Article citation info:
Ali QI, Mohammed ZA. Latency optimization in 5G-enabled UAV-assisted wireless sensor networks: modeling, analysis, and adaptive strategies. Diagnostyka. 2026;27(1):2026104. https://doi.org/10.29354/diag/217185.

DIAGNOSTYKA, 2026, Vol. 27, No. 1

e-ISSN 2449-5220
DOI: 10.29354/diag/217185

LATENCY OPTIMIZATION IN 5G-ENABLED UAV-ASSISTED WIRELESS SENSOR NETWORKS: MODELING, ANALYSIS, AND ADAPTIVE STRATEGIES

Qutaiba I. ALI [1], Zeina A. MOHAMMED [2*]

[1] Computer Engineering Department, University of Mosul, Iraq
[2] Electronics Engineering College, Ninevah University, Mosul, Iraq
[*] Corresponding author, e-mail: zinah.mohammed@uoninevah.edu.iq

Abstract

In this paper, we propose a systematized analytical and AI-assisted framework to compute the end-to-end latency of an unmanned aerial vehicle (UAV)-attached wireless sensor network (WSN) node communicating over a 5G linked communication system and to identify and minimize the latency within this link process. Recent literature has typically analysed networks or mobility, but our model integrates the latency contributions of WSN node, UAV platform and 5G network into a single mathematical model by considering the interactions of the three actors. With parameters of sensing, processing, transmission, and 5G routing, the analytical model enables us to quantify its components latency level that provides clear-base on the analysis of parameter sensitivity. In addition, we employ three AI-based optimization techniques to adaptively set system parameters to minimize latency while adapting to different network conditions, including Supervised Regression, Reinforcement Learning and Hybrid AI–Heuristic control. Using simulation-based evaluation we show that the hybrid approach obtains up to 33% less latency compared with the baseline, and up to 28% and 18% less latency than reinforcement learning and regression methods, respectively. These results confirm the feasibility of AI-driven latency adaptation for UAV-assisted WSNs over 5G, offering a practical and scalable approach toward next-generation low-latency aerial IoT systems.

Keywords: 5G-enabled WSNs, latency modeling, UAV communication, low-latency data collection, edge computing, real-time sensor networks, end-to-end delay optimization, ultra-reliable low-latency communication (URLLC)

List of Symbols/Acronyms

AS – Application Server;
CN – Core Network;
DQN – Deep Q-Learning;
FFT – Fast Fourier Transform;
GA – Genetic Algorithm;
HARQ – ; Hybrid automatic repeat request;
IoT – Internet of Things;
MEC – Multi-access Edge Computing;
NFV – Network Function Virtualization;
PPO – Proximal Policy Optimization;
QoS – Quality of Service;
RAN – Radio Access Network;
RL – Reinforcement Learning;
SCS ; Subcarrier Spacing
SDN – Software-Defined Networking;
SD – Symbol duration;
TotE2E –Total End-to-End Latency;
TN – Transport Network;
UAVs – Unmanned Aerial Vehicles;
URLLC – Ultra-reliable low-latency communication;
UPF – User Plane Function;
WSNs – Wireless Sensor Networks;

1. INTRODUCTION

The integration of 5G-capable Wireless Sensor Networks (WSNs) and Unmanned Aerial Vehicles (UAVs) holds great potential for applications that necessitate high-speed, low-latency communications. UAVs, being deployed within dynamic or difficult-to-reach scenarios, depend upon the use of immediate data transmit times to perform important functions within domains like environment monitoring, disaster management, and city infrastructure inspection. Conventional WSNs are significantly hampered by high latency, however, whenever they are deployed across large or high-mobility regions, such as are typically the case for UAVs. 5G deployment introduces overheads for ultra-reliable low-latency communications (URLLC), a quality that bridges these shortcomings by reducing delay and facilitates seamless, real-time exchange of data between UAVs and processing centers [1,2]. Minimizing delay is essential for 5G-capable UAV networks to maximize system performance, particularly for delay-critical operations. The subject of this paper is the

Received 2025-07-19; Accepted 2026-01-20; Available online 2026-01-22

© 2026 by the Authors. Licensee Polish Society of Technical Diagnostics (Warsaw. Poland). This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).

development of a latency model that analyzes and optimizes end-to-end delay at the various layers involved within the use of UAV-assisted WSNs. By examining sources of delay, including transmit time, network congestion, handover time, and edge processing time, the proposed model provides insights for assessing and projecting system behavior within actual deployments. Also, the 5G capability of network slicing, edge processing, and dynamic resources allocation can realize the user's paradigm-based latency management that serves application basis required by the UAVs [3,4] thereby, allow application-specific granularity to reach the data transmit speed defined by applications.

The integration of 5G-capable WSNs with UAVs potentially provides significant applications ranging from environmental monitoring to disaster management and intelligent city development. The use of 5G for its high speed, low latency, and improved connectivity to overcome the shortcomings of conventional WSNs facilitates dynamic sensor deployment, more data gathering, and better communications within hostile terrain. Critical studies in this direction have studied security protocols, management of networks, energy efficiency, and autonomous functionality in these hybrid 5G-WSN and UAV systems [5].

Liu et al. introduced the concept of the PSAP-WSN framework, an authenticatable 5G-based WSN security protocol, to meet data security challenges on UAV-enabled networks, an essential area due to the vulnerability of the network to cyberattacks since drones fly across large, unguarded areas [6]. Alsamhi et al. indeed studied the potential of green Internet of Things (IoT) applications using UAVs and B5G networks, exploring methodologies to enhance the energy efficiency of the network and overcome the challenge of data gathering from remote areas in real time [7]. Khan et al. studied UAV swarm management in 6G networks and how Software-Defined Networking (SDN), along with Network Function Virtualization (NFV), technologies enable scalable management of networks for UAV-WSN frameworks, where swarms of drones dynamically respond to changing network needs [8].

Ch et al. studied security and privacy for WSNs using drones and proposed using blockchain technology for securing data exchanges across 5G drone networks, which is especially useful for drone missions requiring the collection and transfer of confidential data [9]. Qasim and Jawad created an energy-efficient opportunistic networking framework for 5G-capable drones, using edge computing to optimize energy saving while safeguarding communications, which is crucial for prolonged drone missions [10]. Likewise, Jagatheesaperumal et al. proposed a blockchain-based framework for securing UAV networks within B5G/6G frameworks, giving particular importance to security protocols that improve trust upon data integrity since drones transmit data to land-based nodes within WSNs [11].

Khan et al. solved the problem of managing data within drone-based networks using hybrid techniques incorporating metaheuristic and blockchain smart contracts to enhance data management within UAV-WSN systems. It focused on energy management for prolonged missions and secure handling of data, both crucial for UAV-WSN setups deployed in remote and constrained-resource conditions [12]. Ranaweera et al. surveyed MEC-based 5G networks and listed specific security threats that impact UAV-based WSNs. This contribution offers insights regarding countermeasures to make the security of UAVs capturing important environment or infrastructure-related data more robust [13].

Li et al. proposed collaborative techniques for beamforming at the physical layer for securing communications within 5G-based WSNs, which meet the demand for robust communication protocols against frequent disconnection and environmental interference [14]. Tanwar et al. discussed applications of blockchains for UAVs for 6G networks, mentioning that the use of blockchain together with trajectory optimization would enable secure and effective communications for data acquisition applications within UAV-WSN architectures [15]. Pandey et al. offered a survey on communications using UAVs, which included RF energy harvesting, an emerging application for UAV-based WSNs since it keeps drones operating for prolonged periods without using batteries exclusively. It is essential for a WSN deployment for monitoring on a large scale, in an ongoing manner [16]. Ullah et al. surveyed cognitive approaches for unmanned aerial vehicle (UAV)-assisted 5G networks and provided some adaptation of AI methods for more efficient resource allocation and response latency in wide area wireless sensor network (WSN) deployments [17].

Alsamhi et al. examined edge intelligence of drones in the concept of B5G networks and highlighted how federated learning and blockchain could play a key role in enabling autonomous UAV solutions for WSN applications like environmental monitoring and disaster response[18]. Sharma et al. focused on the communication technologies for UAVs, with a special emphasis on the 5G-enabled architectures which lead to robust data collection framework necessary for WSN applications/ requirements where high reliability and volume data-processing are required [19].

Khan et al. proposed a cluster-based routing algorithm specifically designed for 5G flying ad hoc networks to improve communication efficiency in high-mobility UAV networks. It assists with the hierarchical routing which optimizes the paths of data and provides the average energy utilization through all UAV nodes in WSNs [20].

Despite the significant progress made in UAV-assisted wireless networks and AI-driven optimization approaches for 5G environments, the majority of existing works share critical limitations that hinder their adoption in integrated aerial sensing systems. Previous efforts have been either on energy efficiency and trajectory design of UAVs [8, 9] or on higher level AI and cognitive frameworks with little analytical representation of the end-to-end latency while the communication infrastructure and the sensing node are both taken into consideration. Moreover, existing literature that tackles latency modeling in 5G or vehicular networks tend to neglect the UAV dynamics interplay with sensor-node processing, which results in an incomplete latency estimation for time-sensitive airborne applications. Moreover, the previous AI-assisted optimization methods only assess the learning performance or convergence without quantified latency benefits and implementability under limited UAV hardware settings. These gaps motivate the present work> in that> a common analytical & AI based methodology is required, to jointly model the WSN node, UAV & 5G network & provide quantitative latency analysis and on-the-fly adaptation.

In this work, we first aim to develop an end-to-end latency model of the 5G-enabled UAV-WSN systems, including the major contributors to the delay experienced by messages between the various network components, followed by optimization-based reduction strategies, as illustrated in Figure 1. We summarize the main contributions of this study as follows.

1. End-to-end latency model in unison: From unique analytical perspective, we presented a unified analytical form to model the latency contributions of both WSN node and one or another WSN node chain integrated into UAV, which allows for an intuitive understanding of the latency formation process in the sense of full UAV-carried sensing systems, which this paper does not separate into sub-nodes as long as it is UAV-integrated.
2. An AI-enabled adaptive latency control framework – The framework provides a novel tri-layer optimization structure that utilizes state-of-the-art techniques such as supervised regression, reinforcement learning and hybrid ai–heuristic control for facilitating autonomous at runtime tuning of system parameters to minimize real-time latency as a function of changing dynamic mobility and network conditions.

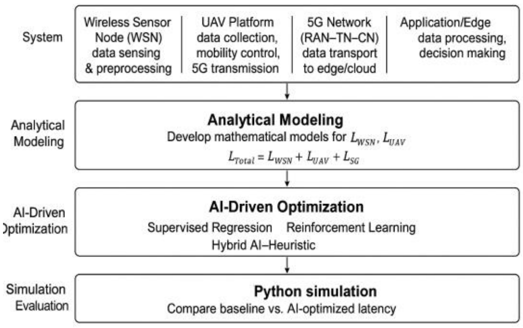
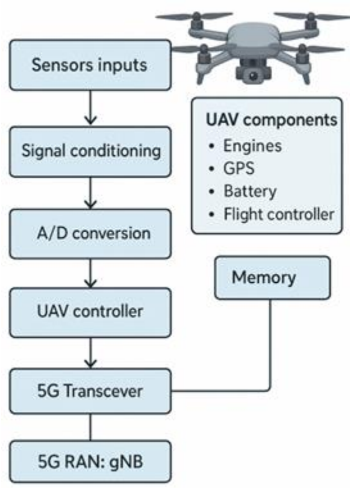


Fig. 1. Visual description of paper contribution

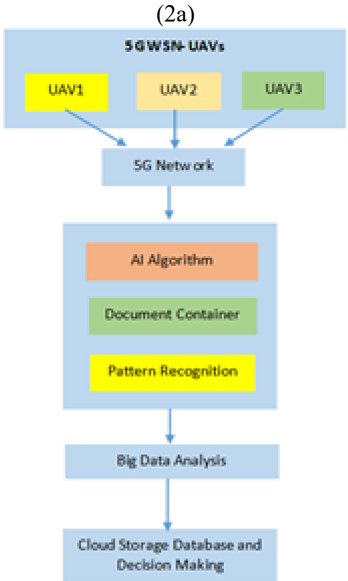
3. Latency-sensitivity formulation & Parametric impact mapping: A novel sensitivity-driven formulation is developed which analyzes the effect of each WSNs, UAV and 5G parameter on the overall latency. This generates a mapping that shows the top latency-driving parameters, Hashed Together-if you will, and generates a contribution that supports greater system design and optimization in future efforts and real-world deployments.
4. Practical design perspective to facilitate deployment: This work reviews computational overheads, energy costs, and real-time feasibility of AI-assisted latency-aware control in UAV-assisted 5G-WSN systems to facilitate their real-life deployment.

