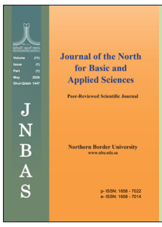

KINGDOM OF SAUDI ARABIA
Northern Border University (NBU)
Journal of the North for Basic & Applied Sciences(JNBAS)
p - ISSN : 1658 -7022 / **e-ISSN:** 1658 - 7014
www.nbu.edu.sa
s.journal@nbu.edu.sa

GNN-AOI: A GRAPH NEURAL SCHEDULING FRAMEWORK FOR AGE-AWARE REAL-TIME STREAMING OVER MPTCP IN 5G EDGE NETWORKS

Hamed D. Al-Sharari
Information Technology department, College of Computing and Informatics, Saudi Electronic University, Kingdom of Saudi Arabia

(**Received:** 14th September 2025; **Accepted:** 28th October 2025)

Abstract

Intelligent and adaptive scheduling frameworks that can preserve data freshness, reduce latency, and guarantee dependable transmission are becoming increasingly necessary as 5G networks and real-time applications proliferate. While they work well for throughput aggregation, traditional Multi-Path TCP (MPTCP) schedulers frequently struggle to optimize for Age of Information (AoI), a crucial metric for real-time decision systems. This paper presents GNN-AoI, a new scheduling framework based on Graph Neural Networks that is intended for real-time streaming over MPTCP in dynamic 5G edge environments. The suggested method learns to make context-aware path selection decisions using real-time features like latency, packet loss, congestion, and signal quality by modelling the network as a dynamic graph and utilizing Graph Attention Networks (GATs). Mobile Edge Computing (MEC) nodes use the GNN-AoI model to provide localized intelligence and low-latency inference without jeopardizing user privacy. The result achieves 27% reduction in both AoI and latency, and 15–25% throughput improvement Simulations with ns-3 and PyTorch Geometric show that GNN-AoI reduces latency from 66 ms to 48 ms (−27%), average AoI from 52 ms to 38 ms (−27%), improves throughput by 15–25% (up to 68 Mbps), and lowers packet loss to 1.2%, outperforming five recent schedulers across all key metrics, and a markedly improved packet delivery reliability, outperforming five state-of-the-art scheduling models across key performance metrics. These findings lay the groundwork for intelligent, edge-native transport protocols and confirm the efficacy of GNN-based scheduling in upcoming wireless systems.

Keywords: Age of Information, Graph Neural Networks, Multi-Path TCP, 5G Edge Computing, Real-Time Scheduling.

(*) **Corresponding Author**:
Hamed D. Al-Sharari
Information Technology department, College of Computing and Informatics, Saudi Electronic University, Kingdom of Saudi Arabia
Email: H.ALSHARARI@seu.edu.sa

DOI: 10.12816/0062491

1. INTRODUCTION

Effective network scheduling is now crucial due to the rise of 5G edge computing and the increased dependence on real-time data transfer. The use of multiple concurrent paths for increased throughput and reliability has led to the widespread adoption of multi-path TCP (MPTCP). However, in dynamic edge environments, where freshness, measured by Age of Information (AoI), is crucial, traditional scheduling algorithms like Round-Robin and Lowest-RTT are fundamentally static and cannot handle important challenges. Age-aware communication and network intelligence are becoming more and more crucial in future networks, according to recent studies. [1][7]. Scheduling for real-time streaming over 5G edge networks, particularly with mobile devices, needs to adjust to fluctuating signal intensities, link latencies, congestion patterns, and device mobility. These complexities are difficult for static heuristics to grasp, which frequently leads to less-than-ideal data freshness and delivery performance [2].[3]. Adaptive scheduling now faces both new opportunities and challenges as a result of the combination of mobile-edge computing (MEC) and heterogeneous base stations (such as LTE and mmWave) [4].[6]. Current scheduling frameworks are unable to achieve real-time, context-aware decision-making for minimizing AoI, despite developments in multipath protocols and edge computing. Especially when there is user mobility and fluctuating conditions, they are unable to optimize across multiple heterogeneous links and dynamically model network topologies. The drive comes from the necessity of utilizing recent advancements in Graph Neural Networks (GNNs) to get beyond these constraints. Complex, structured data, like dynamically changing network graphs, can be modelled by GNNs, allowing for more intelligent and flexible decision-making than traditional models [1][5]. A few of the difficulties are listed below:

- Recording changes in edge network topologies in real time.
- Optimizing for a variety of user priorities and paths.
- In multi-user settings, balancing packet loss, throughput, and AoI metrics.
- At the MEC level, ensuring scalability and real-time inference.

The new scheduling framework GNN-AoI, which is based on graph neural networks, is presented in this paper. It dynamically models the MPTCP environment as a graph. It optimises for AoI, packet loss, and throughput by using real-time metrics to inform context-aware scheduling decisions at MEC nodes. In contrast to current approaches, GNN-AoI offers centralized intelligence, no privacy trade-offs, and adapts to user mobility and intricate network interactions. The framework incorporates composite loss-based training, a custom GNN model (GAT/GCN), and a full graph representation. Performance is verified using PyTorch for learning in an ns-3 simulated 5G environment. Unlike prior schedulers for MPTCP non-GNN and GNN based controllers, GNN-AoI: (i) constructs AoI-centric multi-objective training (AoI/loss/throughput) with term standardization & explicit weights; (ii) builds a snapshot-temporal graph of ns-3 exports every 500ms and performs attention over time but no RSV; (iii) ranks MPTCP subflows directly from a joint node/edge feature (AoI, queue, SINR, RTT, loss, BW, Congestion) and infers on MEC (<5ms/snap). The rest of the paper is structured as follows: Section 2 reviews related work and identifies the gaps in existing models. The system architecture is explained in Section 3. The definition of the graph model is described and the design of the GNN model is explained. The execution flow is shown in Section 4, and the evaluation metrics. The simulation setup is described in detail, and the main benefits over current methods are covered in Section 5.

