Journal of Engineering Research (Libya)

QoS PERFORMANCE OF IPV4 AND IPV6 FOR MIXED DATA AND MULTIMEDIA TRAFFIC UNDER PQ, WFQ, AND CQ

Elfurjani Sasi Mresa*, Abdelfattah Saleh Ergheegh**

*University of Tripoli, Department of Electrical and Electronic Engineering, Tripoli, Libya.
**University of Tripoli, Department of Computer Science, Tripoli, Libya.
Email: e.mresa@uot.edu.ly
Received 4 April 2026; Revised 19 June 2026; Accepted 20 June 2026; Published 25 August 2026.

تقييم مقارن لأداء جودة الخدمة في بروتوكولي IPv4 و IPv6 لحركة مرور البيانات والوسائط المتعددة المختلطة باستخدام PQ و WFQ و CQ

الفرجانى ساسي مريسه*، عبد الفتاح صالح الرقيق**

*قسم الهندسة الكهربائية والإلكترونية، كلية الهندسة، جامعة طرابلس، ليبيا
** قسم الحاسب الآلي، كلية العلوم، جامعة طرابلس، ليبيا

الملخص

تهدف هذه الورقة إلى تقييم أداء جودة الخدمة (QoS) بين بروتوكولي IPv4 و IPv6 باستخدام ثلاث خوارزميات للجدولة الشائعة، وهي(PQ) Priority Queuing، و Weighted Fair Queuing (WFQ)، و(CQ) Custom Queuing ، وذلك باستخدام برنامج محاكاة الشبكات OPNETلحركة مرور البيانات لوسائط متعددة مختلطة والتي تشمل تطبيقات الصوت (Voice)، الفيديو (Video Conferencing)، HTTP، FTP، البريد الإلكتروني (Email)، وقواعد البيانات. وتحت ظروف حمل مرتفع ومثقل (Heavy Load) تم تقييم مؤشرات الأداء، وتشمل زمن التأخير من طرف إلى طرف، تباين تأخير الحزم، معدل نقل البيانات وزمن استجابة التطبيق. أظهرت النتائج أن تأثير خوارزمية الجدولة (Queuing Algorithm) كان أكبر من تأثير نوع البروتوكول المستخدم (IPv4, IPv6) في تحديد جودة الخدمة (QoS). كما تبين أن أداء IPv4 و IPv6 كان متقارباً في معظم التطبيقات غير الآنية (Non Real-time Traffic)، مع وجود فروق طفيفة تعتمد على نوع خوارزمية الجدولة. وفيما يتعلق بحركة الصوت، حافظ كل من PQ و WFQ على قيم منخفضة لزمن التأخير وتذبذب التأخير، في حين سجل CQ قيما أعلى في متوسط زمن التذبذب. أما في تطبيق الفيديو (Video Conferencing) ، فقد كان تأثير خوارزمية الجدولة أكثر وضوحاً، حيث حققت CQ أقل قيم للتأخير وتذبذب الحزم ضمن بيئة الاختبار، بينما أدى استخدام WFQ إلى تراكم ملحوظ في التأخير والدخول في نمط أداء مختلف تحت ظروف الازدحام العالي. وتشير هذه النتائج إلى أن اختيار خوارزمية الجدولة في الشبكات التي تدعم حركة مرور البيانات لوسائط متعددة ومختلطة Mixed Traffic يعد عاملاً أكثر حسماً من نوع البروتوكول وحده في تحديد مستوى جودة الخدمة.

الكلمات المفتاحية: IPv4، IPv6، QoS، PQ، WFQ، CQ، Voice، http ، FTP، Email، Database

Copyright © 2026 by Faculty of Engineering, University of Tripoli, Libya.
This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

ABSTRACT

This paper compares the Quality of Service (QoS) performance of Internet Protocol version 4 (IPv4) and Internet Protocol version 6 (IPv6) under Priority Queuing (PQ), Weighted Fair Queuing (WFQ), and Custom Queuing (CQ) for mixed traffic comprising voice, video conferencing, Hypertext Transfer Protocol (HTTP), File Transfer Protocol (FTP), email, and database applications in Optimized Network Engineering Tools (OPNET) Modeler. Under identical heavy-load conditions, end-to-end delay, packet delay variation, throughput, and application response time were evaluated. The results show that queueing discipline had a greater influence on QoS than protocol version. IPv4 and IPv6 behaved similarly for most non-real-time applications, with only minor scheduler-dependent differences. For voice traffic, PQ and WFQ maintained low end-to-end delay and stable delay variation, whereas CQ produced higher delay variation. For video conferencing, the scheduler effect was much stronger: CQ produced the lowest delay and packet delay variation in the tested topology, while WFQ caused severe delay build-up and entered a markedly different operating regime under heavy contention. These findings indicate that, in mixed-application networks, scheduler selection is a more decisive factor than protocol version alone in determining QoS.

KEYWORDS: IPv4, IPv6, Quality of Service (QoS), PQ, WFQ, CQ, Voice, Video Conferencing, HTTP, FTP, Email, Database.

INTRODUCTION

The rapid growth of bandwidth-intensive and delay-sensitive services—such as video conferencing, cloud applications, online gaming, and Internet of Things (IoT)—has made Quality of Service (QoS) a central requirement in modern Internet Protocol (IP) networks. Because today's traffic is inherently heterogeneous, with real-time and best-effort flows sharing the same infrastructure, network performance is increasingly determined by the ability to control delay, delay variation, throughput, and fairness under congestion.

The ongoing shift from IPv4 to IPv6 is driven primarily by scalability, but it also introduces protocol changes that can influence QoS behaviour. IPv6 supports features intended to assist traffic handling, such as hierarchical addressing and a flow label, yet it also carries a larger header and may incur additional processing overhead depending on implementation and operating conditions. As a result, whether IPv6 delivers measurable QoS advantages over IPv4 remains scenario-dependent and must be evaluated together with the traffic-management mechanisms used inside the network.