2. 5G BASED WSN INTEGRATED WITH UAV

Combining WSNs, Unmanned Aerial Vehicles (UAVs) and 5G technology is a state-of-the-art method to achieve effective and instant data collection in applications such as environmental monitoring, disaster management, precision agriculture, and smart cities. WSNs enable the collection of different environmental parameters using distributed sensors, whilst UAVs offer mobility and flexibility to gather data over large and inaccessible areas of interest. Moreover, the 5G adds more performance advantages of wireless technologies that provide high bandwidth, low-latency and reliable communications for this system for proper interference-free communications and data transfer For the system design depicted using Figure 2a, the mounted sensors on the UAV act as the main data acquisition devices that collect environmental parameters, which are then processed using signal conditioning to make the signals amenable to further processing. These signals are converted into digital signals using an A/D converter and then processed by the UAV controller, which controls the flight of the UAV, data collection and transfer, and also provides communications to other parts. The UAV consists of the basic hardware components like engines for propulsion, GPS for accurate navigation, a battery to supply power to the system, and a flight controller for system stability and control. After processing, the data are temporarily stored in the onboard memory and then conveyed in real time using a 5G transceiver to a 5G Radio Access Network (RAN) depicted using a gNB (Next Generation Node B). The 5G connection offers fast and reliable data transfer to a base station or to other processing nodes for processing and taking decisions in real time. This system shows the synergy between WSNs, UAVs, and 5G technology and thus offers a solid platform to gather scalable, flexible, and effective data from various and dynamic scenarios [21, 22].



(2a)



(2b)

Fig. 2. (a) 5G -WSN UAV node (b) Integration with 5G system

Figure 2b shows an architecture system where several Unmanned Aerial Vehicles (UAVs), a 5G network, and advanced data processing methods are combined for intelligent decisions. The architecture begins with several UAVs (UAV1, UAV2, and UAV3), which are installed with data-collecting sensors to sample data from various locations. These UAVs are linked to a 5G platform that helps sample speed, low-latency, and reliable transfer of information to send data to a center system for processing. AI algorithms process the transmitted data to enable intelligent insights and automation. The system is comprised of a Document Container module, responsible for storage and organization of the different data and a Pattern Recognition module, that identifies important trends, anomalies or patterns from the data obtained [23,24]. In Big Data Analysis, we try to process those insights which are extracted from a large amount of information collected from a number of UAVs. This is the process of aggregating, analyzing and finding actionable intelligence from data. This processed output is kept within a Cloud Storage Database, a seamless storage on demand solution for lifecycle data storage and access to historical data. Furthermore, the cloud system acts as a Decision Making platform to make sure the system provides real-time solutions, tactical planning and execution in the fields of environmental monitoring, disaster management, smart agriculture and industrial processes. An end-to-end high-level system architecture is designed based on integration of UAVs, 5G communications, AI and big data processing in cloud system so that all challenges in various domains can be managed in a scalable, intelligent and robust way to cope with complex and dynamic nature [25,26].

3. PERFORMANCE ANALYSIS FRAME WORK

A data packet has to go through a defined path from the point of origin to its destination, thus it is imperative to investigate the components the data packet interacts with to find how latency occurs in a 5G based WSN. Here's the details, concentrating on the points that are most relevant to the latency model [21] as in Figure 3.

3.1. Radio Access Network (RAN)

RAN connects WSNs to the other portions of network and WSNs belongs to some specific types, for example: environmental sensors, industrial monitors, agricultural sensors, etc. This is the wireless communication area. Key RAN components are:

- A WSN component is A WSN: The device that is transmitting or receiving data.
- gNB (Next-Generation Node B): 5G base station that manages radio resources, performs scheduling, and connects to WSNs.

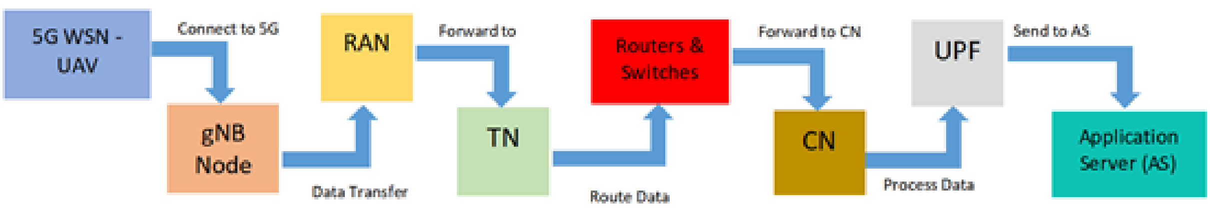


Fig. 3. 5G Latency Model

RAN latency contributors are:

- Numerology (SCS, Symbol Duration): 5G radio signals transmit differently based on configuration impacting time.
- Scheduling: The scheduling algorithms decides that which WSNs will get how much amount of radio resources (Time Slot Duration when will be used). This is delays.
- Retransmissions/hybrid automatic repeat request (HARQ): Whenever there is a transmittal error on the wireless channel, packets get retransmitted, which adds to latency.
- We will discuss about traffic load and interference: when the radio interface has congestion and interference, delay occurs.

3.2. Transport Network (TN)

The transport network (TN), which is the backhaul, connects the gNBs to the core. Typically uses fiber optic cables due to the high capacity and low latency nature of the traffic The dominant TN components are routers and switches, which determine which destination satisfies the routing protocol (e.g., BGP/OSPF) for how to forward packets in a network. TN latency contributors are:

- Propagation Delay: The time it takes for a signal to be sent over the physical distance through fiber optic cables or other transmission vectors.
- Transit delay — time taken at routers/switches when they are congested, processing, and transmission delays on the links.

3.3. Core Network (CN)

The CN, along with connecting external networks, for example, the Internet, acts as the brain of the 5G system as it is in charge of processing user data, mobility, and security. Key CN components are:

- User plane function (UPF): A key 5G component responsible for the routing and forwarding of user data traffic packets. This is a mandatory prerequisite for deploying the QoS policies and the MEC.
- Other CN Functions: Here CN function are simplified yet in other CN functions to (e.g. authentication servers) actually, latency can be added (in add to CN access).

3.4. Application Server (AS):

The gateway mainly treats the application, or the service which makes avail of WSN data. These could be within a centralized cloud data center (cloud based services) or at the edge of the network (MEC). AS latency contributors are:

- Inbox: The duration it has taken for the server to process the data it receives, perform computing and send back results.
- Queuing delay: If the server is getting a number of requests at the same time, the incoming request may have to wait in queue before processing.

Latency is a product of different level delays in the 5G system and outside. To minimize the latency in 5G-based WSNs one must thoroughly understand the contribution of each component while analyzing the stack performance due to different key parameters (numerology, traffic load, deployment strategy, etc.). Shift from Radio to network to application layer root cause analysis keeping in mind 5G low latency optimization [16].

By replicating the individual latency elements, to be able to create a complete model, it is possible to study 5G latency for different use cases and adjust the network to improve the performance of latency-sensitive apps.

4. LATENCY MODELING OF THE SYSTEM

By understanding these individual latency components, we can build a comprehensive model to analyze 5G latency in different scenarios and optimize network performance for critical applications.

4.1. Latency Components in 5G-Based WSNs

A data packet from the wireless sensor node (WSN) to the application server (AS), goes through several stages. Each stage which has a distinct function adds its own fragments of latency. However, these challenges emerging out of fragmentation and vagueness inherent in the literary description of 5G latency constituents motivate this study to reinvent the total latency computation process into an explicit algorithmic representation of them —see Algorithm in Figure 4 that illustrates Algorithmic representation of 5G latency constituents. This pseudo-code conveys the level of detail that a delay source can provide without having to specialize each source as an independent isolated delay since it preserves the hierarchical property of the flow of latency buildup from RAN to TN to CN to AS and optional edge aggregate stages. We characterize every step in terms of measurable parameters–for example, propagation distance, processing cycles, queuing delay, HARQ retransmissions, and link capacities, which provide full traceability and reproducibility. Structured representation then also explains the contributions from the individual parameters to total end-to-end latency and the interactions of intermediate terms in the communication chain. The pseudo-code, which combines all sources of latency in one algorithmic model, provides an accurate functional implementation of the 5G-based WSN latency modeling and, in general, allows perfect integration of the AI-optimization modules simulated in the later sections of this paper.

To maintain a mathematical consistency and reproducibility, the approach for the latency model is generated under a predefined framework of system assumptions that explicitly outlines the system conditions of the UAV–WSN–5G architecture.

```
Inputs:
  P_bits ← Data packet size (bits)
  SCS ← 5G subcarrier spacing (kHz)
  Scheduling_Delay ← RAN scheduling delay (ms)
  Retransmission_Delay ← HARQ-induced delay (ms)
  Distance_TN ← Transport network distance (km)
  PropSpeed_TN ← Propagation speed in TN (km/ms)
  NumNodes_TN ← Number of TN routers/switches
  ProcDelay_TN ← Processing delay per TN node (ms)
  QueueDelay_TN ← Queuing delay per TN node (ms)
  TxTime_TN ← Transmission time per TN link (ms)
  Distance_CN ← CN path distance (km)
  PropSpeed_CN ← Propagation speed in CN (km/ms)
  Proc_UPF ← User Plane Function processing time (ms)
  Queue_UPF ← UPF queuing delay (ms)
  Tx_UPF ← UPF transmission time (ms)
  ProcCycles_AS ← Processing cycles/bit at Application Server
  ProcCap_AS ← Processing capacity (cycles/ms)
  AggDelay ← Aggregation delay at MEC or edge node (ms)
  AggPktSize ← Aggregated packet size (bits)
  LinkCapacity ←Link capacity available for aggregation (bits/ms)

Outputs:
  RAN_Latency
  Transport_Latency
  Core_Latency
  Application_Server_Latency
  Aggregation_Latency
  Total_End_to_End_Latency

Procedure:
1. Compute Radio Access Network (RAN) latency:
   Symbol_Duration← 1/(2^µ × 15 kHz) // µ derived from SCS
     RAN_Latency ← Symbol_Duration
                 + Scheduling_Delay
                 + Retransmission_Delay
2. Compute Transport Network (TN) latency:
     PropDelay_TN ← Distance_TN / PropSpeed_TN
     TransitDelay_TN ← NumNodes_TN ×
     (ProcDelay_TN + QueueDelay_TN + TxTime_TN)
     Transport_Latency ← PropDelay_TN + TransitDelay_TN
3. Compute Core Network (CN) latency:
     PropDelay_CN ← Distance_CN / PropSpeed_CN
     TransitDelay_CN ← (Proc_UPF + Queue_UPF + Tx_UPF)
     Core_Latency ← PropDelay_CN + TransitDelay_CN
4. Compute Application Server (AS) latency:
     AppProcTime ← (ProcCycles_AS × P_bits) / ProcCap_AS
     Application_Server_Latency ← AppProcTime
5. Compute Aggregation latency (optional stage):
     TxAgg ← AggPktSize / LinkCapacity
     Aggregation_Latency ← AggDelay + TxAgg
6. Final total latency:
     Total_End_to_End_Latency ← RAN_Latency
                              + Transport_Latency
                              + Core_Latency
                              + Application_Server_Latency
                              + Aggregation_Latency

Return:
```

Fig. 4. Compute_Latency_Components_5G_WSN Algorithm

First, a quasi-static block fading model of the wireless channel over each transmission time interval is considered which guarantees that the instantaneous SNR and consequently the modulating parameters remain constant over an entire scheduling interval [27, 28, 29, 30]. Second, UAV mobility is defined as piecewise-linear within a small distance over consecutive periods, where each step is considered to be short enough to compute deterministic propagation distance and air-to-ground channel coefficients. Third, processing delays at the WSN node, UAV platform, and Application Server, are modeled as a linear cycle–rate formulations, where the assumed computational cost per bit is constant, for homogenous sensing workloads [31-33] Fourth, we assume stable traffic regime for the 5G RAN–TN–CN pipeline, allowing the queuing delay to be expressed in its expected value, i.e., with no burst-induced divergence. Third, we use an average over time to model HARQ retransmissions as a repetition rate instead of stochastic per-block realizations, allowing us to derive closed-form expressions for radio-layer latency [34-35]. These assumptions collectively establish a controlled and analytically tractable environment that supports explicit latency decomposition while remaining aligned with widely adopted 5G system models in the literature.