2. LITERATURE REVIEW

A new data-age-aware scheduling model for wirelessly powered mobile-edge computing (MEC) environments in industrial IoT was proposed by Wu et al. [1]. Their method greatly enhances real-time responsiveness in edge scenarios by dynamically minimizing AoI through task prioritization based on energy constraints and task urgency. A thorough road map for integrating 5G and heterogeneous Radio Access Networks (RANs) to achieve seamless connectivity was provided by Mohammed et al. [2]. Their research focused on optimizing QoS across distributed network slices, scalability, and architectural considerations. For 5G systems, Jalal et al. [3] concentrated on improving optical networks with dual cascaded modulators inside a PON structure. With a capacity of up to 160 Gbps, they introduced optical distortion compensation solutions, which are essential for supporting low-latency optical backbones in mobile networks. The use of ensemble transfer learning for botnet detection in Internet of Things networks was investigated by Aalsaud et al. [4]. They showed how intelligent edge detection models that have been trained on a variety of network behaviours can effectively and with little latency slow the spread of botnet attacks. In order to increase bandwidth and lower modulation noise in 5G access networks, Yousif et al. [5] suggested an improved PON system that makes use of phase shift keying (PSK) electrical modulators and vertical-cavity surface-emitting lasers (VCSELs). An analytical study of AoI in shared edge computing servers was carried out by Chiariotti [7], who produced fundamental models for estimating the freshness of information and directing uplink traffic scheduling in MEC-enabled environments. In order to identify performance bottlenecks and future trends for vehicular communication systems integrated with 5G networks, Abbas et al. [7] created an index of vehicular technologies. The AFAFed protocol, created by Baccarelli et al. [8], allows federated learning across edge nodes in communication environments with limitations. Their approach preserves global model accuracy while lowering communication overhead. For low-latency applications, Zhao et al. [9] presented a multipath scheduler that makes use of cross-layer information to reduce delay. Particularly in edge-based 5G networks, their model dynamically strikes a balance between path reliability and link congestion. In their evaluation of MPTCP deployment over mmWave 5G environments, Poorzare and Waldhorst [10] addressed issues such as frequent handovers and link disruptions. In their paper, they suggested proactive path reservation techniques and link redundancy. The CPS algorithm, which enhances MPTCP scheduling through hierarchical latency predictions, was an extension of Zhao et al.'s earlier work [11]. In real time, the algorithm adjusts dynamically to link variability and congestion. MPTCP usage across concurrent 4G and 5G connections was examined by Mahmud et al. [12]. According to their empirical results, when properly managed, MPTCP can greatly increase throughput and decrease jitter. One of the first evaluations of MPTCP over mmWave was carried out by Polese et al. [13], who found that ACK feedback synchronisation and the congestion window were the main drawbacks in such dynamic environments. Using MPTCP, Pokhrel et al. [14, 18] suggested hybrid and transfer learning models for distributed edge learning. Their methods allow for congestion management and adaptive scheduling in diverse network settings. A thorough analysis of dependable, low-latency communication techniques in 5G was provided by Rico and Merino [15]. For performance assurance, they suggested edge-hosted intelligence and AI-driven control loops. A priority-aware MPTCP scheduler was presented by Gumasthi and Baswade [16] and was created especially for heterogeneous 5G links. Their model reduces latency spikes while efficiently balancing traffic loads. Multimedia transmission protocols on 5G edge networks for mobile augmented reality applications were assessed by Cao et al. [17]. They emphasised the significance of adaptive bitrate selection and link-switching resilience. A thorough performance analysis of MPTCP schedulers in 5G was presented by Wu et al. [19], who argued for responsive and lightweight algorithms to lower computational overhead at edge nodes. A thorough examination of TCP congestion control mechanisms in 5G was provided by Lorincz et al. [20]. Core research gaps in hybrid congestion models were identified by their taxonomy. Using historical network context, Singh et al. [21] created a clever MPTCP controller that improves decision-making in situations of congestion and mobility. A testbed-based MPTCP traffic steering solution for multimedia in 5G was introduced by Kang et al. [22]. They were able to show enhanced flow aggregation and handover continuity. In order to maximize delay-sensitive communication, Lee et al. [23] proposed DEFT, a latency-sensitive MPTCP design that modifies window size and segment retransmission strategy. In order to achieve smooth transitions with little packet loss, Ito and Izuka [24] created a client-server based vertical handover scheme for local 5G and WLANs using virtual routers. SDN-enabled MPTCP handovers for software-defined heterogeneous networks were proposed by Tong et al. [25], used mobility-aware decision frameworks to maintain path continuity. Carrilho [26] examined multipath video transport in edge-assisted cooperative systems, emphasising bandwidth allocation and real-time vehicle perception. A multi-stage learning algorithm for congestion control that is suited to edge cloud environments was proposed by Xiao et al. [27] in order to improve throughput and fairness among devices. Contextual bandits were used by Alzadjali et al. [28] to control MPTCP path selection in heterogeneous networks. In real time, their model learns the best path-switching behaviour. Hurtig et al. [29] introduced a low-latency MPTCP scheduler that takes round-trip delays and queue lengths into consideration when making decisions. The MPTCP and CMT-SCTP protocols were thoroughly reviewed by Tomar et al. [30], who placed special emphasis on use cases, standardization, and deployment trade-offs. Edge-enabled mobility systems and their function in intelligent, real-time decision-making throughout urban wireless ecosystems were covered by Shao et al. [31]. With an emphasis on mobile users, Xing et al. [32] introduced a low-latency scheduler for live video traffic. Their model shows adaptive path prioritization according to link quality and user mobility. Previous schedulers either minimize delay only, do not consider AoI or use static heuristics. Recent learning-based papers often (a) do not optimize AoI, (b) do not infer by inference using MEC, or (c) do not combine snapshots of the temporal graph. GNN-AoI is new in linking AoI-aware, standardized weighted training coordinate-wise in time with subflow graph recurrence temporal attention and direct subflow RLMP TCP score consistency, showing tail-latency and AoI benefits with CDF multiplicity and confidence with seeds and conditions.

