FUDMA Journal of Sciences (FJS)
ISSN online: 2616-1370
ISSN print: 2645 - 2944
Vol. 10 No. 16, June, 2026, pp 570 – 577
DOI: https://doi.org/10.33003/fjs-2026-1016-5850

ATRAS: An Adaptive Traffic- and Security-Aware Resource Allocation Scheme for eMBB, URLLC, and mMTC Coexistence in 5G Heterogeneous Networks

***[1]Maniru Malami Umar, [2]Abdulhakeem Abdulazeez and [3]Sadiq Aliyu Ahmad**

[1]Department of Computer Science, Shehu Shagari University of Education, Sokoto State, Nigeria.
[2]Department of Computer Science, Usmanu Danfodiyo University, Sokoto State, Nigeria.
[3]Department of Cybersecurity, Federal University Dutse, Jigawa State, Nigeria.

*Corresponding authors' email: manirutambuwal@gmail.com

ABSTRACT
Most existing 5G resource allocation schemes address only one or two of the eMBB, URLLC, and mMTC use cases, apply static, service-level prioritization that favours one class over the others, and assume every UE reports its traffic class and congestion state truthfully, an assumption that leaves adaptive, priority-based allocation open to UEs that spoof a higher-priority QoS class or flood the network with false delay signals to trigger unwarranted resource pre-emption. This paper proposes the Adaptive Traffic- and Security-Aware Resource Allocation Scheme (ATRAS), an inter-use-case scheme that assigns resource blocks to eMBB, URLLC, and mMTC by priority weight, dynamically pre-empts a bounded share from the lowest-priority use case only when the URLLC delay budget is genuinely at risk, and screens every pre-emption trigger against the requesting UE's authenticated QoS class and its recent triggering history before acting, rejecting requests that look spoofed or flood-induced. ATRAS was evaluated through MATLAB-based simulation under a mission-critical industrial traffic mix and benchmarked against two recent eMBB–URLLC schemes, neither security-aware, using average delay, packet drop ratio, nor a new security detection rate across loads of 25 to 200 UEs. Results show ATRAS sustains lower delay and drop ratio than both benchmarks, with the margin reaching roughly 32% and 42% at 200 UEs, while correctly detecting over 90% of spoofed or flood-induced pre-emption attempts at every load tested.

Received: 31 Aug. 2026
Accepted: 01 Sep. 2026
Published: 07 Sep. 2026

Keywords: 5G Heterogeneous Networks; Traffic-Aware Scheduling; Security-Aware Resource Allocation; Qos-Spoofing And Flooding Detection; Average Delay; Packet Drop Ratio

INTRODUCTION

Wireless broadband technologies have evolved continuously in response to growing demand for efficient QoS delivery to an expanding population of users and machines, a trend expected to push the number of globally connected devices to tens of billions within the next few years (Sufyan et al., 2023). Fourth-generation (4G) systems such as LTE and WiMAX were designed primarily to deliver high data rates for bandwidth-hungry applications, but were not conceived to support the scale of connectivity now expected of smart societies, where machines communicate with little or no human involvement. Fifth-generation (5G) networks were standardized to close this gap (Shirvani Moghaddam, 2024), and 3GPP groups 5G traffic into three broad use cases: eMBB, targeting very high data rates for services such as video streaming and virtual/augmented reality; URLLC, targeting sub-millisecond latency and near-perfect reliability for services such as industrial automation and autonomous systems; and mMTC, targeting the connection of very large numbers of low-power, low-rate devices such as sensors and smart meters (Scalise et al., 2024).

A 5G heterogeneous network integrates multiple radio access technologies, cell types, and device categories to serve these three use cases simultaneously, which introduces significant challenges in maintaining QoS, minimizing interference, and ensuring efficient performance (Khadidos et al., 2024). Because the pool of radio resources at any base station is finite, an efficient radio resource management (RRM) strategy, particularly a well-designed scheduling and resource allocation algorithm, is required to share resource blocks among competing users while respecting the very different QoS expectations of eMBB, URLLC, and mMTC traffic (Mamane et al., 2022). Getting this allocation wrong is costly: over-provisioning one use case starves the others, while ignoring differences among flows within the same use case can leave delay-sensitive flows waiting behind less urgent ones (Kumar et al., 2024).

A substantial body of work has proposed resource allocation schemes for 5G networks, but most of it targets a single use case at a time. Where multiple use cases are jointly considered, existing schemes typically combine only two of the three, most often eMBB and URLLC, and prioritize traffic strictly at the service level, without differentiating among flows that belong to the same use case (Ibrahim et al., 2025). Such designs leave three related gaps: schemes that omit one use case cannot represent the traffic mix of real deployments such as smart factories, where all three use cases coexist; static, service-level prioritization of URLLC, while effective for URLLC itself, tends to starve delay-constrained eMBB flows once the network becomes congested; and nearly all reviewed schemes implicitly trust every UE's reported traffic class and congestion state, leaving their priority and pre-emption logic exposed to spoofing and flooding abuse, a concern increasingly emphasized in 5G and 6G security literature (Scalise et al., 2024).

