

Enhanced Distributed Bandwidth Allocation Algorithm for Network Latency Reduction in IEEE 802.17 Rings

Fahd Alharbi

Electrical Engineering Department-Faculty of Engineering, Rabigh, King Abdulaziz University, Jeddah 21589, Saudi Arabia

Abstract—The IEEE 802.17 Resilient Packet Ring (RPR) technology is a cornerstone for the rapid expansion of metropolitan digital infrastructure. It is recognized for its robustness and spatial reuse capabilities. However, under highly dynamic and asymmetric traffic loads, standard RPR fairness algorithms often suffer from bandwidth oscillations, leading to unpredictable queuing delays and increased network latency. To address this limitation, this study introduces an Enhanced Distributed Bandwidth Allocation (EDBA) algorithm designed specifically for network latency reduction. By incorporating a filtered transit queue length into the fair rate calculation, EDBA dynamically calibrates bandwidth allocation at the source nodes before congestion propagates across the transit paths. Performance evaluations via network simulations demonstrate that the proposed algorithm significantly reduces average packet latency by up to 94%, while maintaining fairness and high throughput across the ring topology.

Keywords—RPR; IEEE 802.17; bandwidth; latency; fairness

I. INTRODUCTION

Smart cities handle the challenges of urban growth through technological integration, thereby enhancing the quality of life, upgrading security measures, and maximizing resource efficiency [1]. Smart environments are experiencing a rapid proliferation of time-critical applications. These include intelligent transportation systems (ITS), surveillance systems, real-time video streaming, grid management, financial trading, and telemedicine [2]. With the rise of real-time applications that necessitate rapid and efficient data transmission, latency is a critical performance metric in optical networks. Fundamentally, latency represents the time delay experienced between data transmission and reception. Within optical networks, this total delay comprises several components, including propagation delay through the fiber, queuing delay, and processing time at individual network nodes [3]. Metropolitan area networks (MANs) widely adopt ring aggregation architectures to ensure high availability and robust fault tolerance [4]. Resilient Packet Ring (RPR), standardized under IEEE 802.17, is a specialized ring topology designed primarily for Metropolitan Area Networks (MANs) and Wide Area Networks (WANs). The defining feature of RPR is its exceptional fault tolerance, allowing uninterrupted data transmission even during severe link failures. This robust resilience has driven its widespread adoption among MAN and WAN service providers, who traditionally rely on legacy systems like Synchronous Optical Network (SONET) or Synchronous Digital Hierarchy (SDH) architectures [5].

The Resilient Packet Ring (RPR) architecture (Fig. 1) employs dual counter-rotating optical rings to ensure robust protection and high bandwidth utilization. Each RPR node includes dedicated buffers for both local and transit traffic (Fig. 2). The primary buffer manages high-quality traffic requiring guaranteed data rates and low jitter, whereas the secondary buffer handles low-quality traffic regulated by a fairness algorithm. The primary buffer has priority to be served first, then the local buffer, as long as the secondary buffer length is less than a threshold of $1/4$ of the buffer size. If the secondary buffer length exceeds the threshold, it will have a priority over the local buffer. The fairness algorithm operates via two distinct mechanisms: the aggressive method, which triggers congestion control when the secondary buffer exceeds a specific threshold, and the conservative method, which detects congestion when the combined local and transit traffic exceeds the available bandwidth for low-priority data [6].

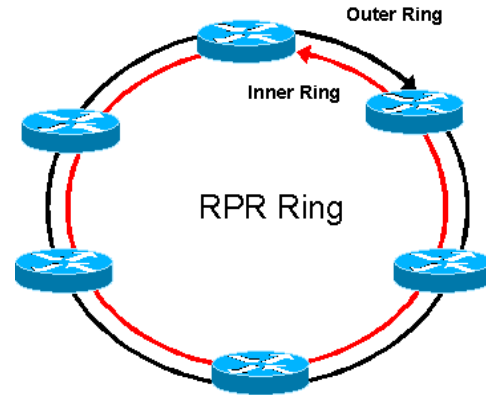


Fig. 1. The packet ring.

