Device Location-Based Link Layer Scheduling for Large-Scale Dense Industrial IOT Networks

Ahmed Ajeena
Marquette University

DEVICE LOCATION-BASED LINK LAYER SCHEDULING FOR LARGE-SCALE DENSE INDUSTRIAL IOT NETWORKS

by

Ahmed Ajeena

A Thesis Submitted to the Faculty of the Graduate School,
Marquette University,
in Partial Fulfillment of the Requirements for
the Degree of Master of Science in Electrical and Computer Engineering.

Milwaukee, Wisconsin

August, 2022

ABSTRACT

DEVICE LOCATION-BASED LINK LAYER SCHEDULING FOR LARGE-SCALE DENSE INDUSTRIAL IOT NETWORKS

Ahmed Ajeena

Marquette University, 2022

A medium access control (MAC) protocol design is proposed in this paper for the industrial internet of things (IIoT) networks with massive connections. Considering a non-fully connected network with multiple access points (APs), we aim to connect a large number of IIoT devices densely populating the network coverage area, achieve low latency, and avoid collisions. To achieve this objective, we adopt a time-slotted carrier sensing-based medium access control design and propose a device location-based time slot assignment scheme for channel access scheduling. The proposed scheme assigns devices to time slots based on their locations to eliminate collisions caused by hidden and exposed terminals. To demonstrate the performance of the proposed design, we analyze the average delay each device experiences with the proposed scheduling scheme and verify the analysis via numerical simulations of an IIoT network with 19 APs and over 17000 devices. The results show the effectiveness of the proposed design in supporting massive connections and achieving low delay.

ACKNOWLEDGMENTS

Ahmed Ajeena

I want to give my immense gratitude and thanks to God Almighty for all His blessings. This journey would not have happened without continuous help from the people around me. I am thankful to my advisor Dr. Jie Gao for his unwavering supervision, valuable and ongoing support, encouragement, and patience. His guidance has been invaluable throughout my master's program and in all aspects of my research. I would like to give my genuine gratitude and thanks to my family for their continuous support. I would not have achieved my goals without their kindness and dedicated support. I would also like to thank my friends who supported me, both friends from back home and the new friends I made here in the US.

I finally would like to dedicate this work to my late father. May God bless his soul and let heaven be his final resting place.

TABLE OF CONTENTS

List of Figures

List of Tables

List of Abbreviations and Acronyms

Abbreviation	Meaning
IoT	Internet of Things
IIoT	Industrial Internet of Things
MTC	Machine-type Communications
MAC	Medium Access Control
LLC	Logical Link Control
OSI	Open Systems Interconnection
mMTC	Massive Machine-type Communications
URLLC	Ultra-reliable Low Latency Communications
M2M	Machine-to-Machine
HTC	Human Type Communication
H2H	Human-to-Human
AP	Access Point
BS	Base Station
OZ	Overlapping Zone
NOZ	Non-overlapping Zone
DTMC	Discrete-time Markov Chain
QoS	Quality of Service
CSMA	Carrier Sensing Multiple Access
CSMA/CA	Carrier-sense multiple access with collision avoidance
ALOHA	Additive Links On-line Hawaii Area
TDMA	Time-division Multiple Access
FDMA	frequency-division Multiple Access
CDMA	Code-division Multiple Access
OFDM	Orthogonal Frequency-division Multiplexing

SPS	Semi-Persistent Scheduling
TTI	Transmission Time Interval
DSME	Deterministic and Synchronous Multi-channel Extension
RB	Resource Block
RTS	Request to Send
MsCS	Mini-slot Based Carrier Sensing
SyncCS	Synchronization Carrier Sensing
SMsA	Superimposed Mini-slot Assignment
GPS	Global Positioning System
QZSS	Quasi-Zenith Satellite System
SC-FDMA	Single Carrier Frequency Division Multiple Access
LPG	Liquefied Petroleum Gas

Chapter 1: INTRODUCTION

1.1 Motivation

The Industrial Internet of Things (IIoT) is the key drive for the upcoming industrial transformation that revolutionizes manufacturing, logistics, mining, etc., around the world. Global investment in IIoT is expected to surpass 260 billion USD by 2026, and the IIoT market is projected to occupy 33% of the Internet of Things (IoT) market around the same time [1], [2].

Envisioned to reshape industries, IIoT focuses on boosting the productivity and efficiency of industrial processes by integrating sensing, communication, computing, and automatic control. IIoT allows for intelligent, remote, and real-time process management of industrial assets, which improves efficiency, safety, and the working environment. The foundation of IIoT is based on the connectivity of a large number of devices such as sensors, actuators, controllers, robots, and automated guided vehicles. Both general IoT and IIoT face numerous challenges, such as the connected devices' energy consumption, diverse latency requirements for end devices, network scalability, and security. In contrast to communications in general IoT scenarios, IIoT communications are subject to more stringent reliability and latency requirements. Specifically, an IIoT network must support massive connections, high reliability, and low delay at the same time, which crosses over between massive machine-type communications (mMTC) and ultra-reliable low latency communications (URLLC). Therefore, dedicated connectivity solutions are necessary to support IIoT due to such a unique characteristic.

1.2 Background

1.2.1 Industrial Internet of Things (IIoT)

The Internet of Things (IoT) is a network of connected objects that can interact without the need for human involvement. By processing real-time generated data and enhancing access to information, IoT will enable fast, and precise decision-making [3]. The IoT is expected to impact all aspects of life in the near future, becoming a major enabler for smart homes and smart cities, with applications in industry, healthcare, utilities, etc. IoT in the different applications features a self-regulated infrastructure that supports communication among the end devices and the surrounding world [4].

Among all IoT-based services, healthcare and manufacturing applications are expected to have the most significant economic growth. Specifically, the healthcare and manufacturing sectors will constitute around 74% of the IoT-based applications in market value by 2025 [3].

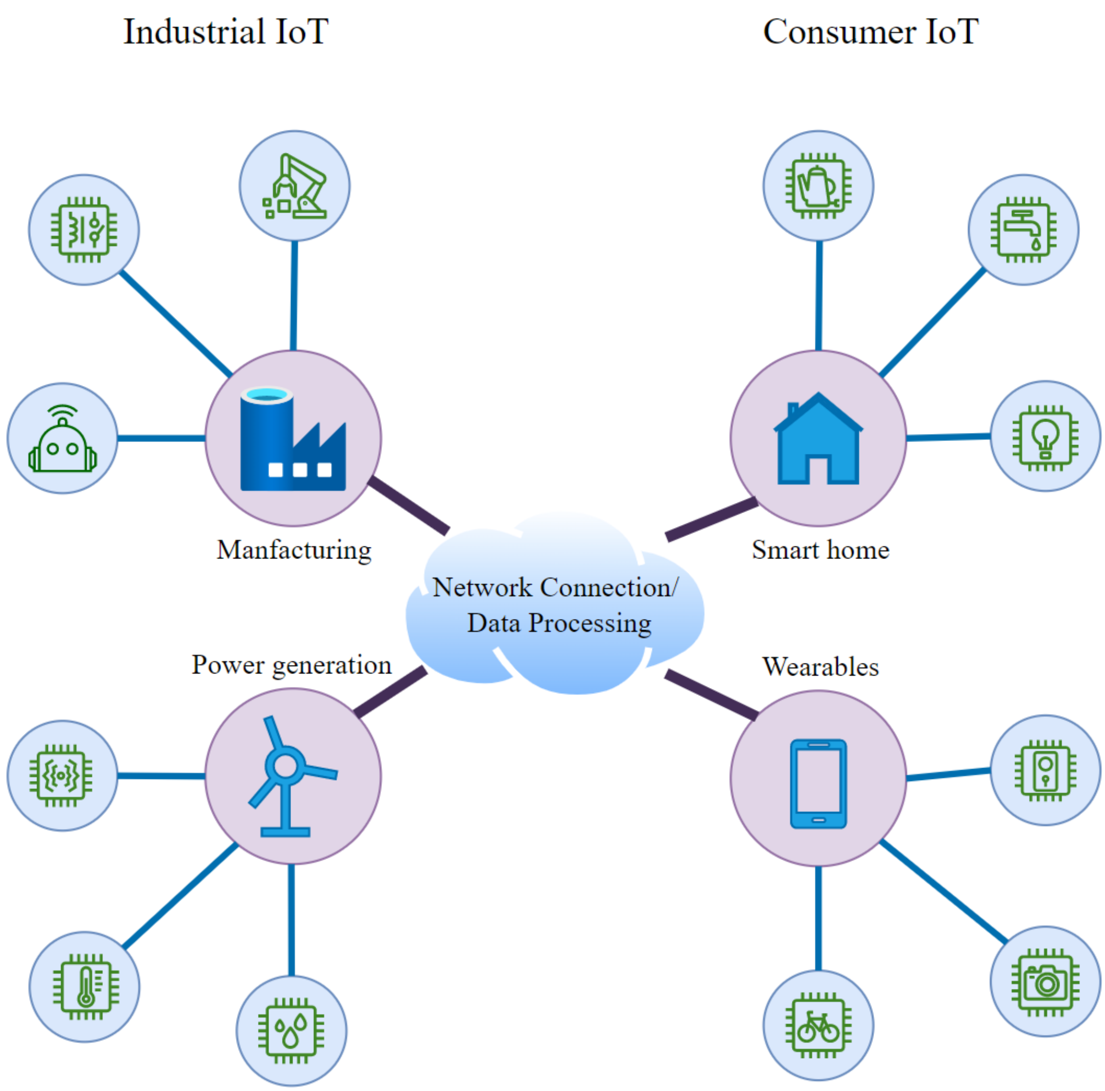

Figure 1.1: Example applications of IIoT and consumer IoT.

IoT can be divided into two subcategories; consumer IoT and IIoT, as shown in Fig. 1.1. Consumer IoT seeks to connect end-user devices such as wearables and appliances to the Internet for a more convenient and comfortable lifestyle. Consumer IoT will help realize concepts such as smart homes and smart cities.

IIoT connects machines and equipment in industrial businesses to boost production and increase profit. IIoT will lead to the creation of smart factories and intelligent manufacturing [5]. IIoT is applicable to production plants, warehouses, oil refineries, power plants, etc. In this thesis, the focus is on IIoT.

IIoT requires the integration of sensing, computation, and communication technologies [6]. The development of these technologies is paramount to the full realization of IIoT deployment. Although some requirements for IoT and IIoT are similar (i.e., both aim to create scalable, energy-efficient, and cost-effective networks), IoT and IIoT paradigms can have very different communication requirements in terms of latency, reliability, and security. A consumer IoT device such as a thermostat meter in a smart office may generate one message per hour [7], while an IIoT device may generate more than one packet per second. Therefore, new communication protocols are being designed for the specific connection and incorporation of industrial equipment and machines to improve reliability [8].

Some of the major communication challenges IIoT faces are end-to-end delay, transmission reliability, and network scalability. A typical IIoT network has to support devices with different latency requirements. Depending on the function, a device could tolerate relatively high delays, such as a humidity sensor installed in a greenhouse. On the other hand, a safety valve in a hazardous area of a chemical powerplant must have an extremely low end-to-end delay. In addition to latency, the communication between the devices and an access point (AP) needs to be reliable. When a device conducts a transmission, the transmitted payload is not lost or corrupted because of environmental conditions (i.e., distances, building materials, sudden changes in pressure or temperature, etc.) or interference with other devices. Typically, delay requirements for IIoT applications range between 10 and 100 milliseconds, with some applications requiring submillisecond delay, while throughput requirements must exceed 99% [9]. Scalability poses another problem for IIoT networks since an IIoT network can incorporate thousands of devices, making it more challenging to maintain low latency and reliability [2], [10]. Two 3GPP defined use cases are relevant to IIoT, machine type communication (MTC) and ultra-reliable low latency communication (URLLC). The goal of MTC technology is to support a large number of devices without impairing network performance. URLLC, on the other hand, supports extremely low end-to-end delays for the generated packets without compromising the throughput [11].

1.2.2 Machine-Type Communication (MTC)

MTC, also referred to as Machine-to-Machine (M2M) communications, is the direct connectivity among devices over any communication medium, including both wired and wireless links. MTC allows machines or devices to communicate without the need for human interaction. MTC connections can involve a vast number of devices in a variety of application domains, hence acting as a pillar for the establishment of IoT. Industrial automation can use MTC to allow a sensor or a meter to convey the data it collects (i.e., pressure readings, flow speed readings, etc.) to a controller that can utilize it (e.g., adjusting an industrial process based on pressure) [12]. Human type communication (HTC), also referred to as human-to-human (H2H) communications, such as computer and handheld devices communications, fundamentally differ from MTC applications because the amount of uplink traffic is often lower than the amount of downlink traffic. Therefore, networks in HTC communications are primarily designed to handle downlinks. On the other hand, MTC applications may generate more data traffic in uplink transmissions than data traffic in downlink transmissions [13], [12]. The prominent uplink transmissions coupled with a large number of devices and massive amounts of generated data cause a significant scheduling overhead, which is a major issue in MTC when deployed in IIoT networks [14].

MTC devices can be categorized based on their attributes and applications. Attributes include devices' hardware complexity, energy efficiency, and mobility, while applications include environmental monitoring, industrial automation, and medical instrumentation. The large number of devices in MTC generate massive amounts of data. The data generation could be periodic or event-based, depending on the services and applications for which the MTC devices are used. The immense amount of data and the various data generation traffic patterns pose a substantial difficulty with respect to collecting and processing the generated data in a timely and efficient manner [15].

The generated packets from MTC devices have specific characteristics compared to other types of communications, such as HTC. The MTC features small packets as small as a few bytes in size, as is common with smart meters. The transmissions are mostly sporadic (i.e., a sensor may send readings at irregular time intervals between one reading and the next). In addition to MTC having fundamental differences compared to the other types of communications, MTC communication characteristics vary depending on the various applications [13]. In this work, we differentiate between MTC, which supports consumer IoT, and MTC, which supports IIoT, given that MTC protocol designs have different considerations depending on the application's requirements.

Table 1.1: MTC design requirements in IIoT as compared with IoT.

Network requirements	IIoT	IoT
Number of devices	Large to very large	Small to medium
Mobility	Low to very low	High
Delay tolerance	Low to very low	Medium to high
Data generation rate	Large to very large volume	low to medium volume

Table 1.1 shows some of the primary differences regarding network requirements between IoT and IIoT. The number of devices deployed per square kilometer in IIoT networks is significantly higher than that for consumer IoT. For instance, the concentration of sensory equipment and controllers in a factory is much higher than meters in a house or speed cameras on a highway. MTC must be designed to support medium to high mobility in IoT. In contrast, when designed for IIoT, mobility is not a major concern since most devices in IIoT are either stationary or have low mobility. IIoT devices have very low to low latency tolerance, while IoT devices can tolerate medium to high latency [13]. Data arrival rates of IIoT devices (e.g., a flow sensor in an LPG pipe) are considerably higher than those of IoT devices (e.g., a smoke detector in an office) [16]. The magnitude of generated data is an important aspect to consider in MTC. The amount of data generated in IIoT is substantially higher than in regular IoT, and this aspect correlates with the data generation rates and the number of devices in IIoT, as mentioned above.

1.2.3 Medium Access Control

The medium access control (MAC) and the logical link control (LLC) are two sublayers of the second layer (i.e., the data link layer) in the Open Systems Interconnection (OSI) model. The MAC's duty is to provide flow control and multiplexing for the transmission medium, while the LLC is responsible for providing flow control and multiplexing for the logical link [17]. Fig. 1.2 shows where MAC is within the 7-layers of the OSI conceptual model.

The wireless medium is accessible to all devices, therefore, communication will be ineffective without well-designed rules regulating devices' access to the medium. A MAC protocol controls access to the shared media by establishing guidelines that permit devices to transmit their packets over the network in an organized and effective way. Thus, by guaranteeing effective and equitable usage of the limited wireless bandwidth, MAC protocols take on a vital role in utilizing the channel resources [18].