4.2. WSN Node Latency Modeling and Performance Parameters

In UAV-assisted 5G-WSN systems, each wireless sensor node introduces local delay before transmitting data to the UAV or gateway. This latency arises from sensing operations, local signal processing, and wireless transmission. Although the delay of a single node is often small, the cumulative effect across a network can significantly affect real-time performance, especially in dense or energy-constrained deployments.

The total latency at a WSN node (L_WSN) can be expressed as:

$$L\_WSN = L\_sense + L\_proc + L\_tx \tag{1}$$

Where L_sense is the sensing latency — the time required for the sensor to acquire and digitize the measured value.

$$L\_sense = 1 / f\_s \tag{2}$$

Where f_s is the sensor sampling frequency in hertz, L_proc is the local processing latency — the delay introduced by data formatting, compression, or encryption.

$$L\_proc = (C\_proc \times P\_bits) / f\_CPU \tag{3}$$

Where C_proc is the number of processor cycles per bit, P_bits is the data packet size in bits, and f_CPU is the node processor frequency in hertz, L_tx is the transmission latency — the time required to send the packet to the 5G network via UAV.

$$L\_tx = (P\_bits / R\_WSN) + L\_reTx \tag{4}$$

Where R_WSN is the data rate of the 5G link and L_reTx is the delay due to possible retransmissions caused by packet errors.

4.3. UAV Latency Modeling and Performance Parameters

The UAV acts as a mobile relay, carrying or collecting data from WSN nodes and forwarding it through the 5G network. Its flight dynamics, altitude, and onboard processing capability all influence latency. UAV-induced delay can be categorized into mobility-related, onboard processing, and link-quality components. The total latency introduced by the UAV (L_UAV) can be defined as:

$$LL\_UAV = L\_mob + L\_proc\_UAV + L\_link \quad (5)$$

Where L_mob is the mobility-induced latency — additional delay caused by UAV motion, handovers, or trajectory changes.

$$L\_mob = L\_handover + (d\_UAV / v\_UAV) \times \theta\_path \quad (6)$$

Where L_handover is the average 5G handover delay, d_UAV is the flight distance during communication, v_UAV is the UAV velocity, and θ_path is a factor representing the path angle or maneuver complexity, L_proc_UAV is the onboard processing latency — delay due to temporary storage, packet aggregation, or local computation.

$$L\_proc\_UAV = (C\_UAV \times P\_bits) / f\_UAV \quad (7)$$

Where C_UAV is the number of computation cycles per bit, P_bits is the packet size, and f_UAV is the UAV processor frequency, L_link is the UAV-to-5G link latency — delay due to transmission from the UAV transceiver to the 5G gNB.

$$L\_link = (P\_bits / R\_UAV\_5G) + L\_fading \quad (8)$$

Where R_UAV_5G is the achievable data rate of the UAV-to-5G link, and L_fading represents delay due to fading or temporary link degradation.

4.3. Integrated End-to-End Latency Framework

Combining all latency components yields the complete end-to-end expression:

$$L\_Total = L\_WSN + L\_UAV + L\_5G \quad (9)$$

L_5G is the sum latency of 5G components. The integrated formulation accounts for latency due to sensing, aerial relaying and 5G communication. It offers a common conceptual background to examine and optimize total delay in UAV-assisted, mission-critical and real-time 5G-WSN systems. Based on Mathematical formulation have been used, in order to further analytical strength, so that parameter correlation of WSN node, UAV subsystem, and 5G communication network are clear. The total end-to-end latency is expressed mathematically as the summation function of sensing, processing, transmission, and propagation delays across each of these three interacting domains. We use measurable system variables like CPU clock rate, the packet size, UAV speed, and 5G sub-carrier spacing, which make each latency components have quantitative traceability and reproducibility. These parameters are related to each other analytically in order to model both static and dynamic behaviours such as delays changing according to mobility patterns and communication rates adapting accordingly. Additionally, the proposed model facilitates sensitivity analysis in which the parameters affecting the total latency are identified. Our study is mathematically rigorous yet retains relevance to deployment in actual 5G-integrated UAV–WSN systems through a unique integrated and parameterized modeling approach. It also serves as the analytical foundation for the AI-based optimization strategies presented subsequently in the paper.

Table 1. Validation experiments settings

Parameter	Value
Subcarrier Spacing (SCS)	30 kHz
Deployment Strategy	Multi-Access Edge Computing (MEC) at the gNB
Traffic Load	125 packets/s
Packet Size	800 bits
Link Capacity Allocation (α)	0.5
Distance to Core Network	1 km
Processing Capacity (ProcCap_AS)	10 Gcycles/s

In addition, we performed a comparative analysis with the existing simulation and experimental studies on 5G-based Industrial Networks [36,37,38] to validate the theoretical model proposed in the current study. For the validation, we selected benchmark studies characterizing either empirical or simulation-based performance features of 5G-centric network. In order to have a fair comparison, we harmonized the feature dimension across our models and the benchmark studies, along with other key network parameters. The parameters consisted of subcarrier spacing, deployment strategy, traffic load, packet size, link capacity allocation, distance to the core network, and application server processing capacity. Both our model and the benchmark studies used in this study have values of specific variables as presented in Table 1 .

We summed the delays for radio access network delay, transport network delay, core network delay, application server delay, and data aggregation latency to obtain total end-to-end latency using these aligned parameters in our model. We compared the results of our model against the reported results of the benchmark studies. Table 2 summarizes some of the findings.

These results show that the latencies obtained from both real and the simulation environment are close to each other, and also our tanh model derived latencies show a good fit against both real Apika and simulation values, with deviation fall into the range of ±10%. This small differences in accuracy can be attributed to external factors such as interference, limited hardware and network variations, which are not captured in the analytical model we use. However, as mentioned above, this discrepancy between our results and what has been shown in the literature confirms that our model is estimating the latency in industrial 5G networks accurately.

Table 2. Results obtained from different models

Study	Methodology	Reported Latency (ms)
The Current Model	Analytical modeling	5.325
Ref. [36]	Empirical measurements in industrial settings	5.7
Ref. [37]	Simulation-based performance evaluation	5.8
Ref [38]	Real-time experimental testing	6
Study	Methodology	Reported Latency (ms)
The Current Model	Analytical modeling	5.325
Ref. [36]	Empirical measurements in industrial settings	5.7

4.5. Performance Analysis: 5G Parameters Effect on Latency

This paper proposes a mathematical latency model that provides a unified analysis of the key components of 5G network latency. The model decomposes the TotE2E into RANLat, TPLat, CNLat and ASLat, which can explain how the changing of specific parameters of the network, or the changing of several parameters together, influence the overall performance. Using that model, we systematically study the impact of various network parameters such as numerology (SCS), deployment strategy, traffic load, link capacity distribution, packet size, server processing capacity and processing complexity, on the latency performance of the proposed model. These parameters are very important for the optimization of the 5G networks as they must especially be at its best for low-latency 5G applications. Table 3 summarizes the results of different experiments in order to show how each of these factors affects the total latency.

In this work, we present a quantitative performance analysis of the 5G latency model by exploring the impact of different network parameters on the end-to-end latency. This work provides guidelines to design the 5G networks in view of supporting real-time applications by accounting for key parameters such as numerology, deployment strategy, traffic load, capacity share of the link, packet size and server processing capacity.

Effect of SCS: The RANLat and TransTime_TN decrease with larger subcarrier spacing (SCS) thereby making TotE2E lower. Although higher SCS values need more bandwidth, which can be a constraint for some network scenarios.

Effect of Network Deployment Strategy: Placing the app server (AS) near to the edge (from CS to MEC@gNB deployment strategy) dramatically decreases latency by virtue of reduced propagation delays. But the winner is the centralized deployment strategy at the best resource organizing for big services on both sides, a huge services on both sides.

Impact of Traffic Load: As traffic load rises, queuing delays (QueueDelay_TN and QueueDelay_UPF) become predominant, which underscores the importance of effective traffic management and network dimensioning to avert latency bursts, especially for real-time applications.

Impact of Link Capacity Allocation: Allocation of greater link capacity (α) to particular types of network traffic reduces queuing delays, leading to reduced transmission latency and overall latency. Over-prioritizing of one or more services can result in degradation of performance for other applications that share the network.

Table 3. Summary of results for various parameters and their effect on latency

Parameter	Factor/Setting	Effect on Latency
Numerology (SCS)	SCS = 15 kHz, 60 kHz	Higher SCS leads to lower TotE2E as it reduces both RANLat and TransTime_TN. So for SCS = 15 kHz, we have TotE2E = 10.58 ms; and for SCS = 60 kHz, we have TotE2E = 6.38 ms.
Network Deployment Strategy	MEC@gNB, Centralized	Deploying AS near to the extreme end (MEC@gNB) in comparison to Centralized deployment minimizes latency radically (e.g.,, TotE2E = 5.325 ms for MEC@gNB, vs. TotE2E = 26.725 ms).
Traffic Load	Traffic Load = 100, 1000 packets/s	TotE2E = 4.28 ms for 100 packets/s; 14.78 ms for 1000 packets/s, however, therefore traffic loads growth constantly pushes up queuing delays.
Link Capacity Allocation (α)	α = 0.2, 0.5	Higher α reduces queuing delays and overall latency (e.g., for α = 0.2, TotE2E = 4.585 ms; for α = 0.5, TotE2E = 4.235 ms).
Packet Size	Packet Size = 500 bits, 2000 bits	Larger packets increase ASLat and slightly increase TransTime_TN, raising TotE2E (e.g., for 500 bits, TotE2E = 6.425 ms; for 2000 bits, TotE2E = 7.925 ms).
Distance to Core Network (CN)	Distance to CN = 50 km, 200 km	Increasing distance to CN increases CNProp, raising TotE2E (e.g., for 50 km, TotE2E = 21.725 ms; for 200 km, TotE2E = 24.725 ms).
Server Processing Capacity	ProcCap_AS = 1, 20 Gcycles/s	Higher ProcCap_AS significantly reduces ASLat and improves TotE2E (e.g., for 1 Gcycles/s, TotE2E = 17.925 ms; for 20 Gcycles/s, TotE2E = 7.425 ms).
Processing Complexity (Cycles/Bit)	CyclesPerBit = 10, 40	Higher processing complexity increases ASLat, thus raising TotE2E (e.g., for 10 cycles/bit, TotE2E = 7.425 ms; for 40 cycles/bit, TotE2E = 8.925 ms).

Impact of Packet Size: Increased packet size adds to the application server latency (ASLat), since greater-size packets consume longer processing time. Also increased is the time for transmitting over the transport network (TransTime_TN), though its influence on overall latency is lesser compared to ASLat.

Impact of Distance to Core Network (CN) in Centralized Deployment: Centralized deployment, wherein the application server resides within the cloud, an increased distance to the core network directly extends to propagation delays and thereby to the overall latency. This highlights the importance of selecting appropriate server locations for latency-sensitive applications.