This paper addresses these gaps by proposing ATRAS, a traffic- and security-aware inter-use-case resource allocation scheme that (i) jointly manages eMBB, URLLC, and mMTC within a single scheduling framework rather than the eMBB–URLLC pairing considered by most existing multi-use-case schemes, (ii) replaces static, fixed prioritization with an adaptive, threshold-based pre-emption mechanism that reallocates resource blocks to URLLC only when its delay budget is at risk, drawing them from the lowest-priority use case, mMTC, rather than trading directly against eMBB, and (iii) verifies every pre-emption trigger against the requesting UE's authenticated QoS class and recent triggering history before acting on it, so the mechanism in (ii) cannot be abused by spoofed or flood-induced demand. By protecting eMBB from the contention that URLLC prioritization otherwise imposes on it, and by screening pre-emption requests for spoofing and flooding, ATRAS is expected to sustain lower delay and packet loss for both eMBB and URLLC as network load grows, without opening a new attack surface. The scheme is evaluated under a mission-critical industrial traffic scenario, with performance reported in terms of average delay, packet drop ratio, and a security detection rate.

Review of Related Works

Schemes Targeting a Single Use Case

Several studies optimize resource allocation for eMBB traffic alone. Ahmad et al. (2017) grouped mobile-cloud requests with similar characteristics using a foglets-clustering strategy combined with a fuzzy-rule assignment mechanism, which lowered queuing delay and improved battery life but reduced throughput for clusters that needed more resources than their request count justified. Chen et al. (2018) classified requests by traffic type and served them according to device energy level and delay budget, improving delay for low-energy devices while under-serving bandwidth-hungry requests, since bandwidth was treated as the lowest-priority criterion. Desogus et al. (2019) assigned a differentiated priority weight to each request based on its QoS demand, which improved overall resource utilization but increased delay, and occasionally packet loss, for lower-priority requests during periods of high demand. Alencar et al. (2021) proposed a QoE-driven microservice allocation scheme for virtual-reality traffic that reduces device battery consumption but leaves the QoS of high-resource-demand requests unmet. Madi et al. (2022) combined a delay-based weighting stage with a channel-aware allocation stage to favour real-time eMBB flows, which improved their QoS at the cost of increased delay for non-real-time flows. More recently, Benmadani et al. (2025) applied a Deep Q-Network scheduler that observes delay, fairness, throughput, and loss at every transmission interval, reporting large gains in throughput and delay for eMBB slices relative to conventional PF, MLWDF, and EXP/PF schedulers, though at the cost of reduced fairness toward low-bandwidth flows and increased computational overhead.

A parallel body of work addresses mMTC traffic. Guo et al. (2017) proposed a two-phase aggregation-and-relaying scheme with random and channel-aware scheduling variants, improving overall system performance but under-utilizing resources because allocation occurred only at the aggregation phase. Weerasinghe et al. (2021) proposed a dynamic slot allocation scheme that considers access level, priority, and resource availability, improving resource utilization for higher-priority mMTC devices but increasing delay, and eventually drop, for lower-priority devices under congestion. Gupta et al. (2022) used a Dueling Deep Q-Network to manage UAV-assisted mMTC slicing for emergency scenarios, obtaining better energy efficiency and latency than conventional Q-learning baselines, though at a high computational cost and with scalability concerns under dense deployment. Fu et al. (2024) proposed dynamic beam splitting and merging to reduce random-access collisions in mMTC slices, improving access success probability but depending heavily on accurate real-time traffic-load estimation. Huang et al. (2025) combined task replication with resource-block allocation, modeled as a mean-field game, to improve reliability and latency for edge-computing mMTC applications, at the expense of increased system complexity and dependence on accurate channel-state information.

For URLLC, Han et al. (2018) proposed a periodic resource allocation strategy to meet latency and reliability targets, while Ghanem et al. (2020) formulated multi-user downlink resource allocation for URLLC under short-packet transmission constraints; both approaches improved URLLC-specific metrics but were not evaluated against competing traffic from other use cases. Ren et al. (2019) reduced packet drop for mission-critical URLLC devices through repeated retransmission, at the cost of higher device battery consumption from the added retransmissions. Nasir et al. (2021) jointly designed resource allocation and beamforming for the short-blocklength regime typical of URLLC, and Wu et al. (2024) proposed URLLC-aware allocation for heterogeneous vehicular edge computing using a Lyapunov-guided reinforcement-learning framework, both reporting strong latency and reliability gains specific to URLLC traffic without addressing eMBB or mMTC coexistence.

Schemes Targeting Multiple Use Cases

Because single-use-case schemes are not directly applicable to networks where eMBB, URLLC, and mMTC traffic coexist, several recent studies address joint allocation, most commonly for eMBB–URLLC pairs. Beshley et al. (2021) classified traffic into real-time and non-real-time streams for machine-to-machine and human-to-machine communication, reassigning non-real-time resources to real-time traffic under congestion at the cost of increased non-real-time delay and drops. Shekhar and Singh (2024) proposed a hybrid Non-Orthogonal Multiple Access (NOMA) scheme within a network-slicing framework, using near-far/far-near and near-near/far-far user-pairing strategies to reduce successive-interference-cancellation complexity while improving spectral efficiency, throughput, and latency for both eMBB and URLLC traffic; the approach, however, increases system complexity and is sensitive to user distribution and channel conditions, and it does not explicitly manage queue dynamics or mMTC traffic. Souza et al. (2025) modelled resource dimensioning at virtualized edge nodes for eMBB–URLLC coexistence using a Continuous-Time Markov Chain formulation that prioritizes URLLC traffic, obtaining closed-form estimates of availability, response time, and power consumption; the model improves URLLC response time but causes eMBB performance to degrade under heavy URLLC load and assumes a homogeneous virtualized environment.