At intermediate routers, queueing disciplines directly determine packet service order and resource allocation. Priority Queuing (PQ) can minimize latency for high-priority flows but may starve lower-priority traffic under heavy load. Weighted Fair Queuing (WFQ) provides weighted bandwidth sharing to improve fairness across competing flows, while Custom Queuing (CQ) allocates predefined service portions to traffic classes to maintain predictable behaviour. Since these schedulers shape congestion dynamics, they can amplify or mask protocol-level differences between IPv4 and IPv6.

Although previous work has separately examined IPv4/IPv6 performance and queueing algorithms, the mechanism linking these two factors has not always been clearly distinguished. PQ, WFQ, and CQ do not fundamentally change their service rules according to the IP version: their operation is primarily determined by traffic classification, queue occupancy, configured priorities or weights, and outgoing-link

capacity. Nevertheless, IPv4 and IPv6 may indirectly produce different QoS measurements because of differences in header overhead, packet processing, fragmentation behaviour, and the availability of protocol-specific classification fields. The minimum IPv4 header is 20 octets, whereas the IPv6 base header is fixed at 40 octets [1], [2]. Consequently, for the same application payload, IPv6 may consume slightly more link capacity and serialization time, particularly for small packets or heavily congested low-capacity links. Conversely, the IPv6 Traffic Class and Flow Label fields can support flow identification, but they affect queueing performance only when explicitly used by the traffic-classification and forwarding configuration.

The research problem addressed in this study is therefore not whether IPv6 inherently changes the operation of PQ, WFQ, or CQ, but whether protocol-level differences remain practically observable when the two IP versions carry the same mixed application traffic under identical scheduler and congestion conditions. The principal research question is: *Under an identical mixed-traffic load, does changing from IPv4 to IPv6 produce a consistent and practically discernible change in QoS, or is performance determined predominantly by queueing discipline and traffic type?*

Three hypotheses guide the evaluation. First, queueing discipline is expected to produce substantial differences in QoS, especially for delay-sensitive voice and video traffic under congestion. Second, when topology, offered load, link capacity, classification rules, and scheduler configuration are held constant, IPv4 and IPv6 are expected to exhibit broadly comparable queueing performance, with protocol-related differences being smaller and less consistent than scheduler-related differences. Third, any observed IPv4–IPv6 difference is expected to depend on traffic type and scheduler configuration rather than constitute a universal advantage of either protocol. The study tests these expectations using PQ, WFQ, and CQ across real-time and non-real-time applications in a controlled OPNET environment.

RELATED WORK

Ensuring reliable QoS in modern IP networks has long been a focal point for researchers. Junaid and Parvez [3] evaluated First-In, First-Out (FIFO), WFQ, and PQ in IPv6 environments, finding that PQ achieved the lowest end-to-end delay but suffered from bandwidth inequity. Alajeeli and Madani [4] compared WFQ, Modified Deficit Round Robin (MDRR), and Class-Based Weighted Fair Queuing (CBWFQ) across IPv4 and IPv6 using OPNET and reported that WFQ yielded smoother jitter for mixed multimedia traffic. Al-Fayyadh [5] confirmed PQ's superiority in minimizing latency while highlighting WFQ's strength in maintaining fairness. Eltaib et al. [6] tested six QoS mechanisms and identified WFQ with Low-Latency Queuing (LLQ) as the best trade-off under congestion. Ibhaze et al. [7] reported slightly higher HTTP/FTP performance under IPv6, attributing this to IPv6 design features, although outcomes vary with topology and configuration. Mresa and Ergheegh [8] investigated the combined effects of queuing algorithms (WFQ, PQ, and FIFO) and routing protocols, namely Routing Information Protocol (RIP), Open Shortest Path First (OSPF), and Enhanced Interior Gateway Routing Protocol (EIGRP), on the QoS of video traffic over IPv4 and IPv6 networks. Recent works by Y. Su et al. [9] and Li et al. [10] explored adaptive and Artificial Intelligence (AI)-driven queueing strategies, reporting 10–20 % improvements in throughput and delay over static approaches. Similarly, Park [11] introduced a triple-metric-based Active Queue Management scheme that classifies traffic according to delay, throughput, and

queue length, effectively emulating Differentiated Services (DiffServ's) per-class treatment for multimedia flows. Rehman et al. [12] observed that IPv6 may exhibit slightly higher delay but greater congestion tolerance in multi-hop scenarios when paired with adaptive WFQ.

Collectively, prior studies show that both protocol version and queue discipline can affect QoS behaviour. However, much of the literature either isolates one factor, focuses on limited applications, or evaluates queueing schemes under non-uniform conditions. In contrast, the present study provides a unified OPNET-based comparison of IPv4 and IPv6 under three queueing disciplines across real-time and non-real-time services, allowing the relative impact of protocol version and scheduler choice to be assessed under the same heavy-load scenario.

The literature therefore provides competing theoretical expectations. IPv6's larger base header can increase per-packet overhead relative to minimum-header IPv4, which may become measurable for small packets or under severe congestion. However, queueing disciplines allocate service according to class, priority, weight, or configured quota rather than IP version itself. Moreover, IPv6-specific fields such as the Flow Label cannot improve QoS automatically; a router must be configured to incorporate them into packet classification or forwarding treatment. Accordingly, a consistent protocol-version effect is not necessarily expected when both protocol scenarios use equivalent traffic classifications and scheduler settings. This motivates evaluating both the main effect of IP version and its interaction with scheduler and application type under controlled conditions.