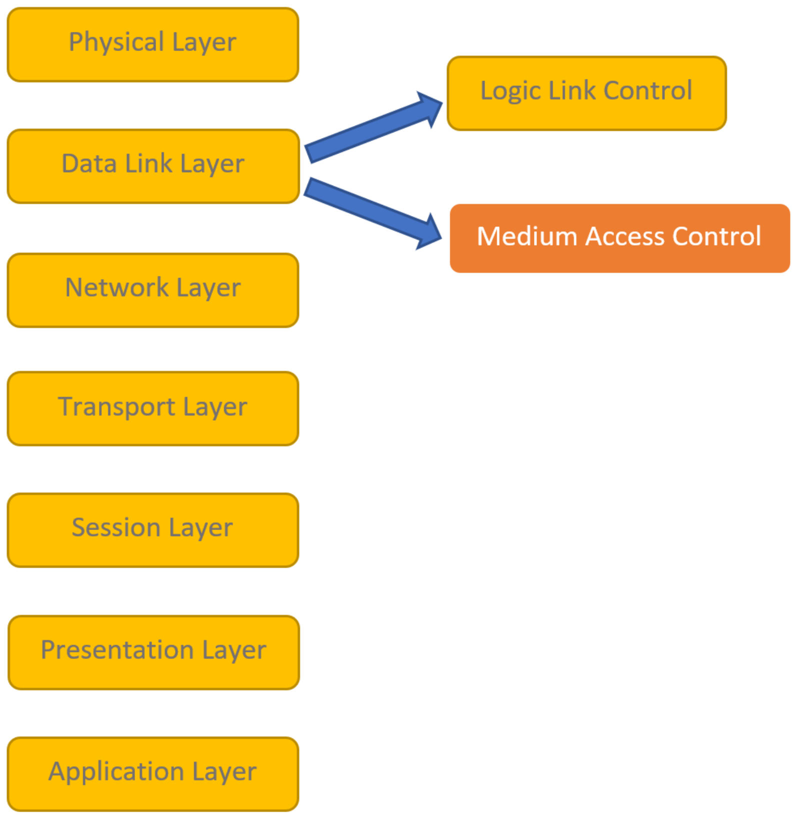

Figure 1.2: MAC in OSI Model.

MAC protocol design is more problematic in wireless communication than in wired communication due to the nature of the wireless medium [19], [20]. For instance, in wireless communication, a device usually cannot send and receive simultaneously due to self-interference between the received and the transmitted signals, as the latter has significantly higher levels of power, making it impossible for the device to receive other signals while conducting a transmission. Another major issue in wireless communication is related to carrier signal sensing, which is a location-dependent mechanism. The position of the receiver with respect to the position of the transmitter affects carrier sensing. Because of the propagation loss, the signal strength diminishes with increasing distance. Only devices within the transmitter's range are able to detect its carrier signal. A device uses carrier sensing to determine whether other devices in the network have an ongoing transmission. In the carrier sensing mechanism, the device detects the carrier signal frequency, which has a higher frequency than the input signal to determine whether another device in the shared medium is currently transmitting. If the device does not detect the carrier signal, it initiates a transmission. Otherwise, it defers its transmission to another time. Two problems arise in carrier sensing:

- **Hidden terminal problem:** A hidden terminal (i.e., wireless device) is a terminal within the intended recipient's range but out of the sender's range. The range refers to the node's transmission range and sensing range which are presumed to be equal. Consider the case shown in Fig. 1.3. Node A is transmitting to node B, while node C simultaneously attempts a transmission to node B. Node C cannot sense the transmission from A when it senses the channel for the carrier signal since node A is not within node C's range. Hence, node C falsely assumes that the channel is idle. If node C starts a transmission while A is transmitting, it will cause a collision with the packet transmitted from node A at the receiver node B. In this case, devices A and C are mutually hidden to each other with respect to the receiver B.

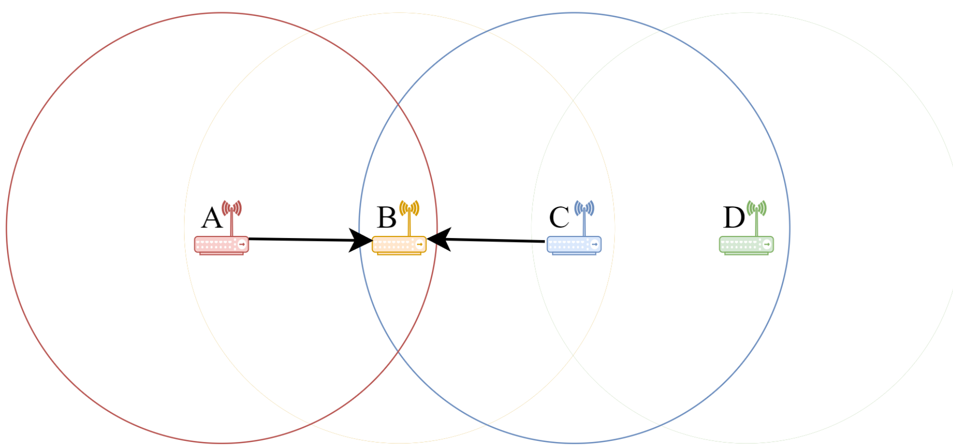

Figure 1.3: Hidden terminal problem. The circles represent the sensing ranges of the wireless nodes.

- **Exposed terminal problem:** An exposed terminal is a terminal within the sender's range but out of the receiver's range. Consider the case shown in Fig. 1.4. Node B is transmitting to node A, while node C intends to attempt a transmission to node D. Node C can sense the transmission being carried out by node B when it senses the channel for the carrier signal. Hence, node C learns that the channel is busy and does not initiate a transmission to node D. Nevertheless, node C transmissions cannot reach node A and cause collisions. In principle, node C can transmit to node D without issues. In this case, node B and node C are mutually exposed to each other given their respective receivers.

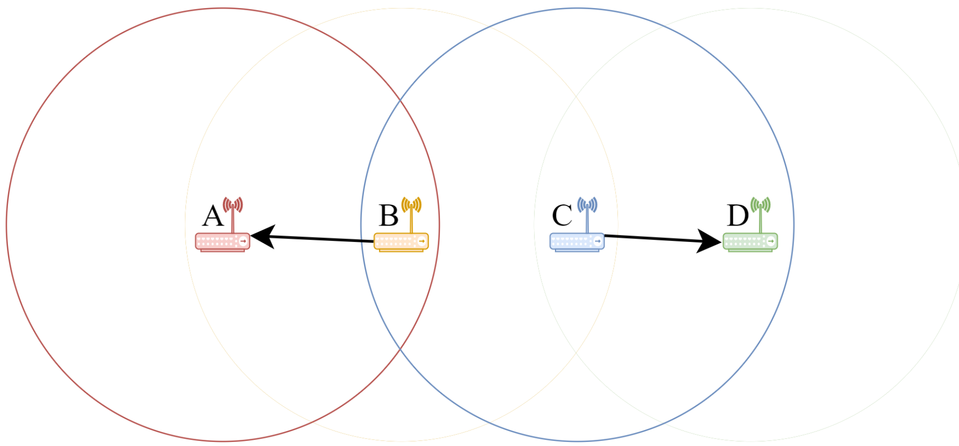

Figure 1.4: Exposed terminal problem.

In addition to hidden and exposed terminal problems, MAC protocol design in wireless communication aims to resolve other challenges, such as unfair resource allocation, channel access delay, and goodput. Unfair resource allocation means the communication medium is not equally shared among all the devices during a designated time. Unfair resource allocation can degrade throughput and cause increased packet delay for a portion of the devices in the network. Channel access delay is defined as the average time from the instant a device attempts to transmit a packet until the start of the successful packet transmission. Goodput is the amount of data successfully received at the destination, not including any lost, corrupted, or overhead packets. Goodput is the throughput perceived by the receiver [21]. A well-designed MAC protocol can achieve the desired system performance, improve reliability, and guarantee the quality of service (QoS).

MAC protocols can be divided into three main categories, contention-based channel access, contention-free channel access, and hybrid channel access [22], [21]. Multiple devices can attempt transmissions simultaneously without centralized control in contention-based channel access. The multiple access to the channel can cause severe drawbacks when the traffic load in the network is high. Under low or medium loads, contention-based access can perform well. Most notably, packet collision (i.e., two transmitters send to the same receiver) and starvation (i.e., a device might not get to transmit because there are always other devices transmitting). Therefore, MAC protocols under this category implement mechanisms to prevent or reduce the aforementioned issues. MAC protocols under this category include but are not limited to Carrier Sensing Multiple Access (CSMA) protocols and Additive Links On-line Hawaii Area (ALOHA) protocols.

In contention-free channel access, a central controller establishes rules for the nodes in the network to access the channel. Therefore, a device can only access the network in accordance with its allocated resources (i.e., only one device transmits over the same channel at any time). The need for a central controller means all the network devices must communicate with an AP. Drawbacks in this scheme are the need for network-wide synchronization and the unused pre-allocated resources. MAC protocols that are based on time-division multiple access (TDMA), frequency-division multiple access (FDMA), and code-division multiple access (CDMA) are contention-free protocols.

A hybrid channel access protocol is a combination of contention-based and contention-free channel access and typically includes the advantages of both schemes. Hybrid protocols tend to have complicated designs. An example of hybrid protocols is the IEEE 1901 [23].

We explain TDMA and CSMA in further detail due to their relevance to our work [24]. In TDMA, time is divided into slots of equal length. Each node in the network transmits its data in the time slots assigned to it (i.e., a device could be assigned one or more time slots). Implementing TDMA can completely prevent collisions, and any device can use the entire frequency bandwidth whenever conducting a transmission. However, as in all contention-free protocols, TDMA protocols require a central controller to allocate the time slots to the nodes in the network. Additionally, network-wide synchronization is required (i.e., all the devices in the network are synchronized in time). In Fig. 1.5, a simple example of TDMA is shown. The figure illustrates that the time slot is wasted when a device does not transmit, which could degrade the network performance.

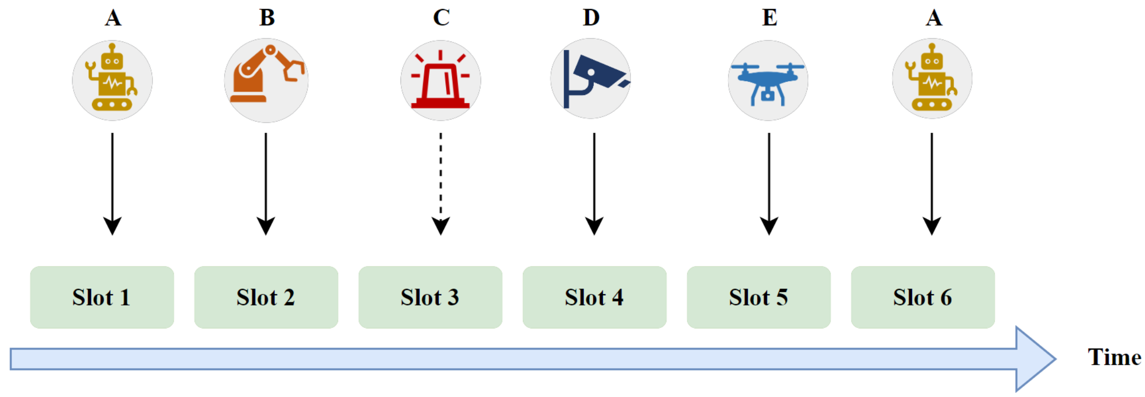

Figure 1.5: An illustration of TDMA scheme. Solid arrows represent ongoing transmissions, and a dashed arrow means the device is idle.

In CSMA protocols, a node senses the medium before initiating a transmission in order to detect the carrier signal and determine whether a transmission is currently taking place by some other device (i.e., check if clear, then send). Unlike TDMA protocols, CSMA protocols do not require centralized controllers since scheduling is unnecessary. CSMA protocols can reduce collisions in MAC but not eliminate them since CSMA protocols are susceptible to hidden terminal problems and simultaneous transmissions. Different protocols have branched from CSMA with different sensing and backoff durations to reduce or mitigate packet collisions. Fig. 1.6 illustrates CSMA.

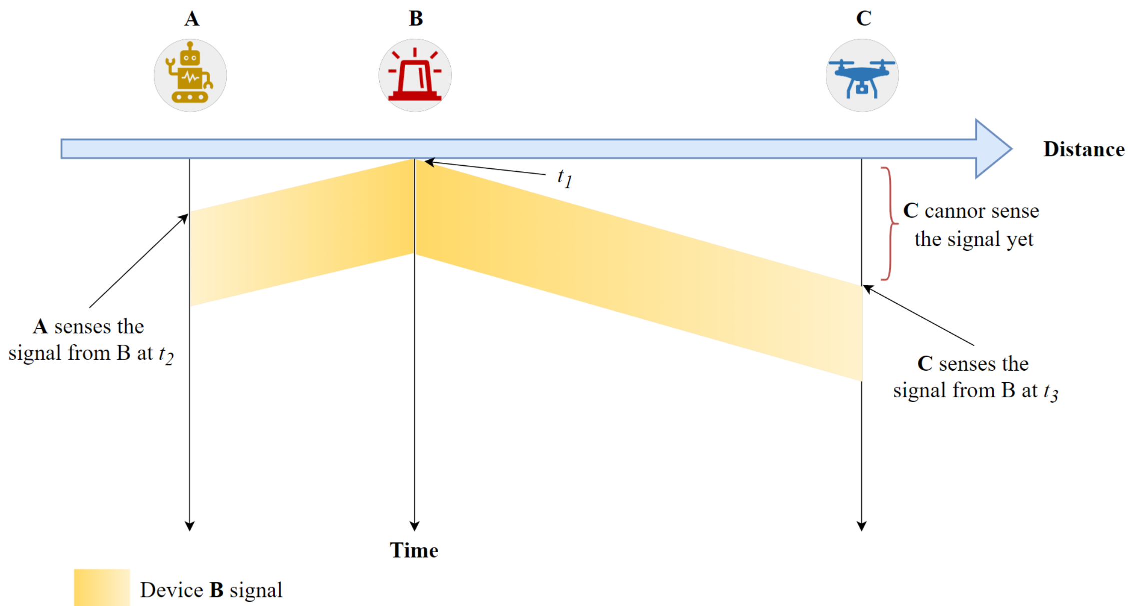

Figure 1.6: An illustration of CSMA scheme. The figure shows that the devices in the network cannot instantly sense a transmission due to the distances among the devices. Therefore, propagation delays need to be accounted for in accordance with the area of the network.

1.3 Objectives

Despite the variety of link-layer solutions in the literature, designing a link-layer solution that simultaneously supports massive connections, low delay, high reliability, and large-scale networks with more than one AP is still an open problem which this thesis aims to solve.

The following are the objectives of the research:

- Examine the current state-of-the-art works that support massive connections and low latency in IIoT networks in order to gain a comprehensive grasp and identify the limitations.
- Develop an exhaustive MAC protocol design that addresses the large network scalability, diverse latency requirements, and high transmission reliability with minimum overhead.
- Evaluate the proposed MAC protocol design to guarantee that the performance requirements can be met.

1.4 Thesis Organization

Chapter Two: In this chapter, we provide a systematic literature review of existing works. The existing works are categorized based on what type of MAC protocol or scheduling scheme they use and which aspect of the IIoT network they try to improve, such as massive connections, low latency, low collision rate, etc.

Chapter Three: In this chapter, we introduce the considered networking scenario in detail. Then we illustrate the scheduling challenges and propose a MAC protocol design that resolves these challenges.

Chapter Four: In this chapter, we present the performance analysis of the proposed MAC design. The analysis includes the packet delay and the throughput. The focus of the analysis is on calculating the average packet delay.

Chapter Five: In this chapter, we present the performance evaluation of the proposed MAC design. In the performance evaluation, we simulate a non-fully connected single-AP network and three scenarios for a non-fully connected multiple-AP network. The results show the average packet delay for low and high packet arrival rates.

Chapter Six: The contributions and the results of the thesis are concluded in this chapter. Additionally, prospective future works and possible extensions of the current work are highlighted.

Chapter 2: LITERATURE REVIEW

2.1 Introduction

In this chapter, we discuss some of the existing works that aim to improve IIoT communication using the data link layer. While connectivity solutions for IIoT can be developed in physical, data link, or even higher layers of the network protocol stack, link-layer solutions usually depend less on hardware and have lower complexity. The works we examine tackle IIoT issues such as latency, massive connection, packet collision, etc.

2.2 Related Works

In this section, works are categorized based on what aspect of the IIoT communication they try to improve or what scheduling approach they used.

2.2.1 Low Latency

Many existing works have studied low latency communication for IIoT applications [25], [26]. In [25], a hybrid MAC protocol design for a single base station (BS) network with M2M communication was proposed. In the proposed method, devices contend for transmission using p-persistent CSMA, and the admitted devices, after the contention period, transmit their packets using TDMA. The aim of this work was to maximize throughput by constructing an optimization problem that can find the optimal tradeoff between the number of time slots allocated to the contention period and the number of time slots in the transmission period since the two components correlate with one another. Only a portion of the devices in the network are presumed to be active, while the rest of the devices are presumed to be in sleep mode. The time frame structure comprises four parts: notification period, contention period, announcement period, and transmission period. In the notification period, the BS broadcasts the commencement of the contention period to all the devices in the network and uses the announcement period to inform the devices that successfully contended for transmission. In the performance evaluation, the average transmission delay (i.e., the time from when a packet is generated until the BS receives it) is shown to be around 100 milliseconds when the number of devices in the network is 500, and the number of active devices is about 30% of the total number of devices.

