Enhancing Real-Time IoT Applications: Latency Reduction Techniques in 5G MIMO Networks

Chrysostomos-Athanasios Katsigiannis[1], Konstantinos Tsachrelias[1], Vasileios Kokkinos[1] , Christos Bouras[1], Apostolos Gkamas[2] and Philippos Pouyioutas [3]

[1] Computer Engineering & Informatics Dept., University of Patras, Patras Greece
{up1096511, bouras}@upatras.gr, up1072490@upnet.gr, kokkinos@cti.gr,
[2] Department of Chemistry, University of Ioannina, Ioannina Greece
gkamas@uoi.gr
[3] Computer Science Department, University of Nicosia, Nicosia, Cyprus
pouyioutas.p@unic.ac.cy

Abstract. In the ever-growing field of 5G technology, Massive Multiple Input Multiple Output (MIMO) systems hold the promise of substantial improvements in network capacity and efficiency. However, the demand for low latency, imposed by real-time Internet of Things (IoT) applications presents some significant obstacles. This study delves into novel approaches to reducing latency within 5G MIMO setups, focusing particularly on the integration of shortened Transmission Time Intervals (TTIs) and preemptive scheduling tactics. Through simulations involving a small-scale MIMO configuration servicing multiple IoT devices, this study thoroughly evaluates the effects of these methods on transmission latency. The desired result is to demonstrate that the adoption of reduced TTIs notably mitigates overall latency, thereby augmenting the efficacy of time-sensitive IoT tasks such as industrial automation and healthcare monitoring systems. This research not only contributes to the theoretical progression of 5G communications but also furnishes practical insights for network operators and system designers trying to fine-tune 5G networks for evolving IoT environments.

Keywords: Latency Reduction, Internet of Things (IoT), 5G Networks, Multiple Input Multiple Output (MIMO)

1 Introduction

The evolution of the fifth generation (5G) of wireless communication networks promises transformative changes across various sectors, including industrial automation, healthcare, and urban mobility. Central to these advancements is the deployment of Massive Multiple Input Multiple Output (MIMO) technologies, which are pivotal in achieving the high data rates, increased capacity, and reduced latency goals of 5G. As the Internet of Things (IoT) continues to expand, with billions of devices interconnected and communicating incessantly, the latency performance of these networks becomes critical. Real-time applications such as autonomous vehicle navigation, remote surgery, and automated production lines demand not only continuous connectivity but also extremely low latency to function effectively and safely.

Despite the substantial capabilities of 5G networks, meeting the ultra-low latency requirements for these critical applications remains challenging. Current 5G implementations prioritize bandwidth and throughput enhancements, often sidelining latency reduction. However, the unique demands of IoT applications, where timely data delivery is paramount, require a shift in focus towards minimizing latency. This paper aims to bridge this research gap by exploring how to further reduce latency to satisfy the stringent requirements of real-time IoT environments. This study introduces innovative techniques aimed at reducing latency in 5G MIMO configurations, primarily through the optimization of Transmission Time Intervals (TTIs) and the adoption of preemptive scheduling strategies. Shortened TTIs can potentially decrease the time taken for a data packet to be transmitted across the network, thus reducing the overall communication delay. On the other hand, preemptive scheduling prioritizes data traffic that is more delay-sensitive, ensuring that critical information is processed and transmitted first, thereby mitigating the latency experienced by crucial IoT devices [1], [2], [3], [4].

By employing a simulation-based methodology, this research evaluates the impact of these proposed latency reduction techniques within a controlled 5G MIMO environment. The results are expected to contribute significantly to the existing knowledge by demonstrating practical ways to enhance the real-time capabilities of 5G networks tailored for IoT applications. This not only supports theoretical advancements in telecommunications but also offers practical insights and recommendations for network operators and system designers striving to optimize the burgeoning IoT ecosystem within the 5G framework. Also, several studies have explored the integration of 5G and IoT to address the demands of modern applications. For instance, paper [5] provides an extensive review of how 5G technology will transform IoT applications. The paper discusses the limitations of 4G LTE in handling the demands of high device connectivity, data rates, and low latency, and argues that 5G is essential for future IoT developments. Furthermore, it covers key technologies such as 5G NR, MIMO, mmWave communication, and HetNets, as well as addressing security challenges and the potential of AR in 5G IoT systems. Paper [6] provides an overview of the 5G and IoT technologies, discussing their evolution and the challenges they face, such as power saving, signal absorption, refraction, and diffraction. The paper reviews D2D communication and LPWAN technologies like ZigBee, SigFox, WiFi, LoRa, and NB-IoT, highlighting the new capabilities enabled by 5G in IoT systems. While previous studies have primarily focused on bandwidth and throughput enhancements in 5G networks, this research shifts the focus towards latency optimization, a critical factor for the efficacy of time-sensitive IoT tasks. The unique approach of integrating shortened TTIs and preemptive scheduling strategies has not been extensively explored in the context of 5G MIMO setups.

