aDapT-XR: Adaptive Data Allocation and Prioritization for Synchronizing Real and Virtual Worlds in XR Digital Twins

Matis Picoreau
School of Electronic Engineering
Dublin City University
Dublin, Ireland
matis.picoreau2@mail.dcu.ie

Anderson Augusto Simiscuka
School of Electronic Engineering
Dublin City University
Dublin, Ireland
andersonaugusto.simiscuka@dcu.ie

Gabriel-Miro Muntean
School of Electronic Engineering
Dublin City University
Dublin, Ireland
gabriel.muntean@dcu.ie

***Abstract*—Digital Twins (DT) and Extended Reality (XR) technologies are leading the way in industry transformation by bridging the real and virtual worlds and enabling real-time user-digital system interaction and simulation. Despite their promise, XR-enabled DTs still face considerable difficulties in ensuring smooth synchronization between the actual and virtual worlds, especially in network-constrained scenarios. This paper addresses these challenges by introducing aDapT-XR (Adaptive Data Allocation and Prioritization Technique for DTs in eXtended Reality), a novel network-aware data prioritization framework designed for heterogeneous data flows in real-time XR environments. aDapT-XR dynamically adjusts data transmission priorities according to network conditions, reducing latency and improving quality of service (QoS) levels. This work addresses the main limitations of existing methodologies and contributes to the advancement of XR-DT system design by offering practical solutions for live shows, industrial monitoring, and remote collaboration.**

***Index Terms*—Digital Twins, Extended Reality, QoS, Data Allocation, Prioritization**

I. Introduction

Digital Twins (DT), virtual representations of physical systems, are now transforming industries by enabling higher levels of efficiency and innovation. DTs have an immense economic potential and estimates vary between $1.5 trillion and $2.6 trillion in annual economic benefits by 2030 [1]. This growth is notably driven by advances in artificial intelligence and increasing connectivity of Internet of Things (IoT) devices. In fact, DTs usage is expected to increase rapidly over time, as a large percentage of industrial companies are reportedly planning to deploy them within the next few years to optimize their supply chain and improve their product design and development processes [2]. These developments are in line with the increasingly expanding demand for smart systems, from manufacturing plants to city infrastructure management.

The rapid rise of Extended Reality (XR) technologies, comprising Virtual Reality, Augmented Reality, and Mixed Reality, is also reshaping a range of other sectors, including education [3], entertainment [4], healthcare [5], and industry [6]. XR is putting the user experience at the center, by blending the physical and digital worlds. By combining XR with DTs, real-time simulations and interactions are now achievable, enhancing XR applications in fields such as remote collaboration and live performances [7].

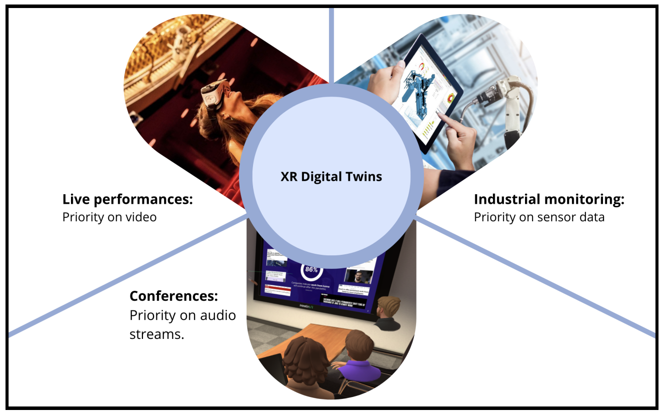

Fig. 1. XR Digital Twin Applications

However, an important challenge in XR-DTs, is to achieve seamless synchronization between the real and virtual world, particularly when real time interactions require low latency and efficient data management over constrained networks [8]. These performance aspects have been studied in existing research focusing on XR performance and resource management within mobile edge networks [9].

Specifically, this paper introduces aDapT-XR, a Network-Aware Data Prioritization framework suitable for coordinating real and virtual environments in XR-DTs, while demonstrating versatility across different scenarios, illustrated in Fig. 1. aDapT-XR employs a prioritization algorithm to adapt the transmission of heterogeneous data streams based on various factors, such as scenario requirements and network congestion, ensuring low latency while maintaining high QoS. The main contribution of our work is to provide dynamic resource management adapted to the needs of different XR-DT environments. Depending on the scenario, the algorithm prioritizes different types of data, such as video for live performances, sensor data for industrial monitoring, or audio streams for multilingual conferences.