In [26], the IEEE 802.15.4e deterministic and synchronous multi-channel extension (DSME) protocol was optimized to support low delay and energy-efficient data collection tasks in a multi-hop IIoT network. The optimal scheduling configuration, where different scheduling techniques are constructed using linear programming, includes the optimization of frame and slot lengths and slot and channel assignments. The optimal configuration is created to minimize the time packets spend at each node and reduce the number of hops. The ad-hoc network is presumed to be homogenous and follows a tree model where all nodes, which are synchronized in time, communicate their data to a central root node. Time is divided into frames, and each frame is further divided into two parts, a contention-based transmission period and a contention-free transmission period. Devices in the contention-based transmission period can send their packets using CSMA/CA. Meanwhile, in the contention-free transmission period, time slots are allocated deterministically to the devices in order for them to send their packets. Additionally, the proposed model uses 16 channels in the 2.4 GHz band. The performance evaluation shows that the proposed protocol design can achieve around 20 milliseconds delay per hop.

2.2.2 Massive Connections

Other works have focused on supporting massive connections for IIoT applications [27], [28]. In [27], a contention-based hybrid CSMA/TDMA MAC protocol was introduced that supports MTC for IoT applications in a centralized network with thousands of devices and a single BS. The time structure of the proposed design is based on frames, and each frame is further divided into four parts: notification period, contention period, announcement period, and transmission period. In the contention period, which is based on CSMA, devices contend for admission into the transmission period. The contention period is further divided into three segments of different time durations. Each segment is allocated to devices with different remaining energy levels. The BS uses the notification period to send a message to all the devices in the network to indicate the following: the start of the frame, the duration of each period, the time interval of each segment, and each segment's designated energy level. Successful devices from the contention period and their corresponding slots in the transmission period are broadcasted by the BS during the announcement period. The transmission period is based on TDMA, where successful contenders are assigned time slots to send their packets. The designated time duration of each segment in the contention period changes every frame. Devices are assigned to different segments depending on their batteries' remaining energy. Devices with lower remaining battery energy levels are assigned to longer time segments since a longer duration can have a lower number of collisions. The simulation results show the average end-to-end delay to be between 90 and 100 milliseconds when the proposed design supports between 1500 and 5000 devices.

In [28], a repetition-based grant-free scheme to support massive MTC was proposed. The design used demodulation reference signals (DMRS) to estimate the uplink channel condition and give the BS a list of the resource blocks (RBs) that a device uses to transmit its data. The BS traces the repetition pattern for the device and applies successive interference cancellation to cancel out the interference in the RBs that arise in massive MTC. The proposed protocol design follows a procedure of three steps, Configuration information broadcast, Repetition-based transmissions, and Successive interference cancellation and acknowledgment transmission. In the configuration information broadcast, the BS uses the system information block to broadcast the DMRS and RB configuration information to all the devices in the cellular network. The end devices, in turn, decode the system information block and start sending packets in a repetition-based transmission. In repetition-based transmission, each device arbitrarily chooses the indices of the RB and the DMRS to send its data repeatedly. The randomization makes it highly unlikely that the data transmission repetition sequences established by two or more devices to be identical. In successive interference cancellation and acknowledgment transmission, the BS tries to identify and decode DMRS data and use it to acquire the channel estimates and the RBs indices of the devices' repeated data transmissions to determine the repetition sequence for each device. Once one of the repeated transmissions for a device is decoded, the BS implements successive interference cancellations for the rest of the RBs for that device. This process, in turn, helps the decoding of other devices' RBs. The simulation results show that the proposed protocol design could achieve around 200 milliseconds delay when 1000 devices are connected to a single BS network.

2.2.3 Low Latency and Massive Connections

There are a few works that target both low delay and massive connections at the same time [29], [30], [31]. In [29], a MAC scheduling solution for uplink and downlink transmissions that supports URLLC and mMTC was proposed in the context of network slicing. Each network slice could either support URLLC or mMTC. The suggested strategy is based on a two-level scheduling approach, the first level is dedicated to scheduling the devices in each slice, and the second level is in charge of resource allocation within each slice. Slices are created, configured, and managed by a dedicated controller. Depending on the slice requirements, there is a different number of radio access network functions for each slice. Therefore, each slice may use different scheduling algorithms to assign channel resources to its devices. In the proposed design, the URLLC slice can achieve low latency for the end devices as long as the number of devices does not exceed a certain threshold. The threshold depends on the slice-dedicated bandwidth and the packet size. The performance evaluation was focused on the URLLC slice, demonstrating that the proposed protocol design can achieve less than 3 milliseconds of end-to-end delay for a slice with 5 active devices, each with a small packet. In comparison, the delay exceeds 40 milliseconds when 30 active devices are involved in the slice, each with a large packet. The performance evaluation did not cover mMTC.

In [30] and [31], a novel MAC protocol and the corresponding scheduling method were designed to support massive connections with low delay and high reliability in a fully-connected IIoT network with one AP. The proposed protocol design includes four components that are designed for small-sized packets. The protocol design comprises a novel time slot structure, channel access procedure, and two approaches that provide diverse QoS and support ultra-dense networks. The proposed protocol may achieve no packet collision or a low packet collision probability when the network device density increases. The four components are Mini-slot based carrier sensing (MsCS), Synchronization carrier sensing (SyncCS), Differentiated assignment cycles, and Superimposed mini-slot assignment (SMsA). In the MsCS, multiple devices are assigned to a time slot. Each device is assigned a mini-slot, which the device uses to sense the channel before initiating a transmission in order to prevent collisions. A device with a lower index mini-slot (e.g., a device assigned the first mini-slot in the time slot) would have a higher priority to transmit than a device assigned a higher mini-slot index (e.g., a device assigned mini-slot five). SyncCS detects and then skips idle slots (i.e., when no device assigned to the slot initiates a transmission). SyncCS requires a fully connected network. In the differentiated assignment cycles, devices with different priorities have different assignment cycles that translate to different frame lengths. This way, devices with higher priority can be allocated shorter frames to guarantee their QoS. At the same time, devices with low priority are assigned to longer frames, which translates to higher delays, where longer frames can support a larger number of devices. In SMsA, instead of assigning one device per mini-slot, SMsA allows multiple devices to be assigned per one mini-slot. This design can significantly increase the number of supported devices in the network at the expense of loosing the no packet collision guarantee. The performance evaluation shows different delay results depending on which of the four components were used, the packet arrival rates, and the number of devices in each assignment cycle (i.e., number of high, regular, and low priority devices), which corresponds to different frame lengths. The simulation was conducted for a single-AP network with 1000 devices, of which 50 devices had stringent delay tolerance, which does not exceed 1 millisecond, while the rest of the devices had regular and high delay tolerance. On average regular delay tolerance is around 10 milliseconds, while high delay tolerance is around 80 milliseconds. The delay results are as low as submillisecond for devices with stringent delay requirements and as high as 120 milliseconds for devices that do not require a very low delay.

2.2.4 Collision-Free Communication in Distributed Networks

In the following group of works, the aim was to design collision-free MAC protocols for ad-hoc wireless networks that support different applications [32], [33], [34]. In [32], a novel collision-free MAC protocol that supports multimedia applications was proposed for a wireless mesh backbone consisting of several routers located at fixed sites and covering a large region. All routers are synchronized in time and use one communication channel. Routers can transmit directly to their one-hop neighbors. Two routers can have simultaneous transmissions if they are at least two hops away from each other. Time is divided into time slots, and each slot is further divided into two parts, sensing and transmission. The sensing part comprises mini-slots, where a mini-slot is assigned to a router. A router sends a jamming signal at its designated mini-slot (e.g., mini-slot one) to inform all its one-hop neighbors that it will initiate a transmission. The neighboring routers assigned to the rest of the mini-slots (i.e., mini-slot two and higher) sense the channel for the jamming signal. Once a jamming signal is detected, they give up their intended transmissions. Routers within two hops can be assigned to the same mini-slot since they can transmit simultaneously without collisions. The proposed MAC protocol scheme ensures a fair opportunity for all neighboring routers to transmit by rotating the mini-slots. Priority access is maintained by adding one or more mini-slots ahead of all the other ordinary access mini-slots. Priority access mini-slots are fixed (i.e., they do not rotate). The performance evaluation shows the data packet access delay increases exponentially when the packet arrival rate approaches 60 packets per second, with a delay of 16 milliseconds. The lowest delay is 2 milliseconds, achieved when the packet arrival rate is 30 packets per second.

For fully connected wireless ad-hoc networks, a novel energy-efficient MAC protocol design was proposed in [33]. The design aims to reduce energy consumption for battery-operated devices as in mobile devices with real-time communication, like voice or video calls, while maintaining low delay and packet loss requirements. The proposed protocol uses a temporary coordinator node to minimize energy consumption by scheduling active and sleep durations for the regular nodes' radio interfaces in a distributed manner, which lowers MAC overhead and transmission collisions across the nodes. When a node intends to send packets, it only has to contend once for transmission. After that, the temporary coordinator node allocates transmission time as long as the contended node has packets ready for transmission. At the beginning of each beacon interval, all nodes are awake (i.e., can sense the network) for a brief period of time to receive the scheduling information from the coordinator node. Typically, nodes are awake at their designated sending and receiving periods. The time frame is split into three parts, announcement period, contention-free period, and contention-based period. In the announcement period, all the network nodes are awake to receive the scheduling information. In the Contention-free period, nodes scheduled for transmission and those for reception are active. In the Contention-based period, nodes with packets ready for transmission can send their transmission requests to the coordinator node. The coordinator node determines the length of each period based on the network's state. Additionally, the current coordinator node chooses a new coordinator and updates it with a table containing the scheduled transmissions and the transmission requests from all the nodes in the network. The performance evaluation shows the average packet delay from the instant the packet is generated until it reaches its destination, which depends on the number of nodes in the network and the traffic load (i.e., packet arrival rate). The lowest average delay is approximately 150 milliseconds and is obtained when the number of nodes in the network is 10, and the traffic load is 200 packets per second. The average delay increases exponentially beyond 1000 milliseconds when the number of nodes in the network is 20, and the traffic load is greater than 600 packets per second.

In [34], a MAC design was proposed for a single-hop, single-channel, non-fully connected wireless ad-hoc network with randomly distributed communication links. For optimal use of the available spectrum and minimal power consumption, the proposed design incorporates the opportunistic access aspect of WiFi networks and the deterministic transmission aspect of cellular networks. Contention-free time slots for data transmissions and receptions are dynamically scheduled by a group of coordinator nodes located across the network, which are presumed to be immobile. The coordination is based on transmission requests from the regular nodes. Each coordinator node schedules the transmissions and receptions of regular nodes in its vicinity (i.e., cell) after it receives their transmission requests. The proposed design eliminates collisions by ensuring that whenever a node is designated to receive a packet, the space around the node would be reserved (i.e., no other node within its communication range receives nor transmits a packet concurrently). Neighboring coordinator nodes share the transmission and reception scheduling table with each other to eliminate transmission interference in the reserved space around receiver nodes since some transmissions can take place from one coordinator node cell to another coordinator node cell. All the nodes in the network are synchronized in time. The time frame has three types of time slots, scheduling slots, contention-free slots, and contention slots. Scheduling slots are used to coordinate transmissions and receptions in the current frame. These slots are received by all neighboring coordinator nodes and regular nodes in the cell. In the contention-free slots, regular nodes conduct their scheduled transmissions and receptions. A node scheduled for transmission can include a request for a future transmission in the header of its packet. In contention slots, regular nodes send their transmission requests to their coordinator nodes. Packet delay was not considered in the performance evaluation of this paper. The considered metrics are throughput, energy consumption, and collision rate, which were compared with protocol designs from similar works.

2.2.5 Location-Based Scheduling

In [35] and [36], methods that employ the location information of end devices were proposed to solve hidden and exposed terminal problems. In [35], a method was proposed to address the hidden and exposed terminal problems for multiple-AP networks using virtual sectors and CSMA/CA protocol. A critical component of the proposed solution is utilizing mobile devices' exact location information, which is gathered through the Global Positioning System (GPS) and the Quasi-Zenith Satellite System (QZSS). The location-based virtual sector method divides each AP's cell (i.e., communication range) into geographical sectors. Mobile devices are grouped into these sectors where they can sense each other when transmitting using CSMA/CA, which resolves the hidden terminal problem within each AP. This solution neglected the hidden terminals that arise from the overlapping cells of neighboring APs. Sectors in adjacent APs are synchronized such that two different sectors from adjacent APs are allocated the same time window for their devices to transmit to reduce exposed terminal problems. All mobile devices are presumed to have GPS and QZSS and can accurately report their locations to the APs in their cells. Each AP broadcasts the sector assignment for the mobile devices in its cell, along with the time slots allocated to each sector.

An extension of the work in [35] is presented in [36], where the focus is on a single-AP network. An optimum allocation strategy for the virtual sector method is proposed to enhance fair channel access among mobile devices by governing the number of time slots allocated to the virtual sectors. Time slots are proportionally allocated to the virtual sectors in accordance with the number of mobile devices in each sector. A minimum amount of time slots is designated for each sector by default, so if a device moves into an empty sector, it could still initiate a transmission. The performance evaluation of this paper presents the throughput and collision rate results while comparing the proposed protocol with similar existing protocols. The throughput significantly degrades when the number of devices in the network is more than 20, and when the device positioning error is greater than 20% of the cell radius. The collision rate is significantly exacerbated due to increased overhead when the duration of the data packet is as small as 0.1 milliseconds in a network with 20 devices, even when the positioning error is 0%.

2.2.6 Semi-Persistent Scheduling (SPS)

In [37], and [38], two scheduling strategies were proposed to support MTC and IoT applications, respectively, using SPS, which is a scheduling mechanism that combines features from both persistent scheduling and dynamic scheduling in order to reduce scheduling overhead at the controller (i.e., AP or BS). A method was proposed in [37] that supports delay reduction for narrowband IoT applications for a network with a single BS. The proposed approach relies on implementing two techniques, SPS and short Transmission Time Interval (sTTI). The time structure is comprised of time frames in which each frame has a duration of 10 milliseconds. Each time frame is further divided into 10 subframes. A sub-frame represents 14 orthogonal frequency-division multiplexing (OFDM) symbols. The subframe is the minimum transmission time interval (TTI) needed for a device to send a packet to the BS. To reduce delay, the author shortens the TTI by using fewer OFDM symbols per TTI. SPS is used for devices with periodic transmission to minimize the unnecessary requests to send (RTSs) packets. In contrast, devices with sporadic transmission continue to use dynamic scheduling since permanently allocating resources for these devices wastes the network resources. The simulation results show that the proposed approach can achieve an uplink latency of under 80 milliseconds for a network of 30 devices.

In [38], the authors proposed a hybrid scheduling LTE strategy for a single BS network that supports HTC and MTC, where the MTC is tailored towards IoT applications. HTC devices are supported by standard LTE dynamic scheduling, while SPS supports MTC devices. The uplink transmission uses single carrier frequency division multiple access (SC-FDMA), where each device is allocated several RBs. The BS uses SPS to pre-assign each MTC device with several RBs. The RBs' assignment continues for a period of time until the BS reconfigures it. Function-wise, the dynamic scheduling used for HTC devices can be divided into time-domain and frequency-domain schedulers. The time-domain scheduler's role is to choose the set of devices to be scheduled for transmission in the upcoming time cycle. The frequency-domain scheduler determines the number of RBs assigned to each device from the set of devices selected by the time-domain scheduler. The simulation results show that the proposed scheduling strategy can support 1100 MTC devices with an average delay of 40 milliseconds. For HTC, the proposed design can support a maximum of 30 devices before surpassing the 150 milliseconds delay threshold for voice packets and the 300 milliseconds delay threshold for video packets.

2.3 Contribution

In this thesis, we propose a MAC protocol design that builds upon the works in [30] and [31] that simultaneously supports massive connections and low average delay. Extending the design of [30] and [31] to fit large-scale networks is of interest but challenging since a direct extension can result in severe hidden terminal and exposed terminal issues. Building on the foundation of the above two works, we design a device location-based time slot assignment scheme to support large-scale non-fully connected IIoT networks with massive connections while maintaining low delay and zero-collision. We first model the network scenario, then we analyze the average packet delay using a discrete-time Markov chain and analyze the idle slot probability. Finally, we validate the performance of our design for different networking scenarios via extensive simulations using MATLAB.

