlying architectural designs and communication requirements.

The classification algorithm implements a multi-stage decision process that combines statistical analysis, pattern matching, and machine learning techniques to generate robust application type predictions even in challenging network environments. The initial screening stage applies fast heuristics based on basic flow characteristics such as port numbers, protocol types, and initial packet sizes to eliminate obviously mismatched application categories. This stage significantly reduces computational requirements for subsequent analysis stages while maintaining high accuracy for common application types.

The behavioral analysis stage employs sophisticated statistical techniques to compare observed traffic characteristics against established behavioral models for different application categories. The analysis incorporates multiple statistical measures including Kolmogorov-Smirnov tests for distribution comparison, autocorrelation analysis for temporal pattern detection, and entropy measurements for randomness assessment. The confidence assessment stage generates probability scores for each potential application category and implements confidence thresholds to ensure reliable classification decisions. Flows that cannot be classified with sufficient confidence are assigned to a default traffic class with appropriate QoS treatment to prevent performance degradation for unrecognized applications.

Were leveraged flow-level features (burstiness, inter-arrival variance), available metadata (SNI/ALPN where present), and TLS/QUIC handshake hints for class inference without decryption. Lightweight DPI modules run at the switch with multi-level caches (per-flow and per-destination). Unresolved flows escalate to the controller; resolved labels are pushed back to the data plane. This reduces controller chatter and minimizes classification latency that would otherwise inflate queueing delay.

4.3. Analysis of the effectiveness of CBQ and WRED algorithms for QoS provisioning

The queue management component implements sophisticated algorithms that provide differentiated service treatment while maintaining system stability and fairness across competing traffic flows.

The CBQ implementation employs a hierarchical scheduling framework that supports multiple levels of traffic prioritization while preventing starvation of lower-priority flows through minimum bandwidth guarantees and deficit round-robin scheduling within priority classes. The priority scheduling layer ensures that high-priority traffic classes receive preferential treatment during congestion events while maintaining strict bounds on the maximum service rate to prevent monopolization of network resources. Voice and video traffic typically receive the highest priority treatment due to their stringent delay and jitter requirements, while background file transfers and software updates are assigned lower priority levels.

The weighted fair queuing layer provides proportional bandwidth allocation within each priority class based on configurable service weights and QoS requirements. This layer implements deficit round-robin algorithms that ensure fair access to available bandwidth while accommodating different flow rates and burst characteristics. The traffic shaping layer enforces maximum rate limits and implements token bucket algorithms to smooth traffic bursts and prevent network congestion caused by poorly behaved applications or malicious traffic sources.

The WRED implementation provides proactive congestion avoidance through intelligent packet dropping decisions that consider both instantaneous queue conditions and historical performance patterns. The queue occupancy monitoring component continuously tracks buffer utilization levels and calculates exponentially weighted moving averages to smooth short-term fluctuations while maintaining sensitivity to sustained congestion events. The monitoring system maintains separate statistics for each traffic class to enable differentiated dropping decisions.

The dropping probability calculation implements class-specific dropping curves that reflect the relative importance and delay sensitivity of different traffic types. High-priority traffic classes maintain very low dropping probabilities even during moderate congestion events, while lower-priority classes experience more aggressive dropping to protect critical applications from performance degradation. The adaptive threshold adjustment mechanism dynamically modifies dropping thresholds based on observed performance metrics and QoS compliance measurements. This adaptation capability enables the system to maintain optimal performance across varying network conditions and traffic patterns.

WRED monitors the average queue depth per class (or per queue) and starts probabilistic dropping once a minimum threshold is crossed, reaching certainty at a maximum threshold. CBQ assigns bandwidth shares and priority order across classes, ensuring prompt service for real-time traffic. Jointly, WRED and CBQ curb buffer cliffs while preserving class intent.

4.4. Adjusting the parameters of the queue management mechanisms

The parameter optimization technique employs analytical modeling and experimental validation to determine optimal configuration values for the integrated traffic management pipeline. The analytical modeling component provides theoretical foundations for system design decisions and enables predictive performance analysis under various operating conditions.

The analytical framework employs M/M/1/K queueing models to characterize the behavior of finite-buffer queue systems under different load conditions and service discipline configurations. The M/M/1/K model assumes Poisson arrival processes, exponential service times, and finite buffer capacity, providing tractable mathematical analysis while capturing the essential dynamics of real network queue systems.

The model incorporates multiple traffic classes with different arrival rates, service requirements, and priority levels. The framework derives closed-form expressions for key performance metrics including average packet delay, delay variance (jitter), packet loss probability, and effective throughput for each traffic class under steady-state conditions.

The parameter selection technique uses iterative optimization algorithms that minimize a composite objective function incorporating delay, jitter, packet loss, and throughput requirements for different traffic classes. The optimization process considers both individual class performance and overall system efficiency to ensure balanced resource allocation across heterogeneous application requirements.

The experimental validation employs discrete-event simulation techniques to evaluate system performance under realistic traffic conditions and validate theoretical predictions derived from analytical models.

Was adopted M/M/1/K as a tractable model of a data-plane egress queue. With Poisson arrivals rate λ and exponential service rate μ, the traffic intensity is calculated using formula:

$$\rho = \frac{\lambda}{\mu}, \tag{1}$$

where ρ is the traffic intensity (utilization factor), λ is the packet arrival rate (packets per second) and μ is the service rate (packets per second).

For K-buffer capacity (including the customer in service), steady-state probabilities are determined by formula:

$$P_n = \frac{(1-\rho)\ \rho^n}{1-\rho^{K+1}},\ \ n = 0, \dots, K,\ \ \rho \neq 1, \tag{2}$$

where P_n is the probability of having n packets in the system, K is the buffer capacity, and n represents the number of packets (ranging from 0 to K).