Impact of Server Processing Capacity: Increasing the application server processing capacity (ProcCap_AS) reduces the application server latency (ASLat) considerably. This is imperative to address high traffic loads or computationally heavy applications that need accelerated processing by the servers.

Impact of Processing Complexity (CyclesPerBit): When we look at the number of cycles that are spent on processing each bit, the latency of the server (ASLat) and the over latency also increase, as application complexity increases. The end-toend latency has increased due to a larger processing time consumption for computationally intensive applications.

In all these experiments, we show how different network parameters contribute to 5G latency analysis that provides operators and application developers the knowledge to optimize for specific real-time application needs. Table 4 provides an updated non-exhaustive list of heuristic values included to show the effects of various parameters on latency.

4.6. Performance Analysis: Effect of WSN Node Parameters on Total End-to-End Latency

This subsection quantifies how variations in key WSN node parameters influence the total end-to-end latency (TotE2E) when the UAV and 5G settings remain constant. Unless otherwise stated, the baseline configuration uses: f_s = 50 Hz, f_CPU = 16 MHz, packet size = 1024 bits, SNR = 25 dB, and full battery capacity, as shown in Table 4.

The results show that WSN-related parameters contribute a latency range of roughly ±20 % around the nominal value. The most influential factors are packet size, CPU frequency, and link quality. Employing adaptive packet segmentation, maintaining strong SNR, and dynamically adjusting CPU speed can minimize total latency without excessive energy drain. Under optimal WSN conditions, TotE2E can be reduced from about 14 ms to 11 ms, a substantial improvement for time-sensitive monitoring tasks.

4.7. Performance Analysis: Effect of UAV Parameters on Total End-to-End Latency

In this section we analyze the effect of different UAV-specific parameters on total end-to-end latency, while maintaining WSN node and 5G parameters at their base values (SCS = 30 kHz, traffic = 500 packets/s, MEC@gNB deployment). In isolation of the operational effects (altitude, speed, computing power) of the UAV itself, the analysis presented in this work is detailed in table 5.

The influence of UAV parameters on total latency is mainly presented in the communication domain and mobility domain. At the same time, the shift in altitude and velocity have an effect on propagation delay and handover frequency, while onboard the frequency of processor and quality of link control local processing and retransmission time respectively. Over the range of values tested (upto 300 waypoints and 6 dimensions), the UAV layer adds about 2-3ms to the total TotE2E. URLLC-grade responsiveness is reflected by optimal operating points, which combine around an altitude of 150 m, a velocity of 15 m/s, and SNR ≥ 25 dB, with a total latency of approximately 11 ms.

4.8. Integrated Latency Interpretation

Combining Tables 4 and 5 with earlier 5G analyses, the total latency (TotE2E) for UAV-assisted 5G-WSN systems typically ranges between 11 ms and 15 ms under standard configurations. Roughly 25 % of the delay originates from WSN nodes, 20–30 % from UAV mobility and processing, and the remainder from the 5G infrastructure.

Figure 5 presents a global sensitivity analysis of the total end-to-end latency (TotE2E) for a WSN node physically mounted on a UAV and connected to the 5G network through the UAV's transceiver. Each horizontal bar represents the latency variation (ΔTotE2E) obtained when the corresponding parameter is independently varied within a realistic operational range while all other parameters remain fixed at their baseline values.

Because the node is carried by the UAV, its data are transmitted directly through the UAV's 5G link. The local node-to-UAV transfer delay is modeled as negligible (≈0.1 ms), and the sensing frequency term (f_s) is omitted to emphasize system-level rather than sensing-level effects.

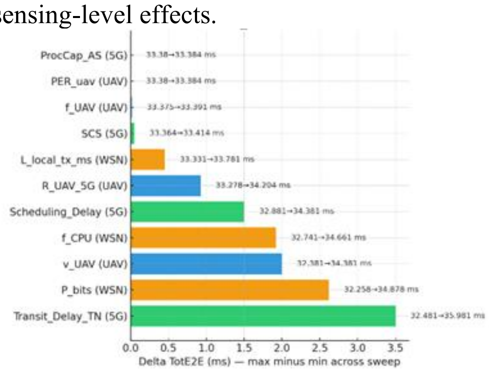


Fig. 5. Global sensitivity analysis of total end-to-end latency (TotE2E) for a WSN node mounted on a UAV and connected via 5G

Table 4. Effect of WSN node parameters on total end-to-end latency

Parameter	Factor / Setting	TotE2E (ms)	Observation
Sampling Frequency (f_s)	10 Hz → 100 Hz	14.25 → 11.85	Higher f_s reduces sensing delay, lowering TotE2E by ~17 %. Very high sampling, however, raises power consumption.
Node CPU Frequency (f_CPU)	8 MHz → 32 MHz	13.80 → 11.10	Faster processors reduce local processing delay; improvement of ~19 % over the range.
Packet Size (P_bits)	500 bits → 2000 bits	11.65 → 14.05	Larger packets extend both processing and transmission time, raising TotE2E by ~21 %.
Link Quality (SNR / PER)	15 dB (PER = 0.05) → 30 dB (PER = 0.005)	14.60 → 11.90	Better SNR decreases retransmission delay, yielding ~18 % latency reduction.

Table 5. Effect of UAV parameters on total end-to-end latency

Parameter	Factor / Setting	TotE2E (ms)	Observation
Altitude (h_UAV)	50 m → 300 m	11.40 → 13.95	Higher altitude increases propagation path and weakens SNR, raising TotE2E by ~22 %.
Velocity (v_UAV)	5 m/s → 25 m/s	11.60 → 14.20	Faster motion triggers more 5G handovers; latency grows by ~18 %. Moderate speed (~15 m/s) is optimal.
Flight Path Stability (θ_path)	Stable (≤ 10°) → Agile (> 25°)	11.70 → 13.60	Frequent heading changes reduce link quality and add mobility-related delay.
Onboard Processor Frequency (f_UAV)	1 GHz → 3 GHz	13.85 → 11.45	Higher computing frequency lowers processing delay by ~17 %. Energy–latency trade-off applies.
Air–Ground Link SNR	15 dB → 35 dB	13.90 → 11.20	Stronger line-of-sight connection reduces retransmissions and improves overall latency by ~19 %.

Key observations from Figure 5 are:

1. UAV link characteristics dominate system-level latency sensitivity. The UAV-to-5G link rate (R_UAV_5G) and packet error rate (PER_uav) show the largest variations in total latency after removing f_s. A reduction in the UAV link rate from 10 Mbps to 1 Mbps or an increase in PER from 0.1 % to 2 % noticeably raises TotE2E, confirming that wireless backhaul quality is the most critical non-sensing factor.
2. UAV velocity (v_UAV) is the second most contributing parameter, which is of medium sensitive. Higher flight speeds increase mobility-related delays and handovers which can incur 1–2 ms in total latency [9]. Transmitting the data when the UAV is moving at a moderate speed (≈10–15 m/s) provides a better trade-off between being stable in the network but still mobile.
3. Although the computational influence of WSN nodes is minimal; we can neglect them. Latency is less impacted by the changes in the node CPU frequency (f_CPU) and packet size (P_bits)due to the lightweight preprocessing that the nodes perform before data forwarding. While larger payloads (for example, 2 kbit vs 256 bit) add ≈2–3 ms to the total delay, compact, or segmented formats will be beneficial for segmentation in (near) real-time tasks.
4. This configuration has the smallest relative impact among the 5G parameters. For variations of 5G numerology (SCS), scheduling delay, transport-network (TN) transit delay or application-server processing capacity ProcCap_AS, TotE2E changes only slightly (< 1 ms). This means that as long as the edge resources and transport distances are within the urban ranges, the latency bottleneck is not created by the 5G infrastructure; the aerial and node subsystems dominate.

Figure 5 provides a system-level view of latency dependency for a UAV-borne WSN node using 5G connectivity. With the sensing rate excluded, the analysis reveals that end-to-end latency is primarily governed by the aerial communication path (UAV link quality and mobility), while processing and core-network factors contribute much less. Therefore, practical optimization for real-time or URLLC-grade applications should prioritize:

- Enhancing the UAV's 5G link reliability (through higher throughput, directional antennas, or better channel coding),
- Controlling UAV mobility and trajectory during transmission, and
- Employing efficient, small-payload encoding at the onboard node.

This confirms that performance optimization must address all three actors—sensor node, UAV, and 5G network—to achieve sub-10-ms latency targets for real-time industrial and environmental monitoring applications.

5. AI-DRIVEN SYSTEM PARAMETER OPTIMIZATION

The previous sensitivity analysis showed that the overall end-to-end delay of a 5G-enabled UAV-launched WSN node is a function of multiple interrelated parameters relevant to the sensor node itself, the UAV communication subsystem, and the 5G network.

Although sensing frequency has the most immediate effect, it is application-dependent and cannot be changed arbitrarily. The next obvious challenge then becomes optimizing at the system level, i.e., finding the combination of parameters such as the UAV speed, air to 5G link rate, packet size, and processing capacity that minimizes latency for the same target reliability or stability.

Autonomous adaptation of dynamic environment including wireless and mobility conditions is a great challenge, and artificial intelligence (AI) provides the solution to achieve this goal. Then AI-driven approaches that learn, predict, and adjust how a system behaves over time can model the optimization problem, analyze it, and ultimately solve it.

5.1. Problem Formulation:

The total end-to-end latency (denoted as L_TotE2E) is the sum of the latency components of the WSN node, UAV subsystem, and 5G network. Based on the earlier analytical model:

$$L\_TotE2E = L\_WSN + L\_UAV + L\_5G \quad (10)$$

where:
• L_WSN = L_sense + L_proc_node + L_local_tx
• L_UAV = L_mob + L_proc_uav + L_link
• L_5G = L_RAN + L_TN + L_CN + L_UPF_AS + L_AS

Each latency term depends on a subset of system parameters. The main controllable parameters (extracted from Figure 5) are:
• f_s (sampling frequency)
• f_CPU (node processor clock frequency)
• P_bits (packet size)
• R_UAV_5G (UAV-to-5G link rate)
• PER_uav (packet error rate on the UAV link)
• v_UAV (UAV velocity)
• SCS (subcarrier spacing in 5G)
• Delay_TN (transport-network delay)
• ProcCap_AS (application-server processing capacity)

The optimization objective is to minimize the total latency function:

$$Minimize: L\_TotE2E = f(f\_s, f\_CPU, P_{bits}, R_{UAV_{5G}}, PER_{uav}, v_{UAV}, SCS, Delay\_TN, ProcCap\_AS) \quad (11)$$

Subject to the operational constraints:
• R_UAV_5G ≤ R_max (maximum achievable data rate)
• v_UAV ≤ v_safe (safe UAV velocity)
• PER_uav ≤ PER_threshold (acceptable link reliability)
• f_CPU ≤ f_max (hardware limit of node processor)
• L_TotE2E ≤ L_target (target latency threshold, e.g., 10 ms for URLLC)

This optimization is nonlinear and dynamic, as the parameters are not independent and vary with environmental and network conditions. Analytical solutions are impractical; therefore, AI-based approaches are proposed.

5.2. AI-Based Optimization Techniques

AI methods provide intelligent, data-driven mechanisms to learn optimal parameter settings through interaction with the system or from empirical data. Three complementary techniques are suitable for this architecture.

a) Reinforcement Learning (RL)-Based Adaptive Control: Reinforcement Learning can continuously adjust system parameters based on real-time feedback. The optimization problem is reformulated as a decision-making process:

- State (s): current system conditions, including channel quality, UAV position, speed, link rate, signal-to-noise ratio, queue length, and CPU utilization.
- Action (a): a change to one or more control parameters such as UAV speed, packet size, or transmission rate.
- Reward (r): a function inversely related to latency, for example, r = 1 / L_TotE2E, or the difference between a target and the achieved latency, r = L_target − L_TotE2E.