Chapter 3: NETWORK SCENARIO & PROTOCOL DESIGN

3.1 Introduction

In this chapter, we define the layout and the characteristics of the IIoT network considered in the proposed design and introduce the time slot structure we utilize in the proposed MAC protocol. Then, we discuss the slot assignment challenges for non-fully connected single-AP and non-fully connected multiple-AP networks. Finally, we present our design that resolves the problematic slot assignment for a non-fully connected single-AP network and a non-fully connected multiple-AP network.

3.2 Network Scenario

We consider a large-scale non-fully connected IIoT network with multiple APs. Each AP has a limited coverage area (e.g., an industrial plant) and connects a large number of devices (e.g., a thousand devices).

The communication and sensing ranges of all devices are assumed to be equal for simplicity of modeling, and the radius of each AP's coverage is the same as the communication range of the devices. To eliminate any coverage hole, the coverage of the APs can overlap. We assume that any device in the network is covered by at least one and at most three APs. The APs are connected, synchronized in time, and able to communicate with each other.

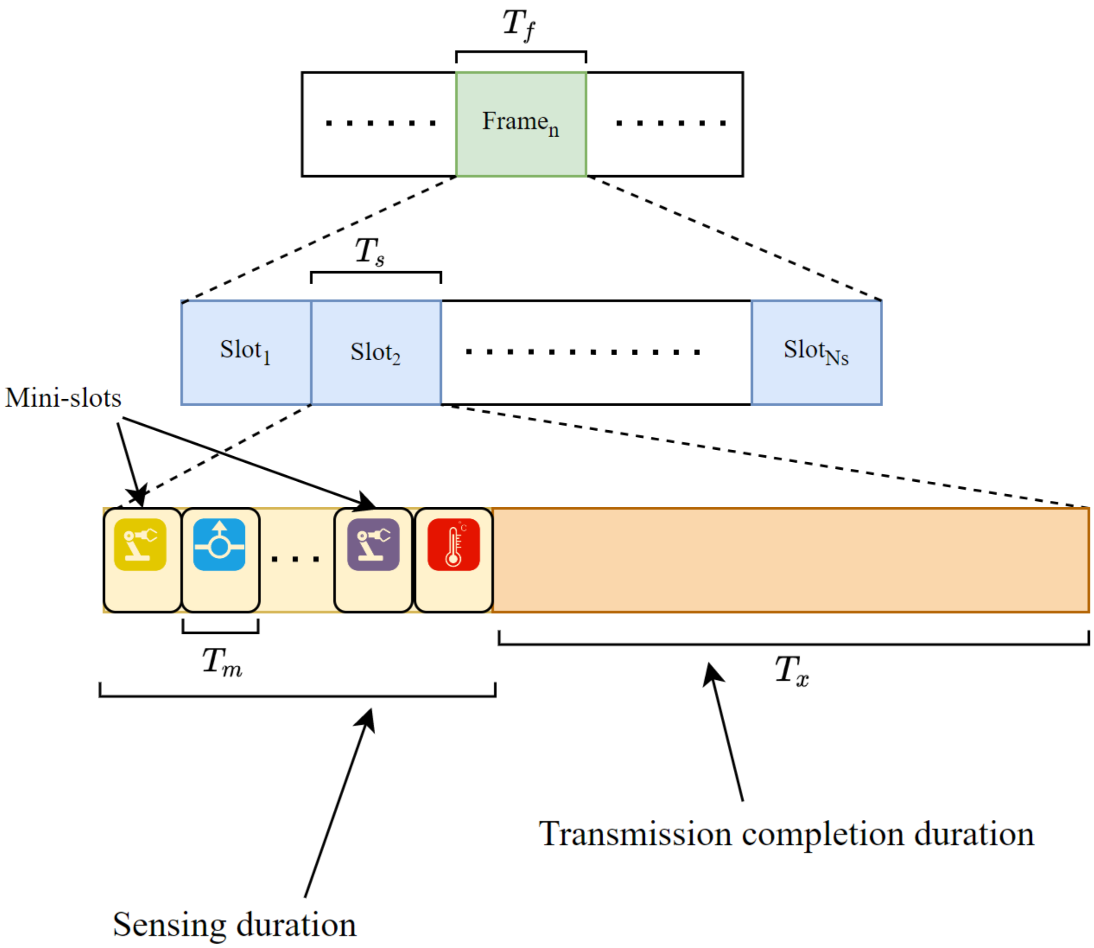

Figure 3.1: The time structure.

In the IIoT use case, devices have no or limited mobility and relatively low packet arrival rates (e.g., compared to the multimedia use cases). Nevertheless, the packet arrival rates of IIoT devices can be higher than general IoT devices such as water or gas meters in a smart home use case, which generates one message per hour or even per day [7]. In this work, the considered IIoT devices have an arrival rate ranging from one packet every several seconds to several packets every second. Due to the nature of machine-type communications, the size of each packet is small (e.g., down to a few bytes) [39].

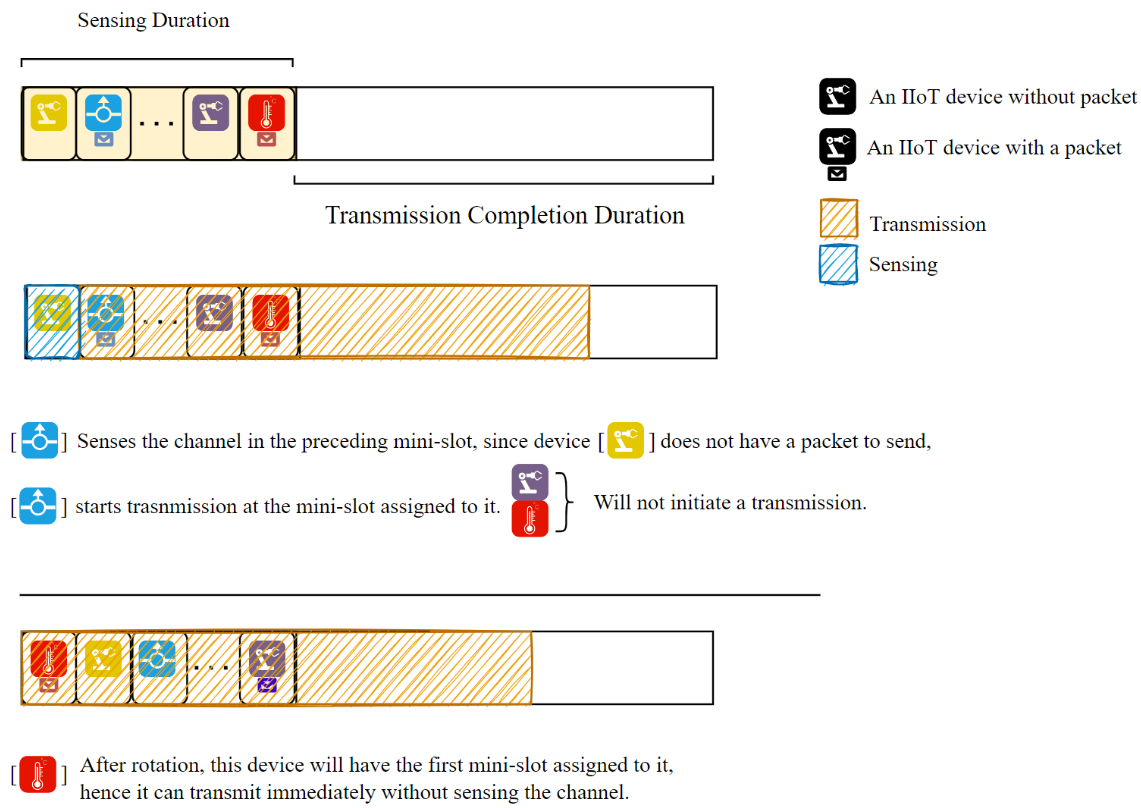

Figure 3.2: The sensing, transmission, and rotation mechanisms.

The frame and slot structure from [30] for time-slotted channel access with carrier sensing is adopted here as the foundation of our design. To build on this foundation, we first briefly introduce its basics as follows. The time structure is divided into frames with duration T_f, and each frame is further divided into N_s slots, the duration of each slot is T_s, as shown in Fig. 3.1. A time slot begins with N_m mini-slots for sensing, each mini-slot has a duration of T_m, followed by the remaining part of the slot with duration T_x for completing a transmission. Each mini-slot is assigned to one device. Any device except the one assigned to the first mini-slot must sense the communication channel in the preceding mini-slot before initializing a transmission. The mini-slots assignment can rotate every frame to ensure fair channel access among all devices sharing the slot, and the sensing and rotation processes are shown in Fig. 3.2. Essentially, the mini-slot structure and the corresponding sensing design allow multiple devices to share a slot without collisions, which significantly increases channel utilization. In addition, channel access is fully distributed after the initial assignment of devices to slots and mini-slots.

In the rest of this thesis, we denote the packet arrival rate for any device in the slot by λ. Then, the expected number of packets to arrive over the duration of a frame at one device is λT_f. The average expected number of packets to arrive at all the devices sharing a slot is $N_m \lambda T_f$.

The APs broadcast the time frames with the slot and mini-slot assignments to the end devices. Since devices are limited to no mobility in the network scenario we consider, the scheduling does not frequently change, which makes downlink transmissions rare compared to uplink transmissions. Therefore, we only evaluate uplink transmissions in this work.

3.3 Protocol Design

In this section, the necessity of location-constrained device scheduling in large-scale and non-fully connected networks is illustrated. Then, our protocol design for large-scale IIoT networks, which avoids collisions while connecting a massive number of devices, is presented. To be comprehensive, we consider two networking scenarios when introducing our protocol design, a non-fully connected single-AP network and a non-fully connected multiple-AP network.

3.3.1 Non-fully Connected Single-AP Network

To illustrate the necessity and idea of location-based device assignment, consider for now a non-fully connected IIoT network with only one AP. When the network devices are assigned time slots in accordance with the frame and slot structure shown in Fig. 3.1 without consideration for their locations, collisions can happen due to hidden terminals. For example, devices x and y covered by AP-A, shown in Fig. 3.3, are distant from each other, and device y cannot sense device x's transmission. If devices x and y are assigned to the same slot, then device y may falsely perceive an idle channel while device x is transmitting, leading to collisions. To solve this problem, the slot assignment must take the locations of the devices into consideration. Specifically, a group of devices can be assigned to the same slot only if they can sense each other. For example, devices x and z in Fig. 3.3, which are close to each other, can be assigned to mini-slots 1 and 2, respectively, of the same slot (e.g., slot 2), while devices x and y cannot be assigned to the same slot.

3.3.2 Non-fully Connected Multiple-AP Network

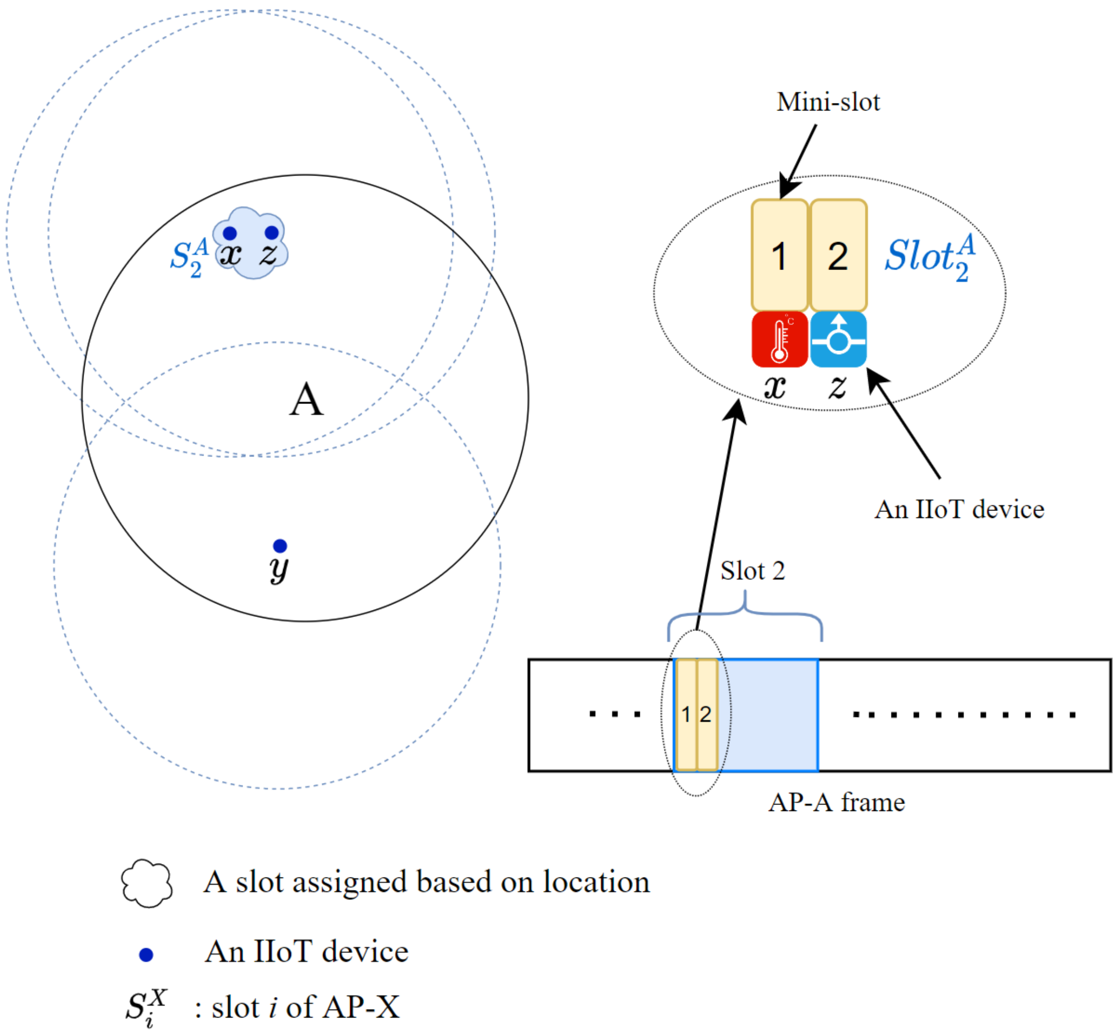

Figure 3.3: Device location-based slot assignment for a single-AP network. Solid and dashed circles represent the AP's coverage and the device sensing range, respectively.