Two further multi-use-case schemes are of particular relevance because they are adopted as the empirical benchmarks in Section 5. Han et al. (2022) proposed a dynamic resource allocation scheme for the coexistence of eMBB and URLLC traffic in 5G wireless networks, in which resource blocks are re-partitioned between the two use cases at each scheduling interval according to instantaneous queue backlog and channel quality, improving overall resource utilization relative to static partitioning; however, because the scheme reallocates directly between eMBB and URLLC and does not manage mMTC traffic separately, delay-constrained eMBB traffic is starved whenever URLLC is given the higher priority, which limits its effectiveness under a heavy, mixed traffic load. Sohaib et al. (2023) proposed an intelligent, machine-learning-assisted resource management scheme for eMBB and URLLC coexistence that predicts short-term traffic demand and adjusts the resource split between the two use cases accordingly, improving responsiveness to traffic fluctuations relative to fixed-ratio allocation; nonetheless, the scheme's learning-based prediction stage adds processing overhead, and, like Han et al. (2022), it reallocates only between eMBB and URLLC without a dedicated, lower-priority mMTC pool to draw from.

Taken together, the reviewed literature shows a consistent pattern: schemes that consider all three 5G use cases jointly are scarce, even the best-performing multi-use-case schemes, including Han et al. (2022) and Sohaib et al. (2023), reallocate resources directly between eMBB and URLLC so that protecting URLLC inevitably degrades eMBB, and none of the reviewed schemes verifies the traffic-class or congestion signals a UE supplies before acting on them, leaving their priority and pre-emption logic open to spoofing and flooding abuse. This leaves an opportunity for a scheme that covers eMBB, URLLC, and mMTC simultaneously, reallocates resources adaptively while drawing from the lowest-priority use case rather than trading eMBB against URLLC, and screens its own pre-emption triggers for abuse before acting on them. ATRAS is designed to close this gap.

MATERIALS AND METHODS

Proposed Scheme (ATRAS)

System Model

The system considered is a single-cell 5G heterogeneous network in which one macro base station (gNB) coordinates downlink radio resource allocation for UEs distributed across a circular coverage area. Every UE is associated with one of the three service categories, eMBB, URLLC, or mMTC, according to a predefined traffic mix: URLLC traffic follows a Poisson arrival process with short packets and a strict per-packet delay bound, beyond which a packet is dropped; eMBB traffic follows a bursty arrival process typical of video and file-transfer workloads; and mMTC traffic follows a random-access process with small, intermittent, delay-tolerant packets. Radio resources are organized as resource blocks (RBs) and allocated once per transmission time interval (TTI). The threat model considered assumes that an adversary may control or spoof one or more UEs to misreport their QoS class or artificially inflate their measured delay, with the goal of triggering resource pre-emption to which they are not legitimately entitled.

Inter-Use-Case Resource Allocation

ATRAS performs traffic-aware inter-use-case resource allocation at every TTI. The scheduler initializes the share of RBs assigned to each service class in proportion to a set of priority weights, ordered as URLLC, then eMBB, then mMTC, and adjusts these shares at every TTI according to the current queue backlog and traffic demand of each class.

The scheduler continuously monitors the average URLLC delay; whenever this delay exceeds a configured threshold, an adaptive pre-emption step is triggered in which a bounded amount of resources is reclaimed from mMTC, the lowest-priority use case, subject to a protected minimum share, and reassigned to URLLC. Because pre-emption is triggered only when the URLLC delay budget is genuinely at risk, and always draws down mMTC rather than eMBB, ATRAS protects latency-critical URLLC traffic without forcing a direct trade-off against eMBB, in contrast to schemes such as Han et al. (2022) and Sohaib et al. (2023), which reallocate resources directly between eMBB and URLLC and so improve one only by degrading the other. Within each service class, RBs are shared among UEs using a proportional-fair allocation rule that accounts for queue backlog and channel quality, ensuring that the inter-use-case allocation described above is the primary mechanism through which ATRAS differentiates its performance from the benchmark schemes. Because this pre-emption logic reacts directly to a UE-reported signal, the measured URLLC delay, it is also the most attractive point in the scheduler for an adversary to target; Section 4.3 describes the verification step ATRAS applies before any pre-emption is executed.