QUEUEING ALGORITHMS AND QoS PERFORMANCE METRICS

Queueing Techniques (PQ, WFQ, CQ)

Priority Queuing (PQ) serves the highest-priority class first, minimizing delay for real-time traffic but potentially starving lower-priority flows under high load. Weighted Fair Queuing (WFQ) distributes service among flows according to assigned weights, providing differentiated yet fair bandwidth sharing. Custom Queuing (CQ) serves classes cyclically using fixed quotas, enabling predictable bandwidth allocation but with possible inefficiency when some queues are empty. Overall, the selected scheduler directly impacts delay, jitter, loss, and bandwidth distribution for mixed traffic.

Performance Metrics (QoS)

QoS in this study is evaluated using end-to-end delay, packet delay variation, throughput, and application-level response times. These metrics were selected because they directly reflect the performance perceived by both real-time and non-real-time applications under different queueing disciplines and IP versions. End-to-end delay and packet delay variation are particularly important for voice and video traffic, while throughput and response time provide useful indicators of performance for data-oriented services such as HTTP, FTP, email, and database applications.

NETWORK TOPOLOGY AND SIMULATION SETUP

Simulations were conducted in OPNET Modeler 14.5A using the topology shown in Figure (1), which consists of six interconnected IP routers linked through Point-to-Point Protocol PPP DS1 connections (1.544 Mbps) and configured to carry heterogeneous traffic including HTTP, FTP, email, database, voice, and video conferencing applications.

Three queueing disciplines—PQ, WFQ, and CQ—were evaluated under both IPv4 and IPv6, with each scenario executed for 15 minutes under identical heavy-load conditions using the predefined application profiles listed in Table (1). To ensure a controlled comparison, the topology, offered traffic, application profiles, and general simulator conditions were kept constant across all scenarios, while only the protocol version and queueing discipline were varied.

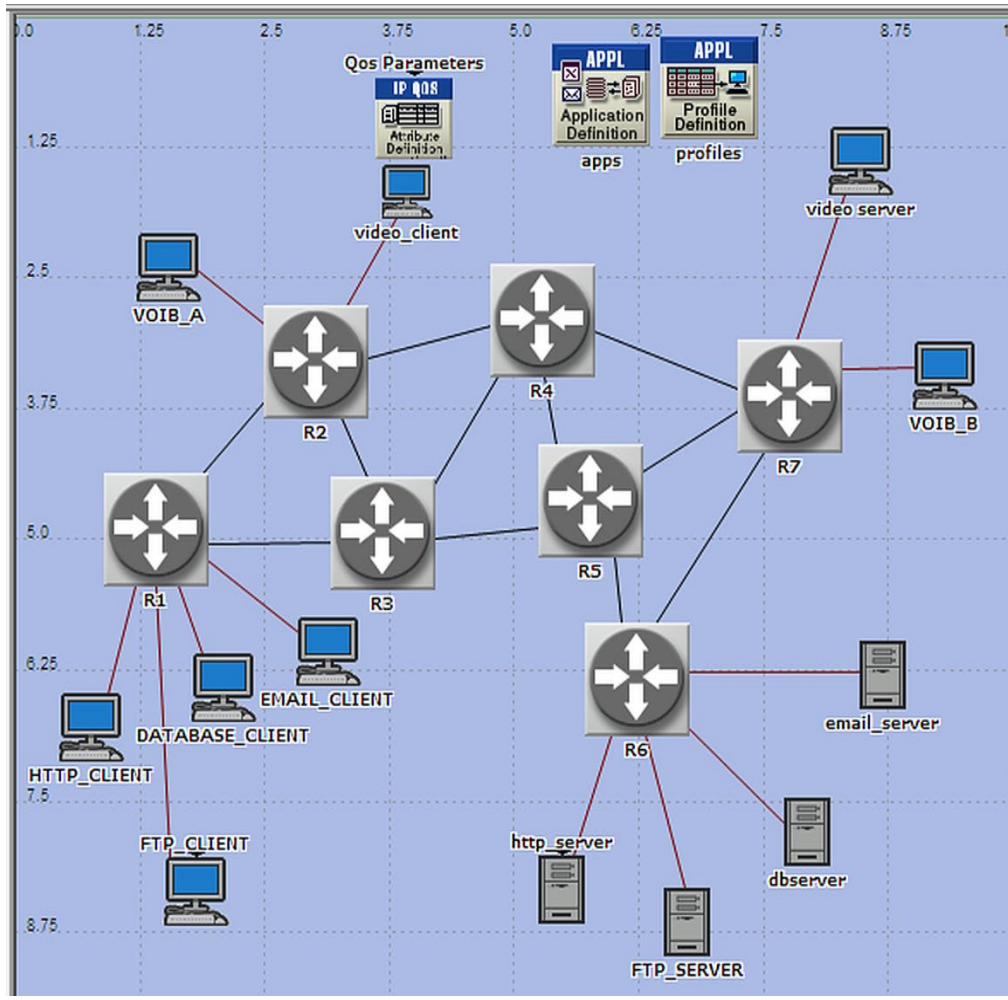


Figure 1: Network Topology Case Study.

Table 1: Traffic Types Configuration.

Application	Type of Traffic
Email	Heavy load
HTTP	HTTP heavy browsing
Video Conferencing	High-Resolution Video128*240 PIXELS, 15 FRAMES/SEC
Voice over Internet Protocol (VoIP)	Pulse-Code Modulation (PCM) Quality Speech Encoder G.711
Database	Heavy load
FTP	Heavy load