In an IIoT network with multiple APs, device location-based slot assignment is insufficient to prevent hidden and exposed terminals. Therefore, slot assignment coordination among all the APs in the network is necessary. As the coverage of the APs overlap, the entire area is divided into overlapping zones (OZs) and non-overlapping zones (NOZs). Although each device is connected to only one AP, a device's transmission can reach more than one AP if it is in an OZ. Suppose the device location-based slot assignment paradigm for the single-AP network, as mentioned in subsection 3.3.1, is applied separately at each AP. Specifically, each AP assigns slots to its connected devices without coordination with other APs. For illustration, AP-A and AP-B, in Fig. 3.4, have one OZ. AP-A assigns slot 3 and slot 2 to its devices in the OZ and NOZ,

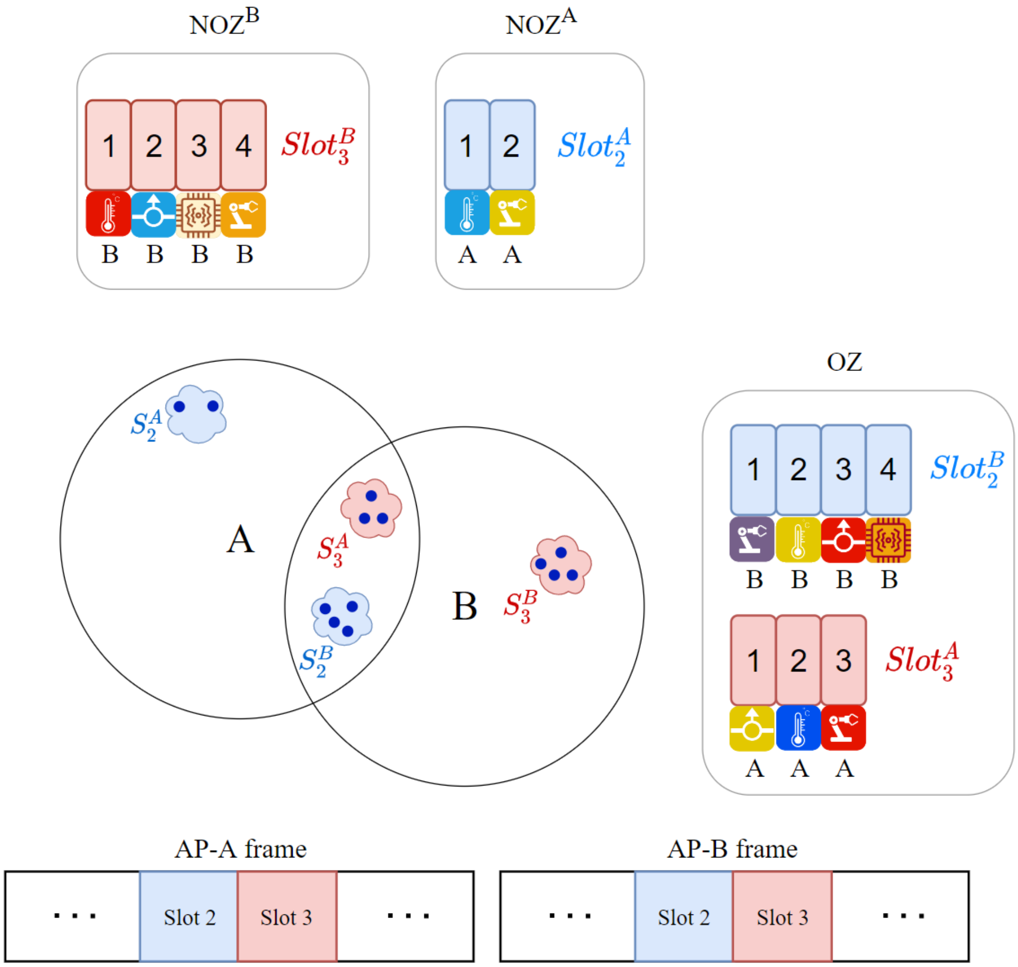

Figure 3.4: Device location-based slot assignment without coordination among the APs in a multiple-AP network. In the figure, different colors represent different time slots. Devices assigned to slots of the same color can transmit simultaneously. The letter under the device refers to which AP the device is connected.

respectively. AP-B assigns slot 2 to its devices in the OZ and slot 3 to the devices in its NOZ. When devices in the OZ transmit to AP-A, they could become hidden terminals to the devices in the NOZ connected to AP-B, causing packet collisions at AP-B and vice-versa.

Applying the device location-based time slot assignment scheme with partial coordination can be problematic. That is both APs in Fig. 3.4 allocate the same slots from each AP's slot assignment to the OZ and the same slots to the NOZs of each AP, as shown in Fig. 3.5, without further coordination between the two APs. In Fig. 3.5, we illustrate the partial coordination, where AP-A assigns the devices connected to it in the OZ slot 5, and AP-B assigns the devices connected to it in the OZ slot 5 as well. Each AP assigns the

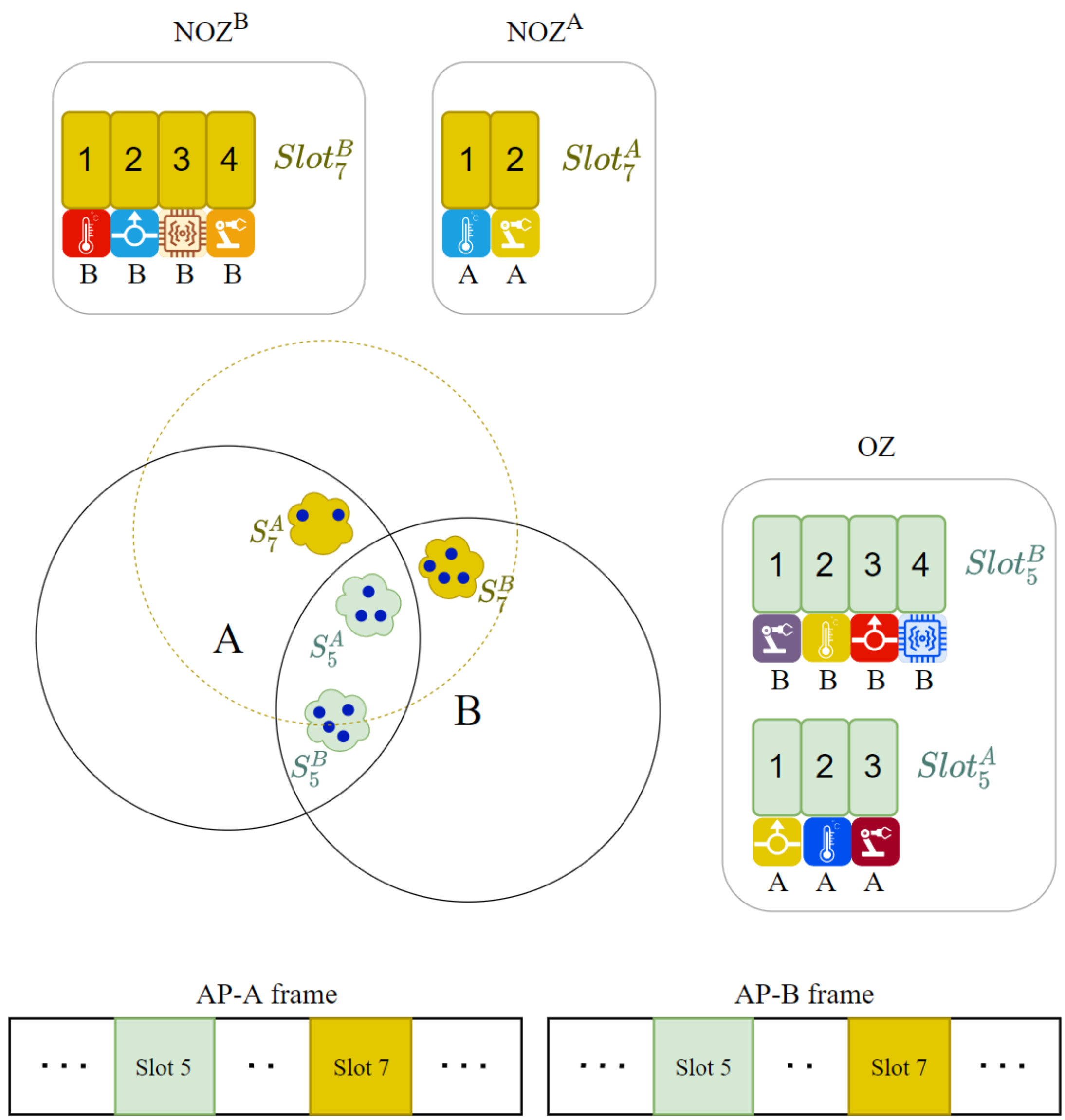

Figure 3.5: Device location-based slot assignment with partial coordination among the APs in a multiple-AP network.

devices in its NOZ slot 7.

Two problems arise in this scenario, one in the OZ and one in the NOZs. In the OZ, suppose two devices have the same mini-slot order (e.g., mini-slot 1) of slot 5, one is connected to AP-A, and the other is connected to AP-B. If both devices transmit simultaneously, both of their packets will be received at AP-A and AP-B, causing collisions at both APs. The two devices cannot sense each other due to the simultaneous transmission. In the NOZ, AP-A devices and AP-B devices can transmit simultaneously without collisions. However, nothing prevents devices from AP-A NOZ and devices from AP-B NOZ from sensing each other.

Suppose a device connected to AP-A assigned to slot 7 initiates a transmission; in that case, a device connected to AP-B also assigned slot 7 can sense its carrier signal and unnecessarily refrain from transmission, i.e., the exposed terminal problem as shown in Fig. 3.5.

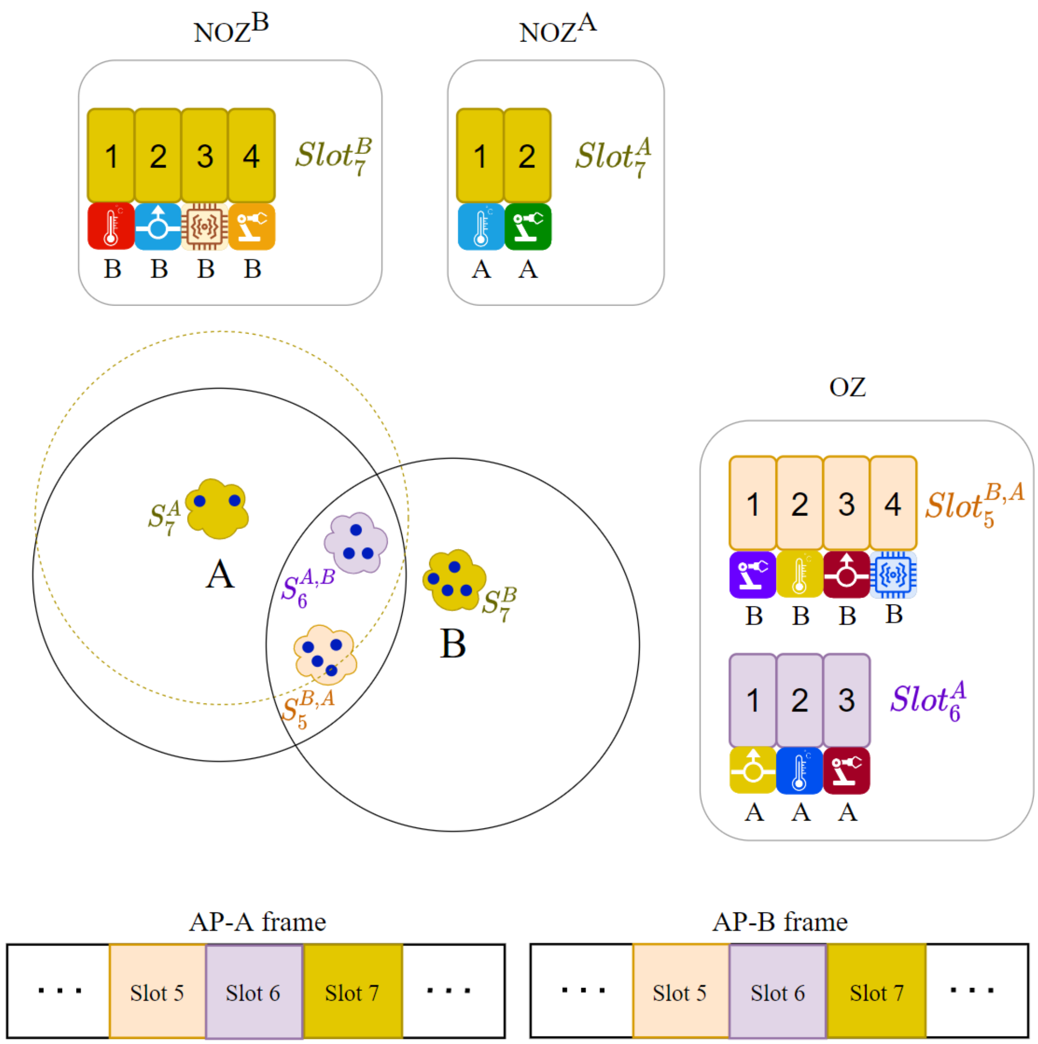

Figure 3.6: Device location-based slot assignment with full coordination among the APs in the network. The figure Shows one slot is added from the OZ to each AP when the APs are fully coordinated.

To completely solve hidden and exposed terminal problems in the non-fully connected network with multiple APs, full coordination among the APs is necessary in addition to the device location-based slot assignment. In the OZ, all the devices are treated as if they belong to a single-AP. Each device in the OZ is assigned the same time slot and mini-slot at all APs covering the OZ, even though it is only connected to one of them. This way, no two devices in the OZ can initiate a transmission simultaneously. Regardless of whether a device is connected to it, each AP will include all devices in the OZ in its slot assignment, as shown in Fig. 3.6. The slot assignment for the devices in the OZ is replicated for all the APs sharing the OZ. Such an arrangement increases the number of required slots to support the devices at each AP and causes a higher ratio of idle slots at each AP, as the cost of eliminating collisions. In Fig. 3.6, the devices in slot 6 are connected to AP-A, and the devices in slot 5 are connected to AP-B. Nevertheless, both devices and slots are included in AP-A's and AP-B's slots assignment.

In the NOZ, the APs coordinate to solve the exposed terminal problem. Devices located in different NOZs can transmit simultaneously without causing collisions. An exposed terminal is created when two devices, each from a different NOZ, are located such that each device can sense the carrier signal of the other device. Each AP assigns the devices in its NOZ after coordinating with the neighboring APs so that all APs ensure the devices that transmit simultaneously are out of each other's coverage. This is shown in Fig. 3.6, where AP-A assigns a group of its devices slot 7, and AP-B assigns a group of its devices slot 7 (i.e., a device from slot 7 in AP-A can transmit simultaneously with a device from slot 7 in AP-B). Through the coordination, slot 7 in each AP is assigned to a group of devices in the NOZ of that AP such that no device in either group can sense the carrier signal of the other group.

The proposed design of this thesis cannot support mobility when devices move across zones, but it can be extended to support mobility. A special category of time slots can be introduced to support mobility by modifying the mini-slots. We will discuss it in the future work section of the conclusions.

3.4 Summary of Design

In this chapter, we defined the network scenario and the time slot assignment, which allows for massive connectivity. Then we discussed the challenges of hidden and exposed terminal problems when time slot assignment is used in a non-fully connected single-AP network and a non-fully connected multiple-AP network. Afterward, we introduced the device location-based time slot assignment. The proposed scheduling scheme resolves the hidden terminal problem in a non-fully connected single-AP network and the hidden and exposed terminal problems in a non-fully connected multiple-AP network. The device location-based time slot assignment requires network-wide APs synchronization and increases frame length.

Chapter 4: PERFORMANCE ANALYSIS

4.1 Introduction

In this chapter, we analyze the packet delay from the instant a packet is generated till the instant when the packet is successfully transmitted and analyze the idle slot probability to evaluate the performance of our device location-based slot assignment scheme. In the analysis, we focus on the devices sharing a time slot, and the result applies to any device in any time slot. Mini-slot rotation is applied to ensure fair channel access among all the devices in the same slot. For simplicity of modeling, we make the following two assumptions. [1]

- All the devices in a time slot have the same packet arrival rate. In practice, a device with a higher packet arrival rate can be assigned to multiple slots.
- Each device buffers at most one packet, including the packet currently being transmitted, i.e., a freshly generated packet replaces an existing packet, if any, in the buffer, unless the existing packet is transmitting. As a result, each device has either 0 or 1 packet in its buffer at any time. This scenario is common for devices such as sensors.

The fully-coordinated device location-based time slot assignment in section 3.3 avoids collisions caused by hidden terminals, while the mini-slot-based carrier sensing avoids other collisions [30], hence all transmissions are collision-free. Additionally, in the scenario considered above, all the devices in the same slot have the same average delay over time as long as the mini-slot rotation is executed for a sufficiently long time.

4.2 Packet Delay

Packet delay is the focus of our performance analysis. We model the packet generation and transmission process of the devices in the target time slot with a virtual transmission queue using a discrete-time Markov chain (DTMC). In the DTMC, the state is the number of packets in the virtual queue. State transitions take

[1] Note that these assumptions are only introduced for the sake of analysis and are not required for our scheduling scheme to be applicable.

place right after the end of the last mini-slot of each slot. The state index ranges from 0 to $N_m - 1$ (i.e., the highest state). The value of the highest state is one less than the number of devices in the slot since exactly one packet is transmitting at the state transition instant when the virtual queue is not empty, as the currently transmitting packet is not considered to be in the queue. The state index can transition to any higher index, remain the same, or decrease by only one, since at most one packet is sent in the target slot in any frame. The following factors determine the state transition in the virtual transmission queue:

- The number of packets currently waiting in the virtual queue, denoted by i, which is the current system's state.
- The number of packets generated from the group of devices sharing the slot, denoted by j, which takes values between zero and N_m.
- The number of generated packets that replace existing packets, denoted by k. A portion of the generated packets can replace existing packets. A generated packet replaces an existing packet in the virtual transmission queue if generated by a device that currently has a packet waiting for transmission.

Without loss of generality, in evaluating the proposed design, the packet arrival rate (i.e., λ) at each device is presumed to follow a Poisson process. Hence, the probability of a packet being generated by any device assigned to the same slot is given by ρ in (4.1).

$$\rho = 1 - e^{-\lambda T_f}. \tag{4.1}$$

The system probability at state i, when j packets are generated, and k packets are replaced is given in (4.2), where $p(i, j, k)$ serves as the basis for calculating the transitional probability from one state to another.

$$p(i, j, k) = \binom{i}{k} \binom{N_m - i}{j - k} \rho^j (1 - \rho)^{N_m - j}. \tag{4.2}$$

The analysis can be extended to support devices with different arrival rates by modifying ρ in (4.2). Due to packet replacement, the transition from state a to state b can happen with a different number of generated packets, (i.e., j in (4.2)), j can take values between zero and N_m. For example, the system may transition from state two to state six when the value of j is greater or equal to 5 and less or equal to 7, depending on how many of the existing packets are replaced.