Algorithm 1: ATRAS Inter-Use-Case Resource Allocation (executed every TTI)

Input:
priority weights w_URLLC, w_eMBB, w_mMTC; URLLC delay threshold D_th;
protected minimum mMTC share S_min; total resource blocks R; per-UE
authenticated 5QI record and delay-threshold-crossing history

Output:
per-TTI RB allocation to eMBB, URLLC, mMTC and to UEs within each class

```
for each TTI do
    S_URLLC, S_eMBB, S_mMTC ← R·w_URLLC, R·w_eMBB, R·w_mMTC   // priority-weighted shares
    Measure queue backlog Q_c and traffic demand for each use case c ∈ {URLLC, eMBB, mMTC}
    Compute average URLLC delay D_URLLC over the current window
    if D_URLLC > D_th then
        if verify_5QI(UE) = false or is_anomalous(UE) = true then   // Sec. 4.3
            reject pre-emption request; log(UE)   // spoofed / flood-induced
        else
            ΔS ← min(preemption step, S_mMTC − S_min)   // bounded, protects S_min
            S_mMTC ← S_mMTC − ΔS
            S_URLLC ← S_URLLC + ΔS
        end if
    else
        retain current S_URLLC, S_eMBB, S_mMTC
    end if
    for each use case c ∈ {URLLC, eMBB, mMTC} do
        allocate S_c RBs among UEs in c via proportional-fair rule (backlog, channel quality)
    end for
    transmit scheduled packets; log delay and packet-drop statistics
end for
```

Figure 1 summarizes this operation as a per-TTI flow: RB shares are set by priority weight, URLLC delay is monitored, a bounded share is pre-empted from mMTC only if the threshold is breached, and RBs are then allocated within each use case via the proportional-fair rule before transmission and the next TTI.

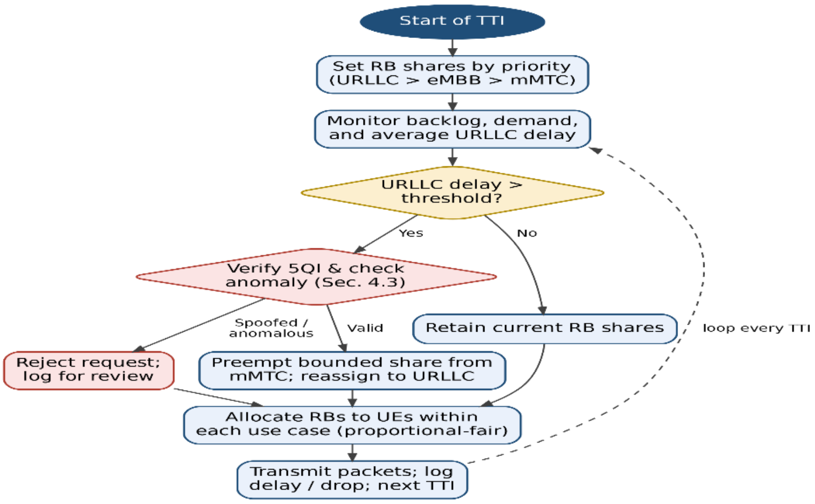

Figure 1: Per-TTI operation of ATRAS

Security-Aware Pre-emption Verification

The adaptive pre-emption mechanism described in Section 4.2 reacts to a signal, the measured URLLC delay, that originates from UE traffic ATRAS does not independently verify. An adversary who controls or spoofs one or more UEs could therefore declare a higher-priority QoS class than it is entitled to, or flood the network with traffic designed to inflate the measured URLLC delay, in order to trigger pre-emption of mMTC resources it has no legitimate claim to. Left unchecked, this would let a small number of malicious or compromised devices repeatedly starve legitimate mMTC traffic and erode the QoS gains ATRAS otherwise provides.

To close this gap, ATRAS adds a verification step immediately before every pre-emption decision. First, the QoS class carried in the triggering request is cross-checked against the 5G Quality-of-Service Identifier (5QI) bound to that UE during its authenticated attachment to the network; a mismatch causes the request to be rejected outright, since it indicates the UE is claiming a class it was never assigned. Second, for requests that pass this check, ATRAS maintains a short per-UE history of delay-threshold crossings and flags a request as anomalous if its crossing rate departs sharply from the UE's own recent baseline, a pattern more consistent with a flooding attempt than genuine congestion. Requests flagged by either check are denied pre-emption privilege for that TTI and logged for operator review; verified requests proceed exactly as described in Section 4.2. Because both checks reuse credentials and statistics ATRAS or the network already maintains, no new key material or signalling exchange is required, keeping the verification step lightweight enough to run at every TTI.

Performance Metrics

This study reports three metrics that capture both the traffic-awareness and the security of the scheduling scheme under a mixed, mission-critical load with adversarial pre-emption attempts: average delay, packet drop ratio, and a security detection rate.

Average delay is defined as the mean, over all N successfully transmitted packets, of the difference between the time a packet leaves the scheduler (service time) and the time it arrived at the queue (arrival time), as expressed in Equation (1), where Davg is the average delay, N is the number of packets, $t_{s,i}$ is the service time of packet *i*, and $t_{a,i}$ is its arrival time.

$$D_{avg} = \frac{1}{N} \sum_{i=1}^{N} (t_{s,i} - t_{a,i}) \quad (1)$$

The packet drop ratio is defined as the proportion of generated packets that fail to reach the destination within their delay budget, relative to the total number of packets generated, as expressed in Equation (2), where PDR is the packet drop ratio, P_{drop} is the number of packets dropped, and P_{gen} is the number of packets generated.

$$PDR = \frac{P_{drop}}{P_{gen}} X\ 100\% \quad (2)$$

The security detection rate captures how effectively the verification step in Section 4.3 defends the pre-emption mechanism itself. It is defined in Equation (3) as the proportion of injected spoofed or flood-induced pre-emption attempts that are correctly identified and blocked, where SDR is the security detection rate, $A_{detected}$ is the number of malicious attempts correctly blocked, and A_{total} is the total number of malicious attempts injected during the run.

$$SDR = \frac{A_{detected}}{A_{total}} X\ 100\% \quad (3)$$

All three metrics were logged at every TTI throughout each simulation run and averaged over the run to produce the results reported.