The remaining sections of this paper are organized as follows. Section II reviews related works, providing context for the proposed approach. Section III introduces the proposed architecture, followed by Section IV, which details the design of the aDapT-XR algorithm. Section V discusses the experimental settings and evaluations. Finally, Section VI concludes the paper and outlines directions for future work.

II. Related Works

The related works discussed in this section provide context on the integration of DTs into XR systems, and explore communication protocols, resource management strategies, and data prioritization techniques crucial for synchronizing real and virtual environments within XR-DTs. These research efforts help situate the contributions of our proposed approach.

A. XR Digital Twin Integration and Use Cases

Recent research on the combined application of XR and DT technologies has received significant attention for improving system performance across diverse domains, including industrial settings and smart cities, often contributing to the vision of the “Industrial Metaverse” [10]. For instance, the integration of XR and DT to overcome resource constraints in industrial environments has been explored in [11], where emerging technologies such as AI, edge computing, and blockchain are employed to improve operational efficiency. Another pertinent work explores XR and DT within industrial control systems, concentrating on real-time control and hardware-in-the-loop experiments for SCADA applications [12]. This study demonstrates the utility of XR and DT in improving operator training and decision-making processes, even though it acknowledges the need for more intuitive and effective user interfaces.

In the domain of power systems, a systematic literature review [13] surveys the applications of DT, XR, and the Metaverse to improve key areas such as energy market management, personnel training, and system maintenance. This research discusses the advantages and limitations associated with each technology, offering recommendations for future inter-disciplinary study, while highlighting current limitations like the high cost of XR hardware and platform compatibility issues. The authors of [14] demonstrate the application of DT and XR technologies in fields such as healthcare and education, emphasizing their ability to facilitate real-time simulations that enhance decision-making and interactive learning. Despite showcasing better user engagement, the study highlights ongoing challenges in achieving high VR realism and reducing motion sickness.

Finally, the application of XR/DT in the context of smart cities is explored in [15], where these technologies are employed to streamline city operations through real-time virtual-physical interactions. This work proposes a novel approach to improve urban infrastructure management by enhancing visualization capabilities and data integration, offering a model for more efficient resource utilization and improved quality of life. However, it also highlights persistent challenges related to hardware limitations and the need for better interoperability, which can complicate the implementation of these technologies in practical, large-scale real-world scenarios.

B. Communication, Resource Management, and Prioritization

Effective communication protocols and robust resource management strategies are fundamental to minimize network delay and ensure efficient and reliable operation of DT systems, particularly when integrated with latency-sensitive XR interfaces.

Several communication protocols have been evaluated for DT systems. A comparison of MQTT, HTTP, and WebSocket for industrial automation, detailed in [16], suggests that MQTT and WebSocket protocols are generally more suitable for real-time applications due to their characteristics such as lower latency and higher throughput. Beyond these, established real-time protocols like RTP and newer transport layer ones such as QUIC offer potential advantages for streaming applications within XR-DTs. For managing potentially large and complex data flows between DT components, middleware platforms such as Apache Kafka are sometimes considered, offering robustness and scalability, although their resource footprint might be significant for edge-driven XR scenarios.

Research has actively focused on minimizing latency and efficiently managing resources. RTT-aware packet delivery prioritization algorithms over MPTCP are proposed in [17], [18] specifically to minimize latency in AR/VR applications by selecting the best transmission subflow for critical data. These approaches demonstrate significant performance improvements in simulations. In [19], the authors discuss the use of DTs in the context of 6G wireless networks and, in particular, how the available resources could be optimized using edge computing. This approach aims to minimize end-to-end latency and enhance user experience, though the computational demands of sophisticated DT models for network optimization can be considerable. Dynamic resource orchestration within DT systems, utilizing techniques like bandwidth aggregation and flexible resource management frameworks, is explored in [20]. While improving overall network efficiency and service performance, challenges related to large-scale deployment and seamless integration across edge and cloud systems persist. Furthermore, methods combining Deep Reinforcement Learning and Lyapunov optimization for time-sensitive resource management in Mobile Edge Computing networks are presented in [21], demonstrating enhanced energy efficiency and reduced task completion times, although scalability for very large IoT networks where possible bottlenecks in data transmissions could appear, might remain a concern.