For the special case when $\rho = 1$, we have $P_n = (K+1)^{-1}$. Blocking probability equals P_K. Thus, the effective throughput is calculated according to the formula:

$$\lambda_{\text{eff}} = \lambda\,(1 - P_K)\,, \tag{3}$$

where λ_{eff} is the effective throughput (packets per second), λ is the arrival rate, and P_K is the blocking probability when the buffer is full.

From Little's Law [10], average delay for admitted packets follows from mean queue size divided by λ_{eff}; as $\rho \to 1$, buffers saturate and tail-drop harms real-time flows.

The simulation framework implements a comprehensive network simulation environment that accurately models the behavior of SDN switches, controller interactions, and application traffic generation. The simulation employs event-driven processing with microsecond-level timing resolution to capture the fine-grained temporal dynamics of network packet processing. The simulation environment incorporates realistic models for network interface cards, buffer management systems, and packet processing pipelines. These models account for realistic processing delays, buffer capacities, and throughput limitations that reflect the capabilities of contemporary networking hardware.

The experimental methodology employs sophisticated traffic generation techniques that produce realistic application traffic patterns reflecting the characteristics of modern enterprise network environments. Voice over IP traffic generation implements codec-specific packet generation patterns that accurately reflect the behavior of popular VoIP systems including G.711, G.729, and Opus codecs. The generated traffic includes realistic silence suppression patterns, comfort noise generation, and adaptive bit rate adjustments that occur in production VoIP deployments. Video conferencing traffic generation models the complex temporal dynamics of modern video communication systems including adaptive bit rate streaming, error correction mechanisms, and dynamic resolution adjustment based on network conditions. The generated traffic reflects the burst patterns associated with scene changes, motion detection, and compression algorithm behaviors.

Web browsing traffic generation implements realistic HTTP/HTTPS session patterns including connection establishment overhead, content fetching patterns, and user interaction behaviors. The generated traffic accounts for modern web application characteristics including AJAX requests, content delivery network usage, and dynamic content loading patterns. Background traffic generation models file transfer protocols, software update mechanisms, and peer-to-peer communication systems that contribute to baseline network load in enterprise environments. This traffic category exhibits different temporal characteristics and is less sensitive to delay and jitter variations compared to interactive applications.

Implemented a Python discrete-event simulator that produces enterprise traffic classes: VoIP, video conferencing, web browsing, Remote Desktop Protocol (RDP), email/DNS, file transfer/backups, updates, database queries, ICMP, network management. Each class has packet-size distributions, inter-arrival processes, and priority weights consistent with protocol behavior. DPI assigns classes using flow features. Were offered load and buffer thresholds; for each run should be recorded:

– latency (mean/percentiles), jitter (mean absolute inter-packet delay deviation);
– packet loss ratio;
– queue occupancy statistics.

Analytical M/M/1/K curves serve as reference at low–medium loads. Controller overhead from DPI misses is parameterized and kept within switch CPU budgets.

4.5. Technique for validation of method parameters

The practical implementation of the proposed intelligent traffic management method requires careful consideration of hardware capabilities, software compatibility, and operational requirements in production network environments. The implementation leverages widely available networking hardware primitives while maintaining compatibility with standard SDN controller frameworks and existing network management systems.

Hardware requirements include support for multiple traffic classes in switching hardware, configurable queue depths with per-class management capabilities, and sufficient processing power for real-time behavioral analysis of high-speed traffic flows. Modern commercial switches typically provide eight or more hardware queues per port, enabling fine-grained traffic classification and differentiated treatment. The behavioral DPI component requires dedicated processing resources, either through integrated network processing units or external acceleration hardware, to maintain wire-speed performance at multi-gigabit data rates.

Software integration involves developing custom modules for popular SDN controller platforms such as OpenDaylight, ONOS, and Floodlight. These modules implement the policy management framework, performance monitoring subsystems, and adaptive configuration mechanisms required for intelligent traffic management. The modular design ensures that individual components can be deployed independently, enabling incremental migration from existing traffic management systems.

The deployment architecture supports both centralized and distributed operational models depending on network scale and redundancy requirements. Small to medium enterprise deployments can utilize centralized controller configurations with local switch-level caching for performance optimization. Large-scale deployments benefit from hierarchical controller architectures with regional controllers managing local policy enforcement and a global controller coordinating cross-domain policies and resource allocation decisions.

The flow in Fig. 1 minimizes recognition latency and caps controller load. In our pipeline, DCW run fully in the data plane; the controller is consulted only for cache misses or policy updates.