The rest of this paper is organized as follows: In Section 2, we introduce the mathematical model utilized in our simulation environment. Moving to Section 3, we delve into the algorithm analysis that forms the basis for constructing our experiment scenarios. Section 4 outlines the simulation environment and methodology employed to assess the performance of the Algorithm. Following that, in Section 5, the simulation results are presented, and a comprehensive analysis of the findings is conducted. Lastly, Section 6 concludes the paper and offers insights into potential avenues for future research.

2 Mathematical Model

In the process of optimizing 5G MIMO systems for real-time IoT applications, it is critical to address the network latency which directly impacts the efficacy and reliability of IoT devices. This mathematical model provides a structured framework for evaluating how modifications in network configurations and protocols affect the overall latency. The model incorporates several components essential for understanding and optimizing the data transmission processes in a 5G MIMO environment.

In the model of data transmission for a 5G MIMO system, each IoT device u receives data xu(t) from the base station at any given time t. The signal yu(t) received by the device is a sum of the contributions from multiple antennas at the base station, each affected by the channel characteristics hu,n between the nth antenna and the device, and the ambient noise ηu(t). This noise represents random fluctuations and interference from other sources that might affect the clarity and integrity of the received signal. The model simplistically assumes linear propagation without considering inter-user interference, which can often complicate transmission in dense network environments. This simplification allows for clearer analysis and understanding of fundamental signal behavior and transmission quality [7], [8], [9]. Assuming a simple linear model without considering inter-user interference, the received signal can be modeled as:

$$y_u(t) = \sum_{n=1}^{N} h_{u,n}\, x_u(t)\eta_u(t) \qquad (1)$$

As for the latency model, the total latency experienced by a packet, denoted as Lu, is the sum of processing, queueing, and transmission latencies. Each component contributes to the delay experienced by the end-user, and minimizing these delays is critical for enhancing user experience and system efficiency in IoT applications enabled by 5G technology. Efficient management of these latency components is essential for ensuring that the network can meet the stringent requirements of emerging IoT technologies and applications. Transmission latency encompasses the time required for a data packet to travel from the base station to an IoT device. This latency component is vital in real-time applications where delays can affect the functionality and responsiveness of the system. Reducing transmission latency is crucial for applications requiring immediate feedback, such as autonomous driving or real-time monitoring systems.

Processing latency refers to the time taken by network infrastructure, particularly the base station, to process data before it is transmitted. This involves encoding, modulation setting, and possibly encryption. Efficient processing is essential to maintain high throughput and system responsiveness, making it a critical factor in overall network performance. Optimizations in hardware and software at the base station can lead to significant reductions in processing latency.

Queueing latency is the delay that occurs when data packets wait in a queue until they can be processed or transmitted. This component of latency is highly variable and depends on the traffic load, network management policies, and prioritization schemes. In 5G networks, where data from numerous devices must be managed simultaneously, effective scheduling strategies can significantly mitigate queueing delays, especially for high-priority traffic in IoT applications. Total latency experienced by a packet sent to user u is:

$$Lu = Lproc + Lqueue + Ltrans \qquad (2)$$

3 Algorithm Analysis

In this section, a comprehensive algorithmic analysis of the code implemented for evaluating latency reduction techniques in 5G MIMO systems tailored for real-time IoT applications, is conducted. This analysis aims to clarify the efficiency, effectiveness, and potential optimizations of the algorithm employed in the simulation experiments. A simplistic pseudoalgorithm of the code used in the experiments follows in Algorithm 1:

Algorithm 1 Latency Reduction Pseudoalgorithm

Step 1: Initialization

Initialize the simulation parameters including the number of antennas at the base station, number of IoT devices, and the TTIs (both standard and reduced).