Specific data prioritization and latency reduction mechanisms tailored for immersive environments and DT contexts have also been investigated. The ADAMS framework [22], for example, provides a Quality of Experience (QoE)-oriented approach for multi-sensory content delivery (video, haptic, olfactory), allowing priorities to be configured based on user preferences and network conditions to improve synchronization and quality. Another approach includes a real-time adaptive streaming system for Cloud VR that uses GPU acceleration to dynamically synchronize video streams with user orientation. This method reduces latency and bandwidth consumption, offering a significant improvement in user experience [23]. The integration of Age of Information metrics into resource management for edge-computing-based DT networks with network slicing ensures data freshness while reducing resource consumption and improving QoS [24]. Other studies focus on resource allocation for wireless VR in DT networks using transfer learning with Echo State Networks to handle network characteristic changes and improve reliability [25], or employ machine learning methods to identify and prioritize delay-sensitive traffic, refining routing strategies for AR/VR services [26]. Novel algorithms also target resource allocation and Field-of-View optimization in VR services over mobile edge networks, aiming to minimize resource consumption via adaptive data management [27].

While these studies present significant advances, challenges related to computational complexity, applicability to real-world scale, and hardware dependencies often remain. This paper builds on these foundational works by introducing a novel prioritization algorithm specifically designed for the synchronization needs of XR-enabled DTs. Our approach distinguishes itself by dynamically adapting prioritization based not only on stream characteristics and network conditions (as explored partially in prior work), but also critically on the specific requirements of diverse operational scenarios, ranging from live performances to industrial monitoring and multilingual conferences.

III. Proposed aDapT-XR Architecture

The proposed architecture for aDapT-XR is illustrated in Fig. 2. It outlines a framework for integrating a DT system within a real-time XR environment and consists of two main components: the Physical Asset and the DT.

A. Physical Asset

The Physical Asset encompasses the device control and monitoring systems, which include sensors and IoT devices. The data collected from these devices is processed in the Processing Layer, where tasks such as compression, data capture, and synchronization take place. The system ensures efficient communication through the Communication Layer, utilizing technologies such as Wi-Fi, 5G, and MPTCP. A key innovation in this architecture is the aDapT-XR, a Latency-Aware and Context-Based Prioritization Algorithm. This solution dynamically adjusts data priorities based on network conditions and adapts to diverse scenarios such as live performances, industrial monitoring, or multilingual conferences. By prioritizing critical data streams according to the context and intended network adaptations, aDapT-XR aims to ensure reduced latency and improved QoS levels.

B. Digital Twin

On the DT side, the Cloud Server serves as the central hub, handling data storage, processing, and control. The Real Time Data Processing layer ensures that the DT accurately reflects the current state of the Physical Asset. The Simulation Layer leverages the data to perform predictive analytics and scenario

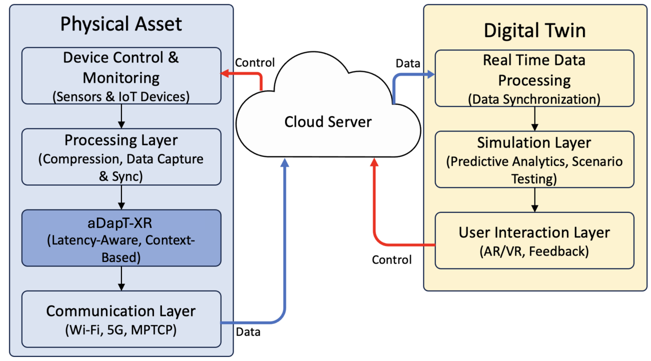

Fig. 2. Prioritization Architecture

testing, optimizing the performance of the Physical Asset. Finally, the User Interaction Layer provides the interface for users to interact with the DT, enabling AR/VR experiences and feedback.

IV. Proposed Algorithm

The proposed aDapT-XR approach dynamically optimizes data transmission in real-time XR environments through a two-stage adaptive process: first adapting to the application scenario, and second adapting to real-time network status.

A. Scenario-Based Weight Adjustment:

The initial importance assigned to each data stream (ω_d) is crucial and context-dependent, directly reflecting the specific XR scenario (S). The algorithm assigns predefined weights based on the stream type according to the scenario's operational needs. For example, in a live performance scenario, video streams are given higher priority, followed by audio and scenography. In a conference scenario, audio and video streams might be prioritized, while in a robot control scenario, control commands and sensor data could take precedence over auxiliary video feeds.

Table I summarizes the key variables and symbols used throughout the description of the aDapT-XR algorithms.

B. Algorithm 1: Initial Priority Calculation and Allocation:

This first algorithmic stage formalizes the calculation of baseline priorities and the initial bandwidth distribution, leveraging the scenario-defined weights ω_d. Priorities (π_d) are calculated for each stream d using the formula:

$$\pi_d = \frac{\omega_d \times S_d}{L_d + \epsilon} \tag{1}$$

where S_d is the data size, L_d is the latency an ϵ a small constant added to prevent division by zero. This formula assigns higher priority scores to streams deemed more important (higher ω_d), those with larger data sizes S_d, or those requiring lower latency (smaller L_d).

These calculated priority scores π_d are then normalized across all streams to determine each stream's proportional weight relative to the total priority sum. Finally, this normalized weight is used to compute the stream's initial bandwidth allocation $\mathcal{B}[d]$ as a fraction of the total available bandwidth B. This initial allocation serves as the starting point for the dynamic adjustment process described in Algorithm 2.

TABLE I
TABLE OF VARIABLES

Variable	Description
Input Parameters & State	
$\mathcal{D}$	Set of data streams (input to Alg. 1)
$\mathcal{N}$	Network state information (contains congestion C, input to Alg.2)
Bw	Total available bandwidth capacity (input to Alg.1)
$\mathcal{S}$	Scenario context (e.g., "conference", input to Alg.1)
τ_{med}	Moderate congestion threshold (parameter for Alg.2)
τ_{high}	High congestion threshold (parameter for Alg.2)
ϵ	Small constant for priority calculation (parameter for Alg.1)
Stream Properties & Derived Weights	
S_d	Characteristic data size for stream d (from $\mathcal{D}$)
L_d	Latency requirement for stream d (from $\mathcal{D}$)
ω_{adj}	Map of scenario-specific base weights per stream type (from $\mathcal{S}$)
ω_d	Effective weight assigned to stream d (from ω_{adj})
Calculated Values	
π_d	Calculated (raw) priority score for stream d (Alg.1)
Π	Map of raw priority scores π_d for all streams (output Alg.1, input Alg.2)
Π_{total}	Sum of all raw priority scores π_d (Alg.1 internal)
$\mathcal{B}$	Map of initial bandwidth allocation per stream (output Alg.1, input Alg.2)
C	Current congestion level measured from $\mathcal{N}$ (used in Alg.2)
$\mathcal{B}_{\text{adj}}$	Map of final adjusted bandwidth allocation per stream (output Alg.2)

Algorithm 1 aDapT-XR: Initial Allocation

Require: Stream set $\mathcal{D}$, Scenario $\mathcal{S}$, Total Bandwidth Bw
Ensure: Priority map Π, Initial Bandwidth Allocation $\mathcal{B}$

```
Π_total ← 0; Π ← {}; B ← {}
ω_adj ← scenario-specific weights from S
for each stream d in D do
    ω_d ← ω_adj[type of d]
    S_d ← D[d]["S"], L_d ← D[d]["L"]
    if L_d = 0 then
        π_d ← ω_d × S_d
    else
        π_d ← (ω_d × S_d)/L_d
    end if
    Π[d] ← π_d; Π_total ← Π_total + π_d
end for
for each stream d in Π do
    Π[d] ← Π[d]/Π_total            ▷ Normalize priority weight
    B[d] ← Π[d] × Bw               ▷ Initial allocation
end for
return Π, B
```

C. Algorithm 2: Congestion-Based Bandwidth Adjustment:

Building upon the initial allocation determined by Algorithm 1, the second stage of the aDapT-XR approach introduces dynamic adaptation based on real-time network conditions. This algorithm adjusts the bandwidth allocated to each stream by monitoring the current network congestion level C and comparing it against predefined thresholds (τ_{med}, τ_{high}).

The core adaptive mechanism operates as follows:

- **High Congestion ($C > \tau_{\text{high}}$):** When the network is heavily congested, the algorithm significantly reduces the allocated bandwidth. This reduction is applied strategically, starting aggressively with streams that have the lowest priority scores π_d (calculated in Algorithm 1) and becoming progressively less severe for higher-priority streams. This aims to preserve resources for the most critical data (e.g., video, essential interactions) by sacrificing bandwidth from less important streams (e.g., secondary background data). The specific reduction logic is detailed in the ApplyReduction procedure.
- **Moderate Congestion ($C \leq \tau_{\text{high}} \& C > \tau_{\text{med}}$):** Under moderate congestion, a similar, but less aggressive bandwidth reduction strategy is used. It still targets lower-priority streams first, but reduces their allocation by a smaller factor compared to the high congestion case.
- **Low Congestion ($C \leq \tau_{\text{med}}$):** If the network is not significantly congested, no adjustments are made. The algorithm maintains the initial proportional bandwidth allocation B derived from Algorithm 1, allowing for a fairer distribution of resources when network capacity is sufficient.

Algorithm 2 aDapT-XR: Congestion Adjustment

Require: Initial Allocation $\mathcal{B}$, Priorities Π, Net State $\mathcal{N}$, $\tau_{\text{med}}, \tau_{\text{high}}$
Ensure: Final Bandwidth Allocation $\mathcal{B}_{\text{adj}}$

```
C ← N["congestion"]
B_adj ← B                          ▷ Start with initial allocation
if C > τ_high then                 ▷ High Congestion: Aggressive Reduction
    rf ← 0.8 ▷ Initial reduction factor Sort streams d by Π[d] ascending (low priority first)
    for each stream d in sorted order do
        if rf ≥ 0.1 then           ▷ Apply if factor above minimum
            B_adj[d] ← B_adj[d] × (1 − rf)     ▷ Reduce BW
            rf ← rf × 0.5 ▷ Decrease factor for next prio
        end if
    end for
else if C > τ_med then ▷ Moderate Congestion: Moderate Reduction
    rf ← 0.4 ▷ Initial reduction factor Sort streams d by Π[d] ascending
    for each stream d in sorted order do
        if rf ≥ 0.05 then          ▷ Apply if factor above minimum
            B_adj[d] ← B_adj[d] × (1 − rf)     ▷ Reduce BW
            rf ← rf × 0.6 ▷ Decrease factor for next prio
        end if
    end for
end if
return B_adj
```

TABLE II
ILLUSTRATIVE WORKLOAD PARAMETERS

Stream	Rate	Size	ω_d	L_d (ms)
Standard Video	20 Mbps	1400 B	0.5	15
High-Priority Audio	3 Mbps	500 B	0.7	50
Low-Priority Data	1.5 Mbps	200 B	0.2	100

Note: Illustrative values for audio priority, $\epsilon = 0.001$.

V. EXPERIMENTAL SETTINGS AND EVALUATION

A. Test Setup

Our evaluation utilizes NS-3 (v3.41), implementing a custom queue discipline ('CustomPriorityQueueDisc') that prioritizes packets according to Eq. (1), as indicated in Algorithm 1[1]. The baseline solution compared against employs a standard Stochastic Fairness Queueing (SFQ)-based queue with an identical buffer size for fair comparison.

The simulated network topology consists of n clients and a single server connected via a central wired Point-to-Point (P2P) link.A P2P setup was chosen to create a controlled environment focused on queueing performance, isolating results from wireless channel effects. This link was configured with bandwidth B (varied, e.g., 100, 45, 30 Mbps) and a 2 ms base delay. Each client node concurrently runs the full set of XR data streams towards the server.

Each client generated XR streams using 'ns3::OnOffApplication' emulating Constant Bit Rate (CBR) sources targeting rates from Table II. Data packets were sent over the User Datagram Protocol (UDP) to server sinks, with packet sizes set according to Table II.

Network congestion was induced by varying the number of concurrent clients n and/or the link bandwidth (B). For results in Figs. 3-5, background traffic was generated by adding competing CBR UDP traffic at varying aggregate rates (between 40 and 70 Mbps) onto the shared link, independent of the main XR streams.

Critically, aDapT-XR's flexibility allows tuning prioritization parameters (ω_d, L_d in Eq. (1)) for different scenarios. The results presented use parameters from Table II to prioritize audio illustratively; these can be easily modified for other streams (e.g., video), but other results are omitted due to space constraints.

System performance was evaluated using the following key metrics: Average End-to-End Packet Delay (ms), Packet Loss Ratio (%), and Average Jitter (ms).

B. Main Results

Our simulation results demonstrate that the proposed prioritization mechanism significantly improves the delivery quality of high-priority streams under congested conditions compared to the baseline SFQ approach. Specifically, we observed substantial reductions in packet loss and delay for the prioritized streams.

[1]Note that this simulation focuses solely on the prioritization aspect and the dynamic bandwidth reallocation described in Algorithm 2 was not tested in this paper.

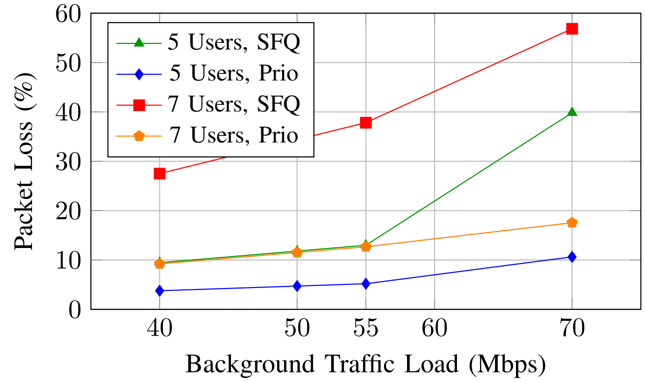

Fig. 3. Audio packet loss comparison vs. background load.

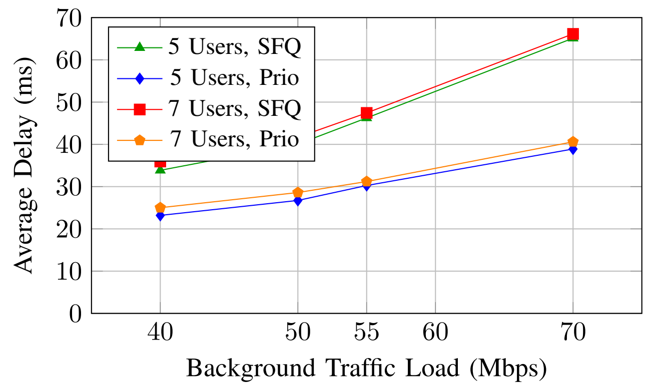

Fig. 4. Average audio delay comparison vs. background load.

Fig. 3 illustrates the audio packet loss ratio against increasing background traffic load. With the prioritization scheme enabled, audio packet loss is considerably lower than that of the baseline SFQ, especially under high contention (7 users). For instance, at a background load of 70 Mbps with 7 users, prioritization reduces the loss from $\sim$57% to 18%.

Similarly, the average end-to-end delay for audio packets, shown in Fig. 4, is consistently lower when using prioritization than when the alternative approach is employed. At high background loads (70 Mbps), the prioritized approach reduces the average audio delay by $\sim$25 ms compared to the values measured when SFQ is used in both 5 and 7 user scenarios.

Figure 5 presents the average audio jitter, revealing a trade-off. Compared to the baseline SFQ's lower jitter, our prioritization scheme introduces significantly higher jitter under high congestion. This increase likely stems from priority scheduling inducing variance in packet inter-arrival times as high-priority packets are served, potentially in bursts. While improved delay and loss (Figs. 3-4) are key benefits, this elevated jitter is a performance cost under heavy load. Future work could explore queue tuning or traffic shaping to mitigate this, though the perceived audio impact depends on application tolerance.

VI. CONCLUSIONS AND FUTURE WORK

In this paper, we presented the aDapT-XR framework, introducing a network-aware data prioritization mechanism for XR-DTs. Our approach dynamically manages heterogeneous data streams by utilizing scenario-specific weights to

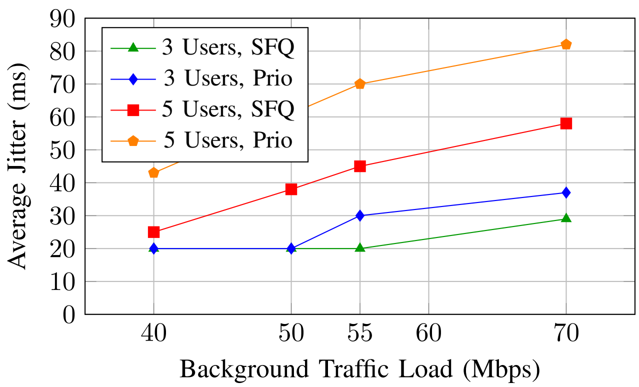

Fig. 5. Average audio jitter comparison vs. background load.

calculate stream priorities. Our simulation results, comparing this prioritization scheme against a baseline SFQ queue under various multi-user load conditions, confirm its effectiveness. We observed significant reductions in both packet loss and average end-to-end delay for prioritized streams, particularly under high network contention. These improvements directly contribute to enhanced QoS for critical XR data flows. It is important to note that these simulations focused on the prioritization stage; the dynamic bandwidth reallocation based on congestion thresholds described in Algorithm 2 was not implemented in this NS-3 model and is left for future work. Future efforts will involve implementing and validating the complete adaptive framework in diverse XR-DT scenarios and real-world testbeds, further investigating its scalability and performance tuning.

Acknowledgment

This work was supported by Research Ireland via grant 12/RC/2289_P2 (INSIGHT), and by the European Union (EU) Horizon Europe via grant 101135637 (HEAT Project).

References

[1] L. M. R. Schuster and C. Penazzo, “Using Digital Twins to Manage Complex Supply Chains,” 2024.

[2] J. D. R. Argolini, F. Bonalumi and S. Pellegrinelli, “Digital Twins: The Key to Smart Product Development,” 2023.

[3] C. H. Muntean, I. Tal, D. Bogusevschi, M. Bratu, T. Bi, and G.-M. Muntean, “Mulseplayer: A multi-sensorial media content delivery solution to enhance end-user quality of experience,” in *IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*, 2024, pp. 1–6.

[4] X. Hao and Y. Wang, “Application of digital twin and meta-universe technology in museum exhibition design,” in *2024 International Conference on Telecommunications and Power Electronics (TELEPE)*, 2024, pp. 262–265.

[5] I. Hussain, M. A. Hossain, and S.-J. Park, “A healthcare digital twin for diagnosis of stroke,” in *2021 IEEE International Conference on Biomedical Engineering, Computer and Information Technology for Health (BECITHCON)*, 2021, pp. 18–21.

[6] F. Pires, A. Cachada, J. Barbosa, A. P. Moreira, and P. Leitão, “Digital twin in industry 4.0: Technologies, applications and challenges,” in *2019 IEEE 17th International Conference on Industrial Informatics (INDIN)*, vol. 1, 2019, pp. 721–726.

[7] M. A. Togou, A. A. Simiscuka, R. Verma, N. E. O’Connor, I. Tamayo, S. Masneri, M. Zorrilla, and G.-M. Muntean, “An innovative adaptive web-based solution for improved remote co-creation and delivery of artistic performances,” *IEEE Transactions on Broadcasting*, vol. 70, no. 2, pp. 719–730, 2024.

[8] A. A. Simiscuka, T. M. Markande, and G.-M. Muntean, “Real-Virtual World Device Synchronization in a Cloud-Enabled Social Virtual Reality IoT Network,” *IEEE Access*, vol. 7, pp. 106 588–106 599, 2019.

[9] E.-S. Lee and B.-S. Shin, “Enhancing the performance of xr environments using fog and cloud computing,” *Applied Sciences*, vol. 13, no. 22, 2023.

[10] X. Yao, N. Ma, J. Zhang, K. Wang, E. Yang, and M. Faccio, “Enhancing wisdom manufacturing as industrial metaverse for industry and society 5.0,” *Journal of Intelligent Manufacturing*, vol. 35, pp. 1–21, 11 2022.

[11] H. M. Kamdjou, D. Baudry, V. Havard, and S. Ouchani, “Resource-constrained extended reality operated with digital twin in industrial internet of things,” *IEEE Open Journal of the Communications Society*, vol. 5, pp. 928–950, 2024.

[12] S. Jeršov and A. Tepljakov, “Digital twins in extended reality for control system applications,” in *2020 43rd International Conference on Telecommunications and Signal Processing (TSP)*, 2020, pp. 274–279.

[13] C. Menezes, B. França, and Y. Lopes, “A survey on extended reality, digital twins, and metaverse applications in power systems,” *IEEE Internet of Things Journal*, vol. 11, no. 21, pp. 34 953–34 977, 2024.

[14] D. Silion, G.-G. Constantinescu, and A. Iftene, “Leveraging digital twin concepts for future applications,” in *International Conf. on INnovations in Intelligent SysTems and Applications (INISTA)*, 2024, pp. 1–5.

[15] S. Gao, “Commercial application of xr and digital twin technology in smart city,” in *2021 3rd International Symposium on Smart and Healthy Cities (ISHC)*, 2021, pp. 84–89.

[16] C. Fongsamut, T. Kanangnanon, B. Jantarakongkul, and P. Jitngernmadan, “Digital twin in automation industry: Optimal communication protocol,” in *2023 7th International Conference on Information Technology (InCIT)*, 2023, pp. 33–37.

[17] F. Silva, D. Bogusevschi, and G.-M. Muntean, “A mptcp-based rtt-aware packet delivery prioritisation algorithm in ar/vr scenarios,” in *IEEE International Wireless Communications & Mobile Computing Conference (IWCMC)*, 2018, pp. 95–100.

[18] A. A. Simiscuka, A. Yaqoob, and G.-M. Muntean, “Utility-Based Multipath Delivery of Prioritized XR Content in a Machine Learning and Network Slicing-enhanced Environment,” in *IEEE International Wireless Communications & Mobile Computing (IWCMC)*, 2024, pp. 1637–1642.

[19] Y. Lu, S. Maharjan, and Y. Zhang, “Adaptive edge association for wireless digital twin networks in 6g,” *IEEE Internet of Things Journal*, vol. 8, no. 22, pp. 16 219–16 230, 2021.

[20] Y. Ma, F. Song, G. Pau, I. You, and H. Zhang, “Adaptive service provisioning for dynamic resource allocation in network digital twin,” *IEEE Network*, vol. 38, no. 1, pp. 61–68, 2024.

[21] Y. Li, J. A. Ansere, O. A. Dobre, and T. Q. Duong, “Real-time optimal resource allocation in multiuser mobile edge computing in digital twin applications with deep reinforcement learning : (invited paper),” in *IEEE 96th Vehicular Technology Conference (VTC2022-Fall)*, 2022, pp. 1–5.

[22] Z. Yuan, G. Ghinea, and G.-M. Muntean, “Beyond multimedia adaptation: Quality of experience-aware multi-sensorial media delivery,” *IEEE Transactions on Multimedia*, vol. 17, no. 1, pp. 104–117, 2015.

[23] S. Zhao, B. Tan, J. Wu, H. Ren, and Z. Zhang, “A real-time virtual reality adaptive streaming system,” in *2020 IEEE 92nd Vehicular Technology Conference (VTC2020-Fall)*, 2020, pp. 1–5.

[24] J. Li, S. Guo, W. Liang, J. Wang, Q. Chen, Z. Hong, Z. Xu, W. Xu, and B. Xiao, “Aoi-aware service provisioning in edge computing for digital twin network slicing requests,” *IEEE Transactions on Mobile Computing*, vol. 23, no. 12, pp. 14 607–14 621, 2024.

[25] M. Chen, W. Saad, C. Yin, and M. Debbah, “Data correlation-aware resource management in wireless virtual reality (vr): An echo state transfer learning approach,” *IEEE Transactions on Communications*, vol. 67, no. 6, pp. 4267–4280, 2019.

[26] Z. Tang, D. Chen, T. Sun, L. Zhang, M. Qi, and X. Wang, “Intelligent awareness of delay-sensitive internet traffic in digital twin network,” *IEEE Journal of Radio Frequency Identification*, vol. 6, pp. 891–895, 2022.

[27] W. Song, Y. Wang, M. Liu, and Z. Fei, “Joint optimization of resource allocation and fov for vr services in mobile edge networks,” in *2020 IEEE 6th International Conference on Computer and Communications (ICCC)*, 2020, pp. 535–541.