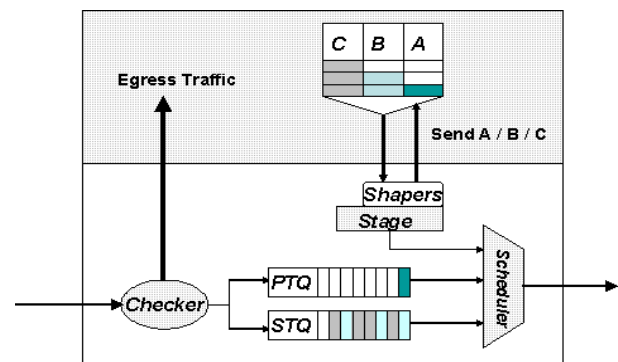


Fig. 2. RPR node architecture.

While RPR preserves the sub-50ms protection of traditional SONET/SDH rings and enhances efficiency through spatial reuse, its standard distributed fairness mechanism degrades under bursty, non-uniform traffic. Intense congestion and bandwidth competition among nodes cause fairness algorithms to oscillate severely. Consequently, this introduces heavy queuing jitter and severe packet latency bottlenecks [7].

To stabilize bandwidth allocation and mitigate these oscillations, several alternative algorithms have been proposed. For instance, the DVSR algorithm [8] monitors network nodes' demands and applies max-min fairness to eliminate oscillations, though it requires a computational complexity of $O(n \log n)$, where n is the number of data flows. Conversely, the Distributed Bandwidth Allocation (DBA) algorithm [9] achieves fairness with a lower complexity of $O(1)$. Control theory has also been applied to optimize fair rate calculations. Researchers have utilized a classical PID controller to stabilize the transit buffer and regulate the bandwidth allocation process [10]. On the other hand, a fuzzy logic controller was designed to monitor the transit buffer, then increase or decrease the fair rate to maintain stability and achieve fairness [11]. Also, a combination of a Smith predictor, P-controller, and fuzzy tuning method was introduced in [6] to handle packet rings with large propagation delays. While these methods regulate bandwidth either by matching injected traffic to available capacity via DVSR and DBA algorithms or stabilizing the low-priority buffer at a target level utilizing PID and fuzzy controllers, transit traffic still accumulates in low-priority buffers during the convergence phase. This accumulation increases queuing delay and exacerbates network latency.

While the expansion of smart city frameworks requires low-latency data processing and optimal resource utilization, traditional bandwidth allocation methods introduce critical bottlenecks [12]. To address these limitations, the Distributed Bandwidth Allocation (DBA) algorithm [9] is leveraged due to its operational efficiency and low computational complexity. The objective is to design an Enhanced Distributed Bandwidth Allocation (EDBA) algorithm. By optimizing the rate-throttling process, EDBA ensures rapid latency reduction without sacrificing the foundational throughput and fairness guarantees of the IEEE 802.17 standard.

II. EDBA ALGORITHM

In this section, we review the Distributed Bandwidth Allocation (DBA) algorithm. To simplify the analysis without loss of generality, we consider a single counter-rotating path: the outer ring comprising N nodes, indexed sequentially along the traffic direction. Let C denote the available capacity allocated for low-priority traffic. The aggregate arrival rate at link k during the n th regulation interval is defined as follows:

$$A(n) = \sum_{i=1}^M R_i \quad (1)$$

where, R_i is the amount of data transmitted from node i and passing through node k , and M is the number of flows. The goal of the fairness algorithm is to maximize the fair rate F_k subject to the following conditions:

$$A \leq C \quad (2)$$

$$R_i \leq F_k, \text{ for all flows going through link } k \quad (3)$$

The algorithm uses the following equation to estimate the fair rate:

$$F_k(n) = F_k(n-1) + [C - A(n)]/M \quad (4)$$

The primary objective of the fair rate estimation equation [Eq. (4)] is to adjust F such that the aggregate arrival rate A matches the available capacity C . Analytical models and experimental results have established that the DBA algorithm converges to the fair rate rapidly [9]. However, as demonstrated in the subsequent section, our simulation experiments reveal that the transit buffer at the congestion point accumulates significantly during this estimation period. Because the DBA algorithm lacks a dedicated queue-draining mechanism, passing packets experience long queuing delays. Consequently, while the DBA algorithm successfully regulates bandwidth allocation, it does so at the expense of increased network latency. This structural limitation reduces the algorithm's ability to guarantee the strict Quality of Service (QoS) requirements of time-critical smart city applications; a deficiency that this work directly addresses.