Step 2: Data Generation

Generate random data payloads for each IoT device within the range of 100 to 1000 units.

Randomly assign each device a type, either as a sensor or a smartphone.

Randomly generate interference levels for each device between 1 and 5.

Randomly assign each device one of three traffic models: CBR, VBR, or bursty.

Randomly assign each device to one of three network slices (high, medium, or low priority).

Step 3: Initialize Latency Measurements

Create arrays to store latency measurements for both standard and reduced TTIs for all users.

Create matrices to store latency measurements for each slice for both standard and reduced TTIs.

Initialize counters to keep track of the number of users in each slice.

Step 4: Simulate Transmission with Standard TTI

For each user, determine the slice to which they belong and update the respective user count.

Calculate the number of resource blocks based on the device type.

Simulate the transmission for each user using the standard TTI by:

Calculating the processing delay.

Calculating the transmission delay based on data size and resource blocks.

Adding the interference delay.

Adjusting the latency based on the slice priority.

Adding the traffic model-specific delay.

Store the calculated latency in the respective arrays and matrices for standard TTI.

Step 5: Simulate Transmission with Reduced TTI

For each user, determine the slice to which they belong.

Calculate the number of resource blocks based on the device type.

Simulate the transmission for each user using the reduced TTI by:

Calculating the processing delay.

Calculating the transmission delay based on data size and resource blocks.

Adding the interference delay.

Adjusting the latency based on the slice priority.

Adding the traffic model-specific delay.

Store the calculated latency in the respective arrays and matrices for reduced TTI.

Step 6: Calculate Average Latencies

Compute the average latency for all users for both standard and reduced TTIs.

Compute the average latency for each network slice (high, medium, low priority) for both standard and reduced TTIs.

Step 7: Display Results

Print the average latency for all users for both standard and reduced TTIs.

Print the average latency for each network slice for both standard and reduced TTIs.

Step 8: Visualization

Plot the latency distributions for standard and reduced TTIs.

Plot the latency distributions for each traffic model (CBR, VBR, bursty).

Plot the average latency for each network slice for both standard and reduced TTIs.

Create scatter plots of latency versus data payload size for both standard and reduced TTIs.

Create scatter plots of latency versus interference levels for both standard and reduced TTIs.

Plot the Cumulative Distribution Function (CDF) of latencies for both standard and reduced TTIs.

Create bar charts to show the latency per user for both standard and reduced TTIs.

Generate box plots to display the distribution of latencies for both standard and reduced TTIs.

The pseudoalgorithm is designed to simulate and evaluate latency reduction techniques in a 5G MIMO network environment. The process is methodically structured to ensure a comprehensive analysis of various factors affecting latency in real-time IoT applications. The simulation begins by initializing several key parameters. The number of antennas at the base station is set to 8, and 10 IoT devices are included in the network. Two TTIs are defined: a standard TTI of 1.0 ms and a reduced TTI of 0.2 ms, aimed at targeting low-latency applications.

Random data payloads are generated for each IoT device, with values ranging from 100 to 1000 units. Devices are randomly categorized as either sensors or smartphones. Additionally, each device is assigned a random interference level between 1 and 5, and one of three traffic models: Constant Bit Rate (CBR), Variable Bit Rate (VBR), or bursty traffic. To simulate network slicing, devices are randomly distributed across three slices with different priority levels: high, medium, and low. The latency measurements are initialized in arrays to store the latency for both standard and reduced TTIs, as well as slice-specific latency measurements. Counters are also set up to track the number of users in each slice. The next phase involves simulating the transmission using the standard TTI. For each device, the network slice is determined, and the corresponding slice user counter is incremented. The number of resource blocks is then calculated based on the device type. The transmission latency is simulated by considering various factors such as data size, TTI, interference, resource blocks, slice priority, and traffic model. The resulting latency is stored in both the overall and slice-specific arrays. A similar procedure is followed for the reduced TTI, where the transmission latency for each device is simulated and stored in the respective arrays. After simulating the latencies, the average latency for both standard and reduced TTIs is calculated across all devices. Additionally, the average latency for each network slice is determined for both TTI configurations. These calculations provide insights into how reduced TTIs, network slicing, and traffic models impact overall latency.