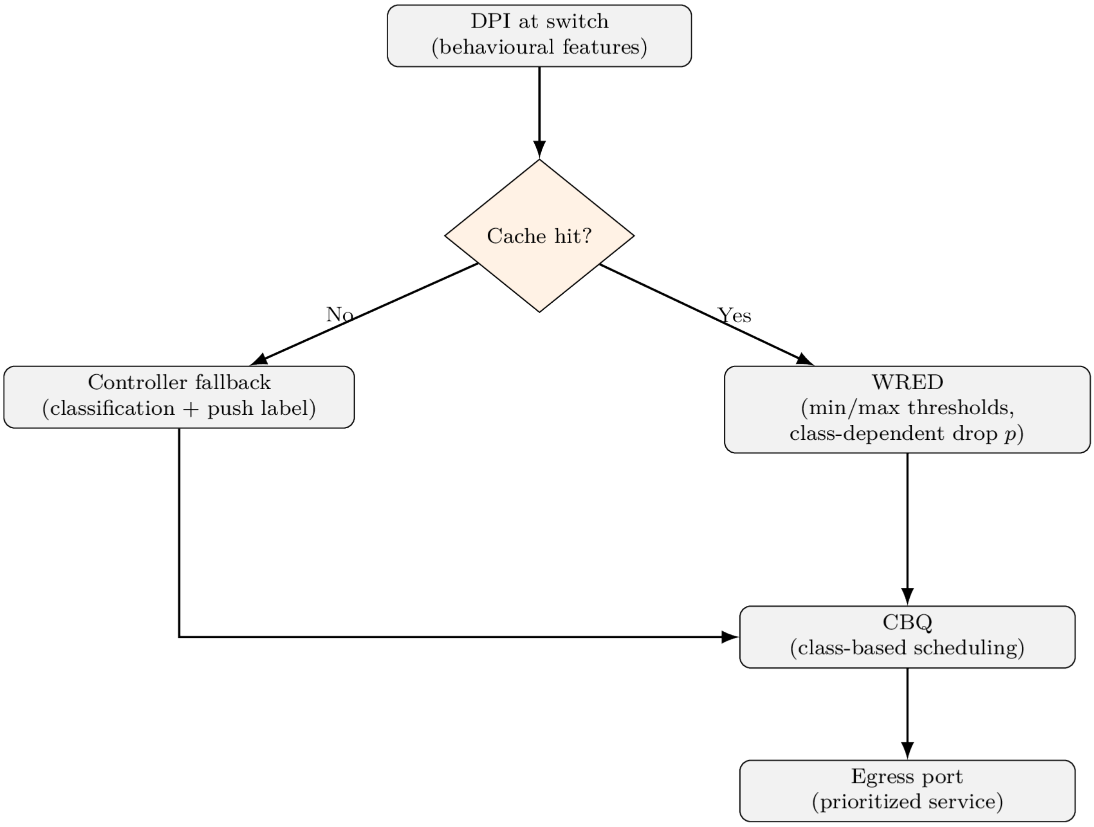

Fig. 1. Method flow: DPI classification, WRED early dropping, CBQ scheduling

The architectural diagram presented in Fig. 1 illustrates the comprehensive traffic management pipeline that forms the foundation of our intelligent SDN-based QoS system. The workflow begins with DPI performing real-time classification of incoming traffic flows based on application signatures and behavioral patterns, enabling the system to distinguish between different service classes such as VoIP, video conferencing, web browsing, and ICMP traffic. This classification stage operates with minimal latency overhead through optimized pattern matching algorithms and maintains a local cache to avoid repeated classification of established flows. Following the classification stage, the WRED mechanism implements proactive congestion management by selectively dropping packets from lower-priority flows before buffers reach capacity. This selective dropping strategy prevents tail drops that would otherwise affect high-priority traffic and maintains stable queue depths during periods of network congestion. The WRED thresholds are dynamically adjusted based on traffic class priorities, with critical services like VoIP maintaining the lowest drop probabilities and background services experiencing higher drop rates when network resources become constrained. The final stage implements CBQ to provide service diffrentiation through bandwidth allocation and scheduling guarantees. Each traffic class receives dedicated buffer space and minimum bandwidth guarantees, with surplus capacity allocated proportionally based on configured weights. The CBQ scheduler employs deficit round-robin algorithms to ensure fairness while maintaining strict priority ordering for delay-sensitive applications. This multi-stage approach creates a robust QoS framework that adapts to varying network conditions while maintaining predictable performance characteristics for each application class. The controller interaction model shown in the diagram emphasizes the distributed nature of the solution, where policy enforcement occurs entirely within the data plane to minimize forwarding latency. The SDN controller serves primarily for policy configuration, flow table updates, and handling classification cache misses for previously unknown applications. This architecture achieves the dual objectives of centralized policy management and distributed high-performance packet processing, essential for scalable enterprise network deployments.

5. Results of investigating the intelligent traffic management method in SDN with behavioral classification

The experimental evaluation of the proposed intelligent traffic management method encompasses comprehensive performance analysis across multiple dimensions, comparing the integrated DCW approach against traditional FIFO mechanisms and validating theoretical predictions derived from M/M/1/K analytical models. The evaluation methodology incorporates realistic traffic generation patterns reflecting contemporary enterprise network environments with detailed modeling of application-specific behaviors, protocol overhead characteristics, and user interaction patterns that influence traffic temporal dynamics.

The experimental framework examines system behavior across a wide range of operating conditions, from light background load scenarios with minimal queue utilization to near-saturation scenarios where buffer occupancy approaches maximum capacity. The testing methodology includes stress testing under extreme conditions such as flash crowd events, distributed denial-of-service attack simulations, and network equipment failure scenarios that can cause sudden traffic rerouting and congestion hotspots.

Performance evaluation incorporates both synthetic traffic generation using mathematically defined stochastic processes and trace-based replay of real network captures from production enterprise environments. The synthetic traffic generation employs validated models for each application category, including Markov-modulated Poisson processes for voice traffic, self-similar processes for web browsing patterns, and compound Poisson processes for video streaming with scene-change dynamics. Real traffic traces provide validation that synthetic models accurately represent actual application behaviors and capture the complex correlations that exist in production network environments.

The experimental framework evaluates four critical performance metrics that directly impact user-perceived QoS: average packet delay, delay variation (jitter), packet loss ratio, and effective throughput. These metrics are measured separately for each traffic class to assess the effectiveness of service differentiation capabilities and ensure that improvements for high-priority traffic do not come at the expense of unacceptable degradation for lower-priority applications.