The objectives of the proposed method are to regulate the fair rate and minimize the packet delay. Accordingly, the buffer length is incorporated in the fair rate calculation equation and is rewritten as follows:

$$F_k(n) = F_k(n-1) + [C - A(n) - \beta Q_f(n)]/M \quad (5)$$

where, Q_f is the filtered version of the low-priority buffer length, and the parameter β controls the speed of clearing the buffer. The filtered buffer length is updated as follows:

$$Q_f(n) = (1 - \alpha)Q_f(n-1) + \alpha Q_i(n) \quad (6)$$

where, Q_i is the instantaneous value of the low-priority buffer length, and the filter coefficient is $\alpha \in [0,1]$ [13].

III. RESULTS AND DISCUSSION

In this section, we conduct two simulation experiments. The first experiment is the simple congestion points scenario shown in Fig. 3, where all links have the same capacity of 622 Mbps, and each link has a propagation delay of 0.1 ms. All data flows are greedy and start at time 0. The measurement time interval is set to $T=1$ ms. The experiment is conducted using the RPR simulator [14].

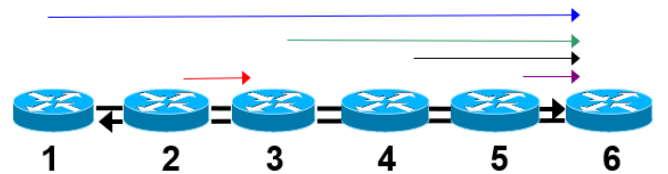


Fig. 3. Simple congestion points scenario.

In this scenario, there are two congestion points. The first congestion point is between switches 5 and 6, where four flows compete for the bandwidth, and a fair share is 155 Mbps. The second point is between switches 2 and 3, where the competing flows are flow (1,6) and flow (2,3). Since flow (1,6) would be limited to the fair share reported by switch 5, flow (2,3) should be able to utilize the remaining bandwidth on link 2-3 and transmit at a rate of 467Mbps. The performance of the

proposed EDBA algorithm is evaluated against two different algorithms. The DBA algorithm is considered as a rate-based algorithm, while the PID controller is considered a queue-based algorithm. The evaluation focuses on the ability to achieve fairness and high resource utilization and also reduce the network latency. The performance of the DBA algorithm, the PID controller, and the proposed EDBA in regulating the bandwidth allocation are shown in Fig. 4, Fig. 5, and Fig. 6, respectively. All algorithms successfully converge very quickly, enabling all flows to achieve their fair shares.

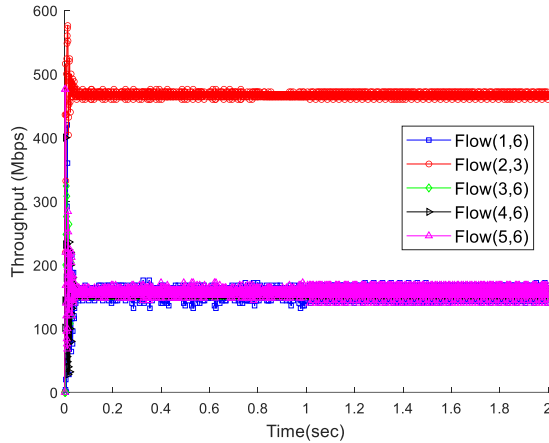


Fig. 4. Simple congestion points scenario: flows convergence (DBA).

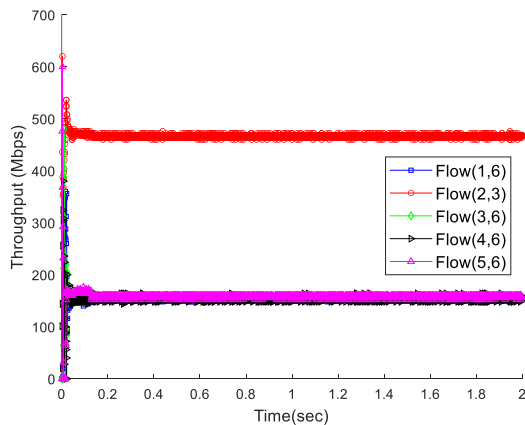


Fig. 5. Simple congestion points scenario: flows convergence (PID).