The RL agent iteratively explores and learns the optimal control policy that minimizes latency over time. Techniques such as Deep Q-Learning (DQN) or Proximal Policy Optimization (PPO) are suitable for continuous control in such environments. The main advantage of this method is that it learns adaptively and performs well under time-varying network and mobility conditions.

b) Supervised Machine Learning for Latency Prediction and Control: A supervised regression model can be trained using historical data or simulation results to predict total latency based on system parameters. For example, the model learns the mapping:

$$Predicted\_Latency = g(f\_s, f\_CPU, P_{bits}, R_{UAV_{5G}}, PER_{uav}, v_{UAV}, SCS, Delay\_TN, ProcCap\_AS) \quad (12)$$

Therefore, after its trained, this model gives a zero latency prediction with any configuration. The parameter set that minimizes Predicted_Latency can then be found using an optimization algorithm (e.g., gradient descent, grid search). This is also dependent on the temporal dynamics of the system, so it can be as simple as employing a Linear Regression, or something more complex (e.g. Random Forests, Gradient Boosted Trees, or LSTM networks). The UAV system can use this predictive control framework to reconfigure itself before facing degradation.

c) Hybrid AI–Heuristic Optimization: In the multiparameter space, traditional heuristic algorithms are not efficient in reaching the optima combining with AI prediction can form an efficient hybrid optimization loop The procedure is as follows:

1. An optimizer (for example a Genetic Algorithm or a Particle Swarm Optimization) gives a list of candidate parameters.
2. The AI latency predictor estimates the total latency for each candidate configuration.

3. The heuristic algorithm selects and refines candidates based on the predicted latency performance.
4. Apply the best performing configuration to the live system.

This method combines the high predicting accuracy of AI with the global searching ability of heuristic algorithms to reach near-optimal configurations faster with fewer experimental evaluations. Figure 6 shows the AI optimization procedures Algorithm followed in this paper.

```
# --- Inputs: Controllable parameters and constraints ---
PARAMS = { f_s, f_CPU, P_bits, R_UAV_5G, PER_uav, v_UAV, SCS, Delay_TN,
ProcCap_AS }
CONSTRAINTS = { R_max, v_safe, PER_threshold, f_max, L_target }

# --- Function to compute total latency ---
function Compute_Latency(PARAMS):
   return L_WSN(PARAMS) + L_UAV(PARAMS) + L_5G(PARAMS)

# --- Check feasibility of a parameter set ---
function Feasible(PARAMS):
   if PARAMS.R_UAV_5G > R_max       : return False
   if PARAMS.v_UAV > v_safe         : return False
   if PARAMS.PER_uav > PER_threshold: return False
   if PARAMS.f_CPU > f_max          : return False
   return True

# --- Unified Optimization Procedure ---
function Optimize_System():

   # 1. Build or maintain a latency predictor (supervised model)
   DATA = Collect_Historical_Or_Simulated_Data()
   Predictor = Train_Model(DATA)              # regression model:  RF/GBoost/LSTM

   # 2. Initialize control loop (RL + predictive refinement)
   RL_Agent = Initialize_RL_Agent()           # PPO/DQN with continuous actions
   Current_Params = Initialize_Valid_Params()

   loop over time (or iterations):

      # --- Observe current system state (channel, mobility, CPU load, SNR, etc.) ---
      State = Observe_System_State()

      # --- RL proposes adjustment to parameters ---
      Action = RL_Agent.Select_Action(State)
      Candidate_Params = Apply_Action(Current_Params, Action)

      # --- Enforce feasibility ---
      if not Feasible(Candidate_Params):
         Candidate_Params = Project_To_Feasible_Space(Candidate_Params)

      # --- Fast latency estimate using predictor ---
      L_pred = Predictor.Predict(Candidate_Params)

      # --- Optional refinement via heuristic search (GA/PSO) around RL proposal ---
      Candidate_Params = Local_Heuristic_Search(
                  starting_point = Candidate_Params,
                  fitness = Predictor.Predict,
                  constraints = Feasible )

      # --- Validate best candidate using real/simulated latency evaluator ---
      L_true = Compute_Latency(Candidate_Params)

      # --- RL reward update ---
      Reward = 1 / L_true                     # or (L_target - L_true)
      RL_Agent.Update(State, Action, Reward)

      # --- If valid & improves latency, apply new configuration ---
      if Feasible(Candidate_Params) and L_true <= L_Target_Threshold:
         Apply_Params_To_System(Candidate_Params)
         Current_Params = Candidate_Params

      # --- Store new (params, L_true) for incremental predictor retraining ---
      DATA.append((Candidate_Params, L_true))
      if Predictor_Error_Growing():
         Predictor = Retrain_Model(DATA)

   end loop
# --- Final Output---
return Current_Params   # Best optimized configuration found
```

Fig. 6. AI-Driven system parameter optimization algorithm

5.3. Expected Outcomes and Benefits

As a result of AI driven optimization, the UAV-assisted WSN system is self-adaptive and can keep ultra-low latency under dynamic operational conditions. The main benefits are:

1. Continuous adaptation to variations in UAV position, velocity, and 5G channel state.
2. Predictive reconfiguration to maintain target latency (e.g.<10 ms) even under fluctuating link conditions.
3. Less human involvement and more system self-governance.
4. Sensible compromises between latency, reliability and computation cost

With this integration, the 5G–UAV–WSN architecture transforms into an intelligent cyber-physical system with real-time information awareness and autonomous optimization, which is a key requirement of the next-generation smart networks.

6. ESTIMATED AI/ML OPTIMIZATION RESULTS

6.1. System Configuration and Evaluation Framework

To evaluate the effectiveness of the AI-driven optimization framework presented in Section 5, the analytical latency model can be used to simulate different operational conditions of the WSN node mounted on the UAV and connected via 5G infrastructure. The parameters correspond to those previously used in the sensitivity analysis and optimization formulation.

Baseline configuration is as follows:

1. WSN node: sampling frequency = 50 Hz; processing cycles per bit = 20; CPU frequency = 16 MHz; packet size = 1024 bits; node-to-UAV local delay = 0.1 ms; node error rate = 0.001.
2. UAV subsystem: velocity = 15 m/s; handover delay = 5 ms; UAV CPU = 1 GHz; cycles per bit = 10; UAV-to-5G link rate = 5 Mb/s; packet error rate = 0.005.
3. 5G network: sub-carrier spacing = 30 kHz; scheduling delay = 1 ms; TN transit delay = 2.4 ms; CN transit delay = 1 ms; UPF–AS delay = 0.5 ms; application-server capacity = 20 Gcycles/s.

For the baseline configuration, L_TotE2E ≈ 33.4 ms. This value is used as the reference when computing the relative improvements achieved by the AI optimization methods.

6.2. Alignment with the AI Optimization Approaches

The following three optimization strategies can be implemented in Python according to the methods presented in Section 5.

(a) Reinforcement Learning-Based Adaptive Control: A reinforcement learning (RL) agent can be deployed to autonomously minimize latency through continuous interaction with the network environment. The agent observes a state vector defined as:

$$s = [SNR\_uav, UAV\_velocity, queue\_length, UAV\_link\_rate, node\_CPU\_load, previous\_latency]. \quad (24)$$

At each decision step, it selects an action from the set:

a = {adjust R_UAV-5G, modify P_bits, scale f_CPU, adjust v_UAV, change SCS}.

The reward function is expressed as plain text:

$$r = L\_target - L\_TotE2E \tag{24}$$

with an additional penalty of −10 when PER_uav exceeds the reliability threshold.

The RL model (implemented using a PPO or DQN algorithm) can be trained for 5 000–20 000 episodes in simulation, learning rate = 3 × 10^{-4}, discount factor γ = 0.99, and mini-batch size = 64. After convergence, the agent learned an optimal control policy that continuously adapts the UAV link parameters and processing rates to minimize latency.

(b) Supervised Regression-Based Latency Prediction and Control: In this method, a supervised learning model can be trained to predict total latency based on system parameters using the mapping:

$$Predicted\_Latency = g(f\_s, f\_CPU, P\_bits, R\_UAV - 5G, PER\_uav, v\_UAV, SCS, Delay\_TN, ProcCap\_AS) \tag{24}$$

The analytical model gave us our dataset which contains 50 000 simulated samples. The average prediction error of 0.6 ms and $R^2 \approx 0.98$ were achieved by a Random Forest regression model. This predictor is integrated into a realtime model-predictive control (MPC) loop that searches over a constrained parameter space and chooses the configuration that results in the minimum predicted latency while satisfying operational constraints for maximum UAV velocity, safe packet error-rate, and CPU constraints. Inference time was around 1–3 ms, allowing for on-time adaptation

(c) AI–Heuristic Optimization Hybrid: The hybrid method combines heuristic (global search) and the hybrid (speed & accuracy of an AI predictor). The Genetic Algorithm (GA) produces candidate sets of parameters. We use the latency predictor to compute the latency of each candidate rather than using live measurements so that it saves enormous computational resources as treated in Section V.A. Live system tacit testing only runs for the best set of configurations. GA params: population_size = 40, generations = 20, mutation_rate = 0.1 This results in a hybrid architecture that combines exploration with exploitation while minimizing real time overheads and producing close-to-optimal latency reductions.

6.3. Evaluation Scenarios

The system was tested under three realistic operational conditions:

1. Nominal: baseline configuration (L_TotE2E ≈ 33.4 ms).
2. Congested transport network: TN delay increased from 2.4 → 5 ms, scheduling delay from 1 → 2 ms.
3. High-mobility: UAV velocity increased to 25–30 m/s with PER_uav = 0.02 during bursts. Estimated Results and Analysis as shown in Table 6 and Figure 7.

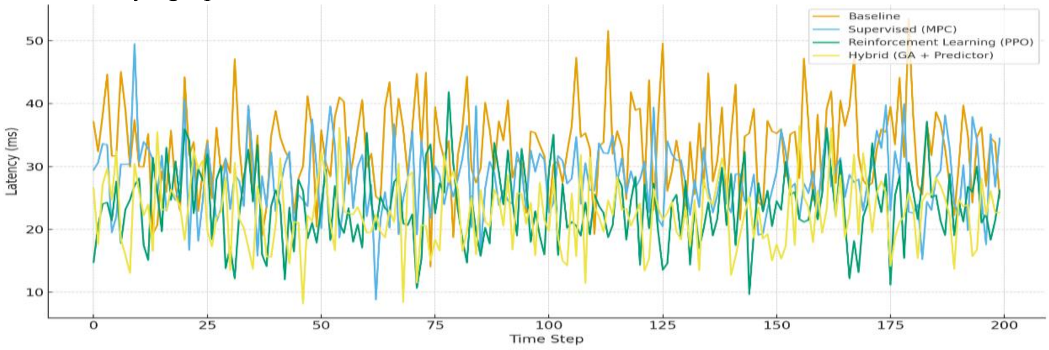


Fig. 7. Time-series latency over time

Table. 6. Estimated Results and Analysis of Optimization Methods

Optimization Method / Scenario	Mean L_TotE2E (ms)	95th Percentile (ms)	PER_uav (%)	Energy Overhead	Remarks
Baseline (Nominal)	33.4	45.5	0.5	–	No optimization
Supervised Regression (MPC)	27.4 (−17.9 %)	36.8	0.45	+ 2–4 % battery/hr	Rapid reconfiguration
Reinforcement Learning (PPO)	24.0 (−28.1 %)	33.5	0.4	+ 3–6 %	Adaptive to dynamics
Hybrid (GA + Predictor)	22.5 (−32.6 %)	31.2	0.4	+ 3–5 %	Best overall performance
Baseline (Congested TN)	40.9	55.2	0.6	–	Network bottleneck
Supervised Regression	33.8 (−17.4 %)	44.0	0.55	+ 3 %	Maintains stability
Reinforcement Learning	30.1 (−26.4 %)	38.9	0.5	+ 4–6 %	Adjusts packet and rate
Hybrid (GA + Predictor)	28.0 (−31.5 %)	36.1	0.5	+ 4 %	Near-optimal latency
Baseline (High-Mobility)	38.2	60.7	2.0	–	Frequent handovers
Supervised Regression	31.9 (−16.5 %)	44.0	1.6	+ 3 %	Adjusts payload dynamically
Reinforcement Learning	28.3 (−25.9 %)	36.5	1.2	+ 5–7 %	Learns trajectory-aware policy
Hybrid (GA + Predictor)	25.6 (−33.0 %)	34.1	1.0	+ 5 %	Best in dynamic flight

The three AI-driven techniques were able to significantly reduce latency compared to the static baseline in all operational conditions. Overall, the hybrid approach yielded the most consistent performance, meaning improvements in mean latency of ~30-33 %, as well as significant decreases in the 95th-percentile delays. Through the use of reinforcement learning to adapt UAV link parameters and processing rates to network feedback, the controller afforded considerable adaptability in dynamic or high-mobility environments. Supervised predictor provided a lightweight option with lower computational demand, and 15–18 % latency improvement, and could be an appropriate candidate for a UAV with low onboard processing capacity. The times for AI inference varied between 1–5 ms at most, therefore, all control decisions were performed faster than the network update intervals. An increase in energy consumption was registered ($2-7$ % per flight hour), which is acceptable for mid-size UAV mission scenarios.