The results are then displayed, showing the average latencies for both TTI configurations and for each network slice. This helps in understanding the effectiveness of reduced TTIs and network slicing in reducing latency.

Finally, the results are visualized through various plots. Latency distributions for standard and reduced TTIs are plotted to show overall performance. The latency distribution for different traffic models is illustrated through histograms, highlighting how different traffic patterns affect latency. A bar chart compares the average latency for each network slice, while scatter plots show the relationship between latency and data payload size, as well as latency and interference levels. A Cumulative Distribution Function (CDF) plot provides a probability distribution of latencies, and a bar chart shows the latency experienced by each user. Additionally, a box plot displays the distribution of latencies, including median, quartiles, and outliers.

The MATLAB functions used in the simulation include simulate_transmission, which calculates the transmission latency based on various factors, get_resource_blocks, which determines the number of resource blocks for each device type, get_slice_priority, which adjusts latency based on network slice priority, and get_traffic_model_delay, which adds delays specific to each traffic model. These functions collectively ensure a detailed and accurate simulation, providing valuable insights for optimizing real-time IoT applications in 5G MIMO networks.

4 Simulation Environment

The simulation concern a 5G MIMO network environment to evaluate the impact of various latency reduction techniques on real-time IoT applications. The simulation setup involves a base station equipped with 8 antennas serving 10 IoT devices. Each device is randomly assigned data payloads ranging from 100 to 1000 units, and their device types are categorized as either sensors or smartphones.

To account for the diverse traffic demands in IoT environments, each device is randomly assigned one of three traffic models: CBR, VBR, or bursty traffic. Additionally, the network is divided into three virtual slices, each representing a different priority level, to simulate network slicing. Users are randomly distributed across these slices to evaluate the effect of prioritization on latency.

The primary objective of this simulation is to compare the performance of standard TTIs and reduced TTIs. Standard TTIs are set at 1.0 ms, while reduced TTIs are set at 0.2 ms to target low-latency applications. The latency measurements consider processing delays, transmission delays based on data size and resource blocks, interference delays, slice priorities, and traffic model-specific delays.

In this setup, the following parameters are defined: the number of antennas at the base station is set to 8, and the number of IoT devices is 10. Each device is assigned data payloads randomly selected between 100 and 1000 units. Device types are assigned randomly as either sensors or smartphones, and interference levels vary between 1 and 5. Traffic models are randomly assigned among CBR, VBR, and bursty. The network is divided into three slices with different priority levels to examine the impact of network slicing on latency. The summary of the parameters can be found in Table 1.

Table 1. Simulation Parameters

Parameter	Value
Number of Antennas (Base Station)	8
Number of IoT Devices	10
Standard TTI	1.0 ms
Reduced TTI	0.2 ms
Data Payloads	Random values between 100 and 1000 units
Device Types	Randomly assigned as either sensors or smartphones
Interference Levels	Random values between 1 and 5
Traffic Models	Randomly assigned as CBR, VBR, or bursty
Network Slices	3 slices (high, medium, and low priority)

The simulation process begins with the generation of random data payloads, device types, and interference levels for each user. Each device is also assigned a specific traffic model and a network slice. The latency for each device is then simulated using both standard and reduced TTIs. This involves calculating the total latency for each device, which includes processing delays, transmission delays based on data size and resource blocks, interference delays, slice priority adjustments, and traffic model-specific delays. The average latency for both standard and reduced TTIs is calculated across all devices. Additionally, the average latency for each network slice (high, medium, low priority) is determined for both TTI configurations. These calculations provide insights into how reduced TTIs, network slicing, and traffic models impact overall latency.

To visualize the results, several plots are generated. These include latency distributions for standard and reduced TTIs, latency distributions for different traffic models (CBR, VBR, bursty), and average latency for each network slice. Scatter plots of latency versus data payload size and latency versus interference levels are also created. Furthermore, CDF plots of latencies, bar charts of latency per user, and box plots of latency distributions for standard and reduced TTIs are presented.

The simulation utilizes MATLAB for numerical computation and visualization. Key MATLAB functions used in the simulation include simulate_transmission, which calculates transmission latency based on various factors, get_resource_blocks, which determines the number of resource blocks for each device type, get_slice_priority, which adjusts latency based on network slice priority, and get_traffic_model_delay, which adds delays specific to each traffic model. This comprehensive simulation environment setup and methodology ensures a detailed evaluation of latency reduction techniques in 5G MIMO networks, providing valuable insights for optimizing real-time IoT applications.