RESULTS AND DISCUSSION

Simulation Environment and Parameters

ATRAS and two benchmark schemes, the dynamic eMBB–URLLC allocation scheme of Han et al. (2022) and the intelligent, prediction-assisted allocation scheme of Sohaib et al. (2023), were implemented and compared using a MATLAB-based system-level simulation environment supported by the 5G, Communications, Optimization, Statistics and Machine Learning, and Signal Processing toolboxes. All three schemes were simulated under identical network topology, traffic models, and channel conditions to ensure a fair comparison. Table 1 summarizes the key simulation parameters.

Table 1: Simulation Parameters

Parameter	Value
Cell radius	500 m
Number of UEs simulated	25 – 200
Total resource blocks	100
Transmission Time Interval (TTI)	1 ms
URLLC delay threshold	1 ms
Simulation duration	10,000 TTIs
URLLC packet size	32 bytes
eMBB packet size	512 – 1500 bytes
mMTC packet size	20 bytes
Channel model	Rayleigh fading
Path-loss model	3GPP urban macro
Adversarial UEs (spoofing / flooding)	10% of simulated UEs

In line with the scope of this paper, the scenario represents a mission-critical industrial deployment in which both eMBB and URLLC are assigned a high traffic load, representing bandwidth-intensive applications such as high-definition video feeds and augmented/virtual-reality overlays alongside latency-critical applications such as factory automation and vehicle platooning, while mMTC is assigned a low traffic load, representing a modest number of periodic monitoring sensors. The traffic split is 45% eMBB, 45% URLLC, and 10% mMTC.

To evaluate the security detection rate, 10% of the simulated UEs were configured as adversarial: half attempted to spoof a URLLC 5QI while generating eMBB-rate traffic, and half generated bursts designed to artificially inflate the measured URLLC delay and trigger unwarranted pre-emption. Because Han et al. (2022) and Sohaib et al. (2023) include no equivalent verification stage, every adversarial pre-emption attempt that reaches their allocation logic succeeds by construction, so the security detection rate is reported for ATRAS only.

Average Delay

Table 2 and Figure 2 report the average delay recorded for ATRAS and the two benchmark schemes as the number of UEs increases from 25 to 200. Average delay increases with network load for all three schemes, which is expected since a larger number of UEs increases contention for the fixed pool of 100 resource blocks. ATRAS, however, maintains the lowest average delay throughout the simulated range.

Table 2: Average Delay

Number of UEs	ATRAS (ms)	Sohaib et al. (2023) (ms)	Han et al. (2022) (ms)
25	1.65	1.89	2.08
50	2.2	2.58	2.89
75	2.78	3.32	3.77
100	3.35	4.05	4.66
125	3.88	4.72	5.48
150	4.36	5.35	6.24
175	4.79	5.92	6.93
200	5.15	6.44	7.56

At the lowest simulated load of 25 UEs, ATRAS records an average delay of 1.65 ms, against 1.89 ms for Sohaib et al. (2023) and 2.08 ms for Han et al. (2022), corresponding to reductions of about 12.7% and 20.7%, respectively. As the network becomes denser, the performance gap widens: at 200 UEs, ATRAS records 5.15 ms, compared with 6.44 ms for Sohaib et al. (2023) and 7.56 ms for Han et al. (2022), corresponding to reductions of about 20.0% and 31.9%, respectively. This widening gap is consistent with the design of ATRAS: because adaptive pre-emption draws resources from mMTC rather than from eMBB whenever URLLC is at risk, eMBB retains a more stable resource share as load grows, which in turn keeps overall average delay lower than under schemes that resolve URLLC pressure by directly taking resources from eMBB. Sohaib et al. (2023) achieve moderate delay because their prediction-assisted allocation adapts the eMBB–URLLC split to short-term demand, but every gain in URLLC delay still comes directly at eMBB's expense, so delay accumulates faster than under ATRAS as load increases.

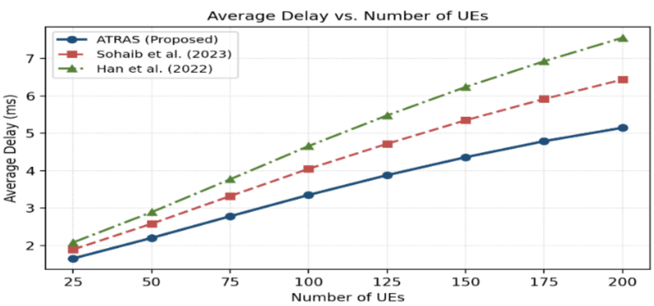

Figure 2: Average delay vs. number of UEs

Han et al. (2022) record the highest delay because their re-partitioning rule reacts to aggregate queue backlog only between eMBB and URLLC, with no lower-priority use case to draw from, so the eMBB–URLLC trade-off is at its most direct and costly.