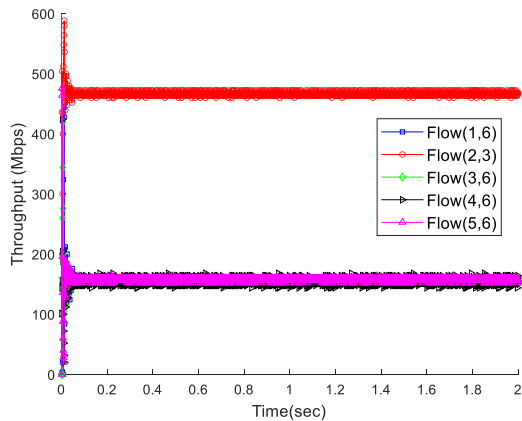


Fig. 6. Simple congestion points scenario: flows convergence (EDBA).

Due to the simultaneous congestion at switches 2 and 5, the low-priority transit buffer occupancy at these nodes accumulates significantly, reaching approximately 25% of the total buffer capacity during the convergence period, as illustrated in Fig. 7. These results demonstrate that the DBA algorithm cannot clear these queues once accumulation occurs.

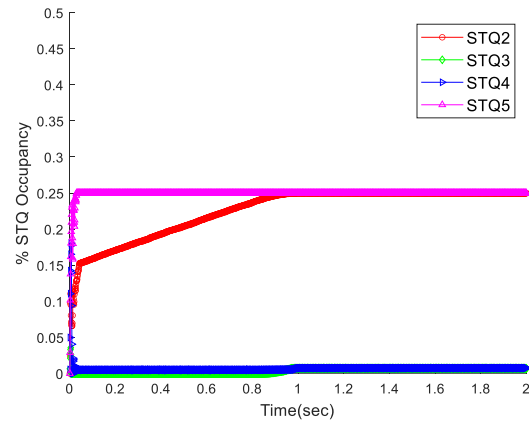


Fig. 7. Simple congestion points scenario: buffer length (DBA).

On the other hand, maintaining a stable queue at a pre-defined target level is a primary objective of the PID controller. Consequently, as illustrated in Fig. 8, the low-priority buffer occupancy at the bottleneck points stabilizes at the target threshold of one-sixteenth (6.25%) of the total buffer capacity. In contrast, the proposed EDBA algorithm successfully empties the accumulated traffic from the low-priority transit buffers, as demonstrated in Fig. 9. By dynamically integrating the filtered buffer occupancy metric Q_f , into the modified fair rate estimation [Eq. (5)], EDBA is capable of clearing the ring transit queues while fully preserving the throughput and fairness mandates of the IEEE 802.17 standard.

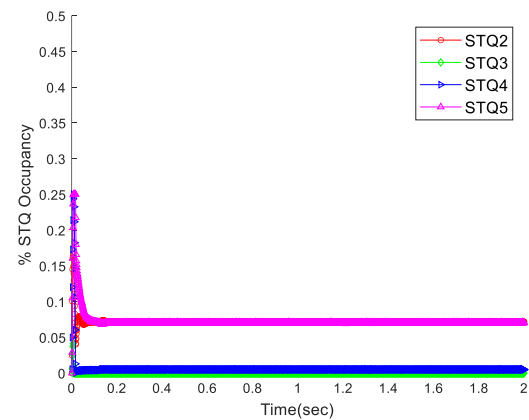


Fig. 8. Simple congestion points scenario: buffer length (PID).

For all algorithms, we compare the delay of packets transmitted from switch 1 to switch 6, where they pass through two congestion points. Fig. 10 illustrates the transient and packet network latency curves for the three evaluated algorithms. As expected from the buffer occupancy behaviors, the DBA algorithm exhibits a severe, continuous growth in network delay during the transient convergence period, ultimately saturating at a high steady-state latency of

approximately 18.124 ms. This rapid, continuous increase confirms the accumulation of low-priority traffic within the transit buffers; without a dedicated queue-clearing mechanism, packets are subjected to persistent, cumulative queuing delays.