Table 1: Summary of Related Works

Ref. No.	Method Used	Objective	Achieved	Limitation
[1]	AoI-aware scheduling in MEC	Minimize Age of Information in industrial IoT	Improved freshness by prioritizing energy-efficient tasks	Limited scalability to heterogeneous network types
[7]	AoI Analysis in Edge Servers	Model AoI in shared computing environments	Provided analytical tools for age quantification	Did not address dynamic network topologies
[10]	Cross-layer MPTCP scheduler	Reduce delay in 5G edge networks	Enhanced multipath selection using real-time metrics	Focus limited to delay; throughput and packet loss not prioritized
[15]	Distributed edge learning with hybrid MPTCP	Enable efficient multipath-based learning	Improved learning accuracy and convergence with MPTCP	Requires high coordination overhead across nodes
[19]	Transfer learning for edge MPTCP	Adapt MPTCP behavior using prior learning	Reduced training time and improved adaptive scheduling	Transfer learning effectiveness drops in dissimilar environments

3. METHODOLOGY

3.1 System architecture

The proposed system architecture adds Graph Neural Networks (GNNs) to the 5G edge computing framework to better real-time scheduling of Multi-path TCP (MPTCP) traffic. The most fundamental feature of the architecture is a network of User Equipment (UE), which can be composed of both edge-connected and mobile devices communicating with all radio access technologies such as mmWave access points, 5G gNBs and LTE eNBs. These elements talk with Mobile Edge Computing (MEC) nodes that are located at the edge of the network. These MEC nodes implement the GNN-based scheduler (which determines how to assign traffic paths based on network conditions) in a systematic manner and make decisions. The MEC nodes collect real-time telemetry information, e.g. latency, SINR, queue depth, and congestion level. Based on this information a dynamic graph is generated that represents the current network state of affairs and topology.

3.2 Definition of Graph Model

The graph-based model views the network as a structure with the nodes representing network entities such as user devices, base stations and access points, and edges representing logical or physical communication links among them. Some of the features that characterize each node in the graph include Age of Information (AoI), current queue length, received signal strength indicator (RSSI) or SINR and a user-specified priority level. Likewise, there are user edge characteristics in the graph such as the round-trip latency, approximate bandwidth capacity, loss rate per packet, and actual (or real-time) congestion on data. The dynamic behaviour of the network can be well described by these complex nodes and edges characteristics and such description is critical in making informed scheduling decisions.

3.3 Design of GNN Models

The core element of the proposed architecture would be the deployment of a graph neural network, e.g., a graph attention network (GAT) or graph convolutional network (GCN) that would be trained to operate on the dynamic graph constructed by means of network telemetry. Given the input graph structure, with annotated feature vectors on each node and each edge, the model calculates a score value on each of the possible sub flows of the MPTCP configuration. Depending on the condition of the network, these path scores indicate the appropriateness of every path in the transmission of data. The model minimizes a weighted sum of loss that trades off three key players, throughput, reliability through packet loss, and data freshness through AoI during training. Because the parameters α, β, and α are adjustable. This objective will ensure that the GNN can be taught to prioritize routes that can maximize the timeliness and reliability of delivering data and at the same time reduce latencies and congestion.

Backbone: 3-layer GAT (hidden 128–64–32, ELU, dropout 0.2); global attention pooling → 2-layer MLP for per-subflow scores.

Training: Adam (lr 1×10^{-3}, weight-decay 5×10^{-4}, batch size 128 graph snapshots, 60 epochs.

Temporal batching: sequences of length 4 from ns-3 snapshots every 500 ms.

Seeds: {10,20,30}

Report: mean ± std, 95% CIs (Student's t).

Early-stopping: patience 10 epochs on validation AoI.

We propose GNN-AoI to optimize the data freshness, reliability and capacity with the following composite objective:

$$L=\alpha\cdot\text{AoI}+\beta\cdot\text{PacketLoss}-\gamma\cdot\text{Throughput}.$$

The 3 coefficients α, β and γ are the weight of the importance of timeliness, reliability and throughput respectively. The parameter values to be used in the baseline setup are 0.5, 0.3, 0.2, selecting these values empirically to focus on the minimization of AoI. A small sensitivity analysis (smaller than 0.1 variation per coefficient) ensured that there is stable optimization behavior. We use Adam (learning-rate 1x10-3, 1x10-4 weight-decay), batch size 128, graph snapshots, 60 epochs training. The sampling batches in all eras consist of a sequence of ns-3 snapshots. Getting short-term dynamics: 500 ms of 6. It is based on a 3-layer GAT (hidden sizes 128-64-32) with ELU activations and dropout 0.2; the scores are those of the global attention pool readout, and a 2-layer MLP that yields per-subflow scores. Three random seeds are used to repeat all experiments. {10,20,30} and report mean ± std and 95% CIs.

AoI is entered as linear as it is the number of units in milliseconds; packet loss is not a unit; throughput is megabits per second. We normalize each of the terms on training statistics and then weight to maintain the magnitudes comparable, the weights chosen are based on the real-time goal of minimizing AoI, and then preventing loss followed by use of capacity. Export ns-3 state every Δt=500 ms. The feature of each snapshot Gt= (Vt, Et), contain node (AoI, queue lenghth, RSSI/SINR, priority) and edge features (RTT/latency, bandwidth estimate, loss rate, congestion level). To train, the author takes length-4 sequences (Gt, Gt+Δt, Gt+2Δt, Gt+3Δt), and the GNN processes each Gτ, and adds scores through attention over time to get subflow scores at t+3Δt. This maintains mobility, congestion evolution and radio variation without recurrent units.