5 Simulation Results

The simulation results provide a detailed analysis of latency reduction techniques in a 5G MIMO network environment [10], focusing on the effects of standard and reduced TTIs, network slicing, and different traffic models on overall network performance.

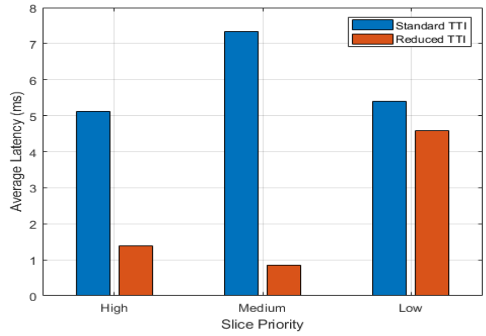

Fig. 1. Average Latency per Slice for Standard and Reduced TTI.

The average latency per network slice for both standard and reduced TTIs is illustrated in Fig. 1. The results indicate that reduced TTIs significantly lower latency across all slice priorities. High-priority slices, benefiting from the highest resource allocation, demonstrate the lowest latency values. In contrast, medium-priority slices experience higher latency, particularly with standard TTIs, due to their balanced resource allocation. Low-priority slices, having the least resource allocation, exhibit the highest latency but still show noticeable improvement with reduced TTIs.

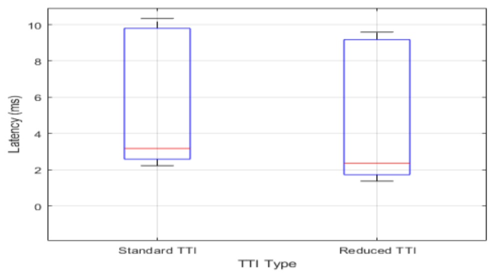

Fig. 2. Box Plot of Latency for Standard and Reduced TTI

The box plot latency in Fig. 2 for standard and reduced TTIs presents a comprehensive view of the latency distribution, including median values, quartiles, and outliers. The data reveals that reduced TTIs consistently achieve lower median latency and narrower interquartile ranges compared to standard TTIs. This indicates not only a reduction in average latency but also a decrease in latency variability, which is crucial for maintaining consistent network performance in real-time IoT applications.

The CDF plot in Fig. 3 provides a probabilistic perspective on latency distribution. The CDF curves for both standard and reduced TTIs show that a higher proportion of data transmissions complete within shorter latency periods when using reduced TTIs. This further corroborates the efficacy of reduced TTIs in enhancing network responsiveness and reliability, as lower latency values are achieved more frequently.

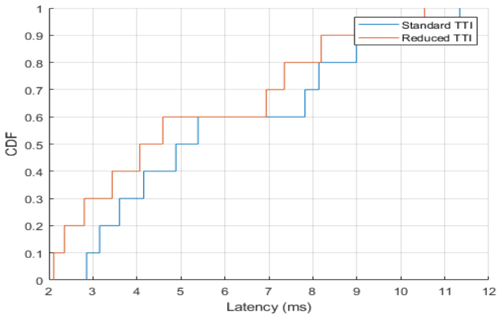

Fig. 3. Cumulative Distribution Function of Latency

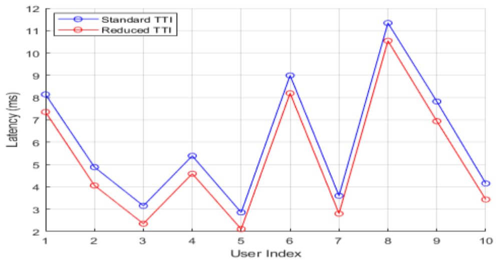

Fig. 4. Latency Comparison between Standard and Reduced TTI.

A direct comparison of latency experienced by each user under standard and reduced TTIs is depicted in Fig. 4. The results consistently demonstrate that reduced TTIs offer lower latency across all users. Variations in latency among users are observed, which can be attributed to differences in data payload sizes, interference levels, and assigned traffic models. Nonetheless, the overall trend highlights the superior performance of reduced TTIs in minimizing transmission delays.