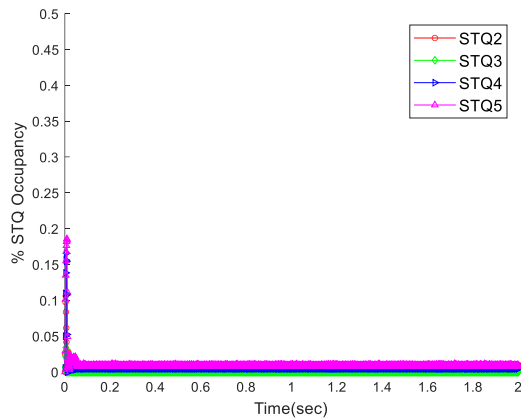


Fig. 9. Simple congestion points scenario: buffer length (EDBA).

In contrast, the queue-based PID controller successfully stabilizes the latency curve, reducing the network delay to roughly 5.383 ms. Because the PID algorithm regulates the transit buffer to a specific target length, it prevents the unbounded delay increase observed in the DBA algorithm. However, it still maintains a permanent, non-zero queuing overhead to enforce its operating setpoint.

Remarkably, the proposed EDBA algorithm outperforms both the DBA algorithm and the PID controller. The EDBA network latency curve drops quickly and stabilizes at approximately 1.104 ms. By emptying the low-priority transit queues using the filtered buffer length variable, Q_f , EDBA almost eliminates queuing delays. This represents an approximate 94% latency reduction compared to DBA and a 79% reduction compared to the PID controller, demonstrating EDBA's capability to sustain the strict low-latency requirements demanded by time-critical smart city applications.

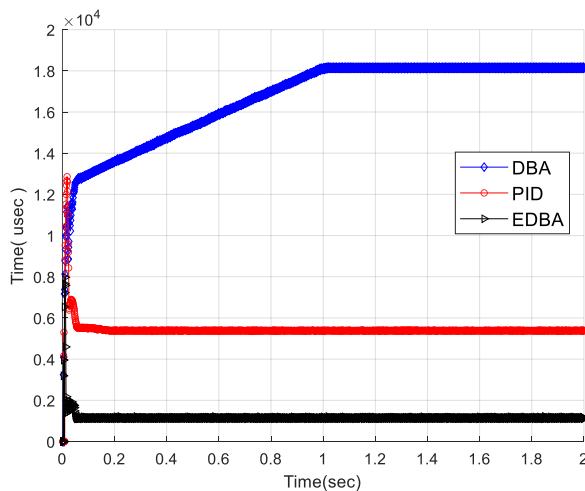


Fig. 10. The transient and packet network latency curves.

In the second experiment (Fig. 11), all flows are greedy and start at time 0. In this scenario, data packets transmitted from Switch 1 to Switch 6 pass through multiple congestion points. The fair-share bandwidth at each congestion point is 311 Mbps because each point is shared by only two flows.

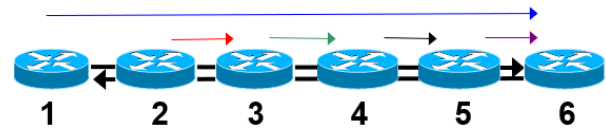


Fig. 11. Multiple congestion points scenario.

Similarly, the DBA algorithm, PID controller, and proposed EDBA algorithm successfully regulate the bandwidth allocation. As depicted in Fig. 12–14, each flow consistently achieves a bandwidth of 311 Mbps.

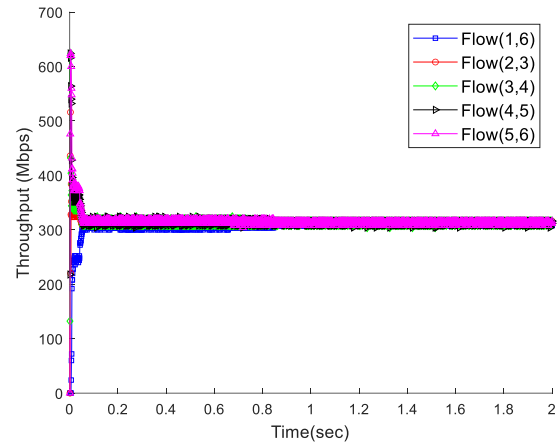


Fig. 12. Multiple congestion points scenario: flows convergence (DBA).