The transitional probability from state i to state h is denoted by $q_{i,h}$, where i is the current state, and h is the future state is given in (4.3). The unique case when the value of i is zero, and the value of h is zero (i.e., the system remains at state zero) is defined separately in (4.3) since it is the only case when no packet is transmitted.

$$q_{i,h} = \begin{cases} 0, & \text{if } i \geq h+2 \\ p(0,0,0) + p(0,1,0), & \text{if } i = 0, h = 0 \\ \sum_{j=j_{\min}}^{j=j_{\max}} p(i,j,k^\star(i,j,h)), & \text{otherwise,} \end{cases} \tag{4.3}$$

where $j_{\max}$, $j_{\min}$, and $k^\star(i,j,h)$ are determined as follows:

$$j_{\max} = \min(N_m, h+1) \tag{4.4a}$$

$$j_{\min} = \max(0, h-(i-1)) \tag{4.4b}$$

$$k^\star(i,j,h) = \min(i, j, j-(i-(h+1))). \tag{4.4c}$$

We define $Q = [q_{i,h}]_{i,h\in\{0,\ldots,N_{m-1}\}}$ as the state transition probability matrix for a given slot with N_m mini-slots. Later in this section, we use Q in (4.7) to find the steady-state probability.

The diagram in Fig. 4.1 shows the DTMC diagram of the state transition model. The diagram represents the number of packets in the system that are queuing for transmission and the transitional probabilities from one state to another.

To elaborate on (4.3), here we illustrate the conditions for transitioning from one state to another. There are two sets of conditions for the system to transition from the current state to a new state. The first set of

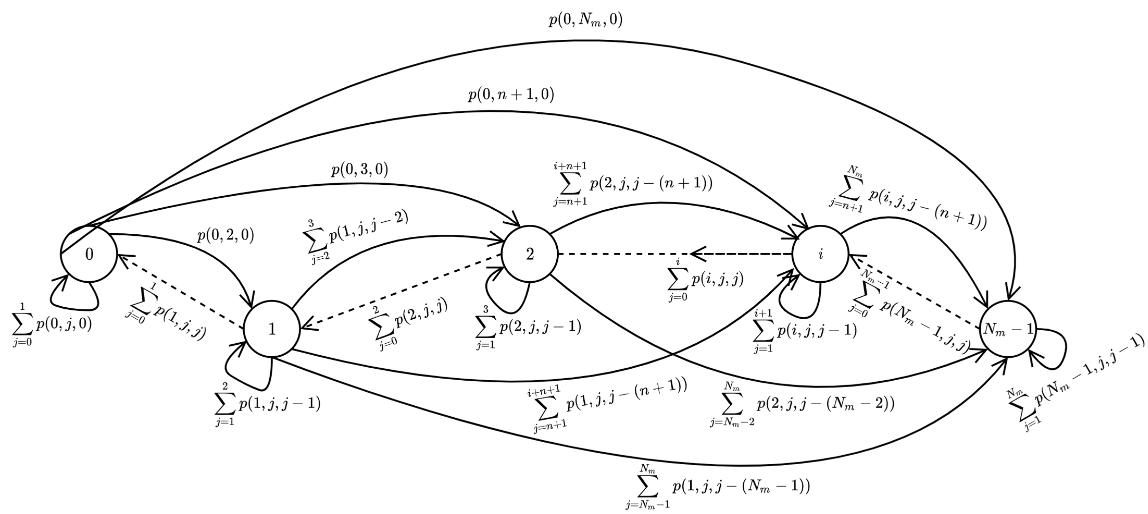

Figure 4.1: The state transition diagram of the system showing the transitional probabilities.

conditions is applied when the system is in state zero (i.e., $i = 0$). This implies that $k = 0$.

$$i_t = i_{t-1} + n, \tag{4.5}$$

where i_t represents the new state, i_{t-1} represents the currents state, and n represents the number of states by which the system increases when j packets arrive. Since there is no packet replacement when $i = 0$, the state transition solely relies on j, according to (4.6a):

$$0 \leq j \leq n + 1. \tag{4.6a}$$

The second set of conditions is applied when the system is in any state other than zero (i.e., i is greater than zero and less than N_m), which depends on how we constructed the DTMC and the instant at which the system changes states. That is, only one packet is transmitted in the target slot every frame, and the system state might increase, stay the same, or decrease by one state. We continue to use (4.5) to express the transitional conditions. The state transition relies on the values of j and k in accordance with (4.6b) and

(4.6c).

$$n + 1 \leq j \leq i + n + 1 \tag{4.6b}$$

$$k = j - (n + 1). \tag{4.6c}$$

The value of n equals the number of states by which the system will shift from its current state. When $n = 0$, the system state does not change, and when $n = -1$, the system state is reduced by one. Similarly, when $n = 3$, the system state increases by 3 and so on. Based on the above conditions, transitioning from one state to another takes on a range of j values in accordance with (4.3) - (4.6c). The Q matrix in (4.7) represents the state transition probability matrix from one state to another. The transitional probability $q_{i,h}$ from (4.3) is an element in Q.

$$Q = \begin{pmatrix} \sum_{j=0}^{1} p(0,j,0) & p(0,2,0) & ... & p(0,n+1,0) & ... & p(0,N_m,0) \\ \sum_{j=0}^{1} p(1,j,j) & \sum_{j=1}^{2} p(1,j,j-1) & ... & \sum_{j=n+1}^{n+2} p(1,j,j-(n+1)) & ... & \sum_{j=N_m-1}^{N_m} p(1,j,j-(N_m-1)) \\ 0 & \sum_{j=0}^{2} p(2,j,j) & ... & \sum_{j=n+1}^{n+3} p(2,j,j-(n+1)) & ... & \sum_{j=N_m-2}^{N_m} p(2,j,j-(N_m-2)) \\ ... & ... & ... & ... & ... & ... \\ ... & ... & ... & ... & ... & ... \\ 0 & ... & \sum_{j=0}^{i} p(i,j,j) & \sum_{j=1}^{i+1} p(i,j,j-1) & ... & \sum_{j=n+1}^{N_m} p(i,j,j-(n+1)) \\ 0 & ... & 0 & 0 & \sum_{j=0}^{N_m-1} p(N_m-1,j,j) & \sum_{j=1}^{N_m} p(N_m-1,j,j-1) \end{pmatrix}. \tag{4.7}$$

To calculate the overall packet delay, first, we calculate the packet delay at the different states, which is given in (4.8). Three delay components are considered when calculating the delay in any particular state. The first component is the base delay, measured from the packet arrival time to when the first mini-slot begins. The second component is the delay the queuing packet accumulates by transitioning from a previous state to the current state. For instance, when the system is currently in state five, and one packet among the five packets was generated when the system was in state two. The third component is the delay the queuing packet accumulates by transitioning from the current state to a future state. For instance, if the system is in state five and it transitions to state four, the packets that do not get transmitted will incur extra delay.

$$d_i = \frac{1}{i}\left(\frac{1}{i}\left(\sum_{u=1}^{i+1} q_{u,i} d_u\right) + \left(1 + \sum_{h=i-1}^{N_m-1} q_{i,h} d_h\right)(i-1) + 1\right), \quad 2 \le i \le N_m - 1, \tag{4.8}$$

where u represents the previous state, h represents the future state, $q_{u,i}$ and $q_{i,h}$ are the transition probabilities from state u to state i and from state i to state h, respectively, and d_i, d_u, and d_h are the packet delays at states i, u, and h, respectively. When the system is in state one, the overall delay is equal to the base delay since there is only one packet in the system. State zero is excluded in (4.8) since it does not contribute to the delay.

Lastly, we calculate the total packet delay. To calculate the total packet delay, we need to find the steady-state probability for Q in (4.7). One method to obtain the steady-state probability is by using the power of the state transition probability matrix in (4.7). When the power of the Q matrix approaches infinity, the system reaches the steady-state, and every row in the state transition probability matrix Q converges to the steady-state probability as given in (4.9).

$$\prod = \lim_{r \to \infty} Q^r. \tag{4.9}$$

When r approaches infinity, all rows in the steady-state probability matrix $\prod$ become identical. Hence, any row vector in $\prod$ represents the steady-state probabilities of the state transitional probability matrix in (4.7). This approach to calculate the steady-state probability is a result of the $Chapman - Kolmogorov$ equation, which finds the state of the system after a certain number of steps (i.e., transitions). We use this method to find the steady-state probability matrix by approximating the number of steps to infinity.

Let π represents the steady-state probability vector, which contains the steady-state probabilities of states 1 to $N_m - 1$. Denoting the steady-state probability at state i given as π_i and using the packet delay at state i (i.e., d_i) from (4.8), we can calculate the average packet delay of all the devices assigned to the considered time slot as follows:

$$D^{Tot} = \sum_{i=1}^{Nm-1} \pi_i d_i, \tag{4.10}$$

where D^{Tot} in (4.10) is the packet delay from the instant a packet is generated until the packet is transmitted for any device sharing the slot.

4.3 Idle Slot Probability

An idle slot is a time slot where no transmission takes place. Conventionally a slot is idle if no device assigned to that slot has a packet to send. The unconventional scenario is when an AP receives transmissions from devices in the OZ connected to other APs. In this case, the AP may not receive the transmitted packets, and the slot is considered idle, although a transmission intended for a different AP happens in the slot. Both the conventional and the unconventional idle slot probability scenarios shall apply to the non-fully connected multiple-AP network.

First, we find the idle slot probability for the conventional scenario. Since the proposed protocol design in section 3.3 eliminates collisions, the idle slot probability can be calculated according to (4.11).

$$\eta_s = 1 - N_m \lambda_s T_f, \tag{4.11}$$

where η_s represents the idle slot probability for a particular slot s in the conventional idle slot probability scenario, $N_m \lambda_s T_f$ is the expected number of packets to arrive at all the devices sharing slot s, and λ_s is the packet arrival rate for any of the devices assigned to slot s.

In the unconventional idle slot probability scenario, some slots are perceived idle by one AP and active by another AP because of the devices in the OZs. Hence, the idle slot probability is calculated for each AP separately. For the AP of interest (i.e., the AP the idle slot probability is being calculated for), the slots in the OZs that are associated with neighboring APs are considered permanently idle when examined by the AP of interest. Therefore, finding each slot's idle probability separately is not enough since some slots are always idle. Instead, we must find the average idle slot probability for all the slots over an entire frame for the AP of interest, which is given as follows:

$$\eta'_{AP_X} = \frac{(N_o + \sum_{s=1}^{N_n} \eta_s)}{N_s}, \tag{4.12}$$

where $\eta^{'}_{AP_X}$ in (4.12) represents the idle slot probability over an entire frame for the AP of interest denoted as AP_X in (4.12). N_o is the number of permanently idle slots that are associated with neighboring APs of AP_X, which are generated from AP_X's OZs. N_n is the number of slots that are not permanently idle (i.e., follow the conventional idle slot probability scenario). N_n represents the non-permanently idle slots from the OZs and NOZ of AP_X. The devices assigned to the N_n slots transmit to AP_X, which processes their packets. N_s is the total number of slots registered with AP_X (i.e., the total number of permanently and non-permanently idle slots).

With no collisions, throughput for a particular AP is the complement of the idle slot probability for that AP as given in (4.13), where θ_{AP_X} represents the throughput of AP_X over the duration of one frame.

$$\theta_{AP_X} = 1 - \eta^{'}_{AP_X}. \tag{4.13}$$

4.4 Summary of Analysis

In this chapter, we analyzed the performance of the proposed protocol design with respect to two metrics packet delay and throughput. Both the delay and the throughput were analyzed for a single slot and the group of devices assigned to that slot. The focus of the performance analysis was the packet delay since the proposed protocol eliminates collisions. For the packet delay, we utilized a DTMC to create a virtual transmission queue, based upon which we developed a closed-form result to calculate the average packet delay for any of the devices sharing the time slot. In our analysis, we assumed that all the devices assigned to the slot have the same packet arrival rate. However, we can straightforwardly extend the design so that different devices in the slot can have different packet arrival rates and, consequently, different packet delays.

Chapter 5: PERFORMANCE EVALUATION

5.1 Introduction

This chapter shows the performance evaluation results of the proposed MAC protocol design for the average packet delay for different packet arrival rates. We compare the analytical results from chapter 4 with the numerical results. The numerical results are based on a computer simulation that mimics the environment of the network scenario we propose in our design. The simulations are performed for two networking scenarios, a non-fully connected single-AP network, and a non-fully connected multiple-AP network. In the simulation results, we examine the protocol design's capability to support massive connectivity while maintaining low latency without causing collisions due to hidden terminals or causing channel underutilization due to exposed terminals.

5.2 Simulation Setup

The work of this thesis extends and builds upon [31], such that similar settings and parameters are adopted. In the simulation, the radius of the coverage of each AP in the network is set to 250 meters, which is approximately the size of a factory. The mini-slot duration T_m needs to be long enough to sense the carrier signal. For the 250 meters radius of an AP's coverage, the propagation delay from edge to edge is under 2 milliseconds. Nevertheless, the mini-slot length is set to 9 milliseconds due to limitations in the physical layer. 9 milliseconds is an adequate duration to sense the carrier signal within the AP's coverage [40]. We consider the packet size to be 50 bytes with a data transmission rate of 3 Mb/s in accordance with the characteristics of MTC. The transmission duration of this packet (i.e., T_x) is 133 milliseconds.

With respect to the numerical simulation, in addition to the previous settings, the system is simulated for 2×10^6 frames. It is a relatively long period of time which gives highly consistent results. The base delay is half a frame (i.e., $T_f/2$). The number of devices varies from one AP to another in order for the simulation to be practical. Additionally, each AP has different numbers of devices connected to it in its OZs and NOZ (i.e., the devices are not evenly distributed throughout the network). All the network scenarios are simulated for low packet arrival rates that range between 0.1 packets per second and 1 packet per second and high packet arrival rates ranging between 1 packet per second and 4 packets per second.

5.3 Non-fully Connected Single-AP Network

For a non-fully connected network with a single-AP as described in section 3.3.1, the network is simulated for 1000 devices connected to one AP. The AP assigns 10 devices to each slot (i.e., each slot has 10 mini-slots). Hence, the frame length is 100 slots. The density of the devices throughout the network is not necessarily evenly distributed. Fig. 5.1 shows the simulation results for low packet arrival rates. When the packet arrival rate is 0.1 packets per second, the probability of two packets arriving at the same frame is very low. Hence, this arrival rate yields the lowest delay possible (i.e., the base delay), around 11 milliseconds. Fig. 5.2 shows the simulation results for the high packet arrival rates. When the packet arrival rate is 4 packets per second, the packet delay is 31.5 milliseconds. In both Fig. 5.1 and Fig. 5.2, the difference between analytical and numerical results is less than 5%.

The non-fully connected single-AP network simulation results are similar to the performance evaluation results in [31], in which the design was for a fully connected single-AP network. The difference is that in [31], the protocol design grants the devices assigned to the same slot different access priorities. In contrast, our design guarantees fair channel access to all the devices assigned to the same slot. The similar results show the validity of our analytical approach in this work since it did not stray from the literature when the applied network settings were the same.

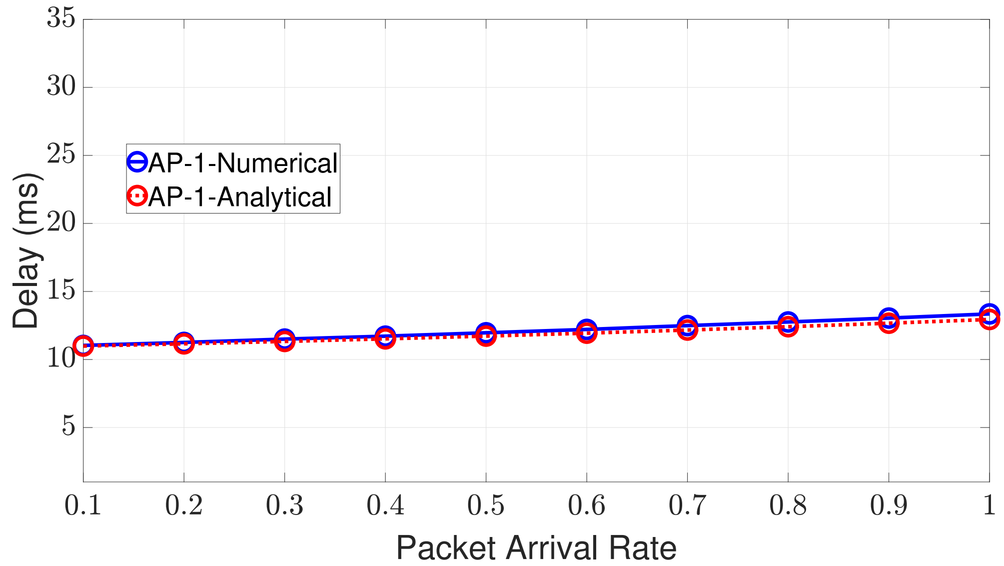

Figure 5.1: Delay Performance for a sigle-AP network with low packet arrival rates.