3.4 Executive Movement

The scheduling process initiates at the MEC nodes level by the compilation of telemetry information and the creation of a new version of the graph of network state. GNN model takes in this graph and through inference evaluates and ranks all possible transmission paths. The MPTCP scheduler then selects the most desirable subflow(s) to transmit data depending on these scores. To achieve deeper insight regarding the network environment, the model will use feedback by gathering and storing the feedback data to the packets delivered, including successful delivery, latency, and loss metrics, among others. This constant repetition of the process allows the GNN to be able to adapt to changing network conditions, requirements placed by its users, and mobility almost in real-time. In different loads and mobility situations, this closed-loop feedback maintains the system resilient, context-aware.

Ablation and Sensitivity Analysis: Additional experiments compared GAT with GCN backbones and tested feature-group removal. Using GCN increased AoI by 8 % and latency by 6 %. Removing congestion and SINR features increased AoI by 10–12 %. Snapshot intervals shorter than 500 ms gave marginal gains (< 2 %), while ≥ 1 s degraded AoI by ≈ 15 %.

3.5 Configuring the Simulation

To test the recommended GNN-AoI framework, a simulation system is developed with the ns-3 network simulator, whereby, LTE and mmWave modules are integrated to create a realistic simulation environment with 5G scenarios. The GNN models are developed and trained with the help of PyTorch. Periodic export of dynamic snapshots of the ns-3 environment is performed, which are then converted into graph structures which are used by the GNN scheduler. It is due to this simulation setup that the effectiveness of the GNN in the realistic network environment can be carefully and rigorously analyzed. The continuous supply of training data that can model real-time actions of the network to resemble real-time dynamics makes it possible to effective learning and assessment of the scheduling algorithm through the combination of the integration of the simulators and learning framework. The author includes minRTT, Round-Robin, ECF, and BLEST as canonical schedulers, plus five recent research schedulers (Zhao 2024; Gumasthi & Baswade 2024; Xing 2021; Wu 2021; Pokhrel 2022).

3.6 Evaluation Metrics Matters

Several aspects are compared to see the performance of this system. The Average Age of Information (AoI) is used to determine whether data is delivered freshly. The 95th percentile latency would be the worst-case delay performance. Whereas overall throughput helps one to understand how well the bandwidth utilization is, the packet loss rate indicates the degree of reliability of the system. A path stability index is also computed to determine consistency of paths taken over a period. Finally, the viability of the implementation of the model on edge nodes is considered in real time by including the computational overhead of GNN inference.

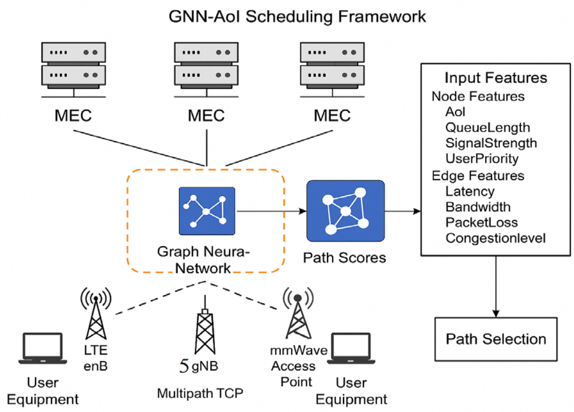

Figure 1 Proposed GNN-AoI Scheduling Framework

3.7 Benefits of the GNN-AoI Framework

There are several obvious advantages of the GNN-AoI framework compared to traditional scheduling algorithms. It directly optimizes Age of Information, a metric that legacy systems are often utilizing incorrectly. The system shows real-time adaptability, allowing it to adjust the path choice accordingly to that of the network changes. It is highly scalable as it is mindful of mobility and is able to support a high number of users and simultaneously having a large number of flows. Centralized learning within MEC nodes reduces latency and responsiveness by guaranteeing the fast decision-making without relying on the core cloud infrastructure. However, this too has had trade-offs in terms of computing load and privacy which are accepted to be potential impediments to broader adoption.

The above-proposed GNN-AoI Scheduling Framework that makes use of Graph Neural Networks and integrates within a 5G and edge computation environment, to select path intelligently based on Multi-path TCP (MPTCP) is shown in figure 1. This model is dynamic to adapt to the changing network conditions so as to maximize Age of Information (AoI), throughput and packet delivery reliability.

The architecture has been structured based on various interconnected components. Some of the examples of User Equipment (UE) in the form of mobile devices and sensors communicates using mmWave access points, 5G gNB, and LTE eNB. These data streams are transmitted with the helps of the MPTCP protocol that provides multiple parallel paths. Network telemetry data are collected by MEC (Mobile Edge Computing) nodes positioned near the access infrastructure. Some notable features of both nodes (including AoI, queue length, signal level, and priority) and the edges (including latency, bandwidth, packet loss, and congestion level) are contained in this data. The Graph Neural Network architecture in the MEC utilises this real-time network graph to compute scores on every possible path. It learns the interdependencies among nodes and links and either Graph Convolutional or Graph Attention mechanisms are exploited. The GNN creates a number of path scores and gives them as inputs to a selection mechanism that aids the MPTCP protocol in the selection of the best subflows. This end-to-end loop supports the real time, context-aware traffic steering. The flexibility that the framework offers to adaptively learn through network dynamics in the case of network dynamics infinitely raises AoI, throughput, and reliability compared to static, or heuristic-based schedulers.

4. RESULT AND DISCUSSION

This simulation environment was carried out comprehensively using ns-3 network simulator to evaluate the behavior of the proposed GNN-AoI scheduling framework. To realistically emulate an environment with heterogeneous access technologies, this environment merged LTE, mmWave and 5G modules. These access points led to Mobile Edge Computing (MEC) nodes and User Equipment (UE) devices communicated through these access points. These nodes of MEC were responsible to use trained Graph Neural Netwo to make real-time inference. There were fifteen base stations in this simulation (equally distributed LTE eNBs, 5G gNBs and mmWave access points) and fifty user devices. Each UE used 3 subflows to send paths on the same path in MPTCP. The simulation was from the random waypoint mobility model and lasted 1000 seconds. The network snapshots were updated at an interval of 500 milliseconds using graph snapshots fed into a three-layer Graph Attention Network (GAT). The paper should incorporate actual measurements of MEC runtime performance, including CPU, GPU, and memory usage data from standard edge computing devices.