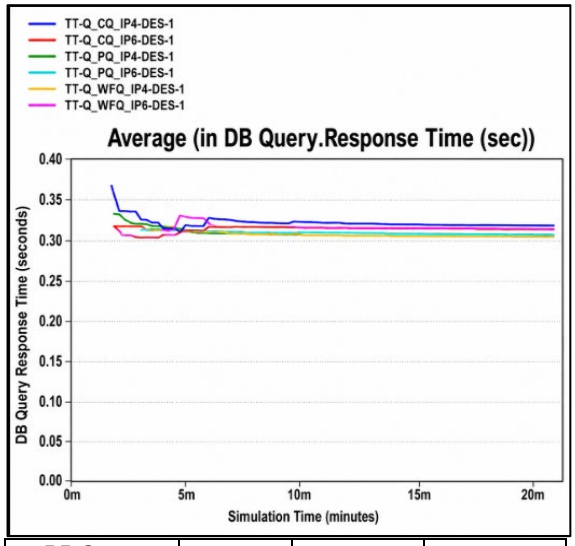
The comparison treats IP version as a protocol-level experimental factor rather than as a modification to the scheduling algorithms. The same application classes and logically equivalent queueing policies were applied in the IPv4 and IPv6 scenarios. No preferential treatment based solely on IP version was configured, and the IPv6 Flow Label was not used to assign an additional scheduling advantage. Therefore, any IPv4–IPv6 differences observed in the simulations may arise from modeled header, packet-processing, or traffic-interaction effects, while larger differences among PQ, WFQ, and CQ represent scheduler-service effects. On a 1.544 Mbps link, the 20-octet difference between the minimum IPv4 header and the IPv6 base header corresponds to approximately 0.104 ms of additional serialization per packet, before considering extension headers or fragmentation. This provides a possible mechanism for small protocol-related differences but is unlikely, by itself, to dominate the effects of persistent queue accumulation under heavy contention. Traffic was classified according to application type, with real-time multimedia traffic receiving higher service priority than lower-priority background traffic. Results are reported as time-averaged values over the simulation duration.

In addition, because the same topology, link capacities, traffic mix, and application profiles were maintained across all simulation runs, the study isolates the effects of queueing discipline and IP version under a common congestion condition. This allows the observed differences in delay, delay variation, throughput, and response time to be linked mainly to scheduler behaviour and protocol characteristics rather than to changes in network structure or traffic load. Consequently, the comparison provides a clearer basis for interpreting the relative influence of IPv4/IPv6 and the selected queueing mechanism.

SIMULATION RESULTS AND ANALYSIS

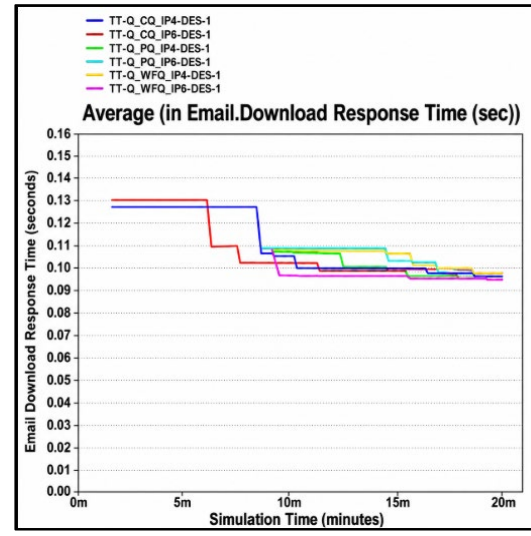
Non-Real-Time Applications (DB, Email, FTP, HTTP)

Figures (2–8) show that the non-real-time results indicate modest differences between IPv4 and IPv6 across PQ, CQ, and WFQ. For database and email traffic, response times remained close across all scenarios, with only minor scheduler-dependent variations and no consistent protocol-level advantage. A similar pattern was observed for FTP and HTTP traffic: although some individual cases favoured IPv4 and others favoured IPv6, the differences were generally small and inconsistent across schedulers. Overall, these results indicate that for non-real-time applications in the evaluated topology, queueing discipline influenced performance more than protocol version, while IPv4 and IPv6 behaved broadly similarly under heavy load.



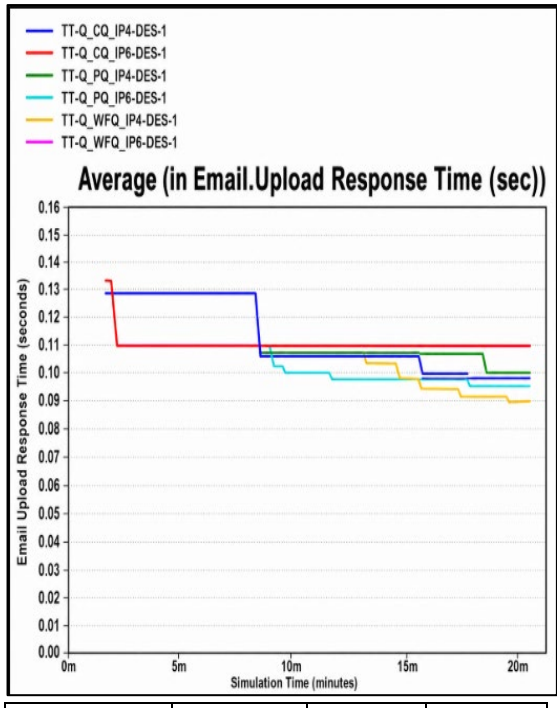
DB Query Response time	PQ	CQ	WFQ
IPv4	0.313455	0.318123	0.314649
IPv6	0.319594	0.317417	0.318763

Figure 2: DB Query response Time.