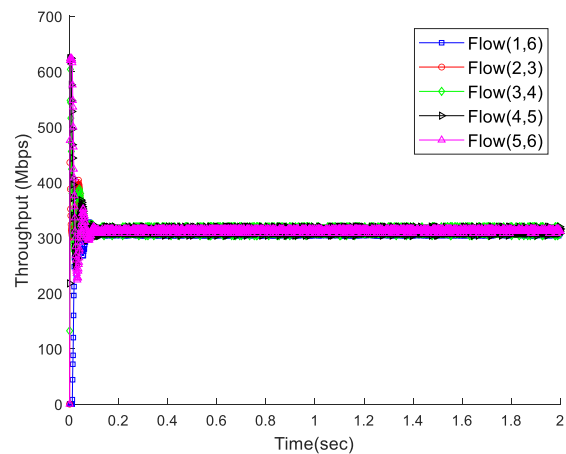


Fig. 13. Multiple congestion points scenario: flows convergence (PID).

This scenario introduces four simultaneous congestion points at switches 2 through 5. As illustrated in Fig. 15, the low-priority transit buffer occupancy at these nodes builds up rapidly, reaching roughly 25% of buffer capacity during convergence. This highlights a key limitation: the DBA algorithm lacks the capability to drain these queues once accumulation takes place.

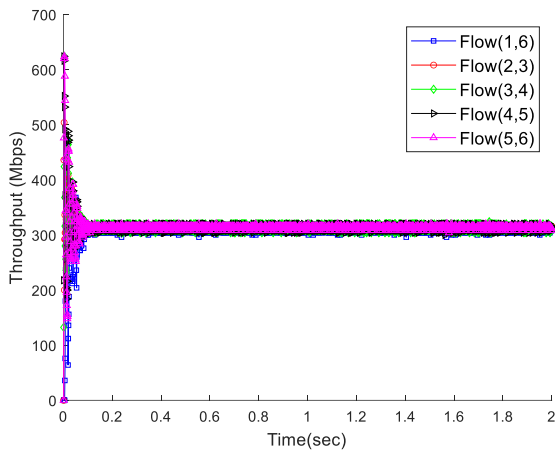


Fig. 14. Multiple congestion points scenario: flows convergence (EDBA).

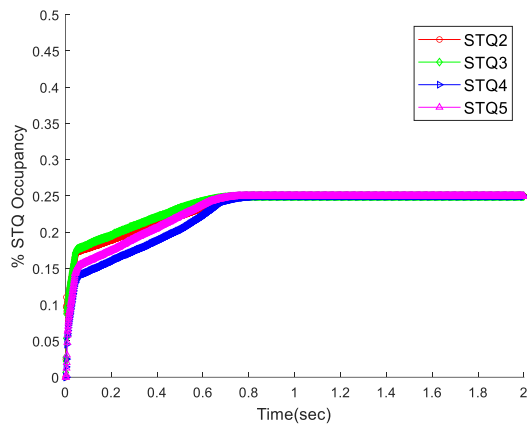


Fig. 15. Multiple congestion points scenario: buffer length (DBA).

While the PID controller focuses primarily on stabilizing the queue at a predefined target level, thereby holding the low-priority buffer occupancy at the bottleneck points to 6.25% of the buffer capacity (Fig. 16), the proposed EDBA algorithm successfully evacuates the accumulated traffic (Fig. 17). This is achieved by dynamically integrating the filtered buffer occupancy measurement Q_f , into the fair rate estimation equation [Eq. (5)]. Consequently, EDBA clears the ring transit queues without compromising the throughput and fairness mandates of the IEEE 802.17 standard.

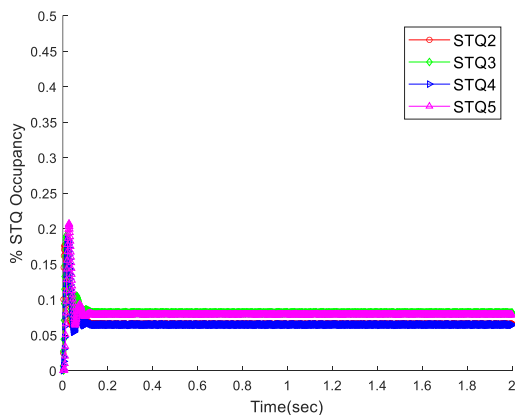


Fig. 16. Multiple congestion points scenario: buffer length (PID).