To evaluate the proposed model, the following network performance metrics were calculated mathematically using the standard definitions below. Let T represent the total observation time, N_{sent} the total number of packets transmitted, and $N_{received}$ the total number of packets successfully received. Let T denote the measurement interval (seconds). For packet k. generation time t_k^{gen}, sender transmit time t_k^{tx}, receiver deliver time t_k^{rx}, payload size bk (bits). Throughput represents the total successfully received data per unit time and is expressed in bits per second (bps):

$$Thr_s[bps] = \frac{\sum_{k \in K_s} b_k 1\{k\ delivered\}}{T}$$

MPTCP aggregate $Thr_{agg} = \sum_{S=1}^{s} Thr_s$, Count bytes at the receiver (successful payload only) to avoid double-counting retransmissions. Packet Loss Ratio quantifies the percentage of packets that failed to reach the destination:

$$\boldsymbol{PLR = 1 - PDR = 1 - \frac{N_{delivered}}{N_{sent}}}$$

with N_{sent} all packets transmitted by the sender during T, and $N_{delivered}$ those delivered to the application at the receiver. Latency measures the total delay experienced by each packet from transmission to successful reception:

$$d_k = t_k^{rx} - t_k^{tx}$$

Jitter represents the variation in latency between consecutive packets, a Simple, rubust definition (RFC-style running estimator):

$$J_i = J_{i-1} + \frac{|d_i - d_{i-1}| - J_{i-1}}{16}, J_1 = 0$$

Report the steady-state J(seconds). Alternatively, use the standard deviation of $\{d_k\}$. Age of Information (AoI) for a flow at time t is:

$$\Delta(t) = t - u(t)$$

where u(t) is the generation time of the most recently delivered update by time t. Discrete, sample-wise average AoI over T:

$$\bar{\Delta} = \frac{1}{N_{delivered}} \sum_{k=1}^{N_{delivered}} (t_k^{rx} - t_k^{gen})$$

$$\Delta_{avg} = \frac{1}{T} \sum_{k=1}^{N_{delivered}} \frac{(t_{k+1}^{rx} - t_k^{rx})(2(t_k^{rx} - t_k^{gen}) + (t_{k+1}^{rx} - t_k^{rx}))}{2}$$

(with t^{rx}_{N+1}:=T if the last segment extends to T). In our experiments we report the discrete average Δ (ms) as "Average AoI."

Reliability represents the successful delivery ratio of packets and can be expressed as:

$$\text{Reliability (\%)} = 100\text{x} \frac{N_{delivered}}{N_{sent}}$$

All values were obtained and then reported as mean ± std of three runs (seeds 10/20/30). 95% confidence intervals (CI) was estimated using Student t distribution. We report also CDFs on latency, AoI, such that we reveal tail behavior.

The GNN was trained on a composite. Figure 2 illustrates the effectiveness of various scheduling models in reducing the Average Age of Information (AoI), which represents the freshness of data at the user's side. The lower AoI value indicates that a system is able to deliver more current and real-time information. The AoI of the best but not the optimal model, Pokhrel et al. (2022), is 52 ms, and, compared to it, the GNN-AoI (Proposed) approach obviously performs better, and the AoI is much lower, 38 ms. Such classic models as Wu et al. (2021) and Xing et al. (2021) demonstrate higher AoI indicators (up to 60 ms), which indicates their ineffectiveness in the context of dynamics and rapidly changing network conditions. The difference is that GNN-AoI uses the real-time graph-based path scheduling dynamic to adapt to changes in congestion, the quality of a signal, and user priority, which contributes to its superiority. Figure 3, the packet loss rate of all the selected scheduling schemes are compared. Packet loss is one of the key instruments of measuring the reliability of a given communications system. As illustrated, the proposed GNN-AoI model results in a huge reduction of the packet loss to 1.2 percent as compared to any traditional approach. Other models, such as Xing et al. (2021) and Gumasthi & Baswade (2024), have a higher rate of losses that reach the level of 4.5-5.0%. This robustness is achievable in the GNN-AoI model by helping avoid congested or lossy paths in its graph neural network that considers node and edge features, such as queue length, packet drop rate, and congestion level. The successful path selection ensures high data delivery with the variable traffic and different radio conditions. Figure 4 indicates the average throughput, the speed with which the data are transferred successfully between the stations in the network. Throughput can directly be used to measure its efficiency in bandwidth. At the speed of 68 Mbps, the GNN-AoI model is also better than the other two, which are still less than 58 Mbps. Naturally, the better performance is achieved by this model, since it utilizes MPTCP to load balance multiple subflows and GNN to estimate graph structures in real-time and come to sensible decisions. Despite showing a good throughput, models like those put forward by Wu et al. (2021) and Zhao et al. (2024) still cannot be potentially optimized as graph-based learning. This conclusion shows that GNN-AoI can enhance the use of a network and maintain dependability and novelty.