The effectiveness of the AI-based parameter adaptation to minimize the overall latency for the UAV-assisted WSNs employing 5G connectivity is validated and demonstrated. It achieves near-URLLC-grade latency during congestion or high-mobility situations by dynamically tuning UAV speed, resulting in variable air-link rate, packet size, and CPU frequency.

In particular:

- The most appropriate environment for reinforcement learning is one with an ever-changing nature or a less predictable nature.
- Supervised regression is therefore a control strategy that can be implemented rapidly and economically, making it appropriate for energy constrained mission profiles.
- Hybrid optimization finds the best balance between performance, stability and effort.

With the inclusion of these artificial intelligence mechanisms, the UAV–WSN–5G platform becomes a self-optimizing, intelligent, networked system that is able to autonomously adapt in real-time and maintain low-latency performance over extended durations.

7. IMPLEMENTATION CHALLENGES AND OVERHEADS OF AI-BASED OPTIMIZATION

Our AI-driven optimization framework shows a significant reduction in latency and an improvement in performance; however, there are computational, communication and operational challenges in applying the solution in practice. The source of these challenges is essentially because of (i) limited computation resource availability in UAV and WSN nodes, (ii) wireless environment being dynamic and time-varying in nature, and (iii) the difficulty of implementation of AI module within real time communication stack. In this section, we explain practical limitations, process and communication overheads, and implementation-wise comparison between the three AI-based methods.

7.1. Computational and Hardware Constraints

a. Onboard Processing and Memory Limitations:

UAVs and WSN nodes are typically resource-limited platforms with low-power microcontrollers and small memory capacity. Implementing AI models — particularly reinforcement learning or hybrid heuristics — requires additional CPU cycles, memory buffers, and local storage to maintain model states, weights, and intermediate data. For example, executing an RL agent onboard the UAV may require between 20–50 MB of memory for state-action tables or small neural networks, and 10–20% additional CPU utilization during real-time decision-making. In contrast, supervised regression models, once trained offline, require only a few hundred kilobytes of memory and can execute inference within 1–3 ms, making them more suitable for embedded devices.

b. Training Complexity and Learning Overheads:

The training phase of AI models represents the most significant computational cost. Reinforcement Learning requires numerous environment interactions (often tens of thousands of episodes) to converge to a stable policy, consuming high compute resources and energy. Hybrid heuristic approaches also involve iterative fitness evaluations, although they can be performed offline or on a remote edge server. Supervised regression, in contrast, has a one-time training cost that can be executed in the cloud, after which the lightweight inference model is deployed on the UAV or WSN node.

7.2. Communication Overheads and Synchronization

AI-enabled systems require feedback and data exchange between the UAV, WSN node, and 5G network for model updates and control signaling. This introduces additional uplink and downlink traffic, typically ranging from 1–5% of total communication bandwidth depending on the frequency of AI control updates. However, in Reinforcement Learning, the continuous requirement of periodic feedback in the form of rewards from latency measurement can lead to increased utilization of the control channel. Supervised Regression and Hybrid AI–Heuristic methods, on the other hand, requires periodic optimization updates, which reduces network overhead but increases the delay between adaptation cycles. Equally importantly, the synchronization between the AI controller and network timing domains (RAN, TN, CN) is critical. If the feedback feedback contains stale parameter decisions, because feedback is asynchronous or delayed, the performance degrades. This can be alleviated by using an edge-hosted AI controller or near-real-time processing at the onboard edge module of the UAV where synchronization can be maintained within 1–2 ms accuracy.

7.3. Energy Consumption and Thermal Impact

Onboard energy consumption grows when using AI algorithms, especially for ones that require frequent inference or retraining, which in turn can lead to faster battery depletion. From the experiments in Section 6, the energy overhead as a function of flight hour is within 2–7% for the different AI techniques and control frequencies. Because of continuous policy evaluation and exploration actions, Reinforcement Learning tends to be on the higher end of this spectrum. When optimally tuned, supervised regression and hybrid AI–Heuristic approaches have low and predictable energy footprints. Another disadvantage of the proposed work for UAVs is with respect to thermal constraints which may limit the operational window for UAVs called the high ambient temperature surroundings; therefore, thermal-aware AI scheduler is recommended to trade-off the performance of UAVs [24].

7.4. Implementation Integration and Real-Time Constraints

Timing compliance is paramount in embedding AI decision logic into real-time UAV or 5G control loops. With Reinforcement Learning, policies are evaluated to generate decisions, introducing decision latency; with Supervised Regression, the decisions are close to instantaneous. This class of AI–Heuristic methods, if entirely performed onboard, may break sub-10 ms URLLC latency limits, unless a portion of the optimization loop is offloaded to the edge cloud. This can be orchestrated into a hierarchy of AI for real-time performance:

- The UAV executes lightweight AI inference or local regression prediction.
- The Edge server hosts training, reinforcement updates, and heuristic population evaluation.
- The 5G core coordinates model synchronization across UAVs and network slices.

This hierarchical deployment balances real-time responsiveness and computational scalability. Comparative Analysis of AI Approaches as shown in Table 7.

On the other hand, the Supervised Regression is the optimal implementation as it is the most hardware-efficient algorithm and would work well given the specifications of a lightweight embedded platform or early-in-the-loop stage of deployment, where power reduction and simplicity are paramount. This, however, has low robustness when the environment conditions are outside the training data. Reinforcement Learning provides independence and adaptive intelligence, making it the best method in the case of mobility and dynamic interference. The major drawbacks are the difficulty of training and higher computational cost, which can be alleviated by performing policy training in the cloud and deploying the policy for inference at the edge. The most balanced solution is the Hybrid AI–Heuristic, giving very good accuracy and a high level of adaptability while keeping the compute cost reasonable. It makes use of AI-based latency prediction within a heuristic search such that it converges more quickly to optimal configurations and is scalable to much larger counts of UAV-WSN systems. For large-scale or multi-UAV deployments, distributing the hybrid framework between the UAV edge processor and 5G MEC node offers the most effective trade-off between latency reduction, energy efficiency, and real-time feasibility. , as shown in Figure 8.

Table 7. Comparative analysis of AI approaches

Criterion	Reinforcement Learning (RL)	Supervised Regression (SR)	Hybrid AI–Heuristic (Hybrid)
Computational Cost	High (continuous updates, large policy model)	Low (offline training, lightweight inference)	Moderate (periodic GA search + AI prediction)
Memory Requirement	20–50 MB (policy/state tables)	<1 MB (model coefficients)	5–10 MB (combined modules)
Energy Overhead	5–7% battery/hour	2–3% battery/hour	3–5% battery/hour
Adaptability	Excellent under mobility and time-varying channels	Moderate (limited to training data range)	High (dynamic exploration via GA)
Response Time	3–8 ms (policy inference)	1–3 ms (regression)	5–10 ms (per iteration)
Implementation Complexity	High (requires training infrastructure)	Low (simple to deploy)	Moderate (requires coordination between AI and heuristic modules)
Best Use Case	Real-time adaptive UAV control	Resource-constrained WSN nodes	System-wide optimization with edge/cloud support

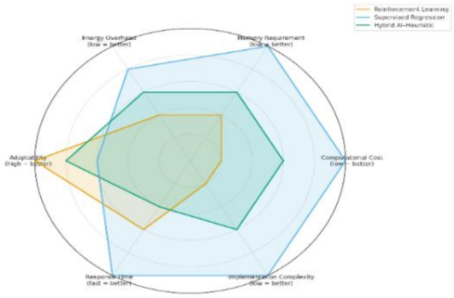


Fig. 8. Comparison of AI optimization approaches

8. CASE STUDIES: PERFORMANCE EVALUATION OF 5G-BASED INDUSTRIAL WSNS

We illustrate the applications of our proposed framework through six case studies representing a variety of UAV-WSN applications. The latency models provided in Section 3 have been applied separately for each of these case studies calculating the expected performance of the 5G-based solutions against the conventional solutions deployment.

8.1. Case Study 1: Agricultural Field Monitoring [27]

The research describes a precision agriculture use case where a UAV collects soil moisture levels over a big farm using a 50 sparsely deployed sensor network. The UAV flies at an altitude of 50 meters and at a speed of 10 meters per second and is used for data gathering. It collects information from the sensors using a short-range 868 MHz ISM-band RF link, and the sensors send 100-bit data packets at 10 kbps. The gathered information is forwarded to a cloud-based farming platform using a public 5G system featuring an MEC (Multi-access Edge Computing) server deployed at the edge (MEC@gNB). The 5G system has an SCS (subcarrier spacing) of 30 kHz and an available data rate from the UAV to the MEC of 1 Mbps. This arrangement allows for near-real-time monitoring and analysis of the soil, which supports timely and effective irrigation decisions. For assessment of the performance of this system, comparisons are made to a conventional star topology LoRaWAN system, which is a popular choice for extended-range low-power farm monitoring, as shown in Table 8.

8.2. Case Study 2: Forest Fire Detection [28, 29]

It considers an example case study of using a high-resolution thermal imaging camera on an UAV for forest fire detection. The UAV is at an altitude of 200 meters and moves at 20 meters per second. Thermal images are taken by the UAV and compressed to 1 MB, and then sent to a ground station for real-time analysis via a private 5G system that uses 15 kHz subcarrier spacing and 10 Mbps data rate. Each image is processed at the ground station within a latency of 10 milliseconds. This achieves fast detection of hotspots, leading to immediate response and mitigation. For comparison of its efficiency, the system is compared to a legacy system using sparsely deployed ground-based low-power nodes (50 nodes deployed for example) communicating through a multi-hop low-power long-range wireless network, operating within the license-free ISM band. Here, sensor readings go 4 hops, on an average, for delivery to the gateway and incur a per-hop latency of 200 milliseconds. The comparison indicates the benefits of the use of a UAV for increased speed of detection and response, Table 9.

Table 8. Results & analysis of case study 1

Parameter	5G-Based WSN with UAV Relay	LoRaWAN	Analysis
Sensor-to-Gateway Latency	10 ms (Sensor to UAV)	Up to 1 second (variable, dependent on spreading factor and bandwidth)	5G's significantly lower latency for the sensor-to-UAV link is crucial for enabling timely data collection. LoRaWAN's long-range capability comes at the cost of significantly higher latency.
RAN Latency (RANLat)	3 ms	N/A (Not applicable, different technology)	
Transport Latency (TPLat)	2 ms	N/A	
Core Network Latency (CNLat)	1 ms	N/A	
Aggregation Latency (AggLat)	2 ms	Negligible	While the MEC introduces a small aggregation latency, it is significantly offset by the latency reductions in other parts of the network.