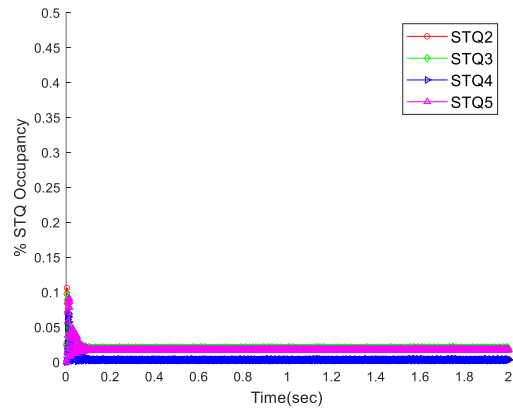


Fig. 17. Multiple congestion points scenario: buffer length (EDBA)

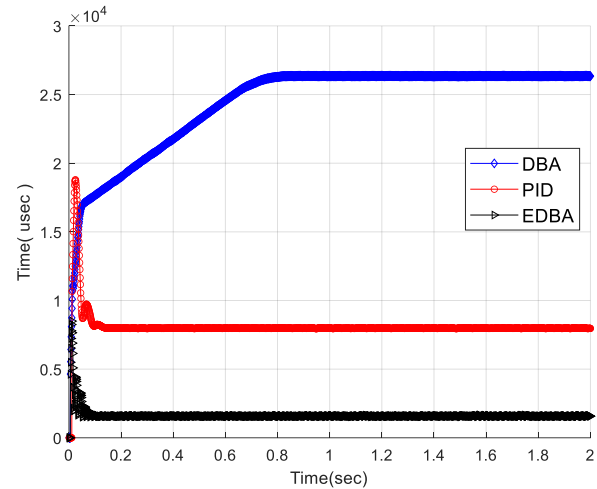


Fig. 18. Flow (1,6) packet delay.

Fig. 18 illustrates the network latency performance for the three evaluated algorithms under the multi-congestion scenario, specifically showing the delay of Flow (1,6) packets transmitted from switch 1 to switch 6. Because these data packets pass through multiple concurrent traffic bottlenecks, the DBA algorithm exhibits a continuous increase in latency during the transient phase, reaching a delay of approximately 26.34 ms. Furthermore, since the rate-based DBA algorithm lacks a queue-draining mechanism, the packet accumulation that builds up in the transit buffers during the fair-rate estimation period exposes all passing packets to high latency.

The queue-based PID controller successfully mitigates network latency growth. At the beginning of congestion, the PID controller experiences a sharp, transient overshoot; however, it rapidly reduces the latency within a few milliseconds, settling at a delay of approximately 7.965 ms. While effective at reducing network delay, the PID controller must maintain its transit queues at a non-zero setpoint to operate, thereby leaving a permanent queuing overhead in the network path

The proposed EDBA algorithm outperforms both approaches. Upon the initialization of concurrent flows, EDBA effectively limits the transient delay spike. Driven by the filtered buffer occupancy metric, Q_f , the rate-throttling

mechanism immediately drains the transient queue accumulation. Consequently, the latency curve decays sharply and stabilizes rapidly, maintaining a remarkably low delay of approximately 1.571 ms.

This optimized behavior translates to substantial performance improvements for delay-sensitive network infrastructures: a 94.0% latency reduction compared to the rate-based DBA algorithm and an 80.0% latency reduction compared to the queue-stabilizing PID controller. By completely evacuating the transient queues rather than containing them, the EDBA algorithm proves highly effective at sustaining the strict, low-latency Quality of Service (QoS) required by next-generation smart city applications.

IV. CONCLUSION

This study presented the Enhanced Dynamic Bandwidth Allocation (EDBA) algorithm, a novel queue-aware framework designed to mitigate transient congestion and optimize latency performance in IEEE 802.17 Resilient Packet Ring (RPR) networks. Experimental evaluations under concurrent multi-congestion scenarios demonstrated that while the conventional rate-based DBA algorithm fails to drain accumulated buffers and the PID controller incurs a permanent queuing overhead, the proposed EDBA algorithm completely evacuates transient queues. Specifically, EDBA maintains a stable, low steady-state delay of approximately 1.571 ms, yielding a 94.0% latency reduction relative to the DBA algorithm and an 80.0% reduction compared to the PID controller. By successfully balancing transient queue evacuation with the strict throughput and fairness mandates of the IEEE 802.17 standard, EDBA proves to be a highly effective solution for sustaining the low-latency Quality of Service (QoS) required by next-generation smart city applications.