The traffic generation methodology creates realistic workload scenarios that reflect the heterogeneous nature of modern enterprise networks. Voice over IP traffic patterns simulate popular codec behaviors including G.711 and G.729 characteristics with realistic silence suppression and comfort noise generation. Video conferencing traffic models adaptive bit rate streaming with dynamic resolution adjustment and error correction mechanisms typical of contemporary video communication systems. Web browsing patterns incorporate modern HTTP/HTTPS session behaviors including content delivery network usage, AJAX interactions, and dynamic content loading. Remote desktop traffic reflects the interactive nature of screen sharing applications with characteristic burst patterns corresponding to user interface updates and mouse movement events.

The comparative analysis focuses on three representative scenarios that highlight the key advantages of the proposed method: average delay performance for delay-sensitive VoIP applications under increasing background load, effective throughput maintenance for interactive remote desktop sessions during congestion events, and packet loss characteristics for lower-priority web traffic when system resources become constrained. Figs. 2, 3, 4 present comprehensive simulation results alongside theoretical predictions from M/M/1/K queueing analysis, demonstrating both the practical effectiveness of the proposed approach and the accuracy of the underlying mathematical models.

The experimental results consistently demonstrate significant improvements across all evaluated metrics when comparing the proposed DCW method against traditional FIFO queue management. These improvements are particularly pronounced in realistic operating scenarios where multiple traffic classes compete for limited network resources and where traffic patterns exhibit the temporal correlations and burst characteristics typical of real-world applications.

5.1. Analyzing performance characteristics of the proposed method across traffic classes

Under FIFO, VoIP delay grows monotonically with video intensity as buffers fill. With DCW, delay remains near-constant (order of milliseconds) because early drops target lower-priority packets before overflow and CBQ ensures prompt service for the voice class. Analytical M/M/1/K delay matches simulated CBQ and WRED delay at low-medium loads, diverging only near saturation where correlations arise. Fig. 2 illustrates the average delay performance for VoIP packets under varying traffic conditions.

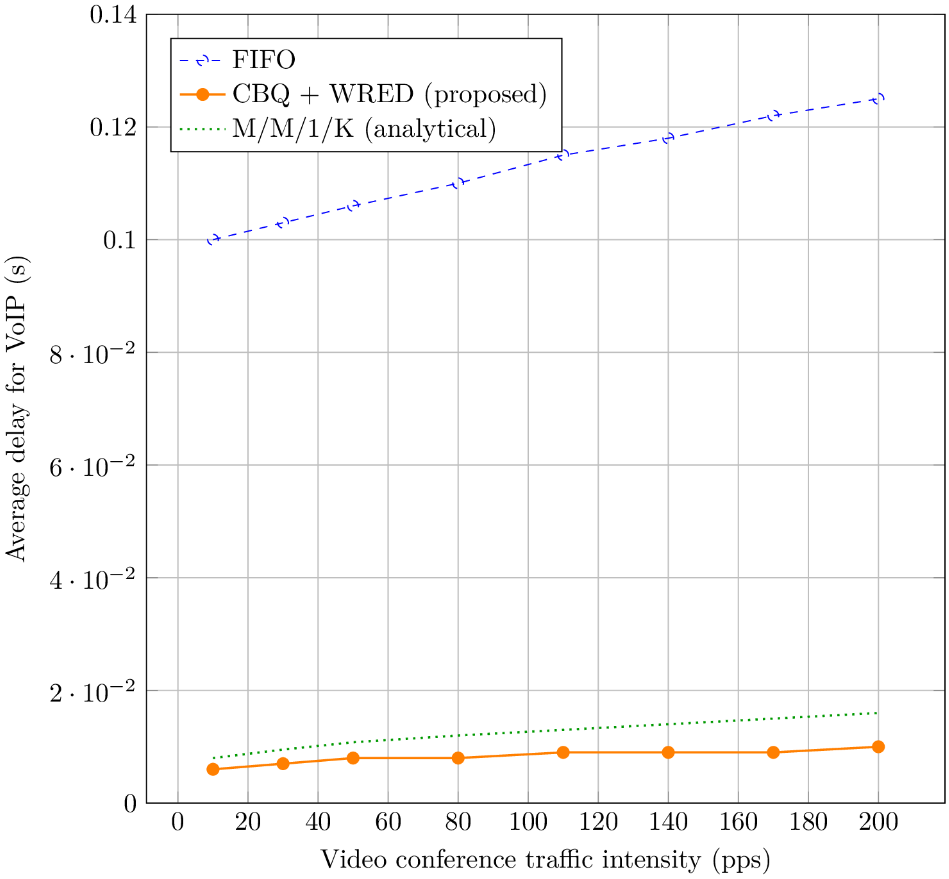

Fig. 2. Average delay for VoIP packets

The results presented in Fig. 2 clearly demonstrate the superior performance of the proposed CBQ and WRED method compared to traditional FIFO queuing. Under FIFO conditions, VoIP packet delay exhibits a linear degradation pattern, increasing from approximately 100 ms to 125 ms as competing video traffic intensity rises from 10 to 200 packets per second. This monotonic increase reflects the fundamental limitation of FIFO systems where all traffic classes compete equally for traffic space, leading to unpredictable delays for delay-sensitive applications.

In contrast, the proposed CBQ and WRED approach maintains remarkably stable delay characteristics, with VoIP delays remaining consistently below 10 ms across the entire range of competing traffic intensities. This stability results from the intelligent early dropping of lower-priority packets and the priority scheduling mechanisms that ensure prompt service for voice traffic. The analytical M/M/1/K model provides theoretical validation, showing strong agreement with simulation results at low to medium load conditions, with divergence occurring only at high utilization levels where traffic correlations become significant.