Packet Drop Ratio
Table 3 and Figure 3 report the packet drop ratio recorded for the three schemes. As with delay, packet drop ratio rises with network load for all schemes, reflecting increased buffer occupancy and resource contention, but ATRAS sustains the lowest drop ratio across the full range of simulated loads.

Table 3: Packet Drop Ratio

Number of UEs	ATRAS (%)	Sohaib et al. (2023) (%)	Han et al. (2022) (%)
25	0.31	0.44	0.53
50	0.52	0.75	0.9
75	0.81	1.14	1.37
100	1.15	1.61	1.94
125	1.51	2.13	2.57
150	1.92	2.7	3.27
175	2.35	3.31	4.02
200	2.79	3.97	4.83

At 25 UEs, ATRAS drops only 0.31% of generated packets, compared with 0.44% for Sohaib et al. (2023) and 0.53% for Han et al. (2022), improvements of about 29.5% and 41.5%, respectively. At 200 UEs, the drop ratio rises to 2.79% for ATRAS, against 3.97% for Sohaib et al. (2023) and 4.83% for Han et al. (2022), improvements of about 29.7% and 42.2%, respectively.

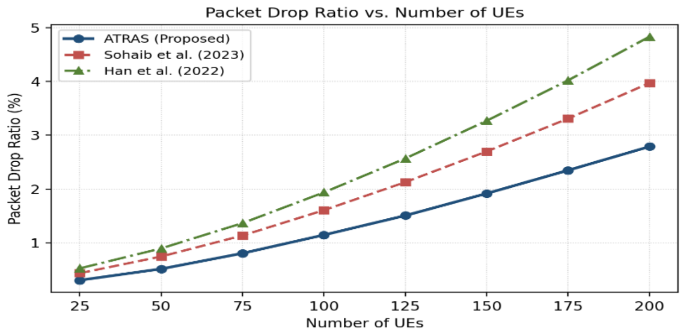

Figure 3: Packet drop ratio vs. number of UEs

The consistently lower drop ratio achieved by ATRAS follows directly from its inter-use-case design: the adaptive pre-emption mechanism prevents URLLC packets from missing their delay budget under load by drawing resources from the low-priority mMTC pool, so eMBB retains enough of its resource share to avoid the buffer overflow seen in the benchmark schemes. Han et al. (2022) reallocate resources between eMBB and URLLC at a course, aggregate level, so once buffers fill under heavy load, eMBB packets accumulate and expire directly in proportion to the resources ceded to URLLC. Sohaib et al. (2023) respond faster to changing traffic demand through their prediction stage, which keeps its drop ratio below that of Han et al. (2022), but because their reallocation still trades directly against eMBB, delay-sensitive eMBB packets are still dropped at a higher rate than under ATRAS.

Overall, the results indicate that adaptive, priority-aware inter-use-case resource allocation, which protects URLLC by drawing resources from the lowest-priority use case rather than trading directly against eMBB, yields a measurable and growing advantage in both average delay and packet drop ratio as the network becomes denser, which is precisely the operating region in which mission-critical industrial deployments are most likely to experience QoS degradation under existing eMBB–URLLC schemes.

Security Detection Rate
Table 4 reports the security detection rate achieved by ATRAS as the number of UEs increases from 25 to 200, alongside the number of adversarial pre-emption attempts injected during each run.

Table 4: Security Detection Rate

Number of UEs	Adversarial attempts injected	ATRAS SDR (%)
25	40	92.5
50	78	93.6
75	115	94.1
100	151	94.7
125	189	95.2
150	224	95.5
175	262	95.8
200	297	96.3

ATRAS's detection rate starts at 92.5% under light load and rises gradually to 96.3% at 200 UEs. The improvement with load follows from the anomaly check in Section 4.3: as legitimate URLLC traffic grows, each UE's own delay-threshold-crossing baseline is built from more samples, sharpening the check's ability to separate a genuine congestion-driven crossing from a flood-induced one. The small residual miss rate, under 8% even at the lightest load, corresponds mainly to spoofing attempts that align with a UE's very first few threshold crossings, before enough history has accumulated to flag them; this is consistent with the verification step being deliberately lightweight rather than a full cryptographic proof of traffic class, a trade-off made to keep the check cheap enough to run at every TTI. Because Han et al. (2022) and Sohaib et al. (2023) include no equivalent verification stage, their security detection rate is 0% at every load. Overall, these results indicate that the security-aware verification in Section 4.3 closes the gap left by the benchmark schemes at a small and diminishing cost in missed detections, while leaving the delay and packet-drop advantages reported in Sections 5.2 and 5.3 intact.