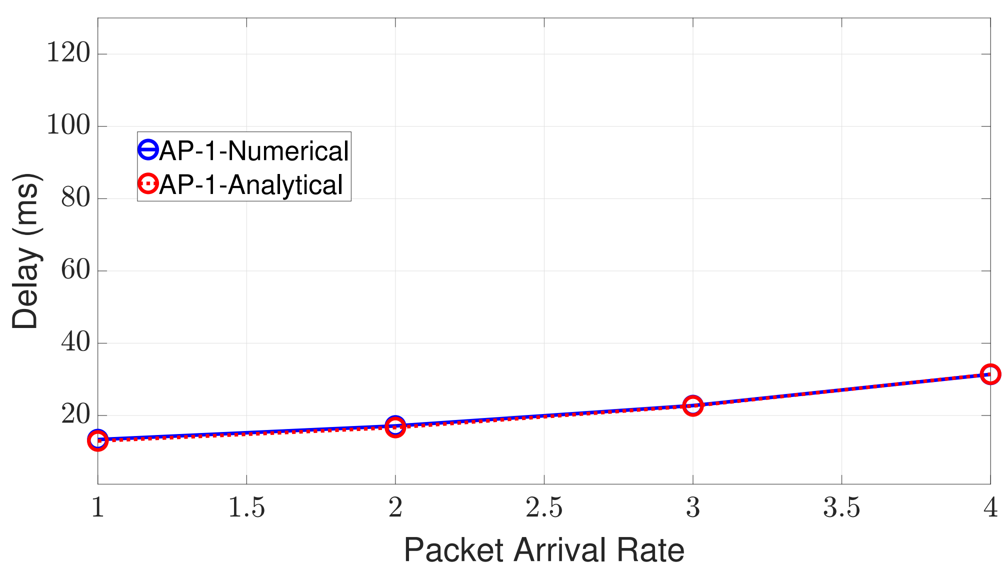

Figure 5.2: Delay Performance for a sigle-AP network with high packet arrival rates.

5.4 Non-fully Connected Multiple-AP Network

For a non-fully connected network with multiple-AP, as described in section 3.3.2, we simulate three different network scenarios, each with a different number of APs, to show in detail the performance and flexibility of the proposed protocol design in this thesis. In the multiple-AP networks, the number of mini-slots per slot varies depending on the number of devices each AP supports. The APs with higher numbers of devices have more mini-slots per slot than the APs with fewer devices.

A slot has at most 10 mini-slots in the simulations in this section. In a multiple-AP network, the AP with the highest number of devices assigns 10 mini-slots to each slot, while the APs with fewer devices assign less than 10 mini-slots per slot. The frame length is fixed for all the APs in the network (i.e., the number of slots and the slot duration are the same for all APs). This is necessary in order for all the APs in the network to be synchronized. Since the slot duration is fixed, the slot transmission duration for the APs with fewer mini-slots per slot will be longer than 133 milliseconds. For example, when the AP with the highest number of devices assigns 10 mini-slots per slot, the transmission duration is 133 milliseconds. In contrast, an AP with fewer devices that assigns 9 mini-slots per slot will have a transmission duration of 142 milliseconds since the duration of one mini-slot is added to the transmission duration.

5.4.1 2-AP Network

In the 2-AP network, one AP has 1000 devices connected to it, and the other AP has 800 devices connected to it. There is one OZ between the two APs. The devices in the OZ are added to the slot assignment of each APs. The number of devices in the OZ affects the packet delay at each AP differently, depending on how many devices from each AP are at the OZ. In this scenario, 30% of AP-1's devices are in the OZ, while 25% of AP-2's devices are in the OZ. Since AP-2 has fewer devices, it assigns fewer mini-slots per slot than AP-1, resulting in lower delays for devices connected to it. The frame length and slot duration are fixed for both APs in the network. Fig. 5.3 shows the result in the case of lower packet arrival rates for both APs in the network. At 0.1 packet per second, the average packet delay for the devices belonging to both APs is at its lowest and equals the base delay. Since the frame length is fixed, the based delay is the same for the deivces of both APs at around 13 milliseconds. With high packet arrival rates, as shown in Fig 5.4, the average packet delay for the devices connected to AP-1 is higher than those connected to AP-2 since AP-1 has more devices and more mini-slots per slot than AP-2. The highest packet delay for the devices connected to AP-1 when the packet arrival rate is 4 packets per second is 48.5 milliseconds, and the highest packet delay for the devices connected to AP-2 for the same packet arrival rate is 39.5

milliseconds.

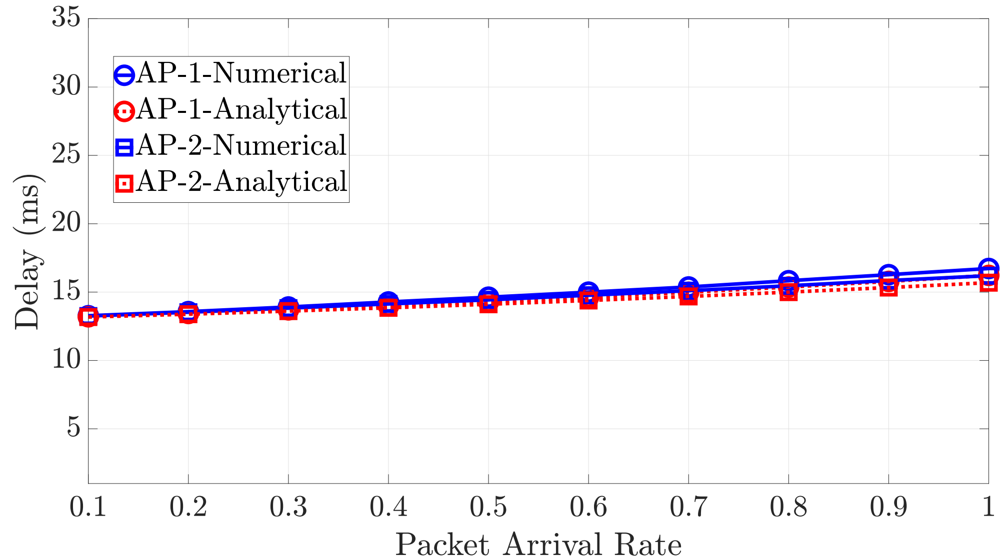

Figure 5.3: Delay Performance for a 2-AP network with low packet arrival rates.

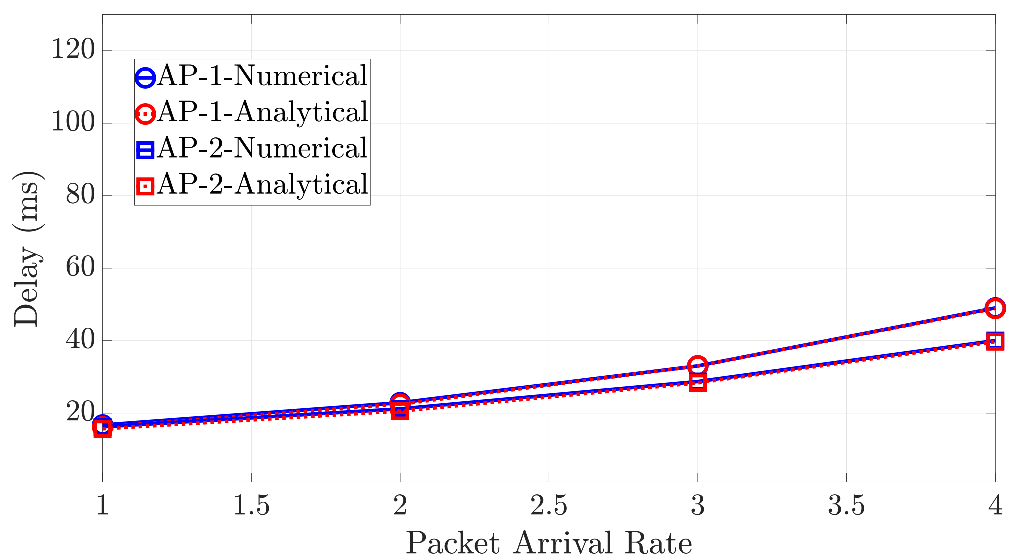

Figure 5.4: Delay Performance for a 2-AP network with high packet arrival rates.

5.4.2 3-AP Network

In the 3-AP network, AP-1 has 1000 devices, AP-2 has 950 devices, and AP-3 has 750 devices connected to each AP. In this scenario, there are four OZ, one between every two APs and one among all three APs. The percentages of devices in the OZs from AP-1, AP-2, and AP-3 are 41%, 23%, and 33%, respectively. Although AP-2 has fewer devices than AP-1, the higher number of devices in the OZs from AP-1 and AP-3 results in more devices added to AP-2's slot assignment, leading AP-2's devices to have the highest packet delay among the three APs. Similar to sub-section 5.4.1, the frame length and the slot duration are fixed for all APs in the network. AP-2 assigns the highest number of mini-slots per slot, and AP-3 assigns the least number of mini-slots per slot since it has the least number of devices among the three APs. Fig. 5.5 shows the low packet arrival rates for the 3-AP network. At 0.1 packet per second, the packet delay of all APs' devices is at its lowest and is equal to the base delay. The base delay is the same for all the APs' devices at around 16.5 milliseconds. With high packet arrival rates, as shown in Fig 5.6, devices connected to AP-2 have higher delays compared to those connected to the other two APs due to the high number of devices in the OZ from AP-1 and AP-3, while AP-3 has the lowest packet delay in the network. At 4 packets per second, the highest packet delay for the devices from AP-1, AP-2, and AP-3 are 65 milliseconds, 82.5 milliseconds, and 51 milliseconds, respectively.

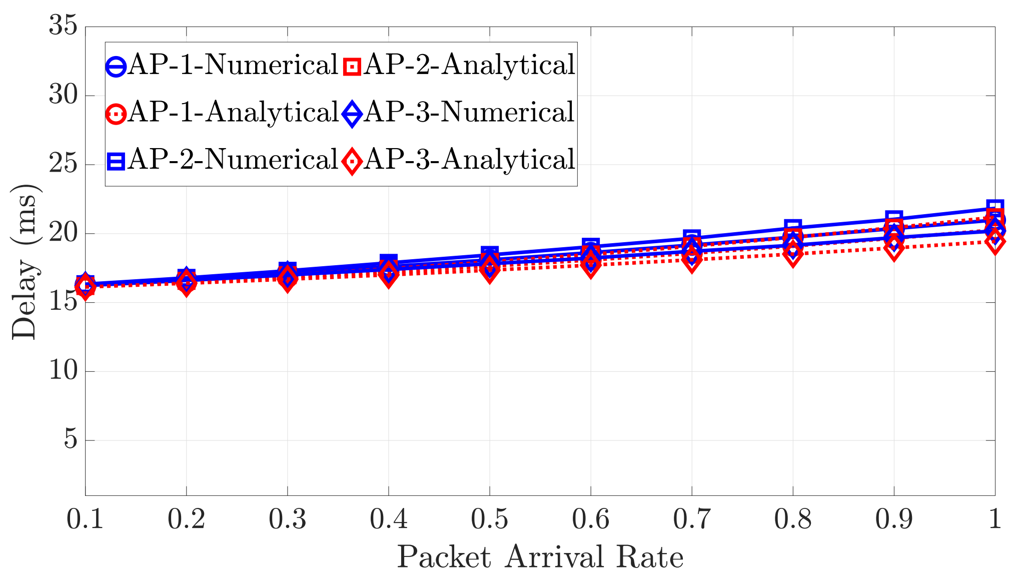

Figure 5.5: Delay Performance for a 3-AP network with low packet arrival rates.

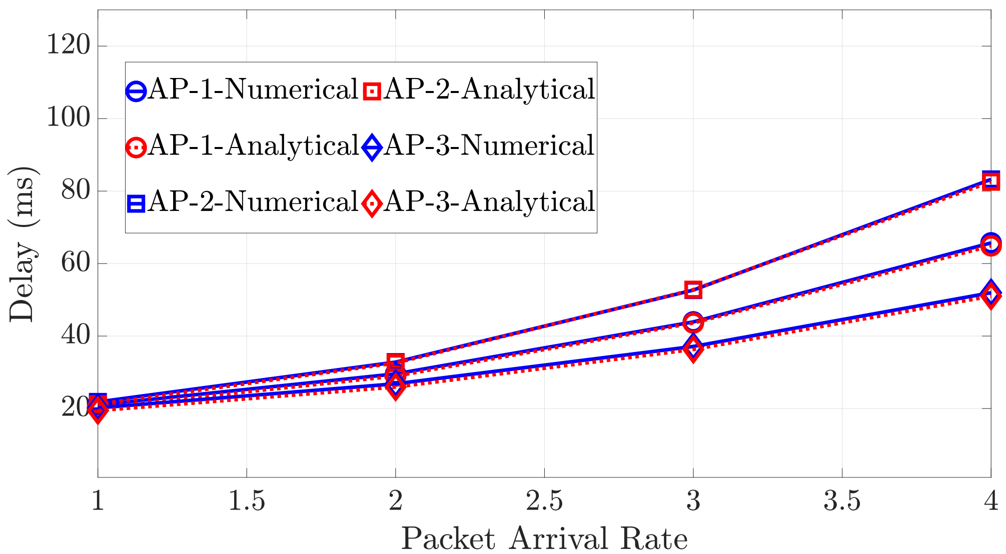

Figure 5.6: Delay Performance for a 3-AP network with high packet arrival rates.

5.4.3 Large-scale Network

The large-scale network scenario is more complex than the two network scenarios in sub-section 5.4.1 and sub-section 5.4.2. In this scenario, the network consists of 19 APs. Each AP has between 800 and 1000 devices connected to it. The total number of devices in the network is 17200 devices. An OZ is shared by a maximum of three APs. The number of devices located within the OZs for any AP constitutes between 19% and 50% of the total number of devices for that AP. Fig. 5.7 presents the simulation results for the low packet arrival rates. The figure shows the AP with the highest packet delay corresponding to the legends with ”AP-MAX”, the AP with the lowest packet delay corresponding to the legends with ”AP-MIN”, and the average packet delay (corresponding to the legends with ”AP-AVG”) for all the APs in the network. Similar to the networks in sub-sections 5.4.1 and 5.4.2 at 0.1 packets per second, the packet delay for all the APs’ devices is the same and is equal to the base delay, which is around 17 milliseconds. Fig. 5.8 depicts the results for the high packet arrival rates, like Fig. 5.7, Fig. 5.8 shows the AP with the highest packet delay, the AP with the lowest packet delay, and the average packet delay for all APs in the network. The delay increases for all the APs when the packet arrival rate increases, but the APs with more devices (i.e., more devices per slot) will suffer higher delays. At 4 packets per second, the packet delay for the AP with the largest number of devices is 88 milliseconds, while the packet delay for the AP with the least number of devices is 32 milliseconds. On average, the packet delay is 54 milliseconds.

Based on the above, an AP can dedicate some slots to have only a few devices assigned to these slots to ensure very low delay for the devices that require low delays.