Fig. 5 shows the latency distribution for three traffic models: CBR, VBR, and bursty traffic. Bursty traffic, which sends data irregularly, has the highest latency and the most variation. This is due to its unpredictable data bursts causing delays. In contrast, CBR traffic, which sends data at a steady rate, has the lowest latency because it allows for more efficient use of network resources. VBR traffic, with its changing data rates, has latency levels between the other two models. The overlapping areas in the histogram suggest that while different traffic models can have similar latency values, each model generally follows its own latency pattern. These results underline the importance of considering traffic types when optimizing network performance, as each type affects latency differently and impacts the overall network efficiency and reliability

The latency experienced in Fig. 6 by individual users for both standard and reduced TTIs is presented in a bar chart. The chart clearly illustrates that reduced TTIs consistently reduce latency for all users, highlighting the uniform benefit of this technique across diverse network conditions. This visualization also helps identify any outliers or users who may experience unusually high latency, which could be investigated further to optimize network performance.

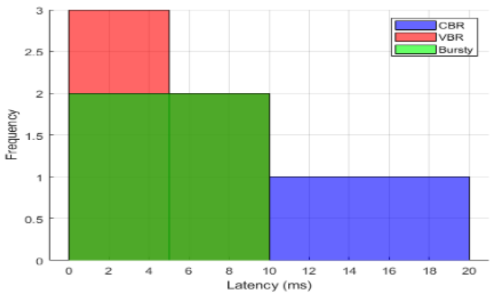

Fig. 5. Latency Distribution per Traffic Model

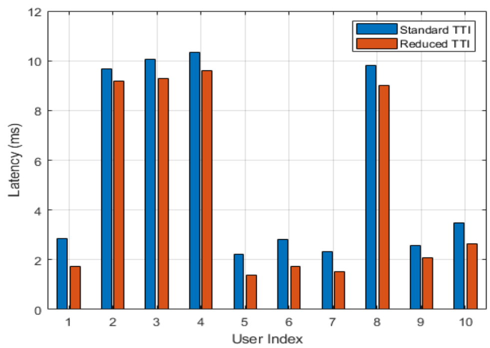

Fig. 6. Latency per User for Standard and Reduced TTI

The scatter plot in Fig. 7 showing the relationship between latency and data payload size reveals that larger data payloads generally incur higher latency. This trend is observed for both standard and reduced TTIs, though reduced TTIs still achieve lower latency overall. This relationship emphasizes the need for efficient data management and transmission strategies, especially for applications handling large volumes of data.

The final scatter plot in Fig. 8 examines the impact of interference levels on latency. Higher interference levels correlate with increased latency, as expected. However, reduced TTIs consistently mitigate the impact of interference, resulting in lower latency across all interference levels. This finding highlights the resilience of reduced TTIs in maintaining network performance even under challenging conditions.

The simulation results clearly demonstrate the advantages of reduced TTIs, network slicing, and traffic model considerations in optimizing latency for 5G MIMO networks. Reduced TTIs not only lower average latency but also enhance the consistency and reliability of network performance. Network slicing effectively prioritizes critical applications, while understanding traffic models helps tailor network strategies to specific use cases. These insights are pivotal for the development of efficient, real-time IoT applications in next-generation networks.