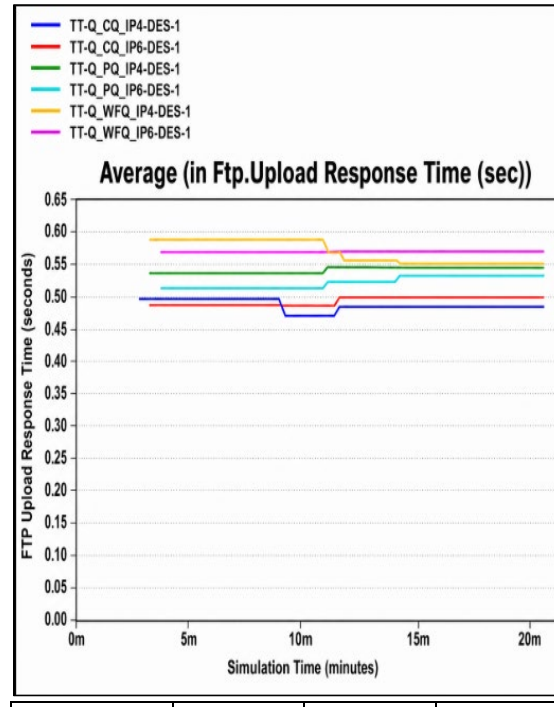
Email Download Response time	PQ	CQ	WFQ
IPv4	0.094563	0.094563	0.094563
IPv6	0.095268	0.095837	0.095268

Figure 3: Email download response Time.



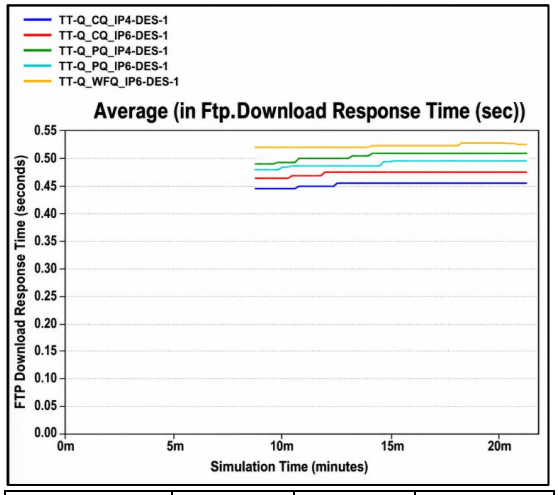
Email Upload Response time	PQ	CQ	WFQ
IPv4	0.0989347	0.0989348	0.0905167
IPv6	0.0961963	0.1096409	0.1096410

Figure 4: Email Upload response time.



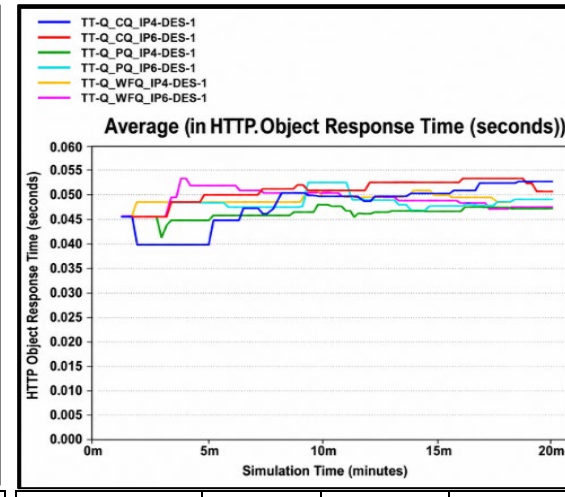
FTP Upload Response time	PQ	CQ	WFQ
IPv4	0.5063213	0.4829537	0.4689161
IPv6	0.4665779	0.4667680	0.4665778

Figure 5: FTP Upload response time.



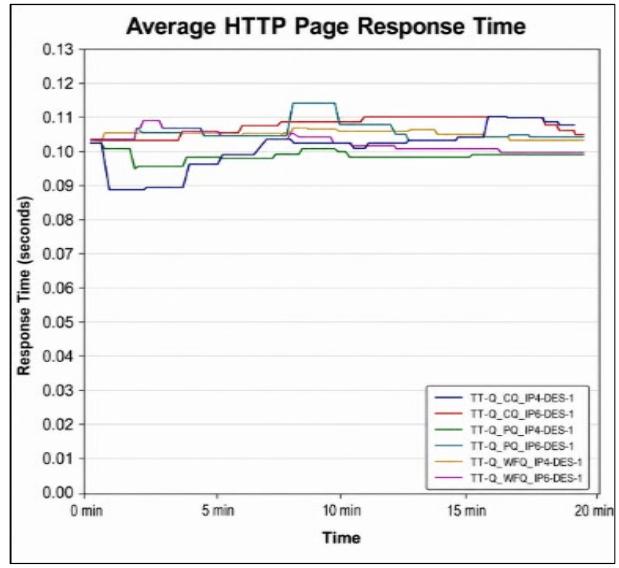
FTP Download Response time	PQ	CQ	WFQ
IPv4	0.4680010	0.4554991	0.4649005
IPv6	0.4654985	0.4654985	0.4649005

Figure 6: FTP Download response time.



HTTP Object Response time	PQ	CQ	WFQ
IPv4	0.0480961	0.0524382	0.0494499
IPv6	0.0494708	0.0507542	0.0482355

Figure 7: HTTP Object response time.



HTTP Page Response time	PQ	CQ	WFQ
IPv4	0.1001384	0.1095080	0.1045428
IPv6	0.1048097	0.1061713	0.1005211

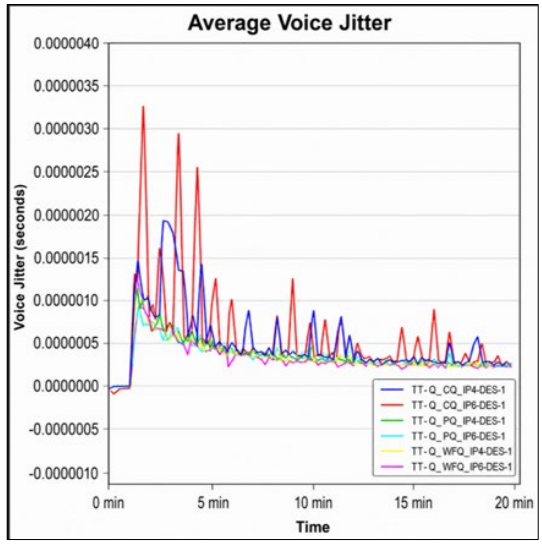
Figure 8: HTTP Page response time.