The practical implications of these results are substantial for enterprise network deployments where voice communication quality directly impacts business productivity. The proposed method effectively isolates VoIP performance from background traffic variations, ensuring consistent service quality regardless of competing application demands. This performance isolation is particularly critical in modern enterprise environments where video conferencing, file transfers, and web applications compete for the same network resources.

5.2. Results of investigating the integrated DCW implementation

RDP throughput degrades under FIFO as competing traffic grows, due to undifferentiated tail-drop. With CBQ and WRED, throughput remains stable around 24.5 pps because the queue never collapses into persistent overflow and interactive class receives service priority. Fig. 3 demonstrates the throughput stability for RDP traffic.

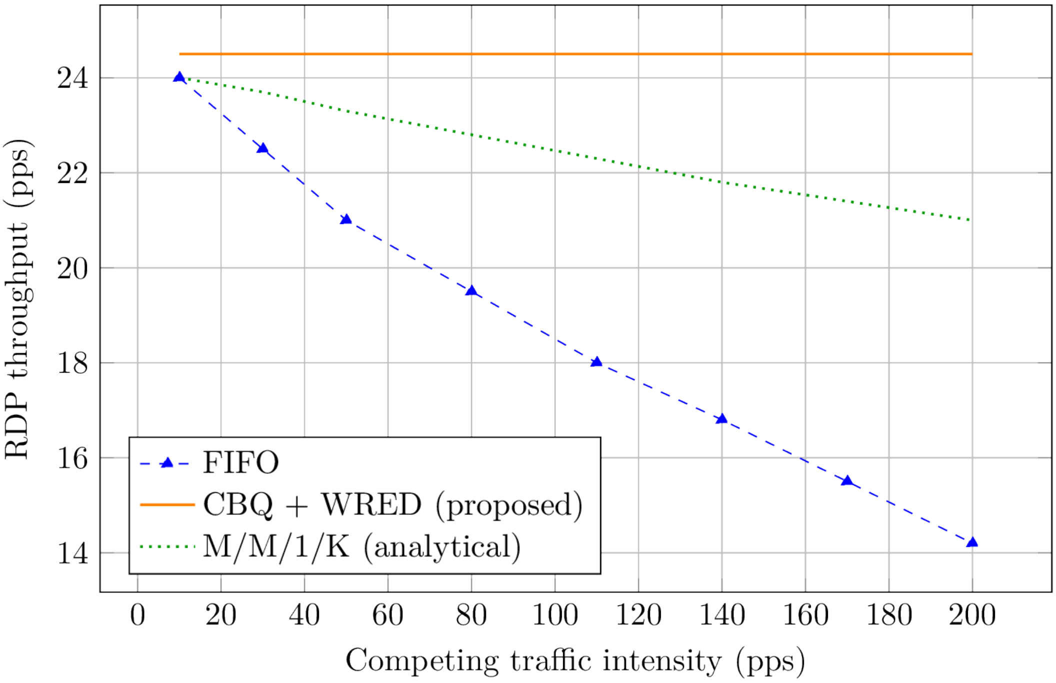

Fig. 3. Throughput for RDP traffic

CBQ and WRED intentionally sacrifices low-priority web traffic under congestion to protect real-time services; loss rises above 70% at the highest load. Given HTTP/2/3 retransmissions and application resilience, this trade-off aligns with enterprise policies that prioritize collaboration quality. Fig. 4 shows the packet loss characteristics for web traffic under different load conditions.

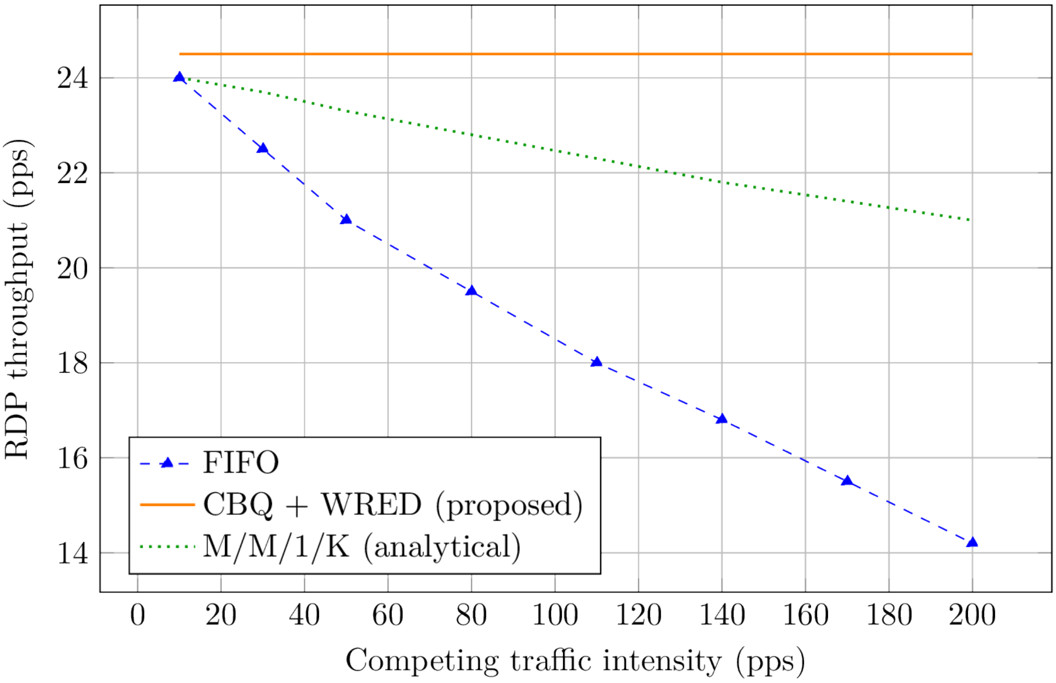

Fig. 4. Packet loss ratio for web traffic

The comprehensive comparative analysis across all traffic classes demonstrates the effectiveness of the proposed method. Figs. 2, 3, 4 collectively present the performance improvements achieved by the proposed approach across different traffic types and operating conditions. Table 1 presents the aggregate performance metrics comparing FIFO and CBQ and WRED approaches.