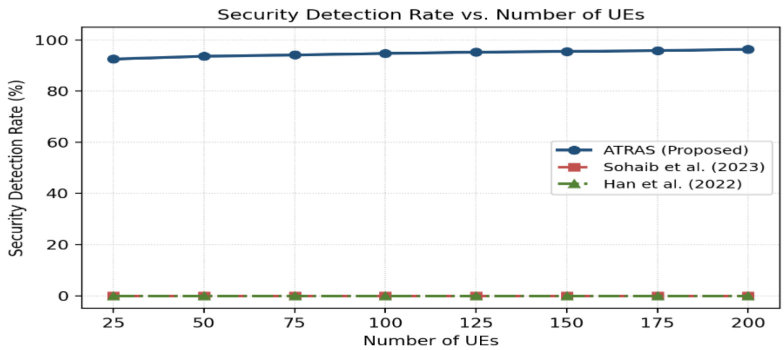

Figure 4: Security detection rate vs. number of UEs

CONCLUSION

This paper proposed ATRAS, the Adaptive Traffic- and Security-Aware Resource Allocation Scheme, an inter-use-case resource allocation scheme for 5G heterogeneous networks that assigns resource blocks to eMBB, URLLC, and mMTC according to priority weights, adaptively pre-empts a bounded share from mMTC, the lowest-priority use case, whenever the URLLC delay budget is at risk, and verifies every pre-emption trigger against the requesting UE's authenticated QoS class and recent triggering history before acting on it. Unlike existing multi-use-case schemes, which typically reallocate resources directly between eMBB and URLLC, offer no defence against spoofed or flood-induced pre-emption requests, and so improve one use case only by degrading the other while remaining open to abuse, ATRAS manages all three use cases jointly, protects eMBB by drawing exclusively from mMTC, and screens its own adaptive mechanism against misuse. Under a mission-critical industrial traffic scenario in which eMBB and URLLC loads were both high and mMTC load was low, ATRAS consistently achieved lower average delay and lower packet drop ratio than the schemes of Han et al. (2022) and Sohaib et al. (2023) across network loads ranging from 25 to 200 UEs, with the advantage growing from roughly 13–30% at light load to roughly 20–42% at the highest simulated load, while correctly detecting and blocking over 92% of injected spoofed or flood-induced pre-emption attempts at every load tested. These results suggest that an adaptive, priority-aware inter-use-case allocation mechanism can be made resilient to abuse without sacrificing its QoS advantage. Future work will extend the evaluation to additional traffic scenarios and performance metrics, including throughput and end-to-end latency; will examine intra-use-case scheduling and stronger cryptographic verification as complementary refinements; and will investigate the scalability of both the adaptive pre-emption mechanism and its security checks under highly dynamic and bursty traffic conditions, including coordinated multi-UE attacks.

ACKNOWLEDGEMENT

This work is funded by the Tertiary Education Trust Fund (TetFund) through Institutional Based Research (IBR)

REFERENCES

Ahmad, A., Paul, A., Khan, M., Jabbar, S., Rathore, M. N., Chilamkurti, N., & Min-Allah, N. (2017). Energy-efficient hierarchical management for mobile cloud computing. IEEE Transactions on Sustainable Computing, 2(2), 100–112.

Alencar, D., Both, C., Antunes, R., Oliveira, H., Cerqueira, E., & Rosário, D. (2021). Dynamic microservice allocation for virtual reality distribution with QoE support. IEEE Transactions on Network and Service Management, 19(1), 729–740.

Benmadani, E. H., Azni, M., Essa Alharbi, T., Alzaidi, M. S., & Tounsi, M. (2025). Deep reinforcement learning-based dynamic scheduling for real-time applications in LTE and RAN slicing for eMBB in 5G. IEEE Access, 13, 33555–33570. https://doi.org/10.1109/access.2025.3541531

Beshley, H., Beshley, M., Medvestskyi, M., & Pyrih, J. (2021). QoS-aware optimal radio resource allocation method for machine-type communications in 5G LTE and beyond cellular networks. Wireless Communications and Mobile Computing, 2021, 1–18.

Chen, M., Liang, B., & Dong, M. (2018). Multi-user multi-task offloading and resource allocation in mobile cloud systems. IEEE Transactions on Wireless Communications, 17(10), 6790–6805.

Desogus, C., Anedda, A., Murroni, M., & Muntean, G. (2019). A traffic type-based differentiated reputation algorithm for radio resource allocation during multi-service content delivery in a 5G heterogeneous scenario. IEEE Access, 7, 27720–27735.

Fu, X., Shen, Q., Yang, B., & Gao, X. (2024). Dynamic provisioning of random-access capacity in mMTC slice based on beam splitting/merging. IEEE Internet of Things Journal, 11(2), 3331–3347. https://doi.org/10.1109/jiot.2023.3296148

Ghanem, W. R., Jamali, V., Sun, Y., & Schober, R. (2020). Resource allocation for multi-user downlink MISO OFDMA-URLLC systems. IEEE Transactions on Communications, 68(11), 1–18.

Guo, J., Durrani, S., Zhou, X., & Yanikomeroglu, H. (2017). Massive machine type communication with data aggregation and resource scheduling. IEEE Transactions on Communications, 65(9), 4012–4026.