Real-Time Applications (Voice, Video conferencing)

Voice Traffic

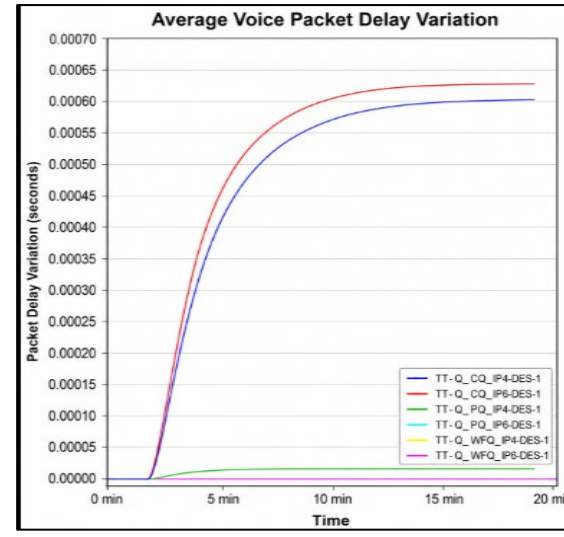
Figures (9–11) present the voice jitter, packet delay variation (PDV), and end-to-end delay obtained under PQ, CQ, and WFQ for both IPv4 and IPv6. The results show that scheduler choice was the dominant factor in voice performance. PQ and WFQ produced almost identical end-to-end delay in both protocol versions, with values around 0.072–

0.073 s, whereas CQ increased delay to approximately 0.102–0.104 s. Voice jitter remained extremely small in all scenarios, although PQ consistently yielded the lowest values. Packet delay variation showed the clearest separation, with CQ producing values roughly 22–23 times higher than PQ and WFQ, while PQ and WFQ remained closely aligned. These results indicate that PQ and WFQ better preserved both delay and delay stability for voice traffic in the evaluated topology, whereas CQ introduced a clear degradation in performance.



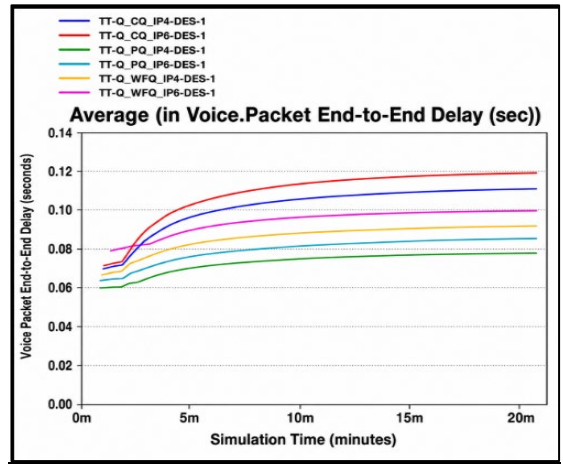
Voice Jitter	PQ	CQ	WFQ
IPv4	0.000000084	0.000000106	0.000000105
IPv6	0.000000099	0.000000110	0.000000116

Figure 9: VOICE Jitter.



Voice Packet Delay variation	PQ	CQ	WFQ
IPv4	0.0000266	0.0006043	0.0000268
IPv6	0.0000270	0.0006270	0.0000278

Figure 10: Voice Packet delay variation.

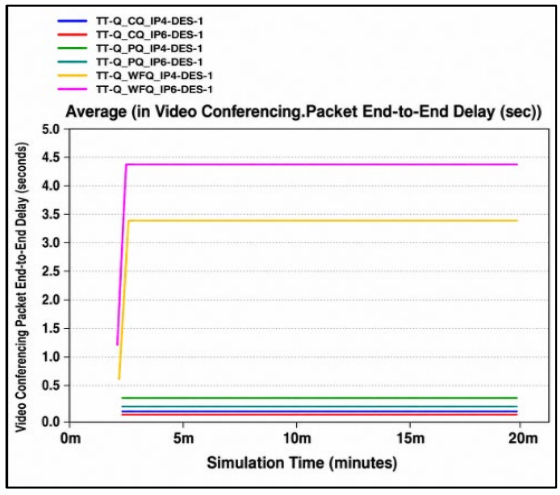


Voice Packet End to End Delay	PQ	CQ	WFQ
IPv4	0.0724232	0.1024506	0.0721849
IPv6	0.0727920	0.1039879	0.0728489

Figure 11: Voice Packet End-to-End delay.

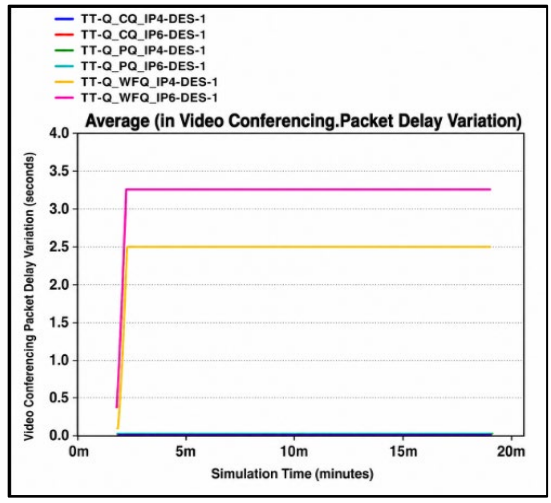
Video Conferencing

Figures (12–13) show a markedly different behaviour for video conferencing traffic. Under PQ and CQ, end-to-end delay remained below 0.32 s, whereas under WFQ it increased sharply to 3.40s for IPv4 and 4.42s for IPv6. Packet delay variation followed the same pattern, rising from low values under PQ and CQ to 2.50 s and 3.25 s under WFQ for IPv4 and IPv6, respectively. Although this observation may appear unexpected, similar findings have been reported in previous studies [5,13]. These results indicate that, under the simulated heavy-load conditions, WFQ placed video traffic in a different operating regime, suggesting persistent queue build-up and severe latency accumulation. By contrast, CQ produced the lowest video delay and packet delay variation in the tested topology, while PQ remained substantially better than WFQ. Therefore, for video conferencing traffic, scheduler behaviour clearly dominated protocol-version effects.