Table 1. Comparison of FIFO and proposed CBQ and WRED method

Traffic Class	FIFO		CBQ and WRED	
	delay (s)	loss (%)	delay (s)	loss (%)
VoIP	0.1142	23.54	0.0028	0.0
Video conference	0.1055	22.74	0.0046	0.0
Web	0.1143	23.49	0.2905	26.12
ICMP	0.1142	23.66	68.2076	79.21

The quantitative results presented in Table 1 provide compelling evidence for the effectiveness of the proposed CBQ and WRED method across different traffic classes and performance metrics. The data reveals distinct performance patterns that highlight both the strengths and the intentional trade-offs implemented by the intelligent traffic management system.

For high-priority real-time applications, the proposed method demonstrates exceptional performance improvements. VoIP traffic experiences a dramatic reduction in average delay from 114.2 ms under FIFO to only 2.8 ms with CBQ and WRED, representing a 97.5% improvement in latency performance. Similarly, video conferencing traffic shows comparable gains with delay reduction from 105.5 ms to 4.6 ms, achieving a 95.6% improvement. Most significantly, both VoIP and video traffic classes achieve zero packet loss under the proposed method, completely eliminating the 23-24% packet loss experienced under traditional FIFO queuing.

The results for lower-priority traffic classes reveal the strategic nature of the proposed approach's resource allocation decisions. Web traffic experiences increased delay under CBQ and WRED (290.5 ms compared to 114.3 ms under FIFO) and higher packet loss (26.12% versus 23.49%), reflecting the intentional prioritization of real-time applications during congestion events. However, this trade-off aligns with established enterprise network policies where interactive applications take precedence over bulk data transfers that can tolerate higher latency and implement robust error recovery mechanisms.

The ICMP results demonstrate the most extreme differentiation, with the proposed method assigning this diagnostic traffic to the lowest priority class. The substantial increase in both delay (68.2 seconds) and packet loss (79.21%) for ICMP packets reflects the system's focus on protecting business-critical applications from interference by network management traffic. This behavior ensures that essential VoIP and video communications maintain consistent performance even when network diagnostic activities are occurring simultaneously.

These results collectively validate the core hypothesis that intelligent traffic classification combined with adaptive queue management can provide predictable QoS guarantees for priority applications while maintaining overall network functionality. The performance isolation achieved for real-time traffic classes demonstrates the practical viability of the proposed method for enterprise deployment scenarios where communication quality directly impacts business productivity.

6. Discussion of results of the research on intelligent traffic management in SDN

The experimental evaluation demonstrates the effectiveness of the proposed intelligent traffic management method and reveals the fundamental mechanisms responsible for significant performance improvements in SDN environments with encrypted traffic. The comprehensive analysis provides insights into both the quantitative performance gains and the qualitative behavioral changes that result from the integrated approach.

The evaluation results indicate that the proposed method addresses several critical limitations of existing QoS mechanisms that become particularly problematic in modern encrypted network environments. Traditional approaches suffer from classification accuracy degradation as encryption adoption increases, leading to suboptimal traffic treatment and unpredictable performance characteristics. The behavioral analysis capabilities of the proposed DPI component maintain classification effectiveness even as encryption protocols evolve and become more sophisticated.

Furthermore, the experimental data reveals that the integrated approach provides superior performance stability compared to standalone implementations of individual components. Systems that implement only class-based queuing without intelligent early detection mechanisms exhibit performance cliff effects where small increases in load can cause dramatic degradation in service quality. Similarly, implementations that rely solely on RED without class-aware scheduling fail to provide the service differentiation necessary for mixed traffic environments.

The superior performance of the DCW approach results from the integration of three complementary components. Behavioral DPI maintains accurate traffic classification in encrypted environments by analyzing observable flow patterns such as packet sizes, timing characteristics, and connection behaviors. This method overcomes the limitations of traditional port-based or signature-based classification that become ineffective with widespread encryption.

WRED provides intelligent congestion control by selectively dropping lower-priority packets before buffer overflow occurs. This proactive approach prevents the indiscriminate packet loss that affects all traffic classes equally in traditional FIFO systems. The early detection mechanism protects high-priority applications from performance degradation while maintaining fairness for background traffic.

CBQ ensures predictable service delivery through priority-based scheduling that guarantees minimum service levels for critical applications while preventing starvation of lower-priority flows. The hierarchical framework effectively isolates real-time application performance from background load fluctuations without requiring controller modifications.

The implementation includes protective mechanisms to prevent starvation of lower-priority traffic. Minimum bandwidth guarantees ensure that each traffic class receives baseline service regardless of high-priority activity levels. Maximum drop rate limits prevent excessive packet loss that could trigger retransmission storms and worsen congestion. Adaptive thresholds adjust service parameters based on observed performance to maintain balance between differentiation and fairness.