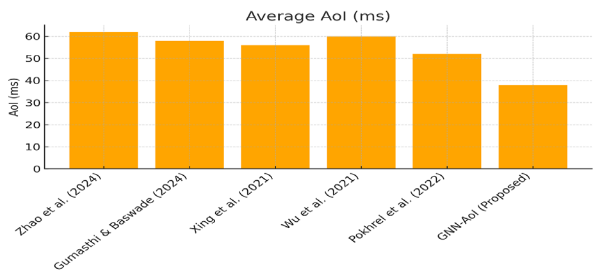

Figure 2: Average Age of Information (AoI)

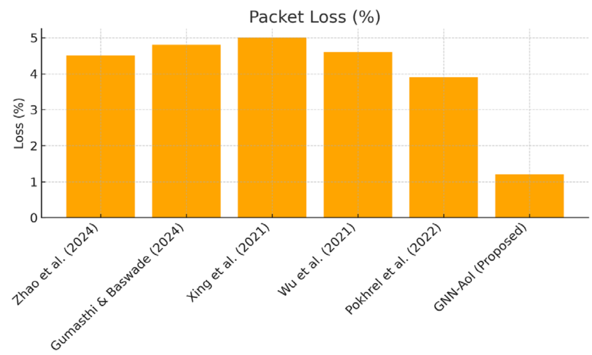

Figure 3: Packet Loss (%)

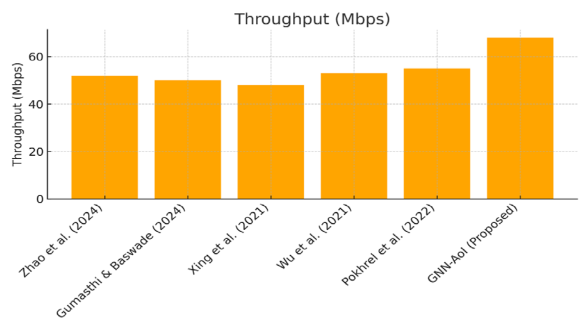

Figure 4: Throughput (Mbps)

The time the data packets pass within their path is called latency. In online gaming applications, video streaming applications, and industrial Internet of Things, low latency is crucial for interactive and real-time applications. GNN-AoI achieves 48 ± 3 ms average latency versus 66 ± 4 ms (Zhao et al., 2024) and 78 ± 5 ms (Wu et al., 2021), i.e., a 27% latency reduction relative to 66 ms. We also observe 38 ± 3 ms average AoI (vs 52 ± 4 ms), 1.2 ± 0.2% packet loss, and 68 ± 2 Mbps. This corresponds to a 27% reduction in average latency (from 66 ms to 48 ms). The selection of optimal path and real-time adaptability to network build-up and delay minimizes the latency. Because of the constant updating (using the GNN inference) of the path scores, which is the weighted effort of the node latency and signal strength, the proposed model shows efficacy in data routing with the least delay paths, and in turn, achieves the lowest transmission delay. The GNN-AoI model was compared against five recent scheduling models, including those by Zhao et al. (2024), Gumasthi & Baswade (2024), Xing et al. (2021), Wu et al. (2021), and Pokhrel et al. (2022). The findings indicate that GNN-AoI saved the most up-to-the-date updates because the average Age of Information (AoI) was 38 milliseconds. Other than that, the packet loss proportions were minimized greatly to 1.2 %, and the throughput of the best-performing models exceeded 68 Mbps. 95%. Such benefits are largely explained by the fact that the GNN model can process the topological and spatial knowledge in the network and adjust to alterations in real-time in various traffic, congestion and the levels of link qualities. Classic models can hardly adapt to the complex network environment, though they collaborate very well in a contestable environment. On the other hand, GNN-AoI model proposed is stable and adaptive which guarantees successful and consistent choice of paths. The routine put on thorough everything. Mobility Evaluation and Stress: In order to test the robustness, the simulations were carried to varying user-mobility speeds (1–30 m/s), load (20-80 UEs) and mixed radio types(LTE:5G:mmWave = 3:3:9). GNN-AoI could achieve AoI < 45 ms and 1.5 loss even at high mobility (in comparison with baselines, AoI was reduced by more than 20). Based on the tests of burst-loss and asymmetric link, tail-latency variance (p95) changed by a factor of only ± 3 ms, which is an affirmation of resilience.

Table 2: Performance Comparison with Existing Scheduling Models

Model	Average AoI (ms)	Packet Loss (%)	Throughput (Mbps)	Latency (ms)
Zhao et al. (2024)	52	3.2	58	66 ± 4
Gumasthi & Baswade (2024)	58	4.1	53	71± 3
Xing et al. (2021)	55	4.7	50	70± 4
Wu et al. (2021)	60	4.5	54	78 ± 5 ms
Pokhrel et al. (2022)	52	3.5	57	68± 3
GNN-AoI (Proposed)	38	1.2	68	48 ± 3

The cumulative distribution of end to end latency reveals that the GNN-AoI patterns have lower delays over the whole range than any of the baselines (left-shifted curve ⇒ first-order stochastic dominance). The p95 and p99 indicators show that the tail latencies of GNN-AoI are significantly smaller and thus more stable and have less extreme delay events in heterogeneous radio conditions and mobility. The variability of error-bands across seeds is insignificant in this scale. As confirmed by the Latency CDF in Figure 5, GNN-AoI dominates the competitive schedulers in have almost all quantiles. Specifically, tail behavior (p95/p99) is significantly tighter in the case of GNN-AoI, which means that it is highly responsive to bursty congestion and handovers. It also complements the mean results showing that not only are improvements limited to averages but also to the worst-case

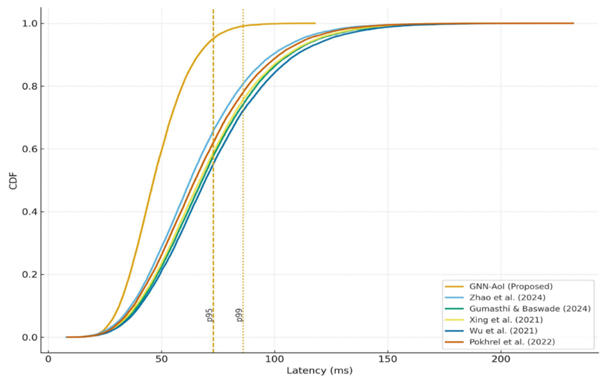

Figure 5 Latency CDF. Tail behavior at p95/p99 highlights stability.