Video Conference End to End Delay	PQ	CQ	WFQ
IPv4	0.3107933	0.1907753	3.3991511
IPv6	0.3194832	0.2042961	4.4212136

Figure 12: Video Packet End-to-End delay.

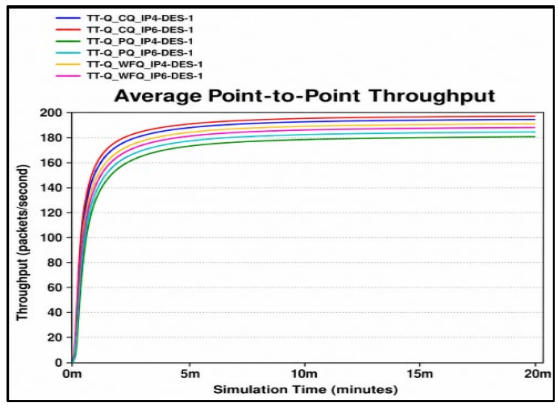


Video Conference Packet Delay variation	PQ	CQ	WFQ
IPv4	0.010859	0.001267	2.500977
IPv6	0.011481	0.001781	3.254557

Figure 13: Video Packet delay variation.

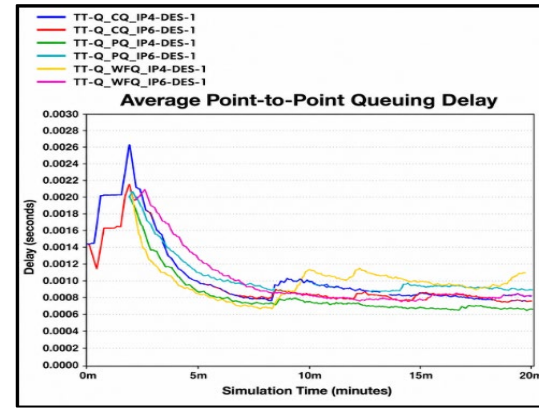
Aggregate Throughput and Queuing Delay

Figures (14–15) compare aggregate point-to-point throughput and queuing delay. IPv4 maintained a small throughput advantage under all three schedulers, with the difference being minor under PQ and CQ and more noticeable under WFQ. However, queuing delay did not favor one protocol uniformly. Under PQ, IPv4 produced the lower delay; under CQ, the difference remained small; and under WFQ, IPv6 produced the lower delay. Therefore, at the aggregate link level, the throughput-delay tradeoff depended more on the selected scheduler than on the IP version itself.



Point to Point Throughput packets/sec	PQ	CQ	WFQ
IPv4	198.061	198.060	197.373
IPv6	197.810	197.800	194.386

Figure 14: Point to Point Throughput (packets/sec).



Point to Point Queuing Delay (sec)	PQ	CQ	WFQ
IPv4	0.000688	0.000768	0.001014
IPv6	0.000874	0.000816	0.000792

Figure 15: Point to Point Queuing Delay.

OVERALL RESULTS

Overall, the results summarized in Table (2) indicate that queueing discipline had a greater impact on QoS than protocol version in the evaluated mixed-traffic environment.

Table 2: Overall Results.

QoS Metric -Traffic Type	PQ		CQ		WFQ	
	IPv4	IPv6	IPv4	IPv6	IPv4	IPv6
End-to-End Delay – Voice (sec)	0.0724	0.0727	0.1024	0.1039	0.0721	0.0728
End-to-End Delay – Video Conf. (sec)	0.3107	0.3194	0.1907	0.2042	3.3991	4.4212
Packet Delay variation – Voice (sec)	0.000026	0.000027	0.000604	0.000627	0.000026	0.000027
Packet Delay variation – Video_Conf. (sec)	0.010859	0.011481	0.001267	0.001781	2.500977	3.254557
Database Query Response Time (sec)	0.313455	0.319594	0.318123	0.317417	0.314649	0.318763
Email Download Response Time (sec)	0.094563	0.095268	0.094563	0.095837	0.094563	0.095268
Email Upload Response Time (sec)	0.098934	0.096196	0.098934	0.109640	0.090516	0.109641
FTP Download Response Time (sec)	0.468001	0.465498	0.455499	0.465498	0.464900	0.464900
FTP Upload Response Time (sec)	0.506321	0.466577	0.482953	0.466768	0.468916	0.466577
HTTP Object Response Time (sec)	0.048096	0.049470	0.052438	0.050754	0.049449	0.048235
HTTP Page Response Time (sec)	0.100138	0.104809	0.109508	0.106171	0.104542	0.100521
Point-to-Point Throughput (Packets/sec)	198.0616	197.8103	198.0608	197.8008	197.3733	194.3866
Point-to-Point Delay (sec)	0.000688	0.000874	0.000768	0.000816	0.001014	0.000792

IPv4 and IPv6 behaved similarly for most non-real-time applications, with only small and inconsistent differences across schedulers. In contrast, real-time traffic was much more sensitive to scheduler behaviour: PQ and WFQ maintained acceptable voice delay, whereas CQ degraded voice delay stability, and WFQ caused severe delay build-up and packet delay variation for video conferencing under heavy load. These findings suggest that QoS performance in mixed-application networks depends primarily on matching the scheduler to the traffic characteristics rather than on protocol migration alone.