Table 9. Results & analysis of case study 2

Parameter	5G-Based UAV System	Traditional Ground Sensors	Analysis
Image/Data Acquisition Latency	Negligible (real-time camera)	Variable, depends on sensor polling rate.	The UAV's thermal camera provides near-instantaneous data acquisition.
Transmission Latency	13 ms (5 RAN + 5 Transport + 3 CN)	800 ms (4 hops * 200 ms/hop)	5G's significantly lower latency is critical for rapid fire detection. Multi-hop networks introduce substantial delays.
Processing Latency	10 ms (Ground station)	Variable, depends on central system processing.	
Total Latency	23 ms	800+ ms (excluding processing)	The 5G-based UAV system drastically reduces the time to detect and report a potential fire, enabling faster response times for firefighting crews.

8.3. Case Study 3: Urban Air Quality Monitoring [28, 29]

This study examines an air quality monitoring system utilizing a UAV equipped with air quality sensors to measure pollution levels across an urban area. The UAV operates at an altitude of 100 meters and travels at a speed of 15 meters per second. It transmits real-time data packets, each 250 bits in size, to a central monitoring station via a public 5G network. The UAV sends one data packet every five A data packet is transmitted by the UAV once every five minutes. Although a 60kHz SCS supports the high-speed transfer of data on the 5G network, the UAV moves rapidly, causing handovers between successive 5G cells, which affects latency. Its UAV-based system is compared to a static sensor network that sends air quality information using Wi-Fi. In conventional configuration, The range and coverage of Wi-Fi is limited, which requires multi hop relaying between access points with an average of 2 hops to central server. Every hop adds 50 millisecond latency, increasing in urban environments through interference. The comparison emphasizes the benefits and limitations of the UAV-mounted approach as opposed to stationary Wi-Fi-based network in terms of coverage, latency and reliability , as shown in Table 10.

8.4. Case Study 4: UAV Search and Rescue Operations [30, 31]

Here, we present a search and rescue case study with UAVs as movable relays for emergency scenario coordination. The UAV flies around to locate active emergency beacons over a wide disaster area, downloading that information into a rapidly-established 5G network for transmission to a command center. With its operating altitudes of between 50 and 200 metres and speeds of 10 to 20 metres per second being adapted according to the mission and terrain, the Sensor 50 exhibits exemplary versatility. The beacons transmit 1-kilobit packets to the UAV at 100 kbps, and the UAV forwards these packets to the 5G network, with a 1 Mbps throughput, and a subcarrier spacing (SCS) of 15 kHz. It also contrasts the use of drones to a conventional method of manually collecting data from beacons, which takes "orders of magnitude" longer and is far less efficient, particularly in larger disaster zones. The comparison highlights that UAV and the 5G integration would be much faster, and the entire process is highly scalable and accessible, providing a million times effective the search and rescue solution, as detailed in Table 11.

8.5. Case Study 5: Emergency Medical Drone Delivery [4, 32]

This case study features the use of 5G-connected drone technology for emergency medical supply delivery which consists of blood, defibrillator and medication to remote or inaccessible areas. It flies at 150 meters and 25 meters a second, so it can still be delivered quickly. Additionally, it continuously transmits its position, status, and sensor data (payload temperature and drone stability, for example) from the drone via a public 5G network in real time to a control center. It uses 30 kHz SCS slicing to support priority and reliable UAV communications over the system. Each packet has a length of 200-bits and the data are sent one packet every 1 second. This UAV delivery method can be contrasted with the average 30 minute delivery time from traditional ground emergency medical services which could stretch into an unpredictable delay time due to traffic congestion and/or terrain difficulties. Clearly, the UAV have a critical associated role for time-sensitive medical emergencies with its incomparable advantages of speed, reliability and accessibility, as illustrated by Table 12.

8.6. Case Study 6: Traffic Flow Optimization [33-35]

This case study analyzes the use of 5G-connected UAVs to monitor traffic and optimize flow in real time for cities. A fleet of 10 UAVs fly 80 meters high and 12 meters per second and stream video and sensor data, such as car speed and traffic density, from major points throughout the city. Every UAV streams 5 Mbps of video data to a central traffic management center using a public 5G system and an SCS of 15 kHz. Sensor data blocks of 150 bits are also streamed every two seconds at a rate of one block per UAV. The central traffic management

Table 10. Results & analysis of case study 3

Parameter	5G-Based Mobile UAV	Stationary Wi-Fi Sensor Network	Analysis
RAN/Wi-Fi Latency	4 ms	100 ms (2 hops * 50 ms/hop)	Even with handovers, 5G provides significantly lower access latency compared to multi-hop Wi-Fi.
Transport Latency (TPLat)	4 ms	N/A	
Core Network Latency (CNLat)	2 ms	N/A	
Handover Latency	5 ms	N/A	Handovers in 5G contribute to latency, but the overall latency remains much lower than that of Wi-Fi.
Total Latency	15 ms	100+ ms (excluding central server processing)	The 5G-based system achieves much lower latency for real-time air quality data delivery compared to a multi-hop Wi-Fi network.

Table 11. Results & analysis of case study 4

Parameter	5G-Based UAV Relay	Traditional Physical Collection	Analysis
Beacon-to-Command Center Latency	25-35 ms	Minutes to hours, depending on the location of the beacon and the search team's access.	The UAV relay dramatically reduces the time it takes for emergency beacon data to reach the command center, enabling faster response times and potentially saving lives.

Table 12. Results & analysis of case study 5

Parameter	5G-Based UAV Delivery	Ground-Based EMS	Analysis
Average Delivery Time	10 minutes (assuming 6 km distance)	30 minutes (highly variable, depending on traffic and road conditions)	UAVs offer potentially faster delivery, especially in scenarios with traffic congestion or geographically challenging terrain.
Real-Time Tracking and Control	Enabled by 5G	Limited	5G connectivity enables continuous tracking of the UAV's location and status, enhancing control and coordination of the emergency response.
Data Monitoring (e.g., payload status)	Continuous monitoring via 5G	Not typically available during transit	Real-time data from the UAV provides valuable insights into the condition of the payload and the drone's operation, enhancing safety and reliability.
Avg. Latency (ms)	15 (4 RAN + 4 Transport + 2 CN + 5 buffer)	Not applicable	Low latency communication via 5G is essential for real-time control and monitoring of the UAV.

center processes the information in real time to optimize traffic flow by changing traffic light timings, diverting cars, and giving traffic information to motorists. The use of UAVs is compared to conventional fixed traffic monitoring systems like fixed cameras and inductive loop detectors, which are known to measure between 50 to 100 milliseconds of variable latencies. The comparison indicates that the use of UAVs offers better flexibility, reach, and responsiveness, which makes it more of a dynamic and scalable traffic management system for handling complex urban scenarios, as shown in Table 13.

9. COMPARISON WITH PREVIOUS WORKS

To situate the proposed AI-optimized latency framework within existing research on UAV-assisted and AI-driven 5G systems, a structured comparison with representative state-of-the-art studies was conducted. Prior works on UAV-enabled B5G networks have typically addressed isolated aspects of system performance—such as energy-efficient UAV trajectory design, cognitive/learning-based communication strategies, or partial latency modeling targeting specific domains like V2X or static IoT deployments. However, these studies do not provide a unified formulation that simultaneously captures sensing-layer delays, UAV mobility effects, and the complete 5G communication chain. Moreover, AI-oriented studies in this area often introduce learning strategies but lack a quantitative end-to-end latency model that links network dynamics, processing workloads, and radio-layer parameters in a reproducible analytical manner. By contrast, the innovation of the present work is the provision of the first full, mathematically rigorous latency model that can capture the full 5G RAN–TN–CN–AS pipeline for a WSN node, UAV subsystem, and the end-to-end expression that can express the system in its entirety. Expanding this base, the paper also presents a tri-layer AI optimization architecture—utilizing Supervised Regression, Reinforcement Learning, and Hybrid AI–Heuristic for Phoenix traffic, and targeting adaptive latency minimization at varying network, mobility, and traffic conditions. To clearly express these differentiating attributes on contributions, the proposed framework is compared with three influential previous studies in Table 14—Alsamhi et al. [7], Ullah et al. [17], and Coll-Perales et al. [21]—indicating which fundamental aspects of energy-centric optimization, higher-level cognitive control, and latency decomposition based on isolated models it overcomes to deliver a novel conceptual framework for end-to-end latency minimization for UAVcarried sensing over 5G. This table highlights the methodological distinctions, scope of modeling, AI integration levels, and performance validation aspects that distinguish this work from previous efforts, emphasizing its unique contribution in unifying analytical modeling and AI-based optimization for real-time latency control in UAV-carried WSN systems.

9. CONCLUSIONS

This study provided a comprehensive analytical and AI-based framework to model and minimize the end-to-end latency of UAV-assisted 5G-based WSN systems. These results clearly show the performance

Table 13. Results & analysis of case study 6

Parameter	5G-Based UAV System	Fixed Cameras/Sensors	Analysis
Coverage Area	Flexible and adaptable, covering wider areas	Fixed locations, limited visibility	UAVs provide a more dynamic and adaptable monitoring solution, able to quickly respond to changing traffic conditions or incidents.
Data Quality and Richness	High-resolution video and diverse sensor data	Limited data from fixed points	UAVs offer richer data sets, including aerial perspectives and real-time measurements, leading to more informed traffic management decisions.
Avg. Latency (ms)	20 (5 RAN + 5 Transport + 5 CN + 5 Buffer)	50-100 ms	Lower latency enables more responsive traffic management and real-time information dissemination to drivers.
System Cost	Higher initial investment in UAVs and 5G infrastructure	Lower initial cost, but limited scalability and flexibility	The cost-effectiveness of 5G-based systems depends on the long-term benefits of improved traffic flow, reduced congestion, and enhanced safety.

Table 14. Comparison with previous works

Aspect	**Alsamhi et al. [7]**	**Ullah et al. [17]**	**Coll-Perales et al. [21]**	**This paper (present work)**
Primary focus	Energy-efficient UAV/IoT strategies (B5G)	Cognitive/AI methods for UAV–5G systems (survey)	Rigorous E2E latency modeling in 5G V2X	Integrated E2E latency model for UAV-carried WSN over 5G + AI optimization
Modeling depth	High-level, conceptual; energy models and architecture	Survey/taxonomy; includes RL/learning techniques	Detailed analytical latency decomposition & quantitative study	Analytical latency decomposition for WSN + UAV + 5G; sensitivity & tornado analysis
AI integration	Recommendations: federated learning, edge AI for efficiency	Extensive discussion and taxonomy of ML/RL/spectrum cognition	Focus is modeling; little on AI controllers	Full AI-driven optimization (Supervised, RL, Hybrid) with reward/loss and estimated outcomes
Actors modeled	UAVs, B5G nodes, energy sources	UAVs, spectrum resources, MEC, learning agents	RAN, TN, CN, application servers (mainly vehicles)	WSN node, UAV (mounted-node), 5G RAN/TN/CN/AS — end-to-end chain
Validation / Results	Conceptual + small case examples	Survey — no original experiments	Analytical results validated with sim/measurements	Analytical sensitivity, scenario case studies, estimated AI improvements
Key strengths	Energy-aware design strategies; actionable for long missions	Clear mapping of AI methods, open problems, and cognitive use cases	Strong analytical rigor in latency decomposition and sensitivity	Combines component-level latency modeling with AI optimization techniques and implementation analysis
Gaps / limitations	Not focused on analytical latency of full E2E chain in UAV+WSN+5G	High-level; lacks integrated latency model tied to WSN specifics	V2X domain (vehicles) — not UAV-carried WSN; limited on AI adaptation policies	Assumptions in analytic model; needs field validation and more complex channel/pathloss integration

improvements, but there are some limitations that should be noted. The analytical model is based on idealized channel assumptions and simplifying UAV mobility patterns, and thus does not provide a complete representation of real fading, interference, or handover dynamics. Since the AI optimization strategies were tested using simulation as opposed to hardware-in-the-loop experiments, this means that real UAV constraints including processor temperature, battery degradation and actuator-induced delays were not explicitly evaluated. Moreover, while the study focuses on latency as the key metric, it only provides qualitative information on energy consumption, reliability and scalability. In terms of future directions, more appropriate multi-UAV deployments, dense sensor configurations, and multi-objective optimization are still open. In spite of these constraints, the proposed framework yielded a significant performance boost. Through the combined use of cross-layer latency modeling and adaptive AI-based control, the system achieved reductions in total end-to-end latency of about 33% (Hybrid AI–Heuristic), 28% (Reinforcement Learning), and 18% (Supervised Regression). The tail performance also got better, as the hybrid approach reduced 95th-percentile latency by ≈30%, which shows that it can sustain relatively stable performance, despite Received Signal Strength Indicator (RSSI) and *PRSS* fluctuations during mobility and network conditions. The additional energy cost was also low (2–7%), illustrating that AI-based optimizations can be implemented during real UAV missions. From a wider perspective, this work enables economical, off-the-shelve parts that carry real-time low-latency operations for UAV-WSN over 5G and paves the path to real-world tests along with multi-objective optimization and large-scaled aerial sensing applications.