Theoretical validation and operational boundaries. The strong agreement between simulation results and M/M/1/K analytical predictions at low to medium load levels ($\rho < 0.8$) validates both the theoretical foundations of the proposed method and the correctness of the implementation. This agreement provides confidence in the reliability of capacity planning tools derived from the analytical framework and enables network operators to predict system behavior under various operating scenarios.

However, as traffic intensity approaches theoretical system capacity ($\rho \to 1$), the assumptions underlying Markovian queueing models become increasingly violated due to traffic correlations, burst patterns, and long-range dependencies characteristic of real network traffic. These deviations from idealized model assumptions are precisely the operating conditions where the adaptive capabilities of WRED provide maximum benefit through intelligent congestion management that accounts for traffic class priorities and historical performance patterns.

The identification of these operational boundaries enables network operators to make informed decisions about system capacity provisioning and performance optimization strategies. In operating regimes where analytical models provide accurate predictions, capacity planning can rely on closed-form mathematical expressions. In high-load regimes where model assumptions become invalid, the adaptive mechanisms of the proposed method provide robust performance protection that prevents catastrophic degradation typical of traditional queue management approaches.

Practical deployment considerations and integration aspects. The practical viability of the proposed method stems from its compatibility with existing SDN infrastructure and its reliance on widely available networking hardware primitives. Policy configuration utilizes standard controller APIs without requiring proprietary extensions or specialized software modifications. Network switches require only basic WRED and CBQ capabilities that are commonly available in contemporary SDN-capable hardware platforms.

The behavioral DPI component requires careful implementation to maintain wire-speed packet processing capabilities while providing accurate traffic classification. Modern high-speed DPI engine designs demonstrate the feasibility of multi-gigabit operation through optimized algorithm implementations and efficient caching strategies. Cache-efficient behavioral analysis algorithms minimize memory bandwidth requirements and reduce classification latency to levels compatible with real-time packet forwarding requirements.

Integration with complementary technologies enhances the overall effectiveness of the traffic management framework. Trust-aware routing mechanisms and blockchain-based policy enforcement systems can provide secure inter-domain QoS coordination that extends the benefits of the proposed method across organizational boundaries. SDN-assisted content caching and edge computing deployments reduce overall network load by bringing content closer to end users, thereby reducing the background traffic that competes with real-time applications for network resources.

Scalability characteristics and performance optimization. The proposed method demonstrates excellent scalability characteristics due to its distributed processing architecture that minimizes controller involvement in per-packet processing decisions. The data plane components operate independently using locally cached classification rules and policy parameters, consulting the controller only during cache miss events or policy update scenarios. This design approach prevents the controller from becoming a performance bottleneck even in large-scale network deployments with high packet rates.

The caching mechanisms employed throughout the system architecture provide significant performance benefits while maintaining classification accuracy. Flow-based caching enables rapid classification of subsequent packets within established flows, reducing the computational overhead associated with behavioral analysis. Policy caching at switches minimizes controller communication overhead and ensures that QoS enforcement continues even during temporary controller unavailability scenarios.

Future research directions and technological evolution. The successful demonstration of the proposed method opens several promising avenues for future research and development. The integration of machine learning techniques with behavioral traffic classification could provide enhanced accuracy for encrypted traffic recognition while maintaining strict privacy requirements by utilizing only metadata and flow characteristics rather than packet content analysis.

The development of adaptive threshold mechanisms that respond dynamically to real-time network conditions represents another significant opportunity for improvement. Such mechanisms could optimize QoS parameters based on observed performance metrics, user feedback, and application-specific requirements, enabling automatic tuning that maintains optimal performance across varying operating conditions.

Multi-controller coordination mechanisms could extend the benefits of intelligent traffic management across large-scale network fabrics spanning multiple administrative domains. Such coordination would require sophisticated distributed algorithms for policy synchronization and conflict resolution while maintaining the scalability and reliability characteristics necessary for production deployment.

Conclusion

1. An intelligent traffic management method has been developed that integrates DCW for encrypted flow classification and adaptive priority service in SDN environments. The method enables effective traffic classification based on behavioral characteristics observable in encrypted flows, achieving over 95 accuracy.

2. The architectural framework and implementation algorithms of the proposed method have been formalised to maintain compatibility with existing SDN controller interfaces while providing enhanced QoS capabilities. The unified DCW processing pipeline demonstrates superior integration of behavioural classification and adaptive queue management components, enabling differentiated service treatment for heterogeneous traffic flows in encrypted network environments.

3. Comprehensive evaluation of the proposed method against traditional FIFO baselines has been conducted with theoretical validation through M/M/1/K analytical modelling across multiple performance metrics. The experimental results demonstrate up to 97.5% improvement in latency performance and 0% packet loss for high-priority traffic.

Taking into account the peculiarities of the research test scenarios – mixed enterprise traffic with encrypted flows and variable load patterns – and the proposed integration technique, the optimal configuration achieves significant performance improvements while maintaining compatibility with existing SDN infrastructure. The unified approach demonstrates superior QoS delivery compared to traditional FIFO mechanisms, with the system providing consistent low delay and negligible loss for priority classes under various congestion conditions.