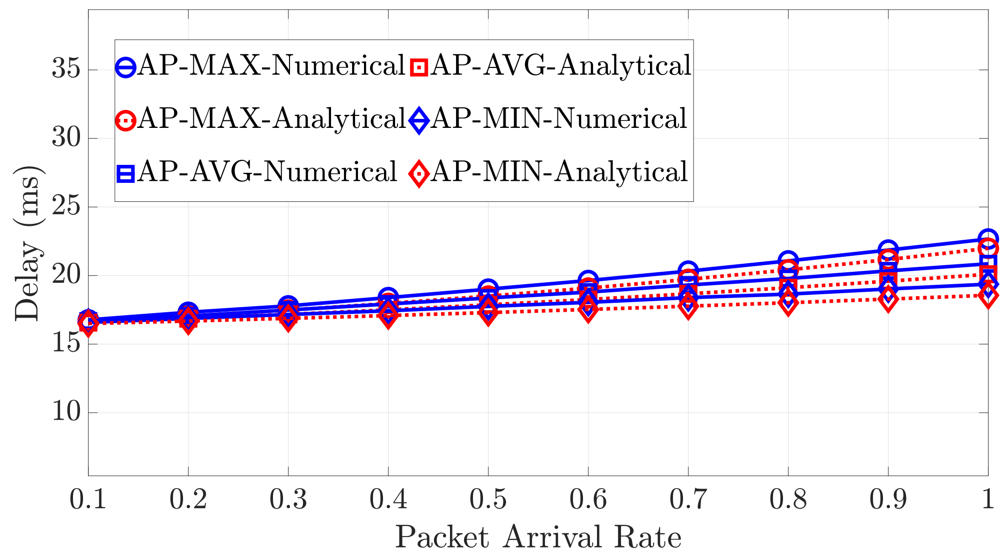

Figure 5.7: Delay Performance for a large-scale network with low packet arrival rates.

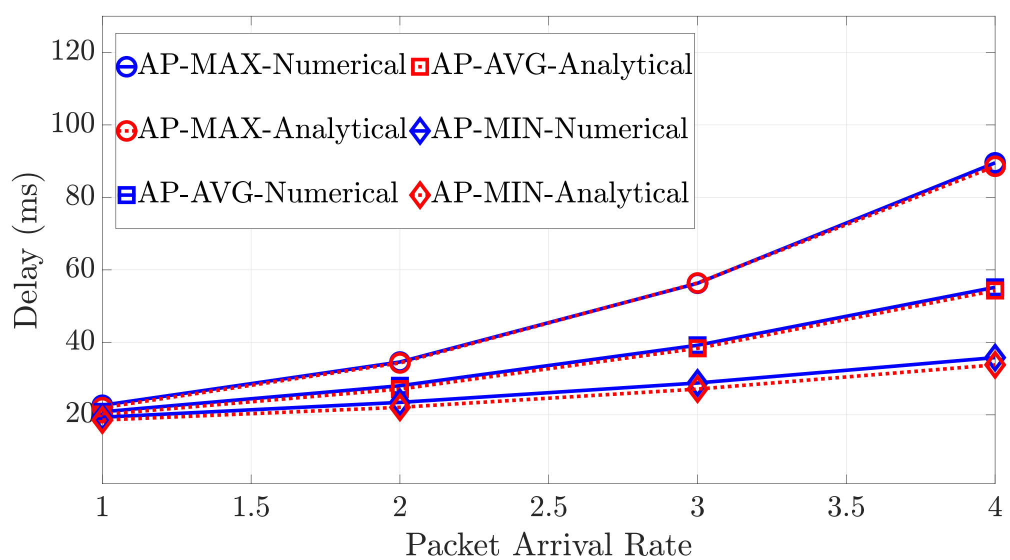

Figure 5.8: Delay Performance for a large scale network with high packet arrival rates.

5.5 Summary of Performance Evaluation

In this chapter, we evaluated the performance of the proposed MAC protocol design. Both numerical and analytical simulations were carried out for one scenario of a non-fully connected single-AP network and three scenarios of non-fully connected multiple-AP networks. There is less than a 5% discrepancy between analytical and numerical results. The simulation results from sub-section 5.4 demonstrate our design capability to provide a low delay to a massive number of devices with zero packet collision. The downside of the proposed scheduling scheme is that it uniformizes the base delay for all the APs in the network, but since the base delay is low, it does not affect the QoS for the connected devices. When the packet arrival rates increase, packet delay increases beyond the base delay. The APs with the highest number of devices (i.e., a higher number of mini-slots per slot) suffer higher packet delays compared to the APs with fewer devices (i.e., fewer mini-slots per slot).

Based on the above, any AP in the network, regardless of the number of devices connected to it, can ensure low latency access for some of its devices. An AP can dedicate some slots to have only a few devices assigned to these slots to provide a very low delay for the devices that require an ultra-low delay.

This chapter demonstrates the effectiveness of our design to ensure no packet collision due to hidden terminal problems and no network underutilization due to exposed terminal problems while maintaining massive access and low latency. The proposed design is implemented at the cost of network-wide synchronization and increased packet delay due to the devices added to the slot assignment from the OZs.

Chapter 6: CONCLUSION AND FUTURE WORK

6.1 Conclusion

In this thesis, we proposed a device location-based scheduling scheme specifically designed for a large-scale non-fully connected IIoT network with multiple APs that allow the connection of a large number of devices while maintaining a low average packet delay. The structure of the multiple-AP network forms overlapping zones and non-overlapping zones among the APs in the network, which in turn can cause hidden terminal problems and exposed terminal problems without well-designed coordination. There are two aspects to the proposed scheduling scheme, the MAC protocol and the slot assignment.

We used a fully distributed time-slotted carrier sensing-based MAC protocol that supports a large number of devices by allowing multiple devices to be assigned to a time slot. The MAC protocol allows multiple devices to share a single time slot, reducing wasted channel resources. Additionally, it eliminates packet collision among the group of devices sharing the slot through the sensing mechanism. This design offers a significant improvement over existing works. For instance, the works that use TDMA-based MAC protocols suffer from wasted channel resources. In contrast, works that use contention-based MAC protocols experience an exponential increase in packet collision and signaling overhead when the number of connected devices increases.

The slot assignment mechanism uses the device location information along with full coordination among the APs to eliminate the hidden and exposed terminal problems in the non-fully connected multiple-AP network, which is not well investigated in the literature. In existing works where the non-fully connected multiple-AP network scenario was studied, the overlapping zones among the APs were ideal and unrealistic in a practical network scenario. Additionally, the scheduling schemes that were proposed did not resolve the hidden and exposed terminal problems.

The simulation results for the multiple-AP IIoT network demonstrate the design's adaptability to support massive connections while maintaining the different delay requirements for IIoT devices by determining the number of devices assigned to a slot. This is illustrated in the performance evaluation, which shows how APs with different numbers of devices can have different packet delays. Any AP can ensure a lower packet delay to some devices by assigning fewer devices per slot. The highest packet delay for a network with 19 APs and 17200 devices was under 100 milliseconds when the packet arrival rate was at its highest, which is significantly less than the average packet arrival interval.

The scheduling scheme proposed in this thesis requires network-wide synchronization among all the APs in the network in order to implement full coordination among the APs. Furthermore, the average packet delay increases when a larger number of devices are connected to the AP of interest or when the number of devices shared from adjacent APs in the OZ of the AP of interest increases.

6.2 Future Work

In our future work, we aim to extend this design in two directions. The first approach is to expand the current scalability and support lower delay requirements. We customize the proposed scheduling scheme to support an ultra-dense large-scale network that could connect approximately 100,000 devices by adjusting the frame length, duration and number of slots, and duration and number of mini-slots per slot. On the other hand, the scheduling scheme can provide a submillisecond average packet delay for devices with extremely stringent delay requirements by assigning these devices to multiple slots in the frame.

Our work at the current stage only considers stationary devices. Hence, in the second approach, we aim to extend our work by enhancing the support for devices with mobility. The introduction of mobility to our design requires modifications to the proposed MAC protocol in order not to cause collisions. The primary approach will focus on creating two categories of slots. The first category of slots is assigned to mobile devices only, while the second category is assigned to stationary devices only. The mini-slots in the catagory of slots that support mobile devices will be specially designed to allow mobile devices to transmit while moving through the network without causing hidden and exposed terminal problems.

Bibliography

[1] R. Huo, S. Zeng, Z. Wang, J. Shang, W. Chen, T. Huang, S. Wang, F. R. Yu, and Y. Liu, "A comprehensive survey on blockchain in industrial internet of things: Motivations, research progresses, and future challenges," *IEEE Communications Surveys & Tutorials*, 2022.

[2] Y. Liu, M. Kashef, K. B. Lee, L. Benmohamed, and R. Candell, "Wireless network design for emerging IIoT applications: Reference framework and use cases," *Proceedings of the IEEE*, vol. 107, no. 6, pp. 1166–1192, 2019.

[3] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies, protocols, and applications," *IEEE communications surveys & tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.

[4] F. Javed, M. K. Afzal, M. Sharif, and B.-S. Kim, "Internet of things (IoT) operating systems support, networking technologies, applications, and challenges: A comparative review," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, pp. 2062–2100, 2018.

[5] M. R. Palattella, M. Dohler, A. Grieco, G. Rizzo, J. Torsner, T. Engel, and L. Ladid, "Internet of things in the 5G era: Enablers, architecture, and business models," *IEEE journal on selected areas in communications*, vol. 34, no. 3, pp. 510–527, 2016.

[6] D. Bandyopadhyay and J. Sen, "Internet of things: Applications and challenges in technology and standardization," *Wireless personal communications*, vol. 58, no. 1, pp. 49–69, 2011.

[7] Ericsson, "Ericsson mobility report: On the pulse of the networked society," *Ericsson Mobility Report*, 2016.

[8] L. Da Xu, W. He, and S. Li, "Internet of things in industries: A survey," *IEEE Transactions on industrial informatics*, vol. 10, no. 4, pp. 2233–2243, 2014.

[9] E. Sisinni, A. Saifullah, S. Han, U. Jennehag, and M. Gidlund, "Industrial internet of things: Challenges, opportunities, and directions," *IEEE transactions on industrial informatics*, vol. 14, no. 11, pp. 4724–4734, 2018.

[10] J. Åkerberg, M. Gidlund, and M. Björkman, "Future research challenges in wireless sensor and actuator networks targeting industrial automation," in *2011 9th IEEE International Conference on Industrial Informatics*. IEEE, 2011, pp. 410–415.

[11] 3GPP, "Study on scenarios and requirements for next generation access technologies," *Technical Specification Group Radio Access Network, Technical Report 38.913*, 2016.

[12] F. Ghavimi and H.-H. Chen, "M2M communications in 3GPP LTE/LTE-A networks: Architectures, service requirements, challenges, and applications," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 2, pp. 525–549, 2014.

[13] S. Chen, R. Ma, H.-H. Chen, H. Zhang, W. Meng, and J. Liu, "Machine-to-machine communications in ultra-dense networks—a survey," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1478–1503, 2017.

[14] Z. Dawy, W. Saad, A. Ghosh, J. G. Andrews, and E. Yaacoub, "Toward massive machine type cellular communications," *IEEE Wireless Communications*, vol. 24, no. 1, pp. 120–128, 2016.

[15] J. Kim, J. Lee, J. Kim, and J. Yun, "M2M service platforms: Survey, issues, and enabling technologies," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 61–76, 2013.

[16] F. Boccardi, R. W. Heath, A. Lozano, T. L. Marzetta, and P. Popovski, "Five disruptive technology directions for 5G," *IEEE communications magazine*, vol. 52, no. 2, pp. 74–80, 2014.

[17] What is the OSI model? Forcepoint. [Online]. Available: https://www.forcepoint.com/cyber-edu/osi-model/(2018)

[18] A. C. V. Gummalla and J. O. Limb, “Wireless medium access control protocols,” *IEEE Communications Surveys & Tutorials*, vol. 3, no. 2, pp. 2–15, 2000.

[19] G. L. Stüber and G. L. Steuber, *Principles of mobile communication*. Springer, 1996, vol. 2.

[20] V. Bharghavan, “A new protocol for medium access in wireless packet networks,” *online document*, 1999.

[21] R. M. de Oliveira, A. B. Vieira, H. A. Latchman, and M. V. Ribeiro, “Medium access control protocols for power line communication: A survey,” *IEEE Communications Surveys & Tutorials*, vol. 21, no. 1, pp. 920–939, 2018.

[22] M. Doudou, D. Djenouri, and N. Badache, “Survey on latency issues of asynchronous MAC protocols in delay-sensitive wireless sensor networks,” *IEEE Communications Surveys & Tutorials*, vol. 15, no. 2, pp. 528–550, 2012.

[23] “IEEE standard for broadband over power line networks: Medium access control and Physical Layer Specifications,” 2010.

[24] Z. Chang and Z. Ting, “Thorough analysis of MAC protocols in wireless sensor networks,” in *2008 4th International Conference on Wireless Communications, Networking and Mobile Computing*. IEEE, 2008, pp. 1–4.

[25] Y. Liu, C. Yuen, J. Chen, and X. Cao, “A scalable hybrid MAC protocol for massive M2M networks,” in *2013 IEEE wireless communications and networking conference (WCNC)*. IEEE, 2013, pp. 250–255.

[26] F. Meyer and V. Turau, “Delay-bounded scheduling in IEEE 802.15. 4e DSME using linear programming,” in *2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS)*. IEEE, 2019, pp. 659–666.

[27] A. Naguib, W. Saad, and M. Shokair, “Remaining energy aware ML-CSMA/TDMA hybrid mac protocol for LTE-M2M wireless network,” in *2019 International Conference on Innovative Trends in Computer Engineering (ITCE)*. IEEE, 2019, pp. 322–327.

[28] S. Moon and J.-W. Lee, “DMRS-Applied Repetition Transmission (DART): Grant-Free Scheme for mMTC,” in *2020 IEEE 3rd 5G World Forum (5GWF)*. IEEE, 2020, pp. 286–290.

[29] A. Ksentini, P. A. Frangoudis, P. Amogh, and N. Nikaein, “Providing low latency guarantees for slicing-ready 5G systems via two-level MAC scheduling,” *IEEE Network*, vol. 32, no. 6, pp. 116–123, 2018.

[30] J. Gao, W. Zhuang, M. Li, X. Shen, and X. Li, “MAC for machine-type communications in industrial IoT—Part I: Protocol design and analysis,” *IEEE Internet of Things Journal*, vol. 8, no. 12, pp. 9945–9957, 2021.

[31] J. Gao, M. Li, W. Zhuang, X. Shen, and X. Li, “MAC for machine-type communications in industrial IoT—Part II: Scheduling and Numerical Results,” *IEEE Internet of Things Journal*, vol. 8, no. 12, pp. 9958–9969, 2020.

[32] P. Wang and W. Zhuang, “A collision-free MAC scheme for multimedia wireless mesh backbone,” *IEEE Transactions on Wireless Communications*, vol. 8, no. 7, pp. 3577–3589, 2009.

[33] K. R. Malekshan, W. Zhuang, and Y. Lostanlen, “An energy efficient mac protocol for fully connected wireless ad hoc networks,” *IEEE Transactions on Wireless Communications*, vol. 13, no. 10, pp. 5729–5740, 2014.

[34] K. R. Malekshan, W. Zhuang, and Y. Lostanlen, “Coordination-based medium access control with space-reservation for wireless ad hoc networks,” *IEEE Transactions on Wireless Communications*, vol. 15, no. 2, pp. 1617–1628, 2015.

[35] K. Akimoto, S. Kameda, A. Taira, N. Suematsu, T. Takagi, and K. Tsubouchi, “Location-based virtual sector method for interference control in WLAN multicell environment,” in *2015 International Conference on Information and Communication Technology Convergence (ICTC)*. IEEE, 2015, pp. 451–456.

[36] K. Akimoto, S. Kameda, and N. Suematsu, “Optimum allocation scheme for user fairness of location-based virtual sector method solving hidden terminal problem in WLAN,” *IEEE transactions on vehicular technology*, vol. 67, no. 9, pp. 8363–8371, 2018.

[37] K. A. Nsiah, Z. Amjad, A. Sikora, B. Hilt, and J.-P. Lauffenburger, “Latency reduction techniques for NB-IoT networks,” in *2019 10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*, vol. 1. IEEE, 2019, pp. 478–482.

[38] A. Hassebo, A. Rezk, and M. A. Ali, “A hybrid uplink scheduling approach for supporting mission-critical smart grid applications in commercial 4G cellular networks,” in *2018 9th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*. IEEE, 2018, pp. 303–308.

[39] C. Bockelmann, N. Pratas, H. Nikopour, K. Au, T. Svensson, C. Stefanovic, P. Popovski, and A. Dekorsy, “Massive machine-type communications in 5g: Physical and mac-layer solutions,” *IEEE Communications Magazine*, vol. 54, no. 9, pp. 59–65, 2016.

[40] S. Yoon, L. E. Li, S. C. Liew, R. R. Choudhury, I. Rhee, and K. Tan, “Quicksense: Fast and energy-efficient channel sensing for dynamic spectrum access networks,” in *2013 Proceedings IEEE INFOCOM*. IEEE, 2013, pp. 2247–2255.