While the proposed EDBA algorithm achieves exceptional performance using a static filtering configuration, adaptive parameter tuning remains an avenue for future exploration. Subsequent research will focus on developing an adaptive tuning mechanism for the buffer clearing parameter, β , and the buffer filter coefficient, α . Dynamically adjusting these parameters based on real-time network dynamics will allow the algorithm to maintain optimal responsiveness across varying traffic intensities.

REFERENCES

- [1] Z. E. Ahmed, A. A. Hashim, R. A. Saeed and M. M. Saeed, "Enhancing Smart City Mobility Using Software Defined Networks," *9th International Conference on Mechatronics Engineering (ICOM)*, Kuala Lumpur, Malaysia, 2024, pp. 299-303.
- [2] Patel, C. M., "Edge Computing for Low-Latency IoT Applications in Smart Cities," *Smart Internet of Things*, 1(4), 2024, pp.282-288.
- [3] "Latency in Optical Networks:How It Impacts Real-Time Applications," *Patsnap.com*,2025. <https://eureka.patsnap.com/article/latency-in-optical-networks-how-it-impacts-real-time-applications> (accessed Aug. 14, 2026).
- [4] Y. Nakayama and N. Oota, "N Rate N+1 Color Marking: Per-Flow Fairness in Ring Aggregation Networks," in *IEEE Transactions on Communications*, vol. 62, no. 12, pp. 4401-4412, Dec. 2014
- [5] Minhas T, Khan S, Arslan F, Ali A, Hussain A, Ali J. A Versatile Resilience Packet Ring Protocol Model for Homogeneous Networks. *Applied Sciences*. 2023; 13(8):4660.
- [6] Fahd Alharbi, "Adaptive Predictive Proportional Controller for IEEE 802.17," *IEEE Access*, vol. 8, pp. 23082-23089, 2020.
- [7] M. M. Iqbal, M. D. Awan, M. Sikander, R. A. Bakar, "VRPR: A New Data Center Protocol for Enhanced Network Performance, Resilience and Recovery," *IEEE Access*, vol. 12, pp. 130524–130545, 2024.
- [8] V. Gambiroza, P. Yuan, L. Balzano, Y. Liu, S. Sheafor, E. Knightly, "Design, Analysis, and Implementation of DVSR: A Fair, High Performance Protocol for Packet Rings," *IEEE/ACM Transactions on Networking*, 12: pp. 85-102, 2024.
- [9] Fahd Alharbi and Nirwan Ansari, "Distributed bandwidth allocation for resilient packet ring networks," *Computer Networks*, vol. 49, pp. 161-171, 2005.
- [10] Tan, Liansheng et al. "PID-RPR: A High Performance Bandwidth Allocation Approach for RPR Networks." *IEICE Trans. Commun.* 88-B (2005): 2872-2878.
- [11] W. -S. Tang, W. -C. Wang, C. -J. Chang, Y. -T. Hsieh, "An effective fair bandwidth allocation in resilient packet ring," *5th International ICST Conference on Communications and Networking in China*, Beijing, China, pp. 1-5, 2010.
- [12] R. Fredyan, L. H. Pramono, and . Khairunnisa, "Adaptive Dynamic Bandwidth Allocation in Smart Cities Using Software-Defined Networking", *Eng. Technol. Appl. Sci. Res.*, vol. 15, no. 4, pp. 24957–24963, Aug. 2025.
- [13] Jiang, F.C., Feng, C.W., Zhu, C., Sun, Y., " Performance analysis of active queue management algorithm based on reinforcement learning,". *Journal Européen des Systèmes Automatisés*, Vol. 53, No. 5, (2020), pp. 637-644.
- [14] S. Gjessing, "The Simula RPR Simulator implemented in Java," *Simula Research Laboratory Technical Report* 2003-12.