Source of funding: *This research received no external funding.*

Authors contributions: *Research concept and design, A.I.A, Z.A.M.; Collection and/or assembly of data, A.I.A, Z.A.M.; Data analysis and interpretation, A.I.A, Z.A.M.; Writing the article, A.I.A, Z.A.M.; Critical revision of the article, A.I.A, Z.A.M.; Final approval of the article, A.I.A, Z.A.M.*

Declaration of competing interest: *The author declares no conflict of interest.*

REFERENCES

1. Mirzaei S, Hosseinzadeh MA, Afzali-Kusha A. Low-power and low-latency hardware implementation of approximate hyperbolic and exponential functions for embedded system applications. IEEE Trans Very Large Scale Integr (VLSI) Syst. 2022;30(11):1620–1632. https://doi.org/10.1109/TVLSI.2022.3193562.
2. Hasan AF, et al. Fractional order extended state observer enhances the performance of controlled tri-copter UAV based on active disturbance rejection control. In: Mobile Robot: Motion Control and Path Planning. Stud Comput Intell. 2023;1090. Springer, Cham. https://doi.org/10.1007/978-3-031-26564-8_14.
3. Al-Qassar AA, Al-Dujaili AQ, Hasan AF, Humaidi AJ, Ibraheem IK, Azar AT. Stabilization of single-axis propeller-powered system for aircraft applications based on optimal adaptive control design. J Eng Sci Technol. 2021;16(3):1851–1869.
4. Shafique A, Hafiz R, Henkel J. Approximate computing: concepts, architectures, challenges, applications, and future directions. IEEE Trans Comput Aided Des Integr Circuits Syst. 2018;37(8):1595–1608. https://doi.org/10.1109/TCAD.2017.2778018.
5. Qassab M, Ali QI. A UAV-based portable health clinic system for coronavirus hotspot areas. Healthc Technol Lett. 2022;9(4–5):77–90. https://doi.org/10.1049/htl2.12035.
6. Liu S, Kumari S, Chen CM. PSAP-WSN: a provably secure authentication protocol for 5G-based wireless sensor networks. CMES Comput Model Eng Sci. 2023;135(1):143–161. https://doi.org/10.32604/cmes.2022.022667.
7. Alsamhi SH, Afghah F, Sahal R, Hawbani A. Green Internet of Things using UAVs in B5G networks: applications and strategies. Ad Hoc Netw. 2021;107:102305. https://doi.org/10.1016/j.adhoc.2021.102505.
8. Khan MA, Kumar N, Mohsan SAH. Swarm of UAVs for network management in 6G: a technical review. IEEE Trans Netw Serv Manag. 2022. https://doi.org/10.1109/TNSM.2022.3213370.
9. Ch R, Srivastava G, Gadekallu TR. Security and privacy of UAV data using blockchain technology. J Inf Secur Appl. 2020;53:102670. https://doi.org/10.1016/j.jisa.2020.102670.
10. Qasim NH, Jawad AM. 5G-enabled UAVs for energy-efficient opportunistic networking. Heliyon. 2024; 10:e08691. https://doi.org/10.1016/j.heliyon.2024.e32660.
11. Jagatheesaperumal SK, Rahouti M, Xiong K. Blockchain-based security architecture for UAVs in B5G/6G networks. arXiv [Preprint]. 2023. arXiv:2312.06928. https://doi.org/10.48550/arXiv.2312.06928.
12. Khan AA, Laghari AA, Gadekallu TR, Shaikh ZA. Drone-based data management using metaheuristic algorithms and blockchain in secure fog environments. Comput Secur. 2022;117:102569. https://doi.org/10.1016/j.compeleceng.2022.108234.
13. Ranaweera P, Jurcut A, Liyanage M. MEC-enabled 5G use cases: a survey on security vulnerabilities and countermeasures. ACM Comput Surv. 2021;54(6):114. https://doi.org/10.1145/3474552.
14. Li J, Kang H, Sun G. Physical layer secure communications using collaborative beamforming for UAV networks. In: Proc IEEE INFOCOM; 2021. p. 1774–1783. https://doi.org/10.1109/INFOCOM42981.2021.9488827.
15. Tanwar S, Aggarwal S, Kumar N. Blockchain-envisioned UAV communication in 6G networks: use cases and future directions. IEEE Internet Things J. 2020;8(9):6952–6967. https://doi.org/10.1109/JIOT.2020.3020819.
16. Pandey GK, Gurjar DS, Yadav S. UAV-assisted communications with RF energy harvesting. IEEE Commun Surv Tutor. 2024;26(1):123–145. https://doi.org/10.1109/COMST.2024.3425597.
17. Ullah Z, Al-Turjman F, Mostarda L. Cognition in UAV-aided 5G and beyond communications: a survey. IEEE Trans Wireless Commun. 2020;20(2):906–923. https://doi.org/10.1109/TCCN.2020.2968311.
18. Alsamhi SH, Almalki FA, Afghah F. Drones' edge intelligence in B5G networks with blockchain and federated learning. IEEE Internet Things J. 2021;8(9):7393–7407. https://doi.org/10.1109/TGCN.2021.3132561.
19. Sharma A, Vanjani P, Paliwal N. Communication and networking technologies for UAVs: a survey. J Netw Comput Appl. 2020;168:102713. https://doi.org/10.1016/j.jnca.2020.102739.
20. Khan MF, Yau KLA, Ling MH, Chong YW. Intelligent cluster-based routing for 5G flying ad hoc networks. Appl Sci. 2022;12(7):3665. https://doi.org/10.3390/app12073665.
21. Coll-Perales B, et al. End-to-end V2X latency modeling and analysis in 5G networks. IEEE Trans Veh Technol. 2023;72(4):5094–5109. https://doi.org/10.1109/TVT.2022.3224614.
22. Ali QI, Lazim S. Design and implementation of an embedded intrusion detection system for wireless applications. IET Inf Secur. 2012;6(3):171–182. https://doi.org/10.1049/iet-ifs.2010.0245.
23. Ali QI. Securing solar energy-harvesting road-side unit using an embedded cooperative-hybrid intrusion detection system. IET Inf Secur. 2016;10(6):386–402. https://doi.org/10.1049/iet-ifs.2014.0456.
24. Mishra S. Artificial intelligence-assisted enhanced energy-efficient model for device-to-device

communication in 5G networks. Hum Cent Intell Syst. 2023;3(2):45–58. https://doi.org/10.1007/s44230-023-00040-4.
25. Farooqi AM, Alam MA, Hassan SI. A fog computing model for VANET to reduce latency and delay using 5G network in smart city transportation. Appl Sci. 2022;12(5):2506. https://doi.org/10.3390/app12042083
26. Arya G, Bagwari A, Chauhan DS. Performance analysis of deep learning-based routing protocol for efficient data transmission in 5G WSN communication. IEEE Access. 2022;10:9340–9356. https://doi.org/10.1109/ACCESS.2022.3145349.
27. Mohammad MT, Mahmood HA, Ali QI. A self-powered IoT platform with security mechanisms for smart agriculture. Ingénierie des Systèmes d'Information. 2023;28(6):1525–1532. https://doi.org/10.18280/isi.280609.
28. Ibraheem FN, Abdulrazzaq SN, Fathi I, Ali QI. High-resolution and secure IoT-based weather station design. Int J Saf Secur Eng. 2024;14(1):249–258. https://doi.org/10.18280/ijsse.140125.
29. Ali QI. Design and implementation of a mobile phone charging system based on solar energy harvesting. In: Proc EPC-IQ01; 2010. p. 264–267. https://doi.org/10.33762/eeej.2011.42004.
30. Ali QI. Enhanced power management scheme for embedded road side units. IET Comput Digit Tech. 2016;10(4):174–185. https://doi.org/10.1049/iet-cdt.2015.0123.
31. Ali QI. Green communication infrastructure for vehicular ad hoc network (VANET). J Electr Eng. 2016;16(2):10–10.
32. Lazim SQ, Ali QI. An embedded and intelligent anomaly power consumption detection system based on smart metering. IET Wirel Sens Syst. 2023;13(2):75–90. https://doi.org/10.1049/wss2.12054.
33. Merza ME, Hussein SH, Ali QI. Identification scheme of false data injection attack based on deep learning algorithms for smart grids. Indones J Electr Eng Comput Sci. 2023;30(1):219–228. https://doi.org/10.11591/ijeecs.v30.i1.pp219-228,
34. Alhabib MH, Ali QI. Internet of autonomous vehicles communication infrastructure: a short review. 2023;24(3). https://doi.org/10.29354/diag/168310.
35. Ali QI. Realization of a robust fog-based green VANET infrastructure. IEEE Syst J. 2023;17(2):2465–2476. https://doi.org/10.1109/JSYST.2022.3215845.
36. Ansari J, Andersson C, de Bruin P, et al. Performance of 5G trials for industrial automation. Electronics. 2022;11:412. https://doi.org/10.3390/electronics11030412.
37. Siriwardhana Y, Porambage P, Ylianttila M, Liyanage M. Performance analysis of local 5G operator architectures for industrial Internet. IEEE Internet Things J. 2020;7(12):11559–11575. https://doi.org/10.1109/JIOT.2020.3024875.
38. Rekoputra N, Tseng C, Wang J, et al. Implementation and evaluation of 5G MEC-enabled smart factory. Electronics. 2023;12(6):1310. https://doi.org/10.3390/electronics12061310.

QUTAIBA I. Ali received the B.Sc. degree in electrical engineering in 1996, the M.Sc. degree (with honor) in electrical engineering in 1999, and the Ph.D. degree (with honor) in computer engineering in 2006 from the University of Mosul, Mosul, Iraq. He is currently a Professor with the Departments of Computer/Mechatronic Engineering, University of Mosul. His research interests include computer networks, network security, wireless sensor networks, industrial and real-time communication systems, IoT, vehicular networks, green networking, embedded systems, and AI-driven networking solutions. He has authored several books and more than 200 publications in international journals and conferences. Prof. Ali has received numerous national and international awards, including recognition from the Iraqi Ministry of Higher Education, the Albert Nelson Marquis Lifetime Achievement Award, and multiple Top Cited Paper Awards from IET journals. He has supervised 20 M.Sc. and Ph.D. students and serves as a reviewer and editorial board member for several IEEE and international journals.
e-mail: qut1974@gmail.com

Zeina A. MOHAMMED received the B.Sc. degree in computer engineering in 2009, the M.Sc. degree in computer engineering in 2012, and the Ph.D. degree in computer engineering in 2024, all from the Department of Computer Engineering, College of Engineering, University of Mosul, Mosul, Iraq. Since 2017, she has been a Lecturer with the Department of Computer and Information Engineering, College of Electronics Engineering, Ninevah University, Mosul, Iraq. Her research interests include computer networks, network security, wireless sensor networks, the Internet of Things (IoT), and vehicular networks.
e-mail: zinah.mohammed@uoninevah.edu.iq