CONCLUSION

This study compared IPv4 and IPv6 under PQ, WFQ, and CQ in an OPNET-based mixed-traffic environment comprising voice, video conferencing, HTTP, FTP, email, and database applications. The principal finding is a bounded null result for protocol version: under the evaluated topology, offered load, link capacity, classification rules, and scheduler configurations, changing between IPv4 and IPv6 did not produce a consistent or practically discernible overall QoS advantage. Individual numerical differences occurred, but their direction and magnitude depended on the application and queueing discipline.

This outcome is theoretically expected because PQ, WFQ, and CQ allocate service according to class priority, flow weight, or configured quota rather than IP version. Differences in network-layer header format may produce modest overhead, but they do not alter the fundamental scheduler rules when traffic is classified equivalently. Under congestion, scheduler-induced queue accumulation consequently had a much greater effect than protocol-header differences. This was particularly clear for real-time traffic: PQ and WFQ maintained low voice delay, CQ increased voice delay variation, and WFQ produced severe delay and packet delay variation for video conferencing under the tested configuration.

The results do not establish universal statistical equivalence between IPv4 and IPv6. Rather, they demonstrate that IP version alone was not a decisive QoS factor within the experimental scope. From an architectural perspective, IPv6 migration should therefore not be expected to improve QoS automatically. Network operators should instead preserve and validate classification rules, queue priorities, weights, service quotas, and buffer policies during migration and select the scheduler according to the requirements and load characteristics of the applications being carried.

FUTURE WORK

Future work should evaluate these schedulers under larger topologies, multiple congestion levels, and different traffic mixes to determine whether the observed trends remain stable across a broader range of operating conditions. Additional mechanisms such as CBWFQ, LLQ, traffic shaping, and policing could also be included for wider comparison. Since WFQ produced severe delay build-up for video conferencing in this study, further investigation of queue weights, service allocation, buffer behaviour, and flow interactions are needed to identify the specific cause of this result. Validation through emulation or real-network testing would further strengthen the practical relevance of the findings.

Repeated independent simulation runs should also be conducted using different random seeds, followed by confidence-interval estimation and, where appropriate,

equivalence testing based on a predefined smallest effect size of practical interest. This would distinguish statistical equivalence from the descriptive absence of a consistent protocol-version effect observed in the present study. Future experiments should additionally vary packet-size distributions, link utilization, IPv6 extension-header use, fragmentation conditions, and Flow Label-based classification to identify the conditions under which protocol-level overhead becomes operationally meaningful.

DECLARATION OF CONFLICTING INTERESTS

The authors declare no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

FUNDING

The authors received no financial support for the research, authorship, and/or publication of this article.

DECLARATION OF GENERATIVE AI AND AI-ASSISTED TECHNOLOGIES IN THE WRITING PROCESS

During the preparation of this work, the authors used ChatGPT to improve language and grammatical correctness. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the final content of the publication.

REFERENCES

[1] J. Postel, "Internet Protocol," RFC 791, *Internet Engineering Task Force*, Sept. 1981.

[2] S. Deering and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification," RFC 8200, *Internet Engineering Task Force*, July 2017.

[3] J. L. Shah and J. Parvez, (2014), "Evaluation of queuing algorithms on QoS-sensitive applications in IPv6 network," *Proc. IEEE Int. Conf. on Advances in Computing, Communication, and Informatics (ICACCI)*.

[4] M. Alajeeli and G. A. Madani, (2022), "Comparison of scheduling schemes in IPv4 and IPv6 to achieve high QoS," Proc. 2nd IEEE Int. *Maghreb Meeting Conf. on Sciences and Techniques of Automatic Control and Computer Engineering (MI-STA)*, Sabratha, Libya.

[5] F. K. Al-Fayyadh, (2018), "A comparative study of QoS over IPv4/IPv6 using Opnet-modeler," *Univ. Thi-Qar J. Sci.*, vol. 6, no. 4, pp. 137–144.

[6] M. O. Eltaib, H. H. Alshammari, A. Boukrara, K. Gasmi, and O. Hrizi, (2022), "Choosing the best quality-of-service algorithm using OPNET simulation," *Int. J. Electr. Comput. Eng. (IJECE)*, vol. 12, no. 4, pp. 4079–4089.

[7] E. Ibhaze, O. Okoyeigbo, Y. A. Samson, P. Obba, and I. K. Okakwu, (2020), "Performance evaluation of IPv6 and IPv4 for future technologies," in *Advances in Intelligent Systems and Computing (AISC)*, vol. 1129, Springer, pp. 15–22.

[8] E. Mresa and A. Ergheegh, (2025), "Assessing the effects of queuing algorithms and routing protocols on quality of service for video traffic over IPv4 and IPv6 networks", *Libyan Journal of Science & Technology*, vol.15, no. 2, pp. 1-7, (2025).

[9] Y. Su and C.-M. Huang, (2018), "QRED: A Q-learning-based active queue management scheme," *J. Internet Technol.*, vol. 19, no. 4, pp. 1189–1200.
[10] Z. Li, Y. Hu, L. Tian, and Z. Lv, (2023), "Packet rank-aware active queue management for programmable flow scheduling," *Computer Networks*, vol. 225, 109632, doi: 10.1016/j.comnet.2023.109632.
[11] G. Park, (2022), "QoS implementation with triple-metric-based active queue management," *Electronics*, vol. 12, no. 1, Art. 23, doi:10.3390/electronics12010023.
[12] M. Rehman, F. Ullah, and S. Khan, (2024), "Comparative evaluation of IPv4 and IPv6 QoS for multimedia streams using adaptive queuing," *Wireless Networks*, vol. 30, pp. 1123–1138.
[13] E. Munthali, S. Tembo, and L. Phiri, (2024), "Impact of VoIP codec schemes and queuing techniques on ftp and video conferencing performance", *International Journal of Networks and Communications*, vol. 13, no. 1, pp. 15-22.