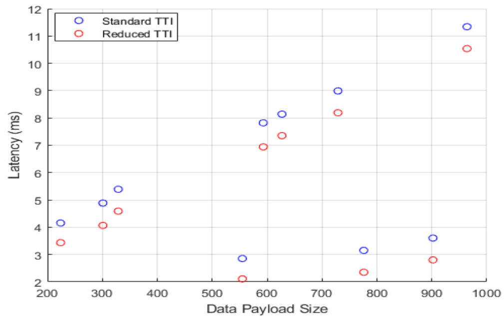

Fig. 7. Latency vs. Data Payload Size

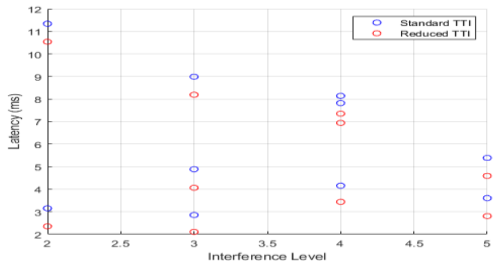

Fig. 8. Latency vs. Data Payload Size

6 Conclusion and Future Work

This research has successfully demonstrated the efficacy of various latency reduction techniques in a 5G MIMO network environment, with a specific focus on real-time IoT applications. The simulation results indicate that reduced Transmission Time Intervals (s) significantly decrease latency across all network slices, making them highly effective for low-latency applications. High-priority slices, which receive more resources, benefit the most, showing the lowest latency values. This prioritization ensures that critical IoT applications, which require real-time responsiveness, maintain optimal performance. The box plot and Cumulative Distribution Function (CDF) analyses further highlight the advantages of reduced TTIs, not only in lowering median latency but also in reducing latency variability.

Moreover, the comparison of latency across different traffic models reveals that the type of traffic significantly affects latency. Constant Bit Rate (CBR) traffic achieves the lowest latency, while bursty traffic exhibits the highest. This insight underscores the importance of considering traffic patterns when designing and optimizing network strategies. Overall, the findings suggest that implementing reduced TTIs, network slicing, and tailored traffic models can vastly improve the performance of 5G MIMO networks, particularly for real-time IoT applications.

While this study provides valuable insights into latency reduction techniques, several avenues for future research remain open. Future research should incorporate more complex interference models to account for inter-user interference in dense network environments. Additionally, investigating how user mobility, including handovers between cells, affects latency will provide a more comprehensive understanding of real-world network performance. Analyzing the trade-offs between latency performance and energy efficiency is crucial, especially for battery-powered IoT devices. Finally, integrating and evaluating sophisticated scheduling algorithms, such as machine learning-based predictive scheduling, and testing these techniques in real-world network environments will provide practical insights and validate the simulation results under actual operating conditions.

Acknowledgements

This research has been co-financed by the Hellenic Foundation for Research & Innovation (H.F.R.I) through the H.F.R.I.'s Research Projects to Support Faculty Members & Researchers (project code: 02440).

References

1. Srinivas K, Aswini .J, Patro P. and Kumar D, "Functional overview of integration of AIML with 5G and beyond the network", 2023 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 2023, pp. 1-5
2. Ghosh A, Maeder A, Baker M. and Chandramouli D, "5G Evolution: A View on 5G Cellular Technology Beyond 3GPP Release 15", in IEEE Access, vol. 7, pp. 127639-127651, 2019
3. Prakash P, Manoj G. and Samsonimmanuel J, "MIMO Antenna System for IoT Applications (5G)", 2022 6th International Conference on Devices, Circuits and Systems (ICDCS), Coimbatore, India, 2022, pp. 178-182, doi: 10.1109/ICDCS54290.2022.9780845.
4. Singh G, Singh J, Mitra D. and Prabha C, "A Roadmap Toward Prospects for IoT Enabled 5G Networks", 2023 7th International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 2023, pp. 1405-1410.
5. Wang L. et al., "Study on Flexible TTI Scheduling for LAA Systems", 2018 15th International Symposium on Wireless Communication Systems (ISWCS), Lisbon, Portugal, 2018, pp. 1-5
6. Zhang Z, Gao Y, Liu Y and Li Z, "Performance evaluation of shortened transmission time interval in LTE networks", 2018 IEEE Wireless Communications and Networking Conference (WCNC), Barcelona, Spain, 2018, pp. 1-5
7. Gkagkas G, Vergados D. and Michalas A, "The advantage of the 5G network for enhancing the Internet of Things", 2021 6th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM), Preveza, Greece, 2021, pp. 1-7.
8. Chettri L. and Bera R, "A Comprehensive Survey on Internet of Things (IoT) Toward 5G Wireless Systems", in IEEE Internet of Things Journal, vol. 7, no. 1, pp. 16-32, Jan. 2020.
9. Yeo J, Kim T, Oh J, Park S, Kim Y, and Lee J, "Advanced Data Transmission Framework for 5G Wireless Communications in the 3GPP New Radio Standard", IEEE Communications Standards Magazine, vol. 3, pp. 38-43, Sep. 2019,
10. Bojovic B. and Lagen S, "MIMO in network simulators: Design, implementation and evaluation of single-user MIMO in ns-3 5G-LENA", preprint arXiv:2404.17472, 2024.