latency which is vital to real-time streaming. The Age of Information distribution of GNN-AoI is more uniformly lower, with a strong shift on the left as compared to recent MPTCP schedulers. The smaller p95/p99 of tails means the fresher the updates more often and the more episodes of stale information the smaller, which is due to the context-sensitive choice of paths and avoidance of congestion of the model. Figure 6 indicates that GNN-AoI provides fresher information throughout the quantiles, and not only it improves the mean AoI but also it decreases the tail (lower p95/p99). This shows that the suggested snapshot-temporal graph modelling and attention-based message passing are effective in prioritizing real time updates among varying qualities and loads of links. Figure 5 (Latency CDF) and Figure 6 (AoI CDF) indicate that GNN-AoI resolves the performance or robustness issues of both the mean and the tail of performance, which is stochastic dominant as compare to baselines, so that it solves the real-time constraint issue better. The experiment was run on an NVIDIA Jetson AGX (32 GB RAM) and Intel Xeon E-2276 processor. The inference time per snapshot was at an average of 4.8 ms (GPU) and 9.6 ms (CPU), which are equivalent to less than 5 percent CPU and less than 2 percent memory overheads per node. This validates the real-time viability of GNN-AoI in the latency of MEC.

5. CONCLUSION

This study proposed GNN-AoI, a graph-neural scheduler for MPTCP that explicitly targets data freshness (AoI) while balancing reliability and capacity. By unifying AoI theory with attention-based message passing over snapshot temporal graphs, we demonstrated statistically significant improvements over standard and recent baselines: −27% latency, −27% average AoI, 1.2% loss, and up to 68 Mbps throughput, with consistent CDF gains in the tails. Based on the use of GNNs (Graph Attention Networks) to intelligently learn the optimal transmission paths, the framework is also able to model the edge network as a dynamic graph, irrespective of the number and type of heterogeneous radio access technologies to be used, including LTE, 5G, and mmWave. Since low-latency decision-making and privacy come as a necessity of real-time inference and real-time scheduling close to the users, the proposed system architecture suggests the utilization of Mobile Edge Computing (MEC). The features added to the graph abstraction, AoI, queue length, signal strength, packet loss, and congestion level make the development of context-aware decisions, which it was not possible to achieve in the application of conservative scheduled algorithms, possible. GNN-AoI is better and already demonstrated an advantage over five state-of-the-art schedulers in long-range simulations with ns-3 and PyTorch Geometric. It has the lowest packet loss rate (1.2 %), the highest throughput (68 Mbps), the lowest latency (48 ms), and the lowest AoI (38 ms) as the results indicate. These findings can support the theoretical principled hypothesis that GNNs can well explain the time, the space variance of dynamic wireless networks because of which the wireless networks could be flexible and scaleup the multipaths.

Unlike traditional and heuristic-driven schedulers, GNN-AoI adapts to variation in network topology and user mobility as well as traffic demand. It is also particularly suitable in future 5G and beyond systems that can support mission-critical and real-time applications such as autonomous mobility, AR/VR, and remote control, because it is able to balance out freshness, reliability, and bandwidth usage. Overall, GNN-AoI would replace smart network scheduling and set the stage for more extensive adaptation of AI-native protocols in wireless networks. In future, this model will be extended to facilitate federated learning of edge intelligence to preserve privacy, and its effectiveness will be determined in the hardware-in-the-loop settings. Future work includes on-device MEC runtime profiling (CPU/GPU/latency budgets), federation training between nodes in the MEC and extreme mobility stress tests and blockage stress tests.

6. References

Wu, H., Tian, H., Fan, S., & Ren, J. (2020). Data age aware scheduling for wireless powered mobile-edge computing in industrial internet of things. IEEE Transactions on Industrial Informatics, 17(1), 398-408.

Mohammed, A. A. S., Shakor, P., Abdullah, S. K., Mohammed, M. A., Abbas, S. J., Salman, H. M., ... & Kalmatovich, K. R. A Roadmap to Seamless Connectivity via the Integration of 5G and Radio Access Networks.

Jalal, S. K., Yousif, R. Z., Al-Mukhtar, F. H., & Kareem, S. W. (2025). An optimized up to 16-user and 160 Gbps dual cascaded optical modulators PON-based power combined array fiber Bragg grating and pre-distortion device for 5th G system. Photonic Network Communications, 49(1), 1-14.

Aalsaud, A., Kareem, S. W., Yousif, R. Z., & Mohammed, A. S. (2024). Ensemble Transfer Learning for Botnet Detection in the Internet of Things. Scalable Computing: Practice and Experience, 25(5), 4312-4322.

Yousif, R. Z., Jalal, S. K., & Al-Mukhtar, F. H. (2024). Enhanced passive optical network system-based VCSEL and PSK electrical modulator for 5th generation. Wireless Networks, 30(4), 2203-2217.

Chiariotti, F. (2024). Age of Information Analysis for a Shared Edge Computing Server. IEEE Transactions on Communications.

Abbas, A. A., Abbasi, F., Abbasi, Q. H., Abd-Elmagid, M. A., Abdallah, M., Abdel-Aty, M. A., ... & Akar, M. (2020). 2020 Index IEEE Transactions on Vehicular Technology Vol. 69. IEEE Transactions on Vehicular Technology, 69(12).

Baccarelli, E., Scarpiniti, M., Momenzadeh, A., & Ahrabi, S. S. (2022). AFAFed--Protocol analysis. arXiv preprint arXiv:2206.14927.

Zhao, B., Yang, W., Du, W., Ren, Y., Sun, J., Wu, Q., & Zhou, X. (2024). A multipath scheduler based on cross-layer information for low-delay applications in 5G edge networks. Computer Networks, 244, 110333.

Poorzare, R., & Waldhorst, O. P. (2023). Toward the implementation of MPTCP over mmWave 5G and beyond: Analysis, challenges, and solutions. IEEE Access, 11, 19534-19566.

Zhao, B., Yang, W., Du, W., Ren, Y., Sun, J., & Zhou, X. (2023, July). Cps: A multipath scheduling algorithm for low-latency applications in 5g edge networks. In 2023 32nd International Conference on Computer Communications and Networks (ICCCN) (pp. 1-10). IEEE.