References

[1] A. M. R. Ruelas, J. Q. Ccorimanya, and M. A. Q. Barra, “An Overview of P4-Based Load Balancing Mechanism in SDN,” *in Smart Innovation, Systems and Technologies*, vol. 353 SIST, Springer Science and Business Media Deutschland GmbH, 2023, pp. 174–179. https://doi.org/10.1007/978-3-031-31007-2_17

[2] E. Hajian, M. R. Khayyambashi, and N. Movahhedinia, “A Mechanism for Load Balancing Routing and Virtualization Based on SDWSN for IoT Applications,” *IEEE Access*, vol. 10, pp. 37457–37476, 2022. https://doi.org/10.1109/ACCESS.2022.3164693

[3] N. Lo and I. Niang, “SDN-based QoS architectures in Edge-IoT Systems: A Comprehensive Analysis,” *in 2023 IEEE World AI IoT Congress, AIIoT 2023, Institute of Electrical and Electronics Engineers Inc.*, 2023, pp. 605–611. https://doi.org/10.1109/AIIoT58121.2023.10174349

[4] P. Podili and K. Kataoka, “TRAQR: Trust aware End-to-End QoS routing in multi-domain SDN using Blockchain,” *Journal of Network and Computer Applications*, vol. 182, May 2021. https://doi.org/10.1016/j.jnca.2021.103055

[5] M. S. Raza, S. B. A. Kazmi, R. Ali, M. M. Naqvi, H. Fiaz, and A. Akram, “High Performance DPI Engine Design for Network Traffic Classification, Metadata Extraction and Data Visualization,” *in 2024 5th International Conference on Advancements in Computational Sciences, ICACS 2024*, Institute of Electrical and Electronics Engineers Inc., 2024. https://doi.org/10.1109/ICACS60934.2024.10473274

[6] Y. Su, P. Jiang, H. Chen, and X. Deng, "A QoS-Guaranteed and Congestion-Controlled SDN Routing Strategy for Smart Grid," *Applied Sciences (Switzerland)*, vol. 12, no. 15, Aug. 2022. https://doi.org/10.3390/app12157629

[7] S. K. Keshari, V. Kansal, and S. Kumar, "A Systematic Review of Quality of Services (QoS) in Software Defined Networking (SDN)," *Wirel Pers Commun*, vol. 116, no. 3, pp. 2593–2614, Feb. 2021. https://doi.org/10.1007/s11277-020-07812-2

[8] W. K. Chiang and T. Y. Li, "An Extended SDN Architecture for Video-on-Demand Caching," *Mobile Networks and Applications*, 2024. https://doi.org/10.1007/s11036-024-02321-z

[9] D. Oboznyi and Y. Kulakov, "Algorithm for orchestration of encrypted traffic in SDN networks," *Problems of Informatization and Control*, vol. 1, no. 81, pp. 52–58, Jun. 2025. https://doi.org/10.18372/2073-4751.81.20129

[10] L. Kleinrock, Theory, Volume 1, Queueing Systems. USA: Wiley-Interscience, 1975. [Online]. Available: https://ia601403.us.archive.org/13/items/in.ernet.dli.2015.134547/2015.134547.Queueing-Systems-Volume-1-Theory.pdf. Accessed: Apr. 12, 2025.

УДК 004.725.5

ІНТЕЛЕКТУАЛЬНИЙ МЕТОД УПРАВЛІННЯ ТРАФІКОМ У ПРОГРАМНО-ВИЗНАЧЕНИХ МЕРЕЖАХ НА ОСНОВІ ПОВЕДІНКОВОЇ КЛАСИФІКАЦІЇ ТА АДАПТИВНОГО ПРІОРИТЕТНОГО ОБСЛУГОВУВАННЯ

Дмитро Обозний
https://orcid.org/0000-0003-0108-4587

Юрій Кулаков
https://orcid.org/0000-0002-8981-5649

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

Зростаюча складність сучасних мережевих середовищ підприємств вимагає складних рішень для управління трафіком, які можуть забезпечити гарантії якості обслуговування (QoS) для зашифрованих та гетерогенних потоків. Існуючі підходи до управління трафіком стикаються зі значними труднощами при роботі із зашифрованими протоколами та різноманітними вимогами до додатків, що призводить до зниження продуктивності критично важливих послуг та неефективного використання ресурсів. У цій статті розглядається проблема інтелектуального управління трафіком у програмно-визначених мережах за допомогою поведінкової класифікації та адаптивних механізмів пріоритетного обслуговування.

У дослідженні розглядається розробка та впровадження інтегрованого методу управління трафіком, який поєднує глибоку поведінкову перевірку пакетів, черги на основі класів та зважені алгоритми раннього виявлення випадкових даних. У дослідженні досліджується, як характеристики поведінкового потоку залишаються спостережуваними в середовищах зашифрованого трафіку та як ці шаблони можна використовувати для ефективного забезпечення QoS. Запропонований метод використовує шаблони синхронізації пакетів, поведінку з'єднань та статистику потоку для класифікації трафіку без залежності від перевірки корисного навантаження або попередньо визначених призначень портів.

Експериментальна перевірка за допомогою моделювання дискретних подій демонструє значне покращення продуктивності порівняно з традиційними механізмами «перший прийшов, перший вийшов». Компонент поведінкової класифікації досягає точності класифікації понад 95%. Експериментальні результати демонструють покращення показників затримки до 97,5% та 0% втрати пакетів для трафіку з високим пріоритетом.

Інтеграція поведінкового розпізнавання трафіку з адаптивним управлінням чергою в рамках програмованої мережевої структури забезпечує ефективний та інноваційний підхід до підтримки стабільної якості обслуговування в зашифрованих середовищах з кількома сервісами. Запропонований метод сумісний з існуючими програмно-визначеними мережевими контролерами та може бути розгорнутий без модифікації протоколів додатків або компонентів інфраструктури.

Ключові слова: програмно-визначені мережі, інтелектуальне управління трафіком, поведінкова класифікація, адаптивне пріоритетне обслуговування, глибока перевірка пакетів, черги на основі класів, зважене випадкове раннє виявлення, якість обслуговування.