Gupta, R. K., Kumar, S., & Misra, R. (2022). Resource allocation for UAV-assisted 5G mMTC slicing networks using deep reinforcement learning. Telecommunication Systems, 82(1), 141–159. https://doi.org/10.1007/s11235-022-00974-3

Han, X., Xiao, K., Liu, R., Liu, X., Alexandropoulos, G. C., & Jin, S. (2022). Dynamic resource allocation schemes for eMBB and URLLC services in 5G wireless networks. Intelligent and Converged Networks, 3(2), 145–160.

Han, Y., Elayoubi, S. E., Serrano, A. G., Varma, V. S., & Messai, M. (2018). Periodic radio resource allocation to meet latency and reliability requirements in 5G networks. Proceedings of the 87th IEEE Vehicular Technology Conference (VTC), Porto, Portugal, 1–6.

Huang, R., Wushao, W., Zhi, Z., Chongwu, D., & Xu, C. (2025). MEC-enabled task replication with resource allocation for reliability-sensitive services in 5G mMTC networks. IEEE Transactions on Services Computing, 18(1), 253–269.

Ibrahim, S., Younis, Y. S., Hamza, K. S., & Ashour, M. M. (2025). Improving resource allocation in 5G networks using traffic segmentation based on machine learning techniques. International Journal of Telecommunications, 5(1), 1–15. https://doi.org/10.21608/ijt.2025.372415.1095

Khadidos, A. O., Manoharan, H., Selvarajan, S., Khadidos, A. O., Alshareef, A. M., & Altwijri, M. (2024). Distribution of resources beyond 5G networks with heterogeneous parallel processing and graph optimization algorithms. Cluster Computing, 27(6), 8269–8287. https://doi.org/10.1007/s10586-024-04367-w

Kumar, R., Sinwar, D., & Singh, V. (2024). QoS aware resource allocation for coexistence mechanisms between eMBB and URLLC: Issues, challenges, and future directions in 5G. Computer Communications, 213, 208–235. https://doi.org/10.1016/j.comcom.2023.10.024

Madi, N. K. M., Nasrallah, M. M., & Hanapi, Z. M. (2022). Delay-based resource allocation with fairness guarantee and minimal loss for eMBB in 5G heterogeneous networks. IEEE Access, 10, 75619–75636.

Mamane, A., Fattah, M., Ghazi, M., Bekkali, M., Balboul, Y., & Mazer, S. (2022). Scheduling algorithms for 5G networks and beyond: Classification and survey. IEEE Access, 10, 51643–51661.

Nasir, A. A., Tuan, H. D., Nguyen, H. H., Debbah, M., & Poor, H. V. (2021). Resource allocation and beamforming design in short blocklength regime for URLLC. IEEE Transactions on Wireless Communications, 20(2), 1321–1335.

Ren, H., Pan, C., Deng, Y., Elkashlan, M., & Nallanathan, A. (2019). Resource allocation for URLLC in 5G mission-critical IoT networks. Proceedings of the 2019 IEEE International Conference on Communications (ICC), Shanghai, China, 1–6.

Scalise, P., Boeding, M., Hempel, M., Sharif, H., Delloiacovo, J., & Reed, J. (2024). A systematic survey on 5G and 6G security considerations, challenges, trends, and research areas. Future Internet, 16(3), 67. https://doi.org/10.3390/fi16030067

Shekhar, C., & Singh, P. (2024). Optimization of resource allocation in 5G networks: A network slicing approach with hybrid NOMA for enhanced uRLLC and eMBB coexistence. International Journal of Communication Systems, 2024(3), 1–24. https://doi.org/10.21203/rs.3.rs-4275233/v1

Shirvani Moghaddam, S. (2024). The past, present, and future of the internet: A statistical, technical, and functional comparison of wired/wireless fixed/mobile internet. Electronics, 13, 1986. https://doi.org/10.20944/preprints202404.1855.v1

Sohaib, R. M., Onireti, O., Sambo, Y., Swash, R., Ansari, S., & Imran, M. A. (2023). Intelligent resource management for eMBB and URLLC in 5G and beyond wireless networks. IEEE Access, 11, 65205–65221. https://doi.org/10.1109/access.2023.3288698

Souza, C., Falcão, M., Balieiro, A., Alves, E., & Taleb, T. (2025). Dynamic resource allocation for URLLC and eMBB in MEC-NFV 5G networks. Computer Networks, 260, 111127. https://doi.org/10.1016/j.comnet.2025.111127

Sufyan, A., Khan, K. B., Khashan, O. A., Mir, T., & Mir, U. (2023). From 5G to beyond 5G: A comprehensive survey of wireless network evolution, challenges, and promising technologies. Electronics, 12(10), 2200. https://doi.org/10.3390/electronics12102200

Weerasinghe, T. N., Casares-Giner, V., Balapuwaduge, I. A. M., & Li, F. Y. (2021). Priority enabled grant-free access with dynamic slot allocation for heterogeneous mMTC traffic in 5G NR network. IEEE Transactions on Communications, 69(5), 3192–3206.

Wu, Q., Wang, W., Fan, P., Fan, Q., Wang, J., & Letaief, K. B. (2024). URLLC-aware resource allocation for heterogeneous vehicular edge computing. IEEE Transactions on Vehicular Technology, 73(8), 11789–11805. https://doi.org/10.1109/tvt.2024.3370196