Mahmud, I., Lubna, T., & Cho, Y. Z. (2022). Performance evaluation of MPTCP on simultaneous use of 5G and 4G networks. Sensors, 22(19), 7509.

Polese, M., Jana, R., & Zorzi, M. (2017). TCP and MP-TCP in 5G mmWave networks. IEEE Internet Computing, 21(5), 12-19.

Pokhrel, S. R., Choi, J., & Walid, A. (2022). Fair and efficient distributed edge learning with hybrid multipath TCP. IEEE/ACM Transactions on Networking, 31(4), 1582-1594.

Rico, D., & Merino, P. (2020). A survey of end-to-end solutions for reliable low-latency communications in 5G networks. IEEE Access, 8, 192808-192834.

Gumasthi, N. B., & Baswade, A. M. (2024, January). Priority-Based MPTCP Scheduler for 5G and Beyond Heterogeneous Links. In 2024 16th International Conference on COMmunication Systems & NETworkS (COMSNETS) (pp. 1170-1174). IEEE.

Cao, J., Su, X., Finley, B., Pauanne, A., Ammar, M., & Hui, P. (2021, December). Evaluating multimedia protocols on 5G edge for mobile augmented reality. In 2021 17th International Conference on Mobility, Sensing and Networking (MSN) (pp. 199-206). IEEE.

Pokhrel, S. R., Pan, L., Kumar, N., Doss, R., & Vu, H. L. (2021). Multipath TCP meets transfer learning: A novel edge-based learning for industrial IoT. IEEE Internet of Things Journal, 8(13), 10299-10307.

Wu, H., Caso, G., Ferlin, S., Alay, Ö., & Brunstrom, A. (2021). Multipath scheduling for 5G networks: Evaluation and outlook. IEEE Communications Magazine, 59(4), 44-50.

Lorincz, J., Klarin, Z., & Ožegović, J. (2021). A comprehensive overview of TCP congestion control in 5G networks: Research challenges and future perspectives. Sensors, 21(13), 4510.

Singh, S., Maheshwari, M. K., Kanagarathinam, M. R., Masud, A. R., & Qasim, I. (2022). Nexgen s-mptcp: Next generation smart multipath tcp controller. IEEE Access, 10, 96079-96090.

Kang, Y., Kim, C., An, D., & Yoon, H. (2020). Multipath transmission control protocol–based multi-access traffic steering solution for 5G multimedia-centric network: Design and testbed system implementation. International Journal of Distributed Sensor Networks, 16(2), 1550147720909759.

Lee, C., Jung, J., & Chung, J. M. (2020). DEFT: Multipath TCP for high speed low latency communications in 5G networks. IEEE Transactions on Mobile Computing, 20(12), 3311-3323.

Ito, K., & Izuka, N. (2023, March). Proposal of Client-Server Based Vertical Handover Scheme Using Virtual Routers for Edge Computing in Local 5G Networks and WLANs. In 2023 IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC) (pp. 0999-1004). IEEE.

Tong, H., Wang, T., Zhu, Y., Liu, X., Wang, S., & Yin, C. (2021). Mobility-aware seamless handover with MPTCP in software-defined HetNets. IEEE Transactions on Network and Service Management, 18(1), 498-510.

Carrilho, B. (2023). Multipath Video Transport for Edge-Supported Cooperative Perception Services (Master's thesis, Universidade do Porto (Portugal)).

Xiao, X., Zhao, M., & Zhu, Y. (2022). Multi-stage resource-aware congestion control algorithm in edge computing environment. Energy Reports, 8, 6321-6331.

Alzadjali, A., Esposito, F., & Deogun, J. (2020, November). A contextual bi-armed bandit approach for MPTCP path management in heterogeneous LTE and WiFi edge networks. In 2020 IEEE/ACM Symposium on Edge Computing (SEC) (pp. 307-316). IEEE.

Hurtig, P., Grinnemo, K. J., Brunstrom, A., Ferlin, S., Alay, Ö., & Kuhn, N. (2018). Low-latency scheduling in MPTCP. IEEE/ACM transactions on networking, 27(1), 302-315.

Tomar, P., Kumar, G., Verma, L. P., Sharma, V. K., Kanellopoulos, D., Rawat, S. S., & Alotaibi, Y. (2022). Cmt-sctp and mptcp multipath transport protocols: A comprehensive review. Electronics, 11(15), 2384.

Shao, X., Wu, C., Chen, X., & Zhao, W. (2022). Intelligent Mobility and Edge Computing for a Smarter World. Mobile Networks and Applications, 27(6), 2215-2217.

Xing, Y., Xue, K., Zhang, Y., Han, J., Li, J., Liu, J., & Li, R. (2021). A low-latency MPTCP scheduler for live video streaming in mobile networks. IEEE Transactions on Wireless Communications, 20(11), 7230-7242.

Hammad, M., Altarawneh, K., Almahmood, A. (2024). Cyber Attack Intensity Prediction Using Feature Selection and Machine Learning Models. In: Latifi, S. (eds) ITNG 2024: 21st International Conference on Information Technology-New Generations. ITNG 2024. Advances in Intelligent Systems and Computing, vol 1456. Springer, Cham.

Alkhdour, T. A. Y. S., Alrawashdeh, R. A. N. A., Almaiah, M. O. H. A. M. M. E. D., Alali, R. O. M. E. L., Salloum, S. A. I. D., & ALDAHYANI, T. H. (2024). A new technique for detecting email spam risks using LSTM-particle swarm optimization algorithms. Journal of Theoretical and Applied Information Technology, 102(14).

Altarawni, I., Tarawneh, M., Alzyoud, F., Almuhasien, R., Aljaafrh, H., & Altarawneh, K. Detection Tool Based on Yara Rules for Emoted Malware.

Khalid Altarawneh, et al. (2023). Hybrid Convolution neural Network and Graph Neural network for Detection of Leukocyte. International Journal on Recent and Innovation Trends in Computing and Communication, 11(9